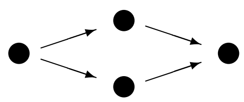
Partial Orders and Fully Abstract Models for Concurrency



Uffe Henrik Engberg

March 26, 1990

Contents

Preface

In this thesis sets of labelled partial orders are employed as fundamental mathematical entities for modelling nondeterministic and concurrent processes thereby obtaining so-called noninterleaving semantics. Based on different closures of sets of labelled partial orders, simple algebraic languages are given denotational models fully abstract w.r.t. corresponding behaviourally motivated equivalences. Some of the equivalences are accompanied by adequate logics and sound axiomatisations of which one is complete.

The majority of the work was done with a scholarship at the computer science department, University of Aarhus, Denmark. The rest was carried out with grant-in-aid from the Danish Research Academy during a visit at the technical University of Munich, Germany, where I enjoyed the hospitality of Wilfrid Brauer and his concurrency group.

The thesis has grown out of inspiring and encouraging talks with my supervisor Mogens Nielsen to whom I give my special thanks. I am also grateful to Kim S. Larsen for the discussions we had when preparing a joint paper with him and Mogens Nielsen. I should like to thank Anders Gammelgård for our discussions and Karen Møller for her part of the typing. Last, not least, thanks go to my parents and my wife Ricarda.

Uffe Henrik Engberg

Resumé

Den foreliggende licentiatafhandling placerer sig inden for området: semantiske modeller for parallelle systemer. En gren heraf er semantisk beskrivelse af konkrete programmeringssprog, hvori parallelisme og nondetermisme kan udtrykkes. Gennem den semantiske beskrivelse fastlægges hvilke processudtryk, der er ækvivalente, således, at det f.eks. giver mening at tale om, hvorvidt et processudtryk er en korrekt implementation af et andet, eller at en proces kan erstatte en anden i en given kontekst. Mange af studierne inden for området har taget udgangspunkt i mere abstrakte processprog som CCS, og de er blevet udstyret med forskellige former for semantik, eksempelvis: operationel, denotationel, aksiomatisk og logisk semantik.
Det er den operationelle semantik, som åbner mulighed for en intuitiv forståelse af, hvad en proces kan gøre, og hvilke egenskaber der er afgøerende for, at to processer opfører sig ens – er ækvivalente. Ofte er intuitionen den, at procesækvivalensen fremkommer i en eksperimental opsætning, hvor en observatør udfører tests på nogle maskiner i henhold til en bestemt "protokol", og hvor, det en maskine kan gøre, er bestemt ud fra det tilhørende procesudtryk.
For at sikre, at en semantik er i overensstemmelse med den operationelle intuition, er det derfor vigtigt med en præcis forbindelse til den operationelle semantik. Ved denotationelle semantikker er det formaliseret ved, at de denotationelle modeller er fuldt abstrakte m.h.t. de tilhørende operationelle ækvivalenser. D.v.s. de operationelle ækvivalenser er kongruenser (bevares i vilkårlige kontekster) og to processer giver anledning til de samme denotationer i modellerne, netop når processerne er operationelt ækvivalente (modellerne er "fully abstract"). For aktiomatiske semantikkers vedkommende er de tilsvarende begreber sundhed og fuldstændighed, hvor et bevissystem er sundt og fuldstændigt, når processer kan bevises ækvivalente, hvis og kun hvis de er operationelt ækvivalente. Ved logisk semantik forlanges typisk, at to processer tilfredstiller de samme logiske formler, præcist når de er ækvivalente.

Den overvejende del, af de sædvanlige operationelle ækvivalenser, adskiller sig primært ved hvilken grad af nondeterminisme de er i stand til at skelne, og har som fællestræk at (endelige) parallelle processer er ækvivalenete med tilsvarende rent nondeterministiske, men sekventielle processer – d.v.s. parallelisme reduceres til nondeterminisme. Flere af disse operationelle ækvivalenser er blevet karakteriseret logisk eller udstyret med sunde og fuldstændige bevissystemer, og nogle af ækvivalenserne er givet denotationelle modeller, som baserer sig på abstraktioner over beregningstræer, og som er fuldt abstrakte m.h.t. ækvivalenserne. Derimod er sådanne resultater mere sjældne, når det drejer sig om de ækvivalenser, hvor parallelisme ikke reduceres til nondeterminisme.

Ved i stedet at fokusere på parallelisme og negligere de nondeterministiske aspekter, når

tests og protokoller for eksperimenterne fastlægges, gives der i afhandlingen flere forskellige operationelt definerede ækvivalenser for simple processprog, og ækvivalenserne udstyres med fuldt abstrakte modeller, hvor mængder af mærkede partielle ordninger, forkortet m.p.o.'er, fungerer som den naturlige modpart til beregningstræer.

Afhandlingen består af en indledende præsentation samt to hoveddele, der hovedsageligst adskiller sig ved om det er testene eller protokollerne, der gøres til genstand for variation, når de operationelle ækvivalenser defineres. De to dele er skrevet uafhængigt af hinanden og kan derfor også læses adskilt. Overordnet følger begge dele den samme linie. Først foretages en isoleret undersøgelse af de objekter, der senere skal danne baggrund for de denotationelle modeller. Derefter gives operationel og denotationel semantik til det pågældende sprog for endelige processer, og det bevises, at de stemmer overens. Hver del afsluttes med tilføjelse af rekursion, og de tidligere denotationelle resultater løftes til den ny opsætning.

I den første del af afhandlingen betragtes et meget simpelt processprog med kombinatorer for (sekventiel) præfiksning af atomare aktioner, nondeterministisk valg, og parallelsammensætning uden auto-parallelitet, d.v.s. at en atomar aktion kun kan optræde i én af to parallelle processer. Gennem et lidt usædvanligt transitionssystem og en fastlagt type tests, designet til at afdække parallelitet, opnås tre forskellige operationelle ækvivalenser ved at betragte, hvordan processer fra sproget kan reagere på eksperimenter med testene. Til ækvivalenserne knyttes denotationelle modeller, der baserer sig på en klasse af m.p.o.'er, kaldet semiord, der reflekterer fravær af auto-parallelitet, og det bevises, at modellerne er fuldt abstrakte m.h.t. ækvivalenserne. Generelle m.p.o.'er, og dermed også semiord, kan bl.a. sammenlignes via to forskellige (partielle) ordninger, som udtaler sig om, hvorvidt én m.p.o. er et præfiks, henholdsvis glattere/ mindre sekventiel end en anden m.p.o.. Det viser sig, at de denotationelle afbildninger kan udtrykkes som bestemte lukninger af en kanonisk associering af semiord til procesudtryk. Disse lukninger er præfikslukninger, som igen, alt afhængig af den aktuelle ækvivalens, er opad-/ nedad-/ konvekslukkede m.h.t. "glatheds" relationen. Desuden gives et sundt bevissystem som ved en mindre udvidelse vises at være fuldstændigt for en af ækvivalenserne.

I den anden del betragtes et mere generelt processprog, der rummer mulighed for autoparallelitet og sekventiel sammensætning af vilkårlige processer. Eksperimenter fastlægges her til at være maksimale sekvenser af direkte tests, og i stedet gøres de direkte tests til genstand for variation. Med en enkelt direkte test undersøges, om visse typer af aktioner kan udføres parallelt på én gang. Hver "naturlig" mængde af direkte tests og tilhørende transitionssystem, giver anledning til en operationel ækvivalens, hvortil der knyttes en fuldt abstrakt model, der p.g.a. af det udvidede processprog, bygger på generelle m.p.o.'er. De denotationelle afbildninger følger samme mønster som i den første del, men de bestemte lukninger er her nedadlukninger, restringeret til lagdelte m.p.o.'er, hvor hvert lag svarer til en af de direkte tests, som er mulige ved den aktuelle ækvivalens. Af disse resultater afledes, at ækvivalenserne danner et gitter med den almindelige (automatteoretiske) strengækvivalens i bunden, som den mindst nuancerede m.h.t. hvilke processer, der kan skelnes. Hver af disse ækvivalenser karakteriseres ved en Hennessy-Milner-lignende lineær modallogik.
Til processproget føjes en forfinelseskombinator, der til hver atomar aktion angiver et procesudtryk (uden forfinelseskombinatorer) som aktionen skal implementeres ved. På

en simpel måde indkopereres den ny kombinator i transitionssystemerne, og det bevises, at den operationelle virkning er, som hvis de enkelte forfinelser på forhånd var tekstuelt substitueret ind for de pågældende atomare aktioner. Derved bliver der mulighed for, at forfinelser af parallelle aktioner kan "overlappe", hvorfor ækvivalenserne ikke bevares under den ny kombinator. Derfor studeres i stedet deres (største konsistente) kongruenser. Herved opnås én enkelt mere nuanceret kongruens. Kongruensen gives en fuldt abstrakt denotationel model, hvor den afgørende forskel er, at nedadlukningerne i stedet bliver restringeret til m.p.o.'er, som ikke kan skelnes ved "overlapning". For et delprocessprog uden auto-parallelitet karakteriseres kongruensen yderligere ved en modallogik, der, til forskel for de ovennævnte, har en ekstra modaloperator, hvormed en slags delvis baglås kan specificeres.

Sammenfattende kan det siges, at afhandlingen fremviser forskellige måder, hvorpå grader af parallelisme ved processer kan skelnes, enten gennem forskellige operationelt motiverede ækvivalenser, eller gennem de præordninger som ækvivalenserne er fremkommet af, og at mærkede partielle ordninger på naturlig måde tjener som hjørnesten i de tilhørende modeller.

Presentation

Introduction

Overall Background

During the last two decades attention to the area of concurrency has increased as programming concepts for handling nondeterministic and concurrent systems have been introduced while advances in hardware technology have made it realistic to use new programming languages incorporating these concepts. A great deal of the research has been made in order to achieve a good understanding of the meaning of concurrent systems and how to reason about them, an understanding comparable to that of sequential systems where e.g. the well-known axiomatic method of Hoare [Hoa69] is applicable for sequential programs. The ongoing research has resulted in a multitude of models for concurrency, for example Kahn-MacQueen networks [KM77], Mazurkiewicz traces [Maz77], Petri nets [Rei85], event structures [NPW81, Win87] and different semantics for process algebras. The main intention of this thesis is to contribute to this line of research.

Principal Confinement

Whereas it is standard to take the meaning of a sequential program as a function from input to output there is no prevailing agreement on what the meaning of concurrent programs should be. As De Nicola and Hennessy reason in [DNH84] it is necessary to search for counterpart to functions when building semantic theories for concurrency. In order not to obscure this task it is common practise to pay less attention to data aspects of concurrent programs and in stead investigate the fundamentals of control since this were the essential nature of concurrency lies. That is, in place of concrete programming languages for concurrency, like Concurrent Pascal, Modula-2 and Ada, abstract languages or process algebras containing combinators for the most fundamental notions of control – sequential, nondeterministic and parallel composition – are taken as starting point for the development of semantic theories for concurrency. This is also the case for the present thesis and deliberately only process languages with these fundamental, more algebraic combinators are studied. Prominent examples of larger process algebras which have been equipped with a broad spectrum of theories are CCS [Mil80, Mil84] and TCSP [Hoa78, BHR84].

General Requirement

Various forms of semantics for process algebras exist including: operational, denotational, axiomatic and logical – each of which contributes to knowledge and insight. Typically through labelled transition systems [Plo81] the operational semantics provide the means for an intuitive understanding of how concurrent processes behave and which processes are behaviourally equivalent. This is one of the main arguments when Milner (in e.g. [Mil83]) and many others argue that a semantic approach should be firmly based on an operational semantics. Consequently it will be a general requirement here too. Due to the importance of the requirement it has got an explicit formulation within the different types of semantics.
In case of denotational semantics it is formalized by the concept of a denotational map being fully abstract w.r.t. an associated behavioural equivalence. I.e. the interpretations of two processes in the denotational domain should be identified exactly when the processes are behaviourally equivalent.
As far as axiomatic semantics are concerned the analogous concepts are soundness and completeness – a proof system being sound when processes are provably equal only if they are behavioural equivalent, and complete if all such processes can be be proved equal.
Regarding semantics by logics one formulation of the requirement is adequacy. That means a logic is adequate when two processes satisfy the same set of formulas exactly when the processes are behaviourally undistinguishable.

Main Objective

The diversity of approaches to concurrency is also reflected in their attitude to the questions as to whether a linear or branching view of nondeterministic and concurrent systems should be taken, and whether concurrent processes should be reducible to purely nondeterministic, but sequential processes. When using a CCS/ TCSP like notation the first question can be illustrated by whether or not

$$(*) \qquad a.(b+c) \text{ and } a.b + a.c$$

should be identified, and similarly for the second whether or not

$$(**) \qquad a \parallel b \text{ and } a.b + b.a$$

should be distinguished. Changing from a look of controversy, the discussions around these questions seem now to have resulted in the understanding that there are no straight answers and that the attitude taken should depend on the situation at hand.

When concurrency is reduced to nondeterminism, concurrent processes are considered equivalent to ones with nondeterministic choice between different sequential shuffles of the individual processes as in (∗∗) above, and the semantics are often described as being interleaving. For CCS, TCSP and other process algebras the question of a linear or branching view has here led to a whole spectrum of behavioural equivalences ranging from trace equivalence (in the classical language theoretic sense – not to be mistaken for Mazurkiewicz traces) [Hoa85, OH86], which identify say (∗), over failure and testing [BHR84, DNH84, OH86] to bisimulation equivalence [Mil80, Par81, Mil84], equivalences which do not identify (∗). Operationally these equivalences differ mainly in their view of the branching structure of the labelled transition system associated with processes. Through the study of degrees of branching some of the equivalences have been given fully abstract denotational models where the counterparts to input-output functions (for sequential programs) can be viewed as abstractions of computation trees (also called synchronization trees) which in turn are slightly modified unfoldings of the corresponding labelled transitions systems.

In other approaches concurrency is independent of nondeterminism and the processes of (∗∗) are distinguished. Among these approaches are the so-called partial order semantics where causality, respectively concurrency, is represented by means of partial orderings of actions. I.e. alternatively to computation trees, constructions containing labelled partial orders (*lpos* for short) are proposed as counterparts to functions. These constructions are often sets of some kind of lpos and so nondeterminism cannot be discriminated in the semantics using them. However, it is possible in the denotational semantics based on a generalization of lpos, labelled event structures, where nondeterminism is dealt with by means of a conflict relation. See [BC87] for a good survey on the rôle of partial orders in semantics for concurrency. Apart from step semantics, different proposals for generalizations of existing behavioural equivalences (for nondeterminism) have been made with time-based equivalence [Hen88b] and distributed bisimulation [CH88] among the most discriminating. See also the final remarks of these papers. In the style of [Jon88, Rei88] the situation can roughly be sketched as:

		(∗) =	(∗) ≠
(∗∗)	=	Trace	Bisimulation
(∗∗)	≠	Step	Distributed Bisimulation

Behavioural process equivalence

		(∗) =	(∗) ≠
(∗∗)	=	Set of words	Computation tree
(∗∗)	≠	Set of lpos	Event structure

Entity modelling processes

Whereas the work on interleaving semantics has led to a number of e.g axiomatisation and full abstractness results, such results are more unusual when it comes to noninterleaving semantics. Motivated by this and the suggestion of Pnueli [Pnu85] to study degrees of concurrency in place of branching the main objective of the thesis is to explore the possibilities of defining "natural" operational semantics for algebraic process languages which open up opportunities for alternative semantics, especially for fully abstract denotational models with lpos as main ingredient of the entities modelling processes. That is to say we are seeking different behavioural equivalences where lpos come "naturally" in to the corresponding models, thereby capturing various degrees of nonsequentiality.

Possible Courses

Looking for ideas of how to modify behavioural equivalences such that the semantics is not interleaving, it immediately appears to try to catch a property which intuitively seems to be a distinctive characteristics of concurrency. To take an example one might argue that if a defect occurs in a subprocess then other concurrent subprocesses are able to run undisturbed (except of course if there is some dependence due to communication). If e.g. $\mho$ denotes the faulty process which cannot do any action and if besides the usual $a.p \stackrel{a}{\rightarrow} p$ also the rule $a.p \stackrel{a}{\rightarrow} \mho$ is used in the definition of the action relation, then many of the known behavioural equivalences would distinguish say $(**)$. In the introduction and final remarks of [Hen88b], Hennessy discusses other ideas and in the same paper and in [CH88, Cas88] the ideas are successfully examined obtaining axiomatisations for generalizations of bisimulation equivalence. However, bearing in mind the difficulties in finding fully abstract models for bisimulation equivalence, we deliberately choose to study degrees of concurrency as "orthogonal" to the existing study of degrees of branching. Taking the lead of [HM80, Mil80, DNH84, Abr87] the intuition will be that of a behavioural equivalence arising in an experimental setting with observers performing tests according to some "protocol" on machines, with operational abilities defined in terms of labelled transition systems. Though omitting branching aspects, the various manners in which to capture degrees of branching can serve as a clue for capturing degrees of nonsequentiality. For example, instead of having tests with different strengths in discovering nondeterminism, tests may in different ways be geared towards parallelism (possibly by departuring from the traditional labelled transition systems). Once tests capable of detecting some kind of concurrency are fixed, variations may be obtained by changing the "protocol" in the style of [DNH84]. Another direction to take is suggested in [Pnu85, BIM88] where increasing discriminating equivalences are obtained from a simple equivalence (trace) by considering the congruence when different combinators are added. So, finding combinators uncovering an aspect of concurrency, the congruence will be forced to take the aspect into account. These directions can be combined in several ways of which we have chosen two and elaborated each in a separate part of the thesis.

Overview and Basic Organization

The thesis is divided in two parts, which mainly differ in whether the tests or “protocols” of the experiments are subject to variations when the behavioural equivalences are defined.

In part I a particular kind of tests suitable to probe concurrency of processes is introduced for a simple process language, PL, and different equivalences are obtained by considering possible outcomes of the experiments. PL contains combinators for prefixing of atomic actions, nondeterministic choice and parallel composition (without communication). The experiments and the labelled transition system is somewhat unconventional. Here an atomic action can be thought of intuitively as connected to a certain resource thereby excluding auto-parallelism [vGV87] (an atomic action can only occur in óne of two parallel processes). When a signal, a, is submitted to initiate the action (ambiguously designated) a, this is noted such that other actions, possible the same, can be signaled to initiate. Each time the action a is completed this is signaled by $\bar{a}$ as response. At first an attempt is made to signal a (multi) set of actions and if this turns out well a test is made on the signaled actions, where the language for specifying tests contains constructs for what Abramsky [Abr87] calls traces and copying. The process may accept the experiment if the actions can be signaled and the following test is successful, and may reject the experiment if the actions can be signaled and the test is not successful. The three equivalences, $\underset{\sim}{\sqsupset\!\sqsubset}$, $\underset{\sim}{\sqsupset\!\sqsubset}_{\mathrm{a}}$ and $\underset{\sim}{\sqsupset\!\sqsubset}_{\mathrm{r}}$, are generated from the preorders $\underset{\sim}{\sqsubset}$, $\underset{\sim}{\sqsubset}_{\mathrm{a}}$ and $\underset{\sim}{\sqsubset}_{\mathrm{r}}$ respectively, where $\underset{\sim}{\sqsubset}$ is the intersection of $\underset{\sim}{\sqsubset}_{\mathrm{a}}$ and $\underset{\sim}{\sqsubset}_{\mathrm{r}}$, and one process is related via $\underset{\sim}{\sqsubset}_{\mathrm{a}}$ ($\underset{\sim}{\sqsubset}_{\mathrm{r}}$) to another if the experiments the first may accept (reject) also may be accepted (rejected) by the other.

Unlike in part one, the tests of the experiments in part II are varied when the different behavioural equivalence are introduced and the basic process language, BL, is slightly more general as auto-parallelism and full sequential composition is possible. Experiments are maximal sequences of direct tests and the variations arrise from the power admitted for the direct tests – with a single action tested as the weakest and a multiset the most powerful. For any “natural” fixed set of direct tests, G, processes are considered behaviourally equivalent, $\underset{\sim}{\gtrless}_{\mathsf{G}}$ (actually generated from a preorder $\lesssim_{\mathsf{G}}$), if they react identically to the same experiments. The equivalences are generalizations of the ordinary (maximal) trace equivalence which appears from the weakest direct tests.
Holding on to the behavioural equivalences BL is extended to RBL by adding a refinement combinator which makes it possible to prescribe through a map, called a BL-refinement, how atomic actions within the scope of the combinator should be refined or implemented in terms of basic processes of BL (change of atomicity). Because the refinement combinator enables “overlapping” of refined actions, the equivalences are not preserved under the new combinator and their finer associated congruences, $\underset{\sim}{\gtrless}^{c}_{\mathsf{G}}$, are considered. This part of the thesis is largely a continuation/ extension of [Lar88] and [NEL89] to cope with

auto-parallelism and recursion.

Both parts follow the same general line. At first lpos, or rather equivalence classes of lpos, are studied in their own right. Operations and the relations, prefixing and "smoother than" (where one lpo is smoother than another if the ordering relation of the first is a superset of that of the other lpo), are introduced and properties are derived – of course selecting certain topics in preparation for the models to come. In part I the study is actually confined to particular equivalence classes of lpos, called semiwords, where equally labelled elements are demanded to be ordered, thereby reflecting absence of auto-parallelism. One important property of semiwords is that they have canonic representatives wherefore definitions and reasoning can be made directly in terms of these. Aiming at similar conditions for the general equivalence classes of lpos, pomsets, elements of representatives are in part II taken from a certain ground set and in fact pomsets can to some extent be handled as smoothly as semiwords. Together pomsets and semiwords will in the rest of the presentation be referred to simply as lpos.

After the initial study of lpos, operational and denotational semantics are given of the process language in question and a connection between them is established. More specifically, the denotational models, which build on different closures of sets of lpos, are proved to be fully abstract w.r.t. the corresponding operational equivalences. Besides this, alternative methods to reason about the processes are given, and links to the equivalences are shown.

Finally each part is ended by adding recursion to the process language, and both the operational semantics and the denotational characterizations are extended accordingly. In part II new behavioural equivalences, $\underset{\approx}{\sqsubset}_{\mathsf{G}}$, come in by relaxing the maximality requirement of sequences (of direct tests). The new equivalences are not preserved in BL or RBL contexts, and their congruences, $\underset{\approx}{\sqsubset}^{c}_{\mathsf{G}}$, are studied. For this purpose a new criterion – a language being expressive w.r.t. a preorder – which ensures algebraicity of precongruences is introduced. More technical prerequisites are necessary in part II and for the same reason they are treated more thoroughly there. For instance two ways of extending (denotational) relations to open expressions are compared and proofs (of results mentioned in [Hen83]) are made in full detail. Acquaintance with standard denotational techniques for dealing with recursion as presented in [Hen88a] is assumed.

The two parts of the thesis are written and may be read independently and hence there is a few differences in notation and some redundancy around the treatment of lpos. As a help for the reader each part is equipped with an index of the most used notions, definitions and symbols. To avoid repeating references a common bibliography is included at the end of this presentation of the thesis.

Summary of results

We shall here briefly state the results of the thesis and start out by looking at the syntactic finite process languages (without recursion constructs) PL, BL and RBL, where PL as previously mentioned has combinators for prefixing, nondeterministic choice and parallel composition (without auto-parallelism and communication), BL in addition has auto-parallelism and full sequential composition and RBL a refinement combinator.

Operationally a new idea is introduced for PL. In the labelled transition system control is divided in two: at first nondeterministic choices are made during the act of signaling actions to initiate. These are in turn later be completed and vanish from the configurations.

For BL the operational capabilities are given via a more standard extended labelled transition system in the style of [Nic87, Hen88a] where an internal step is used to resolve (internal) nondeterministic choice. When it comes to RBL it turns out that a simple operational "lazy substitution" of refinements can be given by means of the internal step relation and this operational "substitution" is shown to coincide with the textual substitutions of refinements.

Looking at the models, we draw the attention to the fact that they consist of finite sets of lpos and that the denotational maps of the different models all can be regarded as some kind of closure of the same canonical association of lpos to process expressions. In addition the denotational maps admit simple compositional definitions, basicly built in terms of the operators used in the canonical maps and the relevant closure at the places where the closure is not preserved.

For PL and $\underset{\sim}{\sqcup}$ ($\underset{\sim}{\sqcup}_{\mathrm{a}}$ or $\underset{\sim}{\sqcup}_{\mathrm{r}}$ respectively) the closure used in the corresponding model, M_χ (M_δ or M_υ), is the prefix- and convex (downwards or upwards) closure w.r.t. the "smoother than" partial ordering of semiwords. The models are shown to provide suitable interpretations of the behavioural equivalences through the full abstractness results. From the models and examples it is seen that both $\underset{\sim}{\sqcup}_{\mathrm{a}}$ and $\underset{\sim}{\sqcup}_{\mathrm{r}}$ are strictly more abstract than $\underset{\sim}{\sqcup}$. Furthermore, a sound proof system, DED_π, is given which makes it possible to show statements concerning "prefix-closure" as well as more ordinary algebraic properties of the combinators such as commutativity and associativity of $+$ and $\|$. Extending DED_π to DED_δ by adding the axiom $a.(x \parallel y) \leq a.x \parallel y$ a sound and complete proof system is obtained for $\underset{\sim}{\sqcup}_{\mathrm{a}}$ (or rather $\underset{\sim}{\sqsubset}_{\mathrm{a}}$). In the style of [Hen88a] the results can be schematized:

SYNTAX:
PL

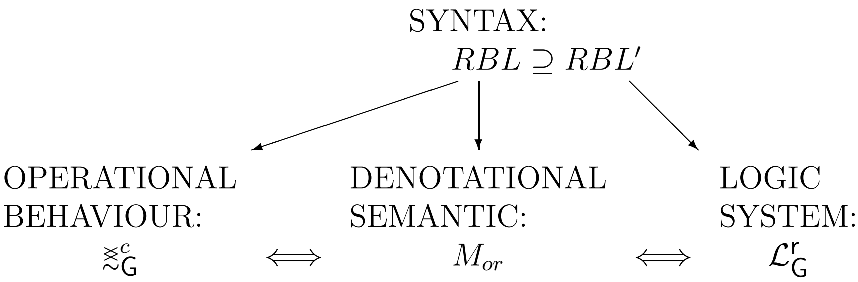
OPERATIONAL BEHAVIOUR:		DENOTATIONAL SEMANTIC:		PROOF SYSTEM:
$\underset{\sim}{\Box}$	$\Longleftrightarrow$	M_{χ}	$\Longleftarrow$	DED_{π}
$\underset{\sim}{\Box}_{\mathrm{r}}$	$\Longleftrightarrow$	M_{υ}	$\Longleftarrow$	DED_{π}
$\underset{\sim}{\Box}_{\mathrm{a}}$	$\Longleftrightarrow$	M_{δ}	$\Longleftrightarrow$	DED_{δ}

Turning to BL and fixing a set of direct tests, G, the closure of the the corresponding fully abstract model is the ordinary "smoother than" downwards closure of pomsets restricted to those pomsets which are "layered" and where each layer resembles a possible direct test from G. Varying G it is seen that the equivalences form a lattice (in the sense of their ability to distinguish processes) with the usual trace/ word equivalence, $\underset{\sim}{\bowtie}_{\mathsf{W}}$, at the bottom and the unrestricted multiset equivalence, $\underset{\sim}{\bowtie}_{\mathsf{M}}$, at the top. Each $\underset{\sim}{\bowtie}_{\mathsf{G}}$-equivalence is given an alternative characterization in terms of an adequate Hennessy-Milner like linear modal logic, $\mathcal{L}_{\mathsf{G}}$, containing a straight forward generalization of the "labelled" necessity modality (box) and atomic propositions expressing termination and non-termination. The results are sketched below:

SYNTAX:
BL

OPERATIONAL BEHAVIOUR:		DENOTATIONAL SEMANTIC:		LOGIC SYSTEM:
$\underset{\sim}{\bowtie}_{\mathsf{M}}$	$\Longleftrightarrow$	M_{M}	$\Longleftrightarrow$	$\mathcal{L}_{\mathsf{M}}$
$\underset{\sim}{\bowtie}_{\mathsf{G}}$	$\Longleftrightarrow$	M_{G}	$\Longleftrightarrow$	$\mathcal{L}_{\mathsf{G}}$
$\underset{\sim}{\bowtie}_{\mathsf{W}}$	$\Longleftrightarrow$	M_{W}	$\Longleftrightarrow$	$\mathcal{L}_{\mathsf{W}}$

The main observation for RBL is that when considering the largest congruences, $\underset{\sim}{\bowtie}^{c}_{\mathsf{G}}$, contained in the equivalences, $\underset{\sim}{\bowtie}_{\mathsf{G}}$, the addition of the refinement combinator collapse the lattice of equivalences into a strictly finer equivalence. Thereby also the result, $\underset{\sim}{\bowtie}^{c}_{\mathsf{W}} = \underset{\sim}{\bowtie}^{c}_{\mathsf{M}}$, which looks like a similar result Hennessy notices in the final remarks of [Hen88b] for time-based bisimulation. The closure used in the fully abstract model for $\underset{\sim}{\bowtie}^{c}_{\mathsf{G}}$ is again the downwards closure of pomsets, but instead restricted to those pomsets where of any two concurrent elements the successors of one also are successors of the other or vice versa. By removing auto-parallelism from RBL a sublanguage, RBL', is obtained which, beside resembling semiword based models, is equipped with an adequate logic, $\mathcal{L}^{\mathsf{r}}_{\mathsf{G}}$. An extra modality for specifying a kind of semi-deadlock is here at disposal. The schematized results are:

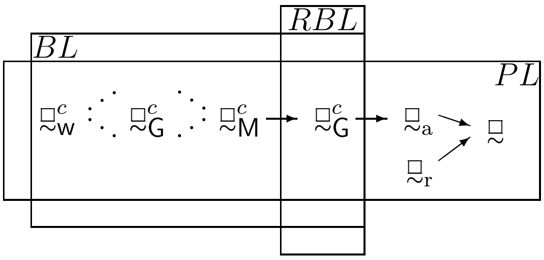


Now for the full process languages of PL, BL and RBL with recursion.

The transition systems for the different process languages are extended in the usual way to cope with recursion and in particular it is noticed for RBL that no extra (internal step) inference rule is needed for the interplay between the refinement combinator and the recursion constructor.

The models remain in principle the same but sets of lpos may now be infinite and the models, M^p_{G} and M^p_{or}, for $\underset{\sim}{\sqsubset}{}^c_{\mathsf{G}}$ w.r.t. BL and RBL respectively, separately carry information concerning approximating sequences. The domains of the finitary models are in a uniform way shown to be algebraic complete partial orders and the achieved models are proved to be fully abstract w.r.t. the corresponding behavioural equivalences. In this course a new criterion for algebraicity of precongruences turn out to be very useful.

PL can, modulo NIL and minor syntactic differences, be considered as a sublanguage of BL which in turn is a sublanguage of RBL. Then from the pleasant fact that both the M^p_{or} and M_δ model are expressed as abstractions over the downwards and prefix closure of a canonical association of lpos with expressions it follows that the relationship between the equivalences roughly can be illustrated as:



where $\rightarrow$ indicates that the equivalence on the left-hand side is strictly more abstract than the one on the right-hand side (the congruence of an equivalence is w.r.t. the language labelling the highest box the equivalence is contained in). Since the equivalence of the two parts only are compared here, we give two expressions, which illustrates that $\underset{\sim}{\sqsubset}{}^c_{\mathsf{G}}$ w.r.t. RBL is strictly more abstract (on PL) than $\underset{\sim}{\sqsubset}_{\mathrm{a}}$ (identified by $\underset{\sim}{\sqsubset}{}^c_{\mathsf{G}}$ but not by $\underset{\sim}{\sqsubset}_{\mathrm{a}}$):

$$(a.b \parallel c.d) + (b \parallel a.d \parallel c) + (a \parallel c.b \parallel d) \quad \text{and} \quad (b \parallel a.d \parallel c) + (a \parallel c.b \parallel d)$$

To sum up the achievements of the thesis one could say that means are brought about for discriminating degrees of concurrency in processes, either through different behavioural equivalences or through the preorders they are generated from, and that labelled partial orders in a natural way serve as cornerstones in the associated models.

Conclusion

The full abstractness results are obtained at the expense of simplified process languages and an undetailed view on branching. We shall here discuss a few ideas to redress some of the shortcomings and their impact on the results.

For PL the requirement of absence of auto-parallelism is crucial. This is best seen in the proofs of full abstractness which rely heavely on the fact that semiwords are characterized by their linearizations and no characterization of the pomsets that are identified by their linearizations is known. But by omitting auto-parallelism, it looks manageable to extend PL to BL and keep the results. Now consider what happens if a refinement combinator which does not introduce auto-parallelism is added, either to PL or the extension. Then it is unlikely that it will have any influence, at least not on the $\underset{\approx}{\sqsubset}_{\mathsf{a}}$-equivalence, since two refined processes (without $+$), which can be distinguished by sequences, already are distinguished by the may-experiments on the unrefined processes.

Whereas the combinators of BL are quite simple this is by no means the case for the refinement combinator of RBL, but it suffers from an effective way to be specified. As it is now, a refinement is given by a function from the (infinite) set of atomic actions to the process expressions of BL. One way to go would be to introduce the notation $[a_1 \rightsquigarrow p_1, \ldots, a_n \rightsquigarrow p_n]$ for the refinement where all actions remain unrefined except that a_1 is refined to p_1, a_2 to p_2, etc. and only allow such refinements. Then it would not be possible to specify fission refinements as they are formulated now, but a closer look at the proofs, where these refinements are used, shows that refinements which "fission" on a finite set will do and so all the results go through. With the refinement combinator it is possible to imitate relabelling by considering the relabelling functions as a special class of BL-refinements (maps to individual atomic processes). Looking at the way relabelling usually is introduced in transition systems, the relabelling combinator is stactic in nature in contrast to the more dynamic nature of the refinement combinator, but this difference cannot be uncovered by the equivalences. Inaction (NIL, $SKIP$) seems also easy to include in RBL. The few proofs, where the refinements are assumed not to make actions disappear (ε-freeness), get more complicated. A (maybe unexpected) consequence of adding NIL would be that expressions like a and $a + NIL$ would be distinguished by $\underset{\sim}{\bowtie}_{\mathsf{G}}$ and also by the congruence of $\underset{\approx}{\sqsubset}_{\mathsf{G}}$ (think of a context where the expressions are sequential composed by another action b). Once inaction is added to RBL it is no problem to simulate hiding of an action a; simply use the refinement combinator $[a \rightsquigarrow NIL]$. However the use of such an abstraction feature is limited as long as parallel processes cannot communicate – a matter we shall address next.

The extensions discussed until now stay so to say within the simplified view on branching.

But if we extend the parallel combinator of RBL such that e.g. synchronization shall happen on all common actions as in TCSP [BHR84] and we still look at maximal sequences, we would at once get a finer view, because the possibility of deadlock forces the model to reflect branching structure – see [Pnu85]. We have on purpose carried out this work on nonsequentiality "orthogonally" to existing work on branching, but it is an intriguing question, whether such an extension could be modeled by a smooth combination of e.g. the M_{or} model and the broom model of Pnueli – capturing aspects of nonsequentiality as well as branching.

We conclude by a simple example which indicates that such a combination in no way is straightforward to obtain. Suppose

$$p = a \parallel b \quad \text{and} \quad q = a.b + b.a + a \parallel b$$

Then p and q are identified in both the M_{or} model and the broom model, but $p' = p[a \rightsquigarrow c.d]$ and $q' = q[a \rightsquigarrow c.d]$ would be distinguishable in a parallel context with $c.b.d$ – c is a possible maximal sequence of $q' \parallel c.b.d$ whereas this is not the case for $p' \parallel c.b.d$. Hence a "conjunction" of the two models would be to abstract for the congruence of $\approx_{\mathsf{G}}$ w.r.t the two combinators.

Bibliography

[Abr87] Samson Abramsky. Observation Equivalence as a Testing Equivalence. *Theoretical Computer Science*, 53:225–241, 1987.

[BC87] Gérad Boudol and Ilaria Castellani. Concurrency and Atomicity. Rapports de Recherche 748, INRIA, 1987.

[BF88] Eike Best and César C. Fernández. *Nonsequential Processes, A Petrinet View*, volume 13 of *EATCS Monogaphs on Theoretical Computer Science*. Springer-Verlag, 1988.

[BHR84] S.D. Brookes, Charles Anthony Richard Hoare, and A.W. Roscoe. A Theory of Communicating Sequential Processes. *Journal of the ACM*, 31(3):560–599, 1984.

[BIM88] Bard Bloom, Sorin Istrail, and Albert R. Meyer. Bisimulation Can't be Traced: Preliminary Report. In *15th Ann. ACM Symp. on Principles of Programming Languages*, pages 229–239, 1988.

[Cas88] Ilaria Castellani. *Bisimulations for Concurrency*. PhD thesis, University of Edinburgh, 1988.

[CH88] Ilaria Castellani and Matthew Hennessy. Distributed Bisimulations. Rapports de Recherche 875, INRIA, 1988.

[DM87] Pierpaola Degano and Ugo Monatanari. Concurrent Histories: A Basis for Observing Distributed Systems. *Journal of Computer and System Sciences*, 34:422–461, 1987.

[DNH84] Rocco De Nicola and M.C.B. Hennessy. Testing Equivalences for Processes. *Theoretical Computer Science*, 34:83–133, 1984.

[Gis84] Jay Loren Gischer. *Partial Orders and the Theory of Shuffle*. PhD thesis, Stanford University, 1984.

[Gis88] Jay Loren Gischer. The Equatinal Theory of Pomsets. *Theoretical Computer Science*, 61(2,3):199–224, 1988.

[Gra81] Jan Grabowski. On Partial Languages. *Annales Societatis Mathematicae Polonae*, IV(2):427–498, 1981.

[GTWW77] J. A. Goguen, J. W. Thatcher, E. G. Wagner, and J. B. Wright. Initial Algebra Semantics and Continuous Algebras. *Journal of the ACM*, 24:68–95, 1977.

[Hen83] Matthew Hennessy. Synchronous and Asynchronous Experiments on Processes. *Information and Control*, 59:36–83, 1983.

[Hen85a] Matthew Hennessy. An Algebraic Theory of Processes. Technical Report DAIMI FN–25, Department of Computer Science , Aarhus University, 1985.

[Hen85b] Matthew Hennessy. An Algebraic Theory of Processes (Part Two). Technical Report DAIMI FN–26, Department of Computer Science , Aarhus University, 1985.

[Hen87a] Matthew Hennessy. An Algebraic Theory of Processes. Preparation of [Hen85a] and [Hen85b] for a Book, 1987.

[Hen87b] Matthew Hennessy. Axiomatising Finite Concurrent Processes. Technical Report 4/84, University of Sussex, 1987.

[Hen87c] Martin C. Henson. *Elements of Functional Languages*. Blackwell Scientific Publications, 1987.

[Hen88a] Matthew Hennessy. *Algebraic Theory of Processes*. Series in the Foundations of Computing. MIT Press, 1988.

[Hen88b] Matthew Hennessy. Axiomatising Finite Concurrent Processes. *SIAM Journal on Computing*, 17(5):997–1017, 1988.

[HM80] Matthew Hennessy and Robin Milner. On Observing Nondeterminism and Concurrency. *LNCS*, 85:299–309, 1980.

[HM85] Matthew Hennessy and Robin Milner. Algebraic Laws for Nondeterminism and Concurrency. *Journal of the ACM*, 32(1):137–161, 1985.

[Hoa69] Charles Anthony Richard Hoare. An Axiomatic Basis for Computer Programming. *Communication of the ACM*, 12(10):576–583, 1969.

[Hoa78] Charles Anthony Richard Hoare. Communicating Sequential Processes. *Communication of the ACM*, 21(8):666–677, 1978.

[Hoa85] Charles Anthony Richard Hoare. *Communicating Sequential Processes*. Series In Computer Science. Prentice Hall, 1985.

[Jon88] Bengt Jonsson. Report on the REX Schooll/ Workshop on Linear Time, Branching Time and Partial Orders in Logics and Models for Concurrency, Nordwijkerhout, The Netherlands, May/ June 1988. *EATCS Bulletin*, 36:261–263, 1988.

[KM77] Gilles Kahn and David B. MacQueen. Corutines and Networks of Parallel Processes. In *IFIP Congress*, pages 993–998. North-Holland, 1977.

[Lar88] Kim Skak Larsen. A Fully Abstract Model for a Processes Algebra with refinement. Master's thesis, Department of Computer Science , Aarhus University, 1988.

[Maz77] Antoni Mazurkiewicz. Concurrent Program Schemes and their Interpretations. Technical Report DAIMI PB–78, Department of Computer Science , Aarhus University, june 1977. Aarhus Workshop on Verification of Parallel Processes.

[Mil80] Robin Milner. *A Calculus of Communicating Systems*, volume 92 of *LNCS*. Springer-Verlag, 1980.

[Mil83] Robin Milner. Calculi for Synchrony and Asynchrony. *Theoretical Computer Science*, 25:267–310, 1983.

[Mil84] Robin Milner. Lectures on a Calculus of Communicating Systems. In *Seminar on Concurrency, Pittsburgh, July 1984*, pages 197–220. Springer-Verlag (*LNCS* 197), 1984.

[NEL89] Mogens Nielsen, Uffe Engberg, and Kim Skak Larsen. A Simpel Process Language with Refinement. In *REX Schooll/ Workshop on Linear Time, Branching Time and Partial Orders in Logics and Models for Concurrency, Nordwijkerhout, The Netherlands, May/ June 1988*, pages 523–548. Springer-Verlag (*LNCS* 354), 1989.

[Nic87] Rocco De Nicola. CCS Without τ's. In *Proc. TAPSOFT 87 (Pisa), vol. 1*, pages 138–152. Springer-Verlag (*LNCS* 249), 1987.

[NPW81] Mogens Nielsen, Gordon D. Plotkin, and Glynn Winskel. Petri Nets, Event Structures and Domains (Part I). *Theoretical Computer Science*, 13(1):85–108, 1981.

[OH86] Ernst-Rüdiger Olderog and Charles Anthony Richard Hoare. Specification-Oriented Semantics for Communicating Processes. *Acta Informatica*, 23:9–66, 1986.

[Par81] David Park. Concurrency and Automata on Infinite Sequences. In *Theoretical Computer Science, 5th GI-Conference, Karlsruhe, March 1981*, pages 167–183. Springer-Verlag (*LNCS* 104), 1981.

[Plo81] Gordon D. Plotkin. A Structural Approach to Operational Semantics. Technical Report DAIMI FN–19, Department of Computer Science , Aarhus University, 1981. Available only from University of Edinburgh.

[Pnu85] Amir Pnueli. Linear and Branching Structures in the Semantics and Logics of Reactive Systems. In *12th ICALP*, pages 15–32. Springer-Verlag (*LNCS* 194), 1985.

[Pra86] Vaughan Pratt. Modeling Concurrency with Partial Orders. *International Journal of Parallel Programming*, 15(1):33–71, 1986.

[Rei85] Wolfgang Reisig. *Petri Nets, An Introduction*, volume 4 of *EATCS Monogaphs on Theoretical Computer Science*. Springer-Verlag, 1985.

[Rei88] Wolfgang Reisig. A Report on REX Schooll/ Workshop on Linear Time, Branching Time and Partial Orders in Logics and Models for Concurrency, Nordwijkerhout, The Netherlands, May/ June 1988. *EATCS Bulletin*, 36:264–267, 1988.

[Sta81] Peter H. Starke. Processes in Petri Nets. *EIK*, 17(8/9):389–416, 1981.

[Sti85] Colin Stirling. A Proof-Theoretic Characterization of Observational Equivalence. *Theoretical Computer Science*, 39:27–45, 1985.

[vGV87] R. van Glabbeek and F. Vaandrager. Petri Net Models for Algebraic Theories of Concurrency. *LNCS*, 259:224–242, 1987.

[Win85] Glynn Winskel. On the Composition and Decomposition of Assertions. Technical Report 59, University of Cambridge, Computer Laboratory, 1985.

[Win87] Glynn Winskel. Event Structures. In *Petri Nets: Application and Relations to Other Models of concurrency, Advances in Petri Nets, Sept. 1986*, pages 325–392. Springer-Verlag (*LNCS* 255), 1987.

Part I

Testing Partial Orders

Chapter 1

Semiwords: *SW*

1.0 Preliminaries

Partial orders are often used to reflect causal relationships between events. In this chapter we shall present a special subclass of labelled partial orders called *semiwords* and find a number properties semiwords enjoys. Roughly speaking a semiword is a labelled partial order where the equal labelled elements are ordered. Before giving the exact definitions of labelled partial orders and semiwords we start out by a few mathematical and other conventions.

Propositions and definitions are numbered within chapters, e.g., definition 1.0.1 (the definition below) where the first number indicates the chapter it appears in and the second is the number of the definition.

If $\leq$ is a partial order over A the downwards closure of an element $a \in A$ w.r.t. $\leq$ will be denoted $DC_{\leq}(a)$, i.e., $DC_{\leq}(a) = \{b \in A \mid b \leq a\}$. Similar $UC_{\leq}(a)$ denote the upwards closure of a w.r.t. $\leq$. We shall often use functions defined on sets, so in order not to write to many parenthesis we shall write fS for the function application $f(S)$ where S is a set and at the same time an element in the domain of f. The standard set, $\{1, \ldots, n\}$, will be denoted $\underline{n}$ and a tuple of the form $(t_1, \ldots, t_k)$ is abbreviated $\bar{t}$.

Definition 1.0.1 Given a nonempty set Δ, a *labelled partial ordering* (lpo for short) *over* Δ is a triple $(A, \leq, \beta)$, where $\beta : A \to \Delta$ is a mapping from A into Δ and $\leq$ partially orders the set A or equally $(A, \leq)$ is a poset, i.e., $\leq$ is a binary relation on A which is reflexive, transitive and antisymmetric. $\square$

A can be regarded as events, i.e., particular occurrences of actions and Δ, the alphabet, as actions, or types of events.

Definition 1.0.2 Two lpos $\rho = (A, \leq, \beta)$ and $\rho' = (A', \leq', \beta')$ are said to be isomorphic (written $\rho \cong \rho'$) *iff* there exists a bijection $\phi : A \to A'$ such that for all $a, b \in A : \beta(a) = \beta(\phi(a))$, and $a \leq b$ *iff* $\phi(a) \leq' \phi(b)$.

The equivalence class under $\cong$ of any lpo ρ is denoted $[\rho]$ i.e., $[\rho] := \{\rho' \mid \rho' \cong \rho\}$ and ρ is called a representative. If $\rho = (A, \leq, \beta)$ we also write the corresponding equivalence class as $[A, \leq, \beta]$.

The subset of the quotient set of the lpos over Δ by $\cong$ where the posets of the representatives are finite are called the set of *partial words* over Δ (written $PW(\Delta)$), i.e., $PW(\Delta) := \{[A, \leq, \beta] \mid (A, \leq) \text{ is a } \textit{finite poset}, \beta : A \to \Delta\}$.

The subset of the partial words over Δ where the equal labelled elements of the representatives are linearly ordered are called the set of *semiwords* over Δ (written $SW(\Delta)$ or SW for short), i.e., $SW(\Delta) := \{[A, \leq, \beta] \in PW(\Delta) \mid \forall a, b \in A : \beta(a) = \beta(b) \Rightarrow a \leq b \vee b \leq a\}$. (So the partial order restricted to equal labelled elements satisfies the trichotomy law.) □

The semiwords were first introduced by Starke [Sta81] and reflects the idea that two occurrences (events) of the same action cannot be concurrent.

Though many of the following notions and results could be formulated and hold for $PW(\Delta)$ we prefer to introduce them for semiwords only. First of all because we are only concerned with semiwords in this work and second, because they have a particularly simple representation which we shall refer to as the canonic representatives.

Canonic representatives

According to Starke [Sta81] the canonic representatives can be characterized as follows:

Let A be a finite subset of $\Delta \times \mathbb{N}^+$ and $\leq$ be a partial order on A. Then $(A, \leq)$ is the canonic representative of a semiword over Δ *iff* for all $a \in \Delta, i, j \in \mathbb{N}^+$ it holds:

$SW1$: $(a, i) \in A \wedge 1 \leq j \leq i \Rightarrow (a, j) \in A$

$SW2$: $(a, i), (a, j) \in A \Rightarrow ((a, i) \leq (a, j) \Leftrightarrow i \leq_{\mathbb{N}} j)$

Intuitively (a, i) denotes the i^{th} occurrence of the action a, i.e., (a, i) is a label a with rank i ($SW1$). Since all equal labelled elements are linearly ordered ($SW2$) this gives sense and from Starke it follows that the mapping which for a canonic representative $(A, \leq)$ gives the semiword $[A, \leq, \beta]$, where $\beta((a, i)) = a$, is an isomorphism.

In the sequel we will identify a semiword $s = [\rho]$, $\rho = (A, \leq, \beta)$ with its canonic representative which we denote $(A_s, \leq_s)$ or just $(A, \leq)$ when it is clear from the context. We shall therefore refer to SW as the subclass of the lpos which satisfies $SW1$ and $SW2$.

1.1 Basic Definitions

Notationally, it will be convenient to let a^i denote (a, i). If the rank of the element is unimportant for an argument or statement, we will simply omit the rank i of a^i and just write a.

If not only equal labelled elements of a semiword but all elements are linearly ordered, we call it a word. Formally:

Definition 1.1.1 Let $s \in SW$. s is a *word* over Δ *iff* $\leq_s$ satisfies the trichotomy law:

$$\forall a, b \in A_s.\ a \leq_s b \vee b \leq_s a$$

The set of all words over Δ is denoted $W(\Delta)$ or W for short. Notice $s \in W$ implies $\leq_s$ is total. □

The one to one correspondence between Δ^* and W should be clear (if not, see [Sta81]) and in the sequel we will often identify their members.

In order to introduce operators on SW it will be useful to define a function ψ, which given a semiword s and an action a yields the maximal rank of an element of s labelled by a. Because of $SW1$ this number equals the number of elements labelled with the action, so we can use this for the formal definition:

Definition 1.1.2 ψ: $SW \times \Delta \longrightarrow I\!N$ is defined by:

$$(s, a) \mapsto |\{b^i \in A_s \mid b = a\}|$$

□

This allows us to introduce some further notions.

Definition 1.1.3 For a poset $(A, \leq)$ and a set B the *restriction of* $(A, \leq)$ *to* B (written $(A, \leq)|_B$) is defined to be the poset $(A|_B, \leq|_{B^2})$.

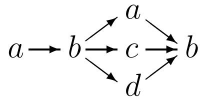
s is a *direct subsemiword of* t *iff* s is a semiword and $s = t|_{A_s}$.

If s is a direct subsemiword of u the *complement semiword* t *of* s (w.r.t. u/in u) is defined to be $t|_{A_u \setminus A_s}$ shifted left according to s. I.e.,

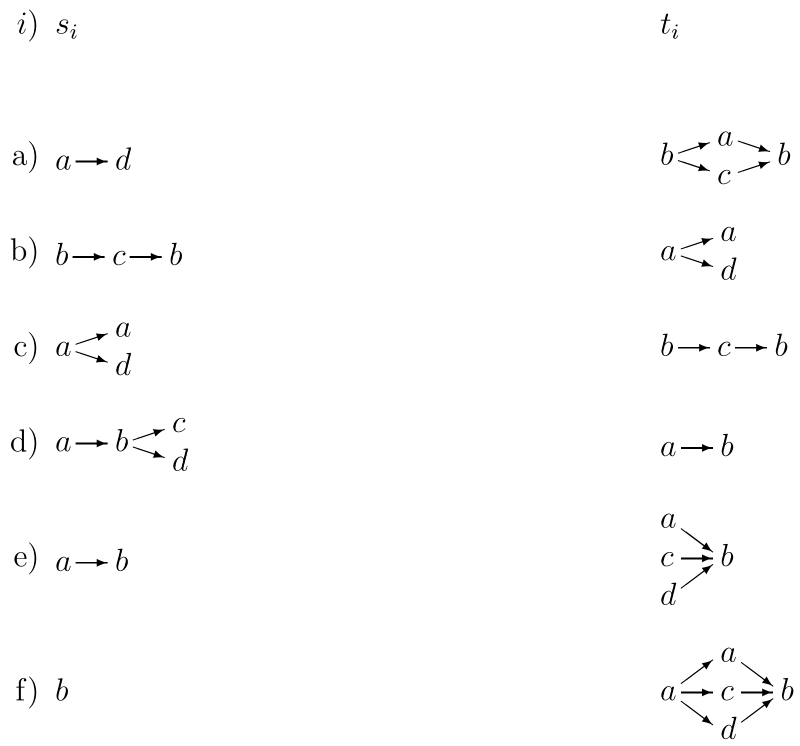
$$A_t := \{a^{i-\psi(s,a)} \mid a^i \in A_u \setminus A_s\}$$
$$\forall a^i, b^j \in A_t.\ a^i \leq_t b^j \text{ iff } a^{i+\psi(s,a)} \leq_u|_{(A_u \setminus A_s)^2} b^{j+\psi(s,b)}$$

For convenience direct subsemiwords will be referred to simply as subsemiwords. □

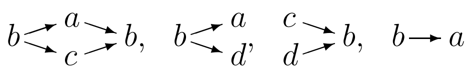
One could have defined a more general notion of subsemiword, but with this definition the subsemiword is directly represented by itself. Furthermore this definition suffice for our purpose.

Example: Let u be the semiword 

s_i below are subsemiwords of u and t_i complement semiword of s_i in u.



The following semiwords are not subsemiwords of u:



From t_a in the example we see that although a semiword is not a subsemiword of a semiword u, it can be a complement semiword. b) and c) shows that a complement semiword and a subsemiword may change rôle, whereas d) and e) shows it is not always the case. Also notice that although e.g., $b \to a$ is a direct part of the picture of u (i.e., $b \to a$ is a subsemiword in a more general sense) it is neither a subsemiword of u nor a complement semiword in u.

Proposition 1.1.4

a) If $A \subseteq A_s$ fulfills $SW1$ then $s|_A$ is a subsemiword.

b) The complement semiword in definition 1.1.3 is in fact a semiword.

Proof
a) We shall show that $\leq_s|_{A^2}$ is a po and fulfills $SW2$.
Since $\leq_s$ is a po on A_s, it must be so on $A \cap A_s$ too. Similar the $SW2$-property must hold on $A \cap A_s$ also.

b) First we prove that A_u fulfills $SW1$.
Let $a^i \in A_t$ and a j such that $1 \leq j \leq i$. We shall prove $a^j \in A_t$.
By definition of t it follows that $a^i \in A_t$ implies $a^{i+\psi(s,a)} \in A_u \setminus A_s \subseteq A_u$. Now $1 \leq j \leq i$

implies $1 \leq j + \psi(s,a) \leq i + \psi(s,a)$, so because u is a semiword and therefore fulfills $SW1$ we have $a^{j+\psi(s,a)} \in A_u$. $a^{j+\psi(s,a)} \notin A_s$ because otherwise $j + \psi(s,a) \leq \psi(s,a)$ which obviously is impossible since $1 \leq j$. Hence $a^{j+\psi(s,a)} \in A_u \setminus A_s$. Then by definition of t: $a^{(j+\psi(s,a))-\psi(s,a)} = a^j \in A_t$.

$\leq_u$ is a po fulfilling $SW2$ so this holds for $\leq_u|_{(A_u \setminus A_s)^2}$ too. By definition of $\leq_t$ this must then also be the case for $\leq_t$. □

Definition 1.1.5 Two semiwords s and t are said to be *disjoint iff* $A_s \cap A_t = \emptyset$. □

Because s, t disjoint implies $\forall a^i \in A_t.\ \psi(s,a) = 0$ we get:

Corollary 1.1.6

a) If s is a subsemiword of u then: $\leq_s \subseteq \leq_u$.

b) and if t is the complement semiword of s in u and disjoint to s then $A_s \cup A_t = A_u$ and $\leq_t \subseteq \leq_u$.

Definition 1.1.7 Given a poset $(A, \leq)$.
$a, b \in A$ are *connected iff* a and b are connected when considering the undirected graph of $\leq$, i.e., when $(a,b) \in (\leq \cup \leq^{-1})^+$.

A poset $(C, \leq')$ is a *maximal connected component* of a poset $(A, \leq)$ *iff* $(C, \leq') = (A, \leq)|_C$ and all elements of C are connected and there is no $a \in C$ and $b \in A \setminus C$ which are connected. For the sake of convenience we will in the sequel just say connected component instead of maximal connected component.

The set of all connected components of a poset $(A, \leq)$ is denoted $\gamma((A, \leq))$.

Since SW is a subclass of the posets, we can talk of the connect components of a semiword as well.

If s is a semiword such that $\gamma(s) = \{\varepsilon, s\}$ we say that s is a *connected semiword.*

□

It is not difficult to see:

Corollary 1.1.8

a) A connected component of a semiword is also a subsemiword of it.

b) For a semiword s, $\gamma(s)$ consists of mutual disjoint semiwords.

c) $s \mapsto \gamma(s)$ can be considered as a function $\gamma : SW \to \mathcal{P}(SW)$.

d) $\{\varepsilon, s\} \subseteq \gamma(s)$.

e) $\gamma(s) = \{\varepsilon, s\}$ *iff* s is a connected component.

where $\mathcal{P}(A)$ denote the power set of A.

1.2 Operations on SW

In this section we shall introduce some of the operators on SW presented by Starke in [Sta81] where he also displays the most fundamental properties of the operators.

Nullary

The semiword with canonic representative $(\emptyset, \emptyset)$ is denoted ε and is called the empty (semi-) word. For every action $a \in \Delta$ we select a corresponding semiword which has the canonic representative $(\{a^1\}, \{(a^1, a^1)\})$ and denote it $\underline{a}$.

Corollary 1.2.1

a) $\gamma(s) = \{\varepsilon\} \Leftrightarrow s = \varepsilon$

b) $\gamma(\underline{a}) = \{\varepsilon, \underline{a}\}$.

Unary

With the previous nullary operators and the concatenation of semiwords defined below, we easily derive an unary operator $a.$ for every $a \in \Delta$. Namely:

Definition 1.2.2 Let $a \in \Delta$. Then $a. : SW \rightarrow SW$ is defined by $s \mapsto \underline{a}s$, where $\underline{a}s$ means the the concatenation of $\underline{a}$ and s. □

From the properties of concatenation we derive:

Corollary 1.2.3

a) $a.s = \underline{a} \Leftrightarrow s = \varepsilon$

b) $a.s = b.t \Leftrightarrow a = b, s = t$

c) $a.s \in W \Leftrightarrow s \in W$

Binary

The definition of *concatenation* displayed as juxtaposition of the operands or placing a . (dot) between the operands is:

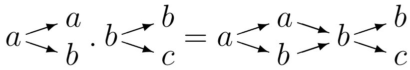
Definition 1.2.4 Concatenation of semiwords, $. : SW \times SW \rightarrow SW$, is defined by $(s, t) \mapsto s.t = st = (A, \leq)$, where

$$
\begin{aligned}
& A = A_s \cup \{a^{i+\psi(s,a)} \mid a^i \in A_t\} \\
& \leq \subseteq A \times A \text{ is defined by:} \\
& \quad \forall a^i, b^j \in A.\ a^i \leq b^j \textit{ iff } \quad i \leq \psi(s,a), j \leq \psi(s,b),\ a^i \leq_s b^j \\
& \quad \quad \text{or} \quad i \leq \psi(s,a), \psi(s,b) < j \\
& \quad \quad \text{or} \quad \psi(s,a) < i, \psi(s,b) < j,\ a^{i-\psi(s,a)} \leq_t b^{j-\psi(s,b)}
\end{aligned}
$$

□

Notice that $A_s \subseteq A_{st}$ and $\forall a \in A_s\ \forall b \in A_{st} \setminus A_s.\ a \leq_{st} b$.

Example:



Corollary 1.2.5 For all $s, t, u \in SW$:

a) $st \in SW$ (well-defined)

b) $s(tu) = (st)u$ (associative)

c) $\varepsilon s = s = s\varepsilon$ (ε unit)

d) $st = su \Rightarrow t = u$ (left cancellation)

e) $ts = us \Rightarrow t = u$ (right cancellation)

Recalling the definition of subsemiword and complement semiword and inspecting the definition of concatenation, we immediately get:

Corollary 1.2.6 s is a subsemiword of st and t the complement semiword of s in st.

The close connection between sub- and complement semiwords of a semiword and concatenation can be further illuminated by the following:

Proposition 1.2.7 Let s be a subsemiword of u. Define t to be the complement semiword of s (w.r.t. u). Then:

a) $A_u = A_{st}$

b) $\forall a, b \in A_s.$ $\quad a \leq_u b \Leftrightarrow a \leq_{st} b$

c) $\forall a \in A_s\ \forall b \in A_u \setminus A_s.$ $\quad a \leq_u b \Rightarrow a \leq_{st} b$

d) $\forall a, b \in A_u \setminus A_s.$ $\quad a \leq_u b \Leftrightarrow a \leq_{st} b$

Notice that we *cannot* conclude $u = st$ from a) – d). Later when dealing with partial orders on SW, we will see some conditions which ensure that there exist such s and t.

Proof

a) $A_{st} = A_s \cup \{a^{i+\psi(s,a)} \mid a^i \in A_t\} = A_s \cup \{a^{i+\psi(s,a)} \mid a^i \in \{a^{j-\psi(s,a)} \mid a^j \in A_u \setminus A_s\}\} = A_s \cup \{a^{(j-\psi(s,a))+\psi(s,a)} \mid a^j \in A_u \setminus A_s\} = A_s \cup (A_u \setminus A_s) = A_u$, where the last equation follows from the fact that s being a subsemiword of u implies $A_s \subseteq A_u$.

b) s is a subsemiword of u wherefore $\leq_s$ agrees with $\leq_u$ restricted to A_s. s is also subsemiword of st and the result follows.

c) Because of a) we see $A_u \setminus A_s = A_{st} \setminus A_s$, so the rest is trivial since we have already noticed by definition of concatenation that $\forall a \in A_s\ \forall b \in A_{st} \setminus A_s.\ a \leq_{st} b$ (no matter whether $a \leq_u b$ or not).

d) Assume a and b actually are a^i and b^j respectively. Since $a^i, b^j \in A_u \setminus A_s$ we have $a^i \leq_u b^j$ *iff* $a^i \leq_u|_{(A_u \setminus A_s)^2} b^j$ which by definition of t is equivalent to $a^{i-\psi(s,a)} \leq_t b^{j-\psi(s,b)}$ (notice that $a^i, b^j \in A_u \setminus A_s, A_s \subseteq A_u \Rightarrow \psi(s,a) < i, \psi(s,b) < j$). This again, by definition of concatenation, is equivalent to $a^i \leq_{st} b^j$. □

Proposition 1.2.8 Let u be a connected nonempty semiword. Then:

a) $\gamma(ut) = \{\varepsilon, ut\}$

b) $\gamma(su) = \{\varepsilon, su\}$

c) $s, t \neq \varepsilon \Rightarrow \gamma(st) = \{\varepsilon, st\}$

Proof

a) Assume $\gamma(u) = \{\varepsilon, u\}$. By corollary 1.1.8.d) $\{\varepsilon, ut\} \subseteq \gamma(ut)$. So what remains to be proved is $r \in \gamma(ut) \Rightarrow r \in \{\varepsilon, ut\}$. One consequence of $r \in \gamma(ut)$ is $r = ut|_{A_r}$. If either $A_r = \emptyset$ or $A_r = A_{ut}$ the result is clear, so assume $\emptyset \neq A_r \neq A_{ut}$.
Then there exist $a \in A_r, b \in A_{ut} \setminus A_r$ and since r is a connected component of ut, a and b cannot be connected. We look at the different possible memberships of a, b w.r.t. A_u and A_{ut}.

$a, b \in A_u \subseteq A_{ut}$: Since u is connected a and b must be connected—a contradiction.

$a \in A_u, b \in A_{ut} \setminus A_u$: Then as noticed by concatenation $a \leq_{ut} b$ and thereby connected—again a contradiction.

$b \in A_u, a \in A_{ut} \setminus A_u$: Similar.

$a, b \in A_{ut} \setminus A_u$: Since $\{\varepsilon, u\} \neq \{\varepsilon\} \Rightarrow u \neq \varepsilon$ there exists a $c \in A_u$. Again as noticed by concatenation $c \leq_{ut} a$ and $c \leq_{ut} b$. Hence a and b are connected and we get a contradiction again.

We have exhausted all possible memberships of a, b and each time got a contradiction, so the assumption $\emptyset \neq A_r \neq A_{ut}$ was wrong. Hence ε, ut are the only connected components of ut.

b) Similar.

c) Let $a, b \in A_{st}$. If we can show that they are connected corollary 1.1.8.e) gives $\gamma(st) = \{\varepsilon, st\}$. Three cases to consider.

$a, b \in A_s$: Since $t \neq \varepsilon$ we have a $c \in A_{st} \setminus A_s$. By proposition 1.2.7.d) $a \leq_{st} c, b \leq_{st} c$, so connected.

$a \in A_s, b \in A_{st} \setminus A_s$: proposition 1.2.7 gives directly that they are connected.

$a, b \in A_{st} \setminus A_s$: Since $s \neq \varepsilon$ we have some $c \in A_s$. Again by proposition 1.2.7 we have $c \leq_{st} a, c \leq_{st} b$ and thereby connected.

□

For words we have the following connection:

Corollary 1.2.9 $st \in W \Leftrightarrow s, t \in W$.

The parallel composition of semiwords is defined:

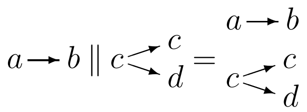
Definition 1.2.10 Let s, t be two disjoint semiwords. Then the *parallel composition* of s and t is:

$$s \parallel t := (A_s \cup A_t, \leq_s \cup \leq_t)$$

□

So parallel composition is only partially defined.

Example:



Corollary 1.2.11 For all $s, t, u \in SW$, mutual disjoint:

a) $s \parallel t \in SW$ (well-defined)

b) $s \parallel t = t \parallel s$ (commutative)

c) $(s \parallel t) \parallel u = s \parallel (t \parallel u)$ (associative)

d) $\varepsilon \parallel s = s = \varepsilon \parallel s$ (ε unit)

e) $s \parallel t = s \parallel u \Rightarrow t = u$ (left cancellation)

f) $t \parallel s = u \parallel s \Rightarrow t = u$ (right cancellation)

Since $\parallel$ is associative we can omit brackets. Furthermore because $\parallel$ additionally is commutative and has ε as neutral element, we can even for a set D of semiwords write $\parallel\{s \mid s \in D\}$ or just $\parallel D$ for short to denote $s_1 \parallel s_2 \ldots \parallel s_n$ where $D = \{s_1, \ldots, s_n\}$. If $D = \emptyset$ then $\parallel D$ denotes ε.

To avoid the proviso of disjointness of semiwords whenever writing expressions involving $\parallel$ we will in the sequel tacitly assume this.

Corollary 1.2.12 If s is a subsemiword of u and t the complement semiword of s in u, similar for s', t', u' then:

a) $s \parallel s'$ is a subsemiword of $u \parallel u'$ and

b) $t \parallel t'$ is the complement semiword of $s \parallel s'$ in $u \parallel u'$.

Proposition 1.2.13

a) $\gamma(s \parallel t) \setminus \{\varepsilon\} = \gamma(s) \setminus \{\varepsilon\} \uplus \gamma(t) \setminus \{\varepsilon\}$ b) $s = \parallel \gamma(s)$

where $\uplus$ means disjoint union of sets.

Proof
a) Trivial.

b) By induction on the size of $\gamma(s)$

$|\gamma(s)| = 1$: Since $\varepsilon \in \gamma(s)$ for all $s \in SW$ we have $s = \varepsilon$ and $\gamma(s) = \{\varepsilon\}$ from which the result follows.

$|\gamma(s)| > 1$: Then there is a $t \in \gamma(s)$ with $t \neq \varepsilon$. Clearly $r = s|_{A_s \setminus A_t}$ is a subsemiword of s (corollary 1.1.8) and $s = t \parallel r$. Since $t \neq \varepsilon$ and hence $A_t \neq \emptyset$ we must have $|\gamma(r)| < |\gamma(s)|$ and the result then follows by applying the inductive hypothesis on r and using a). □

From this proposition and the other concerning γ we obtain the following corollary.

Corollary 1.2.14

a) $a.s = t_1 \parallel t_2 \Rightarrow \begin{cases} t_1 = a.s, t_2 = \varepsilon \\ \text{or} \\ t_1 = \varepsilon, t_2 = a.s \end{cases}$

b) $s_1 \parallel s_2 = t_1 \parallel t_2$
$\Downarrow$
$\exists t_i^j (\in \gamma(t_1) \cup \gamma(t_2))\ i, j \in \underline{2}.\ s_i = t_1^i \parallel t_2^i, t_i = t_i^1 \parallel t_i^2, i \in \underline{2}$

c) $\varepsilon = s \parallel t \Rightarrow s = \varepsilon = t$.

1.3 Partial Orders on SW

There are more natural partial orders on SW of which we shall see two in this section.

1.3.1 Smoother Than

The idea of one semiword, s, being smoother than another, t, i.e., $\leq_s$ is a refinement of $\leq_t$, can be captured formally as follows:

Definition 1.3.1 Let $s, t \in SW$. Then s is *smoother than* t (written $s \preceq t$) *iff* $A_s = A_t$ and $\leq_s \supseteq \leq_t$. □

Example:

$$a \rightarrow b \rightarrow c \quad \preceq \quad a \begin{array}{l} \nearrow b \\ \searrow c \end{array} \quad \preceq \quad \begin{array}{l} a \rightarrow b \\ \quad\;\; c \end{array} \quad \preceq \quad \begin{array}{c} a \\ b \\ c \end{array}$$

Both $=$ and $\subseteq$ are partial orders so evidently:

Corollary 1.3.2 $\preceq$ partial orders SW.

Corollary 1.3.3 If s is a subsemiword of u, t the complement semiword in u and s, t disjoint then $u \preceq s \parallel t$.

The truth of this is evident since s, t disjoint implies $A_s \cup A_t = A_u$, s subsemiword of u implies $\leq_s \subseteq \leq_u$ and t disjoint complement (sub)semiword of s in u implies $\leq_t \subseteq \leq_u$.

Looking at this corollary one might think that s being a subsemiword of u and t the complement implies $st \preceq u$, but this *is not* in general true as can be seen from the following example.

Example: Let $u = \; a \begin{array}{l} \nearrow b \searrow \\ \searrow c \nearrow \end{array} d$. Then $s = \; a \rightarrow b \rightarrow d$ is a subsemiword of u and $t = c$ the complement semiword. But $st \not\preceq u$ because $c \leq_u d$ and $c \not\leq_{st} d$.

Later in proposition 1.3.28 we will se a sufficient condition for $st \preceq u$.

Having defined $\preceq$ we are able to define the set of linearizations or the smoothing of a semiword s, written $\lambda(s)$.

Linearizations: λ

Definition 1.3.4 Define $\lambda : SW \rightarrow \mathcal{P}(W)$ by

$$s \mapsto \{t \in W \mid t \preceq s\}$$

□

Proposition 1.3.5 For all $s, t \in SW$ we have

a) $s \preceq t \Leftrightarrow \lambda(s) \subseteq \lambda(t)$ b) $\lambda(s) \neq \emptyset$

Before we proceed with the proof we need some small lemmas.

Lemma 1.3.6 $\forall s \in SW\ \forall a, b \in A_s.\ (a \not\leq_s b, b \not\leq_s a \Rightarrow \exists t \in SW.\ t \prec s, a \leq_t b)$

Proof The idea is to get a smoothing of $\leq_s$ by adding (a, b) to $\leq_s$ and take the transitive closure. Given $s \in SW$ and $a, b \in A_s$ such that $a \not\leq_s b, b \not\leq_s a$. Define t by $A_t := A_s$, $\leq_t := R^+$, where $R = \leq_s \cup Q$, $Q = \{(a, b)\}$. Clearly $\leq_s \subset \leq_t$ and thereby $t \prec s$, so the only problem is to see $t \in SW$.

$SW1$ holds for t since $A_t = A_s$. Because $\leq_s \subseteq \leq_t$ and $\leq_s$ is reflexive, $SW2$ holds for $\leq_t$. By construction $\leq_t$ is transitive.

Before considering the antisymmetry we prove:

$$\forall c, d \in A_t.\ c\ R^n\ d, c \not\leq_s d \Rightarrow c \leq_s a, b \leq_s d. \tag{1.1}$$

by induction on n.

$n = 1$: Because $c \not\leq_s d$ we must have $c\ Q\ d$. Then $a = c$, $d = b$ and the result follows by the reflexivity of $\leq_s$.

$n > 1$: Then $c\ R^{n-1}\ e, e\ R\ d$ for some $e \in A_t (= A_s)$. Two cases:

$c \leq_s e$: From $c \not\leq_s d$ and the transitivity of $\leq_s$ we conclude $e \not\leq_s d$. By hypothesis of induction $e \leq_s a, b \leq_s d$ and the result follows.

$c \not\leq_s e$: By hypothesis of induction $c \leq_s a, b \leq_s e$. If $e \leq_s d$ then transitivity gives $b \leq_s d$ and if $e \not\leq_s d$ the hypothesis yields $b \leq_s d$ directly.

Now to see:

$$c\ R^n\ d, d\ R^m\ c \Rightarrow c = d \tag{1.2}$$

assume on the contrary that there exists $c, d \in A_t$ such that $c\ R^n\ d, d\ R^m\ c, c \neq d$. Since $\leq_s$ is antisymmetric we have either $c \not\leq_s d$ or $d \not\leq_s c$. We investigate the different cases:

$c \not\leq_s d, d \not\leq_s c$: Since $c\ R^n\ d, d\ R^m\ c$ (1.1) gives $c \leq_s a, b \leq_s d, d \leq_s a, b \leq_s c$. From $b \leq_s c, c \leq_s a$ we get $b \leq_s a$ which is a contradiction to $b \not\leq_s a$.

$c \leq_s d, d \not\leq_s c$: Since $d\ R^n\ c$ we have $d \leq_s a, b \leq_s c$. Collecting these facts: $b \leq_s c \leq_s d \leq_s a$ and thereby $b \leq_s a$—again a contradiction.

$c \not\leq_s d, d \leq_s c$: Similar.

We have exhausted the cases and each time got a contradiction, so the assumption was wrong. From (1.2) the antisymmetry of $\leq_t$ then follows.

□

Since $s \notin W$ implies $\exists a, b \in A_s.\ a \not\leq_s b, b \not\leq_s a$ it follows that:

Corollary 1.3.7 $\forall s \in SW : s \notin W \Rightarrow \exists t \in SW.\ t \prec s.$

Lemma 1.3.8 For all $s \in SW$ we have: $\lambda(s) = \{t \in SW \mid t \preceq s, \nexists t'.\, t' \prec t\}$.

Proof
$\subseteq$: Let $t \in \lambda(s)$ be given. I.e., $t \preceq s, t \in W$.
We shall prove $\nexists t'.\, t' \prec t$. So assume on the contrary that there exists a $t' \prec t$ i.e., $t' \preceq t, t \neq t'$. Now $t' \prec t$ implies $\leq_{t'} \supset \leq_t$. But $\leq_{t'} \supset \leq_t$ means $\exists a, b \in A_{t'}.\, a \leq_{t'} b, a \not\leq_t b$. We cannot have $b \leq_t a$ since this, by $\leq_{t'} \supset \leq_t$, implies $b \leq_{t'} a$, a contradicting the antisymmetry of $\leq_{t'}$. So we have $a \not\leq_t b, b \not\leq_t a$—a contradiction to $t \in W$.

$\supseteq$: Let $t \in SW$ be given such that $t \preceq s$, $\nexists t'.\, t' \prec t$. We only need to show $t \in W$. Now assume on the contrary $t \notin W$. From corollary 1.3.7 we then have there exists a $t' \in SW.\, t' \prec t$—a contradiction. □

We are now ready to prove a part of the last proposition.

Proof (of proposition 1.3.5)
a) $\Rightarrow$: i.e., $s \preceq t \Rightarrow \lambda(s) \subseteq \lambda(t)$.
Let $r \in \lambda(s)$ be given. We shall prove $r \in \lambda(t)$. By definition of λ we have $r \preceq s$ and from the premise $s \preceq t$, so by the transitivity of $\preceq$ we have $r \preceq t$. By the previous lemma we have $r \in \lambda(s) \Rightarrow \nexists r'.\, r' \prec r$. Since $r \preceq t$ it then follows by the same lemma that $r \in \lambda(t)$.

b) $\lambda(s) \neq \emptyset$:
We prove $\forall s \in SW\ \exists t \in W.\, t \preceq s$ by induction on the number $|\delta(s)|$ from which the result follows. The basis must be with $|\delta(s)| = 1$ since $\forall s \in SW.\, s \in \delta(s)$. Furthermore notice that by the previous lemma we have $s \in W \Leftrightarrow |\delta(s)| = 1$.

$|\delta(s)| = 1$: Take t to be s. By reflexivity of $\preceq$ it follows that $t \preceq s$.

$|\delta(s)| > 1$: Then $s \notin W$ and by corollary 1.3.7 $\exists t' \in SW.\, t' \prec s$. Clearly $\delta(t') \subseteq \delta(s)$ and $s \notin \delta(t')$ wherefore $|\delta(t')| < |\delta(s)|$ and we can use the inductive hypothesis to obtain a $t \in W$ such that $t \preceq t'$. By the transitivity of $\preceq$ we get $t \preceq s$. □

Before proving the rest of the proposition we need one more little lemma.

Lemma 1.3.9 For all $s \in SW$ and $a, b \in A_s$ we have:

a) $a \leq_s b \Rightarrow \forall t \in \lambda(s).\, a \leq_t b$ b) $a \not\leq_s b \Rightarrow \exists t \in \lambda(s).\, b \leq_t a$

Proof
a) Immediate since $t \in \lambda(s) \Rightarrow t \preceq s \Rightarrow \leq_s \subseteq \leq_t$.

b) We look at two cases of s.

$s \in W$: Then $\leq_s$ satisfies the trichotomy law. Choose $t = s$.

$s \notin W$: There are two possibilities. Either $b \leq_s a$ or $b \not\leq_s a$. If $b \leq_s a$ the result follows from a) and $\lambda(s) \neq \emptyset$. If $b \not\leq_s a$ we conclude by lemma 1.3.6 that there exists a t' such that $t' \prec s$ and $b \leq_{t'} a$. From a) then $\forall t \in \lambda(t').\, b \leq_t a$. We have already proved $t' \preceq s \Rightarrow \lambda(t') \subseteq \lambda(s)$ so we are done.

□

Proof (of proposition 1.3.5.a) continued)
We shall prove $\lambda(s) \subseteq \lambda(t) \Rightarrow s \preceq t$ which is equivalent to $s \not\preceq t \Rightarrow \lambda(s) \not\subseteq \lambda(t)$. Assume $s \not\preceq t$. Two possibilities.

$A_s \neq A_t$: Then clearly $\lambda(s) \not\subseteq \lambda(t)$ since in general $\forall r \in \lambda(s).\, A_r = A_s$.

$A_s = A_t$: Then we must have $\leq_s \not\supseteq \leq_t$. That means there exists $a, b \in A_t$ such that $a \leq_t b$ but $a \not\leq_s b$. By b) of the previous lemma $a \not\leq_s b$ implies $\exists s' \in \lambda(s).\, b \leq_{s'} a$. We cannot have $s' \in \lambda(t)$. Suppose on the contrary we have $s' \in \lambda(t)$. From a) of the same lemma we have $a \leq_t b$ implies $\forall t' \in \lambda(t).\, a \leq_{t'} b$, hence also $a \leq_{s'} b$. By the antisymmetry of $\leq_{s'}$: $a \leq_{s'} b, b \leq_s a$ implies $a = b$. Then $a \not\leq_s b$ means $a \not\leq_s a$—a contradiction to the reflexivity of $\leq_s$.

□

From this proposition it follows that $s = t$ *iff* $\lambda(s) = \lambda(t)$ and since $\lambda(s)$ is a finite set of words a natural question is "Why not just consider finite sets of words instead of semiwords?". The main reason we are using semiwords, as opposed to finite sets of words over Δ, is the semiwords agreement with our intuition of concurrency—the partial order reflecting the dependencies between occurrences of actions and the lack of such a dependency the concurrency. Furthermore they have a very simple graphical representation which supports the intuition. Also the formal mathematical representation (canonic representation) is simple. A consequence is simplified definitions and proofs. Another technical reason is that not every set of words T (with same multiset) have an $s \in SW$ such that $\lambda(s) = T$. For a more detailed discussion of—and look at pos see [Pra86].

The next proposition is concerned with the $\preceq$-monotonicity of . and $\|$.

Proposition 1.3.10

a) $s \preceq t \Leftrightarrow sr \preceq tr \Leftrightarrow rs \preceq rt$ b) $s \preceq t \Leftrightarrow s \parallel r \preceq t \parallel r$.

Proof
a) We look at the different implications one by one.

$s \preceq t \Rightarrow sr \preceq tr$: Given $s \preceq t$. Shall prove $sr \preceq tr$ or equally $A_{sr} = A_{tr}, \leq_{tr} \subseteq \leq_{sr}$. Since $s \preceq t \Rightarrow A_s = A_t$ we see by the definition of concatenation that, in order to prove $A_{sr} = A_{tr}$, it is enough to prove $\{a^{i+\psi(t,a)} \mid a^i \in A_r\} = \{a^{i+\psi(s,a)} \mid a^i \in A_r\}$, but this follows directly from $A_s = A_t \Leftrightarrow \forall a \in \Delta.\, \psi(s,a) = \psi(t,a)$. To see $\leq_{tr} \subseteq \leq_{sr}$ we must prove $\forall a^i, b^j \in A_{tr} (= A_{sr}).\, a^i \leq_{tr} b^j \Rightarrow a^i \leq_{sr} b^j$. We look at the cases: $i \leq \psi(t,a), j \leq \psi(t,b)$: This implies by the definition of concatenation $a^i \leq_t b^j$ which by $s \preceq t$ implies $a^i \leq_s b^j$. Since $\psi(s,a) = \psi(t,a), \psi(t,b) = \psi(s,b)$ the result follows for this case. The remaining cases $(i \leq \psi(t,a), \psi(t,b) < j)$ and $(\psi(t,a) < i, \psi(t,b) < j)$ follows similarly.

$s \preceq t \Rightarrow rs \preceq rt$: Similar.

$sr \preceq tr \Rightarrow s \preceq t$: $A_s \subseteq A_t$: Assume on the contrary $A_s \not\subseteq A_t$. Then $\exists a^i \in A_s.\ a^i \notin A_t$. This implies $\psi(t,a) < i \leq \psi(s,a)$. Clearly $a^{\psi(r,a)+\psi(s,a)} = a^k \in A_{sr}$. Now $a^k \notin A_t$ because $\psi(t,a) < \psi(s,a) \leq k$. Also $a^k \notin \{a^{j+\psi(t,a)} \mid a^j \in A_r\}$.
If not, then $a^{k-\psi(t,a)} \in A_r$. This gives us $k - \psi(t,a) \leq \psi(r,a)$ or equivalently $\psi(r,a) + \psi(s,a) - \psi(t,a) \leq \psi(r,a)$ and thereby $\psi(s,a) \leq \psi(t,a)$ which contradicts $\psi(t,a) < \psi(s,a)$. So $a^k \notin A_{tr}$. But this contradicts $A_{sr} = A_{tr}$, so the assumption was wrong and we have $A_s \subseteq A_t$.
Similarly, we see $A_t \subseteq A_s$, wherefore $A_s = A_t$. This also implies $\psi(s,a) = \psi(t,a)$ for all $a^i \in A_s (= A_t)$, hence by the definition of concatenation and the fact that $\leq_{tr} \subseteq \leq_{sr}$, it follows $\leq_t \subseteq \leq_s$.

$rs \preceq rt \Rightarrow s \preceq t$: In general we have A_r and $\{a^{i+\psi(r,a)} \mid a^i \in A_u\}$ are disjoint, so $A_{rs} = A_{rt}$ implies $\{a^{i+\psi(r,a)} \mid a^i \in A_s\} = \{a^{i+\psi(r,a)} \mid a^i \in A_t\}$, hence $A_s = A_t$. Similar as above we see $\leq_t \subseteq \leq_s$.

b) Obvious, since in general the disjointness of s and t gives $A_{s\|t} = A_s \uplus A_t$, $\leq_{s\|t} = \leq_s \uplus \leq_t$. $\square$

From this proposition and the transitivity of $\preceq$ we immediately get:

Corollary 1.3.11 If $s_i \preceq t_i$ for $i \in \underline{2}$ then:

a) $s_1 s_2 \preceq t_1 t_2$ b) $s_1 \parallel s_2 \preceq t_1 \parallel t_2$.

The commutativity of $\parallel$ is used in seeing b).

Proposition 1.3.12 For semiwords s, t and u: $u \preceq st \Rightarrow \exists s' \preceq s, t' \preceq t.\ u = s't'$.

Proof Given $u \preceq st$. Define $s' := u|_{A_s}$.
s is a subsemiword of st wherefore A_s fulfills $SW1$. Since $u \preceq st \Rightarrow A_u = A_{st}$ and $A_s \subseteq A_{st}$ it follows that $A_s \subseteq A_u$. Hence s' is a subsemiword of u. Then we can define t' to be the complement semiword of s' w.r.t. u.

$u = s't'$: Since s' is a subsemiword of u and t the complement proposition 1.2.7.a) gives $A_u = A_{s't'}$ and b) – d) most of $\leq_u = \leq_{s't'}$. Only three cases remains to be proved and this in a situation with $a \in A_{s'}, b \in A_u \setminus A_{s'}$. $u \preceq st$ implies $A_u = A_{st}$ and by the definition of s' we have $A_{s'} = A_s$, so the situation can be read as $a \in A_{s'} = A_s, b \in A_{s't'} \setminus A_{s'} = A_{st} \setminus A_s$. As noticed by the definition of concatenation we then have

$$a \leq_{s't'} b \text{ and } a \leq_{st} b \tag{1.3}$$

$a \leq_{s't'} b \Rightarrow a \leq_u b$: From $u \preceq st$ we also have $\leq_{st} \subseteq \leq_u$, wherefore (1.3) gives $a \leq_u b$.

$b \leq_{s't'} a \Rightarrow b \leq_u a$: Trivially true because $b \not\leq_{s't'} a$. To see this assume otherwise $b \leq_{s't'} a$ and (1.3) together with the antisymmetry of $\leq_{s't'}$ would give $a = b$ which contradicts a and b belonging to disjoint sets.

$b \leq_u a \Rightarrow b \leq_{s't'} a$: We cannot have $b \leq_u a$ since (1.3) and the first case implies $a \leq_u b$ which then if $b \leq_u a$ would lead to a contradiction as in the last case.

□

Proposition 1.3.13

a) $u \in \lambda(st) \Leftrightarrow \exists s' \in \lambda(s)\ \exists t' \in \lambda(t).\ u = s't'$

b) $u \in \lambda(s \parallel t) \Leftrightarrow \exists s' \in \lambda(s)\ \exists t' \in \lambda(t).\ u \preceq s' \parallel t',\ \nexists u'.\ u' \prec u.$

Proof
a) $\Rightarrow$: By lemma 1.3.8 $u \in \lambda(st)$ implies $u \preceq st$, $\nexists u'.\ u' \prec u$. From $u \preceq st$ we see from the last proposition that $\exists s' \preceq s, \exists t' \preceq t.\ u = s't'$, so if we can prove $\nexists s''.\ s'' \prec s'$ and $\nexists t''.\ t'' \prec t'$ we are done since, then again by lemma 1.3.8 we have $s' \in \lambda(s)$ and $t' \in \lambda(s)$. To see $\nexists s''.\ s'' \prec s'$ assume on the contrary $\exists s''.\ s'' \prec s'$. Then by proposition 1.3.10 $s''t' \prec s't' = u$—a contradiction to $\nexists u'.\ u' \prec u$. Similarly, we prove $\nexists t''.\ t'' \prec t'$.
$\Leftarrow$: $s' \in \lambda(s), t' \in \lambda(s)$ implies $s' \preceq s$ and $t' \preceq t$ and by corollary 1.3.11 $u = s't' \preceq st$. So in order to have $u \in \lambda(st)$ we just need to prove $\nexists u'.\ u' \prec u$. Assume this is not the case, i.e., $\exists u'.\ u' \prec u = s't'$. By the last proposition we clearly see this must mean $\exists s'' \prec s'$ or $\exists t'' \prec t'$—a contradiction to $s' \in \lambda(s)$ or $t' \in \lambda(s)$.

b) $\Rightarrow$: $u \in \lambda(s \parallel t)$ implies $\nexists u'.\ u' \prec u$. Since $u \in \lambda(s \parallel t)$ means $u \preceq s \parallel t$, we have $A_u = A_s \uplus A_t, \leq_s \uplus \leq_t \subseteq \leq_u$. So $s' := u|_{A_s}$ and $t' := u|_{A_t}$ are indeed subsemiwords of u. At first we prove $s' \preceq s$ and $t' \preceq t$. To see $s' \preceq s$ notice $A_{s'} = A_u|_{A_s} = A_s$ and $\leq_{s\parallel t} \subseteq \leq_u$ implies $\leq_{s\parallel t}|_{{A_s}^2} = \leq_{s'}$. Since $\leq_s = \leq_{s\parallel t}|_{{A_s}^2}$ we have $\leq_s \subseteq \leq_{s'}$ and thereby $s' \preceq s$. $t' \preceq t$ is shown similarly.
Now to see $s' \in \lambda(s)$ we just need to prove $s' \in W$; i.e., $\forall a, t \in A_{s'}.\ a \leq_{s'} b \vee b \leq_{s'} a$. Let $a, b \in A_{s'} \subseteq A_{s'} \uplus A_{t'}$ be given. Since $u \in \lambda(s \parallel t)$ and thereby $u \in W$ we have $a \leq_u b \vee b \leq_u a$. W.l.o.g. assume $a \leq_u b$. From $a, b \in A_s$ we see $a \leq_u b$ implies $a \leq_u|_{{A_s}^2} b$ or what is the same $a \leq_{s'} b$. The proof of $t' \in W$ is done in the same way.

To complete this implication we finally have to show $u \preceq s' \parallel t'$ or what comes to the same: since $A_u = A_{s'} \uplus A_{t'} = A_s \cup A_t$ that $\leq_{s'} \cup \leq_{t'} \subseteq \leq_u$. But this follows evidently since $\leq_{s'} = \leq_u|_{{A_s}^2}$ and $\leq_{t'} = \leq_u|_{{A_t}^2}$.
$\Leftarrow$: Because $\nexists u'.\ u' \prec u$ is assumed we see from lemma 1.3.8 that all we have to show is that $u = s' \parallel t' \preceq s \parallel t$. Since $s' \in \lambda(s)$ and $t' \in \lambda(s)$ imply $s' \preceq s$ and $t' \preceq t$ we immediately get the result from corollary 1.3.11. □

Corollary 1.3.14

a) $\lambda(s)\lambda(t) = \lambda(st)$

b) $a.\lambda(t) = \lambda(a.t)$

$\preceq$-downwards closure: δ

In the following we are concerned with the full $\preceq$-downward closure. We will abbreviate $DC_{\preceq}(s)$ by $\delta(s)$. Notice that $\delta(s)$ is a finite set since A_s is finite and so only finitely many refinements of $\leq_s$ are possible. Also $\lambda(s) = W \cap \delta(s)$. Both δ and λ are extended to sets of semiwords in the natural way. E.g., if S is a set of semiwords then $\delta S = \bigcup_{s \in S} \delta(s)$.

Proposition 1.3.15

a) $s \in \delta(s)$

b) $\delta(\varepsilon) = \{\varepsilon\}$ and $\forall a \in \Delta.\, \delta(\underline{a}) = \{\underline{a}\}$

c) $\delta(st) = \delta(s)\delta(t)$

d) $\delta(s) \parallel \delta(t) \subseteq \delta(s \parallel t)$

Before giving the the proof we observe the following immediate consequence:

Corollary 1.3.16

a) $\delta(a.s) = a.\delta(s)$

b) $\delta(s \parallel t) = \delta(\delta(s) \parallel \delta(t))$

b) of corollary 1.3.16 is seen as follows: $\subseteq$ from d) implies $\delta(\delta(s) \| \delta(t)) \subseteq \delta\delta(s \| t) \subseteq \delta(s \| t)$ and $\supseteq$ by $s \parallel t \in \delta(s) \parallel \delta(t) \Rightarrow \delta(s \parallel t) \subseteq \delta(\delta(s) \parallel \delta(t))$.

Proof (of proposition 1.3.15)
a) By the reflexivity of $\preceq$.

b) Follows directly from a) and the fact that $A_\varepsilon = \emptyset$ ($A_{\underline{a}} = \{a^1\}$) allows no refinement of $\leq_\varepsilon = \emptyset$ ($\leq_{\underline{a}} = \{(a^1, a^1)\}$).

c) Clear from proposition 1.3.12 and corollary 1.3.11

d) $u \in \delta(s) \| \delta(t) \Rightarrow (\exists s', t'.\, s' \preceq s, t' \preceq t, u = s' \parallel t') \Rightarrow u = s' \parallel t' \preceq s \| t \Rightarrow u \in \delta(s \| t)$. □

$\preceq$-upwards closure: υ

Similar to the abbreviation of $DC_\preceq(s)$ by $\delta(s)$ we abbreviate $UC_\preceq(s)$ upwards closure of s w.r.t. $\preceq$ by $\upsilon(s)$ and extend υ to sets in the natural way.. We have already seen that $\delta(s)$ is a finite set. The same turns out to be true for $\upsilon(s)$ because A_s is finite and so only finitely many coarsenings of $\leq_s$ is possible. Whereas every po consistent refinement (i.e., it is reflexive, antisymmetric, transitive) of $\leq_s$ to $\leq_{s'}$ yields a semiword s', this is not the case for every po consistent coarsening. For example, if

$$s = (\{a^1, a^2\}, \{(a^1, a^2), (a^1, a^1), (a^2, a^2)\})$$

then the only possible po consistent coarsening of $\leq_s$ is

$$\{(a^1, a^1), (a^2, a^2)\}$$

which isn't a semiword (violates $SW2$).

Before we continue with properties of υ we prove:

Proposition 1.3.17 If s and t are disjoint we have

$$s \parallel t \preceq u \Rightarrow \exists s', t'.\, s \preceq s', t \preceq t' \text{ and } u = s' \parallel t'$$

Proof Given $s \parallel t \preceq u$. Define $s' := u|_{A_s}$ and $t' := u|_{A_t}$.

At first we show $u = s' \parallel t'$.
By definition $s \parallel t \preceq u$ means $A_{s\parallel t} = A_u$ and $\leq_{s\parallel t} \supseteq \leq_u$. From $A_{s\parallel t} = A_s \uplus A_t$ we see $A_u = A_s \uplus A_t$, so $A_u = A_u|_{A_s} \uplus A_u|_{A_t} = A_{s'} \uplus A_{t'}$. Hence $A_{s'}$, $A_{t'}$ fulfills $SW1$ and by proposition 1.1.4 s' and t' are subsemiwords of u. Clearly they are disjoint, so $s' \parallel t'$ well-defined. Since $A_{s'\parallel t'} = A_{s'} \uplus A_{t'}$ we have $A_{s'\parallel t'} = A_u$. $s \parallel t \preceq u$ implies $\leq_u \subseteq \leq_{s'\parallel t'} = \leq_s \uplus \leq_t = \leq_s|_{A_s{}^2} \uplus \leq_t|_{A_t{}^2}$ which on second thoughts is seen to imply $\leq_u = \leq_u|_{A_s{}^2} \uplus \leq_u|_{A_t{}^2}$. But $\leq_u|_{A_s{}^2} \uplus \leq_u|_{A_t{}^2} = \leq_{s'} \uplus \leq_{t'} = \leq_{s'\parallel t'}$, wherefore $\leq_u = \leq_{s'\parallel t'}$.

Secondly we prove $s \preceq s'$ and $t \preceq t'$.
To see $s \preceq s'$ at first notice $A_s = A_{s'}$ by construction, so the proof of $s \preceq s'$ reduces to $\leq_{s'} \subseteq \leq_s$. $s \parallel t \preceq u$ implies $\leq_u \subseteq \leq_{s\parallel t}$ which again implies $\leq_u|_{A_s{}^2} \subseteq \leq_{A_s{}^2}$. Since $\leq_{s'} = \leq_u|_{A_s{}^2}$ and $\leq_{s\parallel t}|_{A_s{}^2} = \leq_s$ we are done.
$t \preceq t'$ is seen in the same way. □

We are now ready to state and prove the following properties of υ.

Proposition 1.3.18

a) $s \in \upsilon(s)$

b) $\upsilon(\varepsilon) = \{\varepsilon\}$ and $\forall a \in \Delta.\, \upsilon(\underline{a}) = \{\underline{a}\}$

c) $\upsilon(s)\upsilon(t) \subseteq \upsilon(st)$

d) $\upsilon(s \parallel t) = \upsilon(s) \parallel \upsilon(t)$

Corollary 1.3.19

a) $\upsilon(st) = \upsilon(\upsilon(s)\upsilon(t))$

b) $\upsilon(a.s) = \upsilon(a.\upsilon(s))$

$\subseteq$ of a) is seen from $st \in \upsilon(s)\upsilon(t)$ and $\supseteq$ from c) of the proposition using $\upsilon(\upsilon(st)) = \upsilon(st)$.

Proof (of proposition 1.3.18)
a) Follows from the reflexivity of $\preceq$.

b) $A_\varepsilon = \emptyset$ allows no coarsening. No coarsening of $\{(a^1, a^1)\}$ is po consistent—fails the reflexivity.

c) $u \in \upsilon(s)\upsilon(t) \Rightarrow \exists s', t''.\, s \preceq s', t \preceq t', s't' \Rightarrow$ (corollary 1.3.11) $st \preceq s't' = u \Rightarrow u = s't' \in \upsilon(st)$.

d) $\subseteq$ follows from the last proposition and $\supseteq$ from corollary 1.3.11 □

Notice that in general $\delta(s \parallel t) \neq \delta(s) \parallel \delta(t)$ and $\upsilon(st) \neq \upsilon(s)\upsilon(t)$ when $s, t \neq \varepsilon$. This can be seen by $st \in \delta(s \parallel t)$ but $st \notin \delta(s) \parallel \delta(t)$ if $s, t \neq \varepsilon$ and if s and t are disjoint and different from the empty (semi)word. Under the same conditions $s \parallel t \in \upsilon(st)$ but $s \parallel t \notin \upsilon(s)\upsilon(t)$.

$\preceq$-convex closure: χ

The preceding up- and downwards closures w.r.t. $\preceq$, δ and υ ,were defined for single semiwords and extended to sets in the natural way. This cannot be done in the same way for the convex closure, χ, we are going to define now.

Definition 1.3.20 Let T be a (finite) set of semiwords. Then the convex closure of T written χT is defined by:

$$\chi T := \{s \in SW \mid \exists t, t' \in T.\, t \preceq s \preceq t'\}$$

$\square$

From the definition of χ it appears:

Corollary 1.3.21 $\chi T = \delta T \cap \upsilon T$.

As for δ and υ we derive some fundamental properties of χ.

Proposition 1.3.22 For $S, T \subseteq SW$ we have

a) $T \subseteq \chi T$

b) $\chi\{s\} = \{s\}$ for $s \in SW$

c) $\chi S \chi T \subseteq \chi(ST)$

d) $\chi S \parallel \chi T \subseteq \chi(S \parallel T)$

e) $\chi S \cup \chi T \subseteq \chi(S \cup T)$

Since $\chi\chi S = \chi S$ we can use a) to derive the opposite inclusions of c) – e) and so obtain:

Corollary 1.3.23

a) $\chi(ST) = \chi(\chi S \chi T)$

b) $\chi(S \parallel T) = \chi(\chi S \parallel \chi T)$

c) $\chi(S \cup T) = \chi(\chi S \cup \chi T)$

Proof (of proposition 1.3.22)
Now first notice that in general if $\square$—an operator between sets—can be considered as the natural extension of a operator, $\square$, between members of these sets, then:

(1.4) $$(A \cap B) \square (C \cap D) \subseteq (A \square C) \cap (B \square D).$$

a) – b) Immediate.

c) $\chi S \chi T$ = (corollary 1.3.21) $(\delta S \cap \upsilon S)(\delta T \cap \upsilon T) \subseteq$ (by (1.4)) $(\delta S \delta T) \cap (\upsilon S \upsilon T) =$ (proposition 1.3.15.c)) $\delta(ST) \cap (\upsilon S \upsilon T) \subseteq$ (proposition 1.3.18.c)) $\delta(ST) \cap \upsilon(ST) = \chi(ST)$.

d) $\chi S \parallel \chi T = (\delta S \cap \upsilon S) \parallel (\upsilon T \cap \upsilon T) \subseteq (\delta S \parallel \delta T) \cap (\upsilon S \parallel \upsilon T) \subseteq$ (proposition 1.3.15.d)) $\delta(S \parallel T) \cap (\upsilon S \parallel \upsilon T) =$ (proposition 1.3.18.d)) $\delta(S \parallel T) \cap \upsilon(S \parallel T) = \chi(S \parallel T)$.

e) $\chi S \cup \chi T = (\delta S \cap \upsilon S) \cup (\delta T \cap \upsilon T) \subseteq (\delta S \cup \delta T) \cap (\upsilon S \cup \upsilon T) = \delta(S \cup T) \cap \upsilon(S \cup T) = \chi(S \cup T)$.

$\square$

That the opposite inclusion in c) – e) of proposition 1.3.22 does not hold as can be seen by the following counter examples. Let $S = \{\varepsilon, a \rightarrow b \rightarrow c\}$ and $T = \left\{\varepsilon, \begin{smallmatrix} a \rightarrow b \\ c \end{smallmatrix}\right\}$. Then $S \cup T \subseteq ST, S \parallel T$ and $s = a \begin{smallmatrix} \nearrow b \\ \searrow c \end{smallmatrix} \in \chi(ST), \chi(S \parallel T), \chi(S \cup T)$. But $\chi S = S$ and $\chi T = T$, so $s \notin \chi S \chi T$, $\chi S \parallel \chi T$, $\chi S \cup \chi T$.

Now for a special version of corollary 1.3.23.

Proposition 1.3.24

a) $\chi(S \cup \{\varepsilon\}) = \chi S \cup \chi\{\varepsilon\} = \chi S \cup \{\varepsilon\}$ b) $\chi(\{s\}T) = \{s\}\chi T$

Proof

a) Evident since $t = \varepsilon$ is the only semiword for which $t \preceq \varepsilon$ or $\varepsilon \preceq t$.

b) $\supseteq$: $\{s\}\chi T = \chi\{s\}\chi T \subseteq$ (by proposition 1.3.22.c) $\chi(\{s\}T)$.

$\subseteq$: Let $u \in \chi(\{s\}T)$ be given. We shall prove $u \in \{s\}\chi T$.
$u \in \chi(\{s\}T)$ implies $\exists t, t' \in T.\ st \preceq u \preceq st'$. From proposition 1.3.12 and $u \preceq st'$ we see that there exists v, v' such that $u = vv', v \preceq s, v' \preceq t'$. Hence $st \preceq u$ means $st \preceq vv'$. Again by proposition 1.3.12 there must exists s_v and $s_{v'}$ such that $st = s_v s_{v'}$ and $s_v \preceq v, s_{v,} \preceq v'$. Now $s_v \preceq v \preceq s$ implies $A_{s_v} = A_s$. Clearly $A_{s_v} = A_s$ and $st = s_v s_{v'}$ implies $s = s_v, t = s_{v'}$. This again means $s \preceq v \preceq s, t \preceq v' \preceq t'$. $s \preceq v \preceq s$ gives $v = s$, so $u = sv'$ for a v: $t \preceq v' \preceq t'$ or equivalently $us = v'$ for a $v' \in \chi\{t, t'\} \subseteq \chi T$, so $u \in \{s\}\chi T$.
□

Corollary 1.3.25 $\chi a.T = a.\chi T$

1.3.2 Prefix of

We are now going to introduce another partial order on SW which shall be the generalization to SW of the well-known prefix partial order on Δ^* ($\cong W$). It will turn out that in general s is a prefix of st. As for Δ^* we have that s being a prefix of $t \in W$ implies that there exists a t' such that $st' = t$, but this is not in general true for arbitrary $t \in SW$!

Definition 1.3.26 s is a *prefix of* the semiword t (written $s \sqsubseteq t$) *iff* s is a subsemiword of t and $DC_{\leq_t}(A_s) \subseteq A_s$ i.e., $\forall a^i \in A_s(\subseteq A_t)\ \forall b^j \in A_t.\ b^j \leq_t a^i \Rightarrow b^j \in A_s$. □

Notice that for a subsemiword s of t $DC_{\leq_t}(A_s) \subseteq A_s$ *iff* $UC_{\leq_t}(A_t \setminus A_s) \subseteq A_t \setminus A_s$. This makes the connection with the definition of the prefix-po in [Pra86] for pomsets clear. We adopt his abbreviation $\pi(s)$ for $DC_{\sqsubseteq}(s)$ – the $\sqsubseteq$-downwards closure of s.

Example: If $s = \begin{smallmatrix} & & d & \searrow & \\ a & \searrow & & & e \\ & & c & \nearrow & \\ b & \nearrow & & & \end{smallmatrix}$ then:

$$\begin{smallmatrix} a \searrow \\ b \nearrow \end{smallmatrix} c \sqsubseteq s, \ \text{but } t = \begin{smallmatrix} a \searrow \\ b \nearrow \end{smallmatrix} c \rightarrow e \not\sqsubseteq s$$

because $e \in A_t$, $d \leq_s e$ and $d \notin A_t$.

Proposition 1.3.27 $\sqsubseteq$ is a po on SW.

Proof Antisymmetry: $s \sqsubseteq t, t \sqsubseteq s$ implies $A_s \subseteq A_t, A_t \subseteq A_s$. Hence $A_s = A_t$. Then of course $\leq_t|_{A_s{}^2} = \leq_t|_{A_t{}^2} = \leq_t$. Since $s \sqsubseteq t$ implies $\leq_s = \leq_t|_{A_s{}^2}$ we have $\leq_s = \leq_t$ and therefore $s = t$.

Reflexivity: s is a subsemiword of s, and the rest is immediate.

Transitivity: Given $s \sqsubseteq t \sqsubseteq u$. $s \sqsubseteq t$ implies s is a subsemiword of t, so $A_s = A_t|_{A_s{}^2} \subseteq A_t$ and $\leq_s = \leq_t|_{A_s{}^2}$. Similar we see $A_t \subseteq A_u$ and $\leq_t = \leq_u|_{A_t{}^2}$ from $t \sqsubseteq u$.
Because $A_s \subseteq A_t \subseteq A_u$ we have $A_u|_{A_s} = A_t|_{A_s} = A_s$. We also have $\leq_s = \leq_t|_{A_s{}^2} = (\leq_u |_{A_t{}^2})|_{A_s{}^2} =$ (since $A_s \subseteq A_t$) $\leq_u|_{A_s{}^2}$. Hence $s = u|_{A_s}$. s is a semiword wherefore A_s fulfills $SW1$, so by proposition 1.1.4 s is a subsemiword of u.
In order to have $s \sqsubseteq u$ it now remains to prove $DC_{\leq_u}(A_s) \subseteq A_s$. Let $b \in A_s, a \in A_u$ be given such that $a \leq_u b$. As $A_s \subseteq A_t$ we have $b \in A_t$. Since $DC_{\leq_u}(A_t) \subseteq A_t$ it follows that $a \in A_t$, so $(a, b) \in \leq_s|_{A_t{}^2} = \leq_t$. Hence $a \leq_t b$. Because $DC_{\leq_t}(A_s) \subseteq A_s$ it then also follows that $a \in A_s$. □

We now present the proposition promised at example on page 28 which gives a sufficient condition for $st \prec u$.

Proposition 1.3.28 If $s \sqsubseteq u$ and t is the complement semiword of s in u then $st \preceq u$

Proof By proposition 1.2.7 it is only necessary to prove

$$\forall a \in A_s, b \in A_u \setminus A_s.\, b \leq_u a \Rightarrow b \leq_{st} a$$

This follows directly from the fact that $b \not\leq_u a$ for all $a \in A_s, b \in A_u \setminus A_s$. Assume on the contrary $\exists a \in A_s, b \in A_u \setminus A_s.\, b \leq_u a$. Then $b \in DC_{\leq_u}(A_s)$ and $b \notin A_s$ which contradicts the definition of $s \sqsubseteq u$. □

From the proof it is seen that $\{s \mid s \sqsubseteq u\}$ exactly is the set, S, of subsemiwords of u for which $s \in S$ *iff* $st \preceq u$, where t is the complement semiword of s in u. So in this way we have found another characterization of $\sqsubseteq$ (this alternative characterization would not hold with a more general notion of subsemiwords). That $s \sqsubseteq u$ or rather $DC_{\leq_u}(A_s) \subseteq A_s$ is a necessary condition was illustrated in the example on page 28.

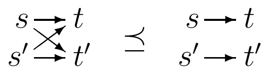
Proposition 1.3.29

a) $a.(s \parallel t) \preceq a.s \parallel t$

b) $s(t \parallel u) \preceq st \parallel u$

c) $(s \parallel s')(t \parallel t') \preceq st \parallel s't'$

provided the expressions are defined.

c) can be visualized as follows:  .

Proof

a) is a special case of b) which in turn is a special case of c).

b) corollary 1.2.6 and corollary 1.2.12 are easely seen to hold for prefixes too. I.e., $s \sqsubseteq st$, t complement semiword of s in st. etc. So $s \parallel s' \sqsubseteq st \parallel s't'$ and $t \parallel t'$ is the complement semiword of $s \parallel s'$ in $st \parallel s't'$. The result then follows from proposition 1.3.28 above. □

Proposition 1.3.30 Let $s, t, u \in SW$. Then:

$$a.u \preceq s \parallel t \ \mathit{iff} \ \begin{cases} \exists s'.\, a.s' \preceq s, u \preceq s' \parallel t, a^1 \notin A_t \\ \text{or} \\ \exists t'.\, a.t' \preceq t, u \preceq s \parallel t', a^1 \notin A_s \end{cases}$$

Proof

if: We only look at the case $\exists s'.\, a.s' \preceq s, u \preceq s' \parallel t, a^1 \notin A_t$ since the other is handled totally symmetric. By corollary 1.3.11 $a.s' \preceq s$ implies $a.s' \parallel t \preceq s \parallel t$ since $a^1 \notin A_t$ and s', t are disjoint. By the same corollary we obtain $a.u \preceq a.(s' \parallel t)$ from $u \preceq s' \parallel t$. Using proposition 1.3.29 we see $a.(s' \parallel t) \preceq a.s' \parallel t$, so collecting the facts we establish $a.u \preceq a.(s' \parallel t') \preceq a.s' \parallel t \preceq s \parallel t$ from which the result follows by the transitivity of $\preceq$.

only if: Consider $A_{\underline{a}} \cap A_s$ and $A_{\underline{a}} \cap A_t$. One of the intersections must be empty - otherwise s and t would not be disjoint which is assumed for $s \parallel t$ to make sense. W.l.o.g. assume the latter is the case i.e., $a^1 \notin A_t$.
Since $a.u \preceq s \parallel t$ implies $A_{a.u} = A_{s \parallel t} = A_s \cup A_t$ we get $a^1 \in A_s$ from $a^1 \notin A_t$ and $a^1 \in A_{a.u}$. So $\underline{a}$ is a subsemiword of s and furthermore $\underline{a} \sqsubseteq s$. Let s' be the complement semiword of $\underline{a}$ in s. By proposition 1.3.28 $a.s' = \underline{a}s' \preceq s$.
To see $u \preceq s' \parallel t$ we prove $A_u = A_{s'} \cup A_t$ and $\leq_{s' \parallel t} \subseteq \leq_u$.
$A_u = A_{s'} \cup A_t$: By corollary 1.2.6 u is the complement semiword of $\underline{a}$ in $\underline{a}u = a.u$, so by definition of complement semiword $A_u = \{b^{i-\psi(\underline{a},b)} \mid b^i \in A_{a.u} \setminus A_{\underline{a}}\} =$ (since $a.u \preceq s \parallel t$) $\{b^{i-\psi(\underline{a},b)} \mid b^i \in A_{s \parallel t} \setminus A_{\underline{a}} = (A_s \cup A_t) \setminus A_{\underline{a}}\} =$ (because $a^1 \notin A_t$) $\{b^{i-\psi(\underline{a},b)} \mid b^i \in (A_s \setminus A_{\underline{a}}) \cup A_t\} =$ (because $a^1 \notin A_t, b^i \in A_t \Rightarrow \psi(\underline{a}, b) = 0$) $\{b^{i-\psi(\underline{a},b)} \mid b^i \in A_s \setminus A_{\underline{a}}\} \cup A_t =$ (by definition of s' being the complement of $\underline{a}$ in s) $A_{s'} \cup A_t$.
$\leq_{s' \parallel t} \subseteq \leq_u$: Let $b^i, c^j \in A_{s' \parallel t}$ with $b^i \leq_{s' \parallel t} c^j$. Clearly only the following two cases can come into consideration.

$b^i, c^j \in A_{s'}$: Then $b^i \leq_{s' \parallel t} c^j$ implies $b^i \leq_{s'} c^j$. Now $b^i \leq_{s'} c^j \Rightarrow$ (again by definition of s' being the complement of $\underline{a}$ in s) $b^{i+\psi(\underline{a},b)} \leq_s|_{(A_s \setminus A_{\underline{a}})^2} c^{j+\psi(\underline{a},c)} \Rightarrow b^{i+\psi(\underline{a},b)} \leq_s c^{j+\psi(\underline{a},c)} \Rightarrow$ (by definition of $\parallel$) $b^{i+\psi(\underline{a},b)} \leq_{s \parallel t} c^{j+\psi(\underline{a},c)} \Rightarrow$ (from $a.u \preceq s \parallel t$) $b^{i+\psi(\underline{a},b)} \leq_{a.u} c^{j+\psi(\underline{a},c)} \Rightarrow$ (by definition of concatenation and $a.$) $b^i \leq_u c^j$.

$b^i, c^j \in A_t$: Then $b^i \leq_{s' \parallel t} c^j$ implies $b^i \leq_t c^j$ from which we get $b^i \leq_{s \parallel t} c^j$. Using $a.u \preceq s \parallel t$ we then see $b^i \leq_{a.u} c^j$. As earlier $a^1 \notin A_t, b^i, c^j \in A_t \Rightarrow \psi(\underline{a}, b) = 0 = \psi(\underline{a}, c)$, so from $b^i \leq_{a.u} c^j$ and the definition of concatenation we get $b^i = b^{i-\psi(\underline{a},b)} \leq_u c^{j-\psi(\underline{a},c)} = c^j$.

$\square$

This proposition can be specialized to W.

Proposition 1.3.31 Let $s, t, u \in W$. Then

$$a.u \preceq s \parallel t \text{ iff } \begin{cases} \exists s' \in W.\, a.s' = s, u \preceq s' \parallel t, a^1 \notin A_t \\ \text{or} \\ \exists t' \in W.\, a.t' = t, u \preceq s \parallel t', a^1 \notin A_s \end{cases}$$

Proof

if: Immediate from the previous proposition since $a.s' = s$ implies $a.s' \preceq s$, so by the previous proposition $a.u \preceq s \parallel t$.

only if: By the same proposition $a.u \preceq s \parallel t$ gives $\exists s'.\, a.s' \preceq s, u \preceq s' \parallel t, a^1 \notin A_t$ or $\exists t'.\, a.t' \preceq t, u \preceq s \parallel t', a^1 \notin A_s$. The result then follows since $s, t \in W$ and $a.s' \preceq s, a.t' \preceq t$ implies $s', t' \in W$. $\square$

Proposition 1.3.32

a) $s \sqsubseteq t \Leftrightarrow us \sqsubseteq ut$

b) $s \sqsubseteq t \Rightarrow s \sqsubseteq tu$

c) $s \sqsubseteq t \Leftrightarrow s \parallel u \sqsubseteq t \parallel u$

Proof

a) Each implication is proven separately.

$s \sqsubseteq t \Rightarrow us \sqsubseteq ut$: Given $s \sqsubseteq t$. We shall prove that us is a subsemiword of ut and that the $\leq_{ut}$-downwards closure of A_{us} is contained in A_{us}.
Since $s \sqsubseteq t$ implies $s = t|_{A_s}$ it follows that us is a subsemiword of ut if we can prove that in general:

(1.5) $$(ut)|_{A_{us}} = u(t|_{A_s}) \; (= us)$$

At first observe that since

$$\begin{aligned} A_{ut} &= A_u \uplus \{a^{i+\psi(u,a)} \mid a^i \in A_t\} \text{ and} \\ A_{us} &= A_u \uplus \{a^{i+\psi(u,a)} \mid a^i \in A_s\} \end{aligned}$$

we have:

$$\begin{aligned} A_{ut}|_{A_{us}} &= A_u \uplus \{a^{i+\psi(u,a)} \mid a^i \in A_t\}|_{\{a^{i+\psi(u,a)} \mid a^i \in A_s\}} \\ &= A_u \uplus \{a^{i+\psi(u,a)} \mid a^i \in A_t|_{A_s}\} \\ &= A_u \uplus \{a^{i+\psi(u,a)} \mid a^i \in A_{t|_{A_s}}\} \end{aligned}$$

It is now evident that $\leq_{(ut)|_{A_{us}}} = \leq_{u(t|_{A_s})}$ by looking at the definition of concatenation thereby establishing (1.5).
It remains to prove $DC_{\leq_{ut}}(A_{us}) \subseteq A_{us}$. So let $a^i \in A_{us}$ and $b^j \in A_{ut}$ with $b^j \leq_t a^i$ be given. If $b^j \in A_u$ then clearly $b^j \in A_{us}$. So assume $b^j \notin A_u$, that is $\psi(u, b) < j$. Then

$b^j \leq_{ut} a^i$ implies $b^{j-\psi(u,b)} \leq_t a^{i-\psi(u,a)}$. From this and $s \sqsubseteq t$ it follows that $b^{j-\psi(u,b)} \in A_s$. By definition of concatenation then $b^j = b^{(j-\psi(u,b))+\psi(u,b)} \in A_{us}$.

$us \sqsubseteq ut \Rightarrow s \sqsubseteq t$: $us \sqsubseteq ut$ implies $us = ut|_{A_u}$ and from (1.5) $(ut)|_{A_{us}} = u(t|_{A_s})$ so we can conclude $s = t|_{A_s}$ a subsemiword of t.
Now let $a^i \in A_s, b^j \in A_t$ be given such that $b^j \leq_t a^i$. Define $k := i + \psi(u,a)$ and $l := j + \psi(u,b)$. Since $a^i \in A_s, b^j \in A_t \Rightarrow i,j \geq 1$ we have $\psi(u,b) < l$ and $\psi(u,a) < k$. Clearly $b^j \leq_t a^i$ is the same as $b^{l-\psi(u,b)} \leq_t a^{k-\psi(u,a)}$ so by the definition of concatenation we have $b^l \leq_{ut} a^k$. Since $a^i \in A_s \Rightarrow a^k \in A_{us}$ we have from $us \sqsubseteq ut$ and $b^l \leq_{ut} a^k$ that $b^l \in A_{us}$. Since $\psi(u,b) < l$ it follows that $b^{l-\psi(u,u)} \in A_s$ which by the definition of l means $b^j \in A_s$.

b) $s \sqsubseteq t \Rightarrow s \sqsubseteq tu$: Given $s \sqsubseteq t$. We shall prove that s is subsemiword of tu and $DC_{\leq_{tu}}(A_s) \subseteq A_s$.
Now $s \sqsubseteq t$ implies $s = t|_{A_s}$ which again implies $A_s \subseteq A_t$. Hence $A_s \subseteq A_t \cup A_{tu} \setminus A_t = A_{tu}$ and since s is a semiword, A_s fulfills $SW1$, wherefore s is a subsemiword of tu (by proposition 1.1.4).
To prove $DC_{\leq_{tu}}(A_s) \subseteq A_s$ let $a \in A_s$ and $b \in A_{tu}$ be given such that $b \leq_{tu} a$.
b cannot be in $A_{tu} \setminus A_t$. If it was $a \in A_s \subseteq A_t$ would imply $a \leq_{tu} b$ as noticed by the definition of concatenation. By the antisymmetry of $\leq_{tu}$ and we would then get $a = b$—a contradiction to a, b belonging to disjoint sets.
So $b \in A_t$. By definition $b \leq_{tu} a$ and $s \in A_s \subseteq A_t$ then implies $b \leq_t a$. $b \in A_s$ then follows from $s \sqsubseteq t$.

c) At first notice $s \sqsubseteq t$ implies $A_s \subseteq A_t$, wherefore t disjoint from u implies s disjoint from u—so well-defined under the proviso. The rest follows directly from $A_{s\|t} = A_s \uplus A_t$. □

Corollary 1.3.33 If $s_i \sqsubseteq t_i$ for $i \in \underline{2}$ then $s_1 \| s_2 \sqsubseteq t_1 \| t_2$ provided t_1 and t_2 are disjoint.

Proposition 1.3.34

a) $u \sqsubseteq st \Rightarrow \begin{cases} u \sqsubseteq s \\ \text{or} \\ \exists t' \sqsubseteq t.\, u = st' \end{cases}$ b) $u \sqsubseteq s \| t \Rightarrow \exists s' \sqsubseteq s, t' \sqsubseteq t.\, u = s' \| t'$

Proof
a) $u \sqsubseteq st \Rightarrow u \sqsubseteq s$ or $\exists t' \sqsubseteq t.\, u = st'$:
Given $u \sqsubseteq st$. Then $u = st|_{A_u}$ and $DC_{\leq_{st}}(A_u) \subseteq A_u$. Since $u = st|_{A_u}$ implies $A_u \subseteq A_{st} = A_s \uplus (A_{st} \setminus A_s)$ we have two principal cases:

$A_u \subseteq A_s$: Claim: then $u \sqsubseteq s$. Clearly $st|_{A_u} = s|_{A_u}$ wherefore u is a subsemiword of s. To see $DC_{\leq_s}(A_u) \subseteq A_u$ let a $a^i \in A_u$ and $b^j \in A_s$ be given such that $b^j \leq_s a^i$. Since $a^i \in A_u \subseteq A_s$, $b^j \in A_s$ implies $i \leq \psi(s,a), j \leq \psi(s,b)$ we have from $b^j \leq_s a^i$ that $b^j \leq_{st} a^i$. Then $a^i \in A_u$ and from $DC_{\leq_{st}}(A_u) \subseteq A_u$ it follows that $b^j \in A_u$ and we are done for this case.

$A_u \not\subseteq A_s$ but $A_u \subseteq A_{st} \setminus A_s$: At first we prove: $A_s \subseteq A_u (\subseteq A_{st})$. Let $a \in A_s$ be given. $A_u \cap (A_{st} \setminus A_s)$ cannot be empty since this would imply $A_u \subseteq A_s$ which we assume is not the case, so there exists a $b \in A_u \cap (A_{st} \setminus A_s)$. As noticed at the definition of concatenation

$a \in A_s, b \in A_{st} \setminus A_s$ implies $a \leq_{st} b$. Since also $b \in A_u$ we have from $DC_{\leq_{st}}(A_u) \subseteq A_u$ that $a \in A_u$.

Because $A_s \subseteq A_u$ and s, u are semiwords, it follows that s is a subsemiword of u, and so $s = u|_{A_s}$. Then we can define t' to be the complement semiword of s w.r.t. u. Recall that this means:

$$A_{t'} := \{a^{i-\psi(s,a)} \mid a^i \in A_u \setminus A_s\}$$
$$\forall a^i, b^i \in A_{t'}.\ a^i \leq_{t'} b^i \textit{ iff } a^{i+\psi(s,a)} \leq_u|_{(A_u \setminus A_s)^2}\ b^{i+\psi(s,b)}$$

Notice $a^k \in A_{t'}$ *iff* $a^{k+\psi(s,a)} \in A_u \setminus A_s$. By the definition of concatenation it follows that $A_{st'} = A_s \cup \{a^{k+\psi(s,a)} \mid a^k \in A_{t'}\} = A_s \cup \{a^{k+\psi(s,a)} \mid a^{k+\psi(s,a)} \in A_u \setminus A_s\} = A_s \cup \{a^i \mid a^i \in A_u \setminus A_s\} = A_s \cup (A_u \setminus A_s) = A_u$—the last equation is a consequence of s being a subsemiword of u.

We now want to prove $\leq_u = \leq_{st'}$, i.e., $\forall a^i, b^j \in A_u\ (= A_{st'})$. $a^i \leq_u b^j \Leftrightarrow a^i \leq_{st'} b^j$.

$\Rightarrow$: Given $a^i, b^j \in A_u$ such that $a^i \leq_u b^j$.
Since s is a subsemiword of u we can compare i and j with $\psi(s,a)$ and $\psi(s,b)$.

$i \leq \psi(s,a), j \leq \psi(s,b)$: Then $a^i, b^j \in A_s$. Hence $a^i \leq_u|_{A_s{}^2}\ b^j$ and by definition of s we have $a^i \leq_s b^j$. From the definition of concatenation we see that this implies $a^i \leq_{st'} b^j$.

$i \leq \psi(s,a), \psi(s,b) < j$: Follows directly by the definition of concatenation.

$\psi(s,a) < i, \psi(s,b) < j$: Then $a^i, b^j \notin A_s$ and so $a^i, b^j \in A_u \setminus A_s$. From $a^i \leq_u b^j$ we then conclude $a^i \leq_u|_{(A_u \setminus A_s)^2}\ b^j$ which by the definition of $\leq_{t'}$ implies $a^{i-\psi(s,a)} \leq_{t'} b^{j-\psi(s,b)}$. By the definition of concatenation we now get $a^i \leq_{st'} b^j$.

$\psi(s,a) < i, j \leq \psi(s,b)$: Then $a^i \in A_u \setminus A_s$ and $b^j \in A_s$.
Now $u \sqsubseteq st$ implies $A_u \subseteq A_s \cup (A_{st} \setminus A_s)$ and since $A_s \subseteq A_u$ it follows that $a^i \in A_u \setminus A_s$ implies $a^i \in A_{st} \setminus A_s$. From $u \sqsubseteq st$ we also see $a^i \leq_u b^j$ only if $a^i \leq_{st} b^j$. On the other hand we noticed at the definition of concatenation that $b^j \in A_s$, $a^i \in A_{st} \setminus A_s$ implies $b^j \leq_{st} a^i$. Since $\leq_{st}$ is antisymmetric we must have $a^i = b^j$—a contradiction to $a^i \in A_u \setminus A_s$ and $b^j \in A_s$, so we can rule out this case.

$\Leftarrow$: Given $a^i, b^j \in A_{st'} (= A_u)$ such that $a^i \leq_{st'} b^j$.
By the definition of concatenation one of the following cases must hold.

$i \leq \psi(s,a), j \leq \psi(s,b), a^i \leq_s b^j$: Since s is a subsemiword of u this implies $a^i \leq_u b^j$.

$i \leq \psi(s,a), \psi(s,b) < j$: Then $a^i \in A_s \subseteq A_u, b^j \in A_u \setminus A_s$. Similar as above we see $b^j \in A_u \setminus A_s$ implies $b^j \in A_{st} \setminus A_s$, wherefore $a^i \leq_{st} b^j$. Since $a^i, b^j \in A_u \subseteq A_{st}$ we then also have $a^i \leq_{st}|_{A_u{}^2}\ b^j$. Because $u \sqsubseteq st$ implies $\leq_u = \leq_{st}|_{A_u{}^2}$ this means $a^i \leq_u b^j$.

$\psi(s,a) < i, \psi(s,b) < j, a^{i-\psi(s,a)} \leq_{t'} b^{j-\psi(s,b)}$: By definition of t' this implies:

$$a^{(i-\psi(s,a))+\psi(s,a)} \leq_u|_{(A_u \setminus A_s)^2}\ b^{(j-\psi(s,b))+\psi(s,b)}$$

or what is the same:

$$a^i \leq_u|_{(A_u \setminus A_s)^2}\ b^j$$

Obviously $\leq_u|_{(A_u \setminus A_s)^2} \subseteq \leq_u$ only if $a^i \leq_u b^j$, so we have now proved: $u = st'$ for the defined t'. Then $u \sqsubseteq st$ reads $st' \sqsubseteq st$ which by the last proposition implies $t' \sqsubseteq t$.

b) $u \sqsubseteq s \parallel t \Rightarrow \exists s' \sqsubseteq s, t' \sqsubseteq t.\ u = s' \parallel t'$:
$u \sqsubseteq s \parallel t$ implies $u = (s \parallel t)|_{A_u}$ which—because s and t are disjoint—equals $s|_{A_u} \parallel t|_{A_u}$. Let $s' = s|_{A_u}, t' = t|_{A_u}$. We then have $u = s' \parallel t'$ and $s' \parallel t' \sqsubseteq s \parallel t$. Since $A_{s'} \subseteq A_s, A_{t'} \subseteq A_t$ and s, t are disjoint, it is trivial to see that $s' \sqsubseteq s$ and $t' \sqsubseteq t$. □

Proposition 1.3.35

a) $\varepsilon, s \in \pi(s)$

b) $\pi(\varepsilon) = \{\varepsilon\}, \forall a \in \Delta.\ \pi(\underline{a}) = \{\varepsilon, \underline{a}\}$

c) $\pi(st) = \pi(s) \cup \{s\}\pi(t)$

d) $\pi(s \parallel t) = \pi(s) \parallel \pi(t)$

Proof
a) Clearly, ε, s are subsemiwords of s and $DC_{\leq_s}(A_\varepsilon) = DC_{\leq_s}(\emptyset) = \emptyset = A_\varepsilon$ so as $DC_{\leq_s}(A_s) \subseteq A_s$.

b) $\pi(\varepsilon) = \{\varepsilon\}$ is evident, and from a) we have $\{\varepsilon, \underline{a}\} \subseteq \pi(\underline{a})$. Since $\varepsilon, \underline{a}$ are the only possible subsemiwords of $\underline{a}$ it follows that $\pi(\underline{a}) \subseteq \{\varepsilon, \underline{a}\}$.

c) $\subseteq$ follows from a) of the last proposition. $\{s\}\pi(t) \subseteq \pi(st)$ follows from proposition 1.3.32.a) and $\pi(s) \subseteq \pi(st)$ from b) of the same.

d) $\subseteq$ follows from b) of the last proposition and $\supseteq$ from the last corollary. □

From c) and $\pi(\varepsilon) = \{\varepsilon\}$ we immediately get the corollary:

Corollary 1.3.36 $\pi(a.s) = a.\pi(s) \cup \{\varepsilon\}$

The next lemma concerned with pos will be used intensively in the proof of proposition 1.3.38 below.

Lemma 1.3.37 Let B be a subset of A and $\leq$ a po on A such that $DC_{\leq}(B) \subseteq B$. Furthermore let Q be a relation such that *either*

a) Q is antisymmetric, transitive and $\leq|_{B^2} \subseteq Q \subseteq B^2$

or

b) $Q \subseteq A \times (A \setminus B)$

Define R to be $\leq \cup\, Q$. Then R^+ is a po on A and $DC_{R^+}(B) \subseteq B$.

Proof Notice that no matter whether a) or b) holds Q is not defined on $(A \setminus B) \times A$. Then we can prove:

(1.6) $$b \in A \setminus B, a \in B \Rightarrow \neg(b\ R^n\ a)$$

by induction on n.

$n = 1$: Assume on the contrary $b \in A \setminus B, a \in B$ and $b\ R\ a$. Since Q is not defined on $(A \setminus B) \times A$ we see that $b\ R\ a$ implies $b \leq a$—a contradiction to $DC_{\leq}(B) \subseteq B$.

$n > 1$: Again suppose on the contrary $b \in A \setminus B, a \in B$ and $b\ R^n\ a$. Then $b\ R^{n-1}\ c$ and $c\ R\ a$ for some $c \in A$. Two cases:

$c \in B$: By hypothesis of induction $\neg(b\ R^{n-1}\ c)$—a contradiction to $b\ R^{n-1}\ c$.

$c \notin B$: Then $c \in A \setminus B$ and by hypothesis $\neg(c\ R\ a)$—a contradiction.

$DC_{R^+}(B) \subseteq B$ now directly follows from (1.6).

Since $\leq$ is reflexive on A^2 and $\leq \subseteq R^+$ this must be the case for R^+ too. By definition R^+ is transitive. We look at a) and b) separately when proving the antisymmetry of R^+.

a) Assume $\leq|_{B^2} \subseteq Q \subseteq B^2$ and Q transitive, antisymmetric. At first we prove:

(1.7) $$a, b \in B, a\ R^n\ b \Rightarrow a\ Q\ b$$

$n = 1$: Follows directly from $\leq|_{B^2} \subseteq Q$.

$n > 1$: Then $a\ R^{n-1}\ c, c\ R\ b$ for some $c \in A$. We must have $c \in B$. Otherwise we would have $c \in A \setminus B$ and from (1.6): $\neg(c\ R\ b)$—a contradiction. So $c \in B$. Then by hypothesis $a\ Q\ c, c\ Q\ b$ and from the transitivity of Q: $a\ Q\ b$.

Next we prove:

(1.8) $$a, b \in A \setminus B, a\ R^n\ b \Rightarrow a \leq b$$

$n = 1$: We must have $a \leq b$ since Q is not defined on $(A \setminus B)^2$.

$n > 1$: Then $a\ R^{n-1}\ c, c\ R\ b$ for some $c \in A$. By (1.6) we cannot have $c \in B$, so $c \in A \setminus B$. By hypothesis of induction and the transitivity of $\leq$ we get $a \leq b$.

From (1.6) – (1.8) and the antisymmetry of Q and $\leq$ we get:

(1.9) $$\forall a, b \in A.\ a\ R^n\ b, b\ R^m\ a \Rightarrow a = b$$

and thereby also the antisymmetry of R^+.

b) Assume $Q \subseteq A \times (A \setminus B)$. At first we prove:

(1.10) $$a, b \in B, a\ R^n\ b \Rightarrow a \leq b$$

$n = 1$: Follows directly since Q is not defined on B^2.

$n > 1$: Then $a\ R^{n-1}\ c, c\ R\ b$. By (1.6) we must have $c \in B$. (1.10) then follows by hypothesis and transitivity of $\leq$.

Similar we prove:

(1.11) $$a, b \in A \setminus B, a\ R^n\ b \Rightarrow a \leq b$$

From (1.6), (1.10), (1.11) and the antisymmetry of $\leq$ we get (1.9). □

Notice that this lemma (with the b) proviso) also could have been used to prove R^+ in lemma 1.3.6 to be a po on A_s by letting $B = DC_{\leq_s}(\{a\})$.

The next proposition says that π distributes over δ and λ and "partly" over υ.

Proposition 1.3.38

a) $\pi\delta(s) = \delta\pi(s)$ b) $\pi\lambda(s) = \lambda\pi(s)$

c) $\pi\upsilon(s) \supseteq \upsilon\pi(s)$

Proof
a) $\pi\delta(s) \subseteq \delta\pi(s)$: Let $t \in \pi\delta(s)$ be given. Then there exists a $t' \in \delta(s)$ such that $t \sqsubseteq t'$. $t' \in \delta(s)$ implies $t' \preceq s$. Consider u defined by $u := s|_{A_t}$. We shall prove that u is a semiword and $t \preceq u \sqsubseteq s$.

Since $t \sqsubseteq t' \preceq s$ implies $A_t \subseteq A_{t'} = A_s$, u must be a subsemiword of s with $A_u = A_t$.
In order to prove $u \sqsubseteq s$ we then just need to prove $DC_{\leq_s}(A_u) \subseteq A_u$ or what is the same $DC_{\leq_s}(A_t) \subseteq A_t$. Let $a \in A_t$ and $b \in A_s$ be given such that $b \leq_s a$. We shall prove $b \in A_t$. Since $t' \preceq s$ implies $\leq_s \subseteq \leq_{t'}$ we have $b \leq_{t'} a$. Because $t \sqsubseteq t'$ implies $DC_{\leq_{t'}}(A_t) \subseteq A_t$ we get $b \in A_t$.
Next we prove $t \preceq u$. Since $A_t = A_u$ we just need to prove $\leq_u \subseteq \leq_t$. Because $t' \preceq s$ implies $\leq_s \subseteq \leq_{t'}$ we get from $A_t \subseteq A_{t'} = A_s$ that $\leq_u = \leq_s|_{A_t^2} \subseteq \leq_{t'}|_{A_t^2}$. Since $t \sqsubseteq t'$ gives $\leq_t = \leq_{t'}|_{A_t^2}$ we are finished.

$\delta\pi(s) \subseteq \pi\delta(s)$: Let $t \in \delta\pi(s)$ be given. I.e., there exists a t' such that $t \preceq t' \sqsubseteq s$. We shall find a semiword u such that $t \sqsubseteq u \preceq s$.
Define u by $A_u := A_s$ and $\leq_u := R^+$, where $R = (\leq_s \cup \leq_t)$.
We first want to examine if u is a semiword. Since $A_u = A_s$ it fulfills $SW1$, and because $\leq_s$ fulfills $SW2$, it follows that $\leq_u$ does so too provided $\leq_u$ is a po.. Now $t' \sqsubseteq s$ implies $\leq_{t'} = \leq_s|_{A_{t'}^2}$ and $t \preceq t'$ implies $A_t = A_{t'}, \leq_{t'} \subseteq \leq_t$, so $\leq_s|_{A_t^2} \subseteq \leq_t$ and we see that a) of lemma 1.3.37 is satisfied. Also $DC_{\leq_s}(A_t) \subseteq A_t$ because $DC_{\leq_s}(A_{t'}) \subseteq A_{t'}$ and $A_t = A_{t'}$. From the lemma we can then conclude u is a semiword and $DC_{\leq_u}(A_t) \subseteq A_t$.

Now clearly $A_u = A_s$ and $\leq_s \subseteq \leq_u$, so $u \preceq s$.

To see $t \sqsubseteq u$ notice that $A_t = A_{t'} = A_s|_{A_{t'}} = A_u|_{A_{t'}} = A_u|_{A_t}$ and $\leq_t \subseteq \leq_u|_{A_t^2}$ by definition of $\leq_u$.
$\leq_u|_{A_t^2} \subseteq \leq_t$ follows from (1.7) of lemma 1.3.37. So t is a subsemiword of u and we already know $DC_{\leq_u}(A_t) \subseteq A_t$.

b) $\pi\lambda(s) \subseteq \lambda\pi(s)$: Follows exactly as $\subseteq$ of a), just notice that for the given t no t' exists such that $t' \prec t$.

$\lambda\pi(s) \subseteq \pi\lambda(s)$: Here we cannot take over the corresponding proof of a) directly, since the semiword u constructed there not necessarily belongs to $\lambda(s)$. For the u of a) we know that $t \sqsubseteq u \preceq s$. The idea is now to choose a $u' \in \lambda(u) \subseteq \lambda(s)$ and prove $t \sqsubseteq u' \preceq s$. But we have to be careful in choosing u'—not every u' of $\lambda(u)$ will do. On the way to find u' we define a $v \preceq u$ which will ensure that every $u' \in \lambda(v)$ will have t as prefix. Let $Q = \{(a, b) \mid a \in A_t, b \in A_u \setminus A_t\}$, $R = \leq_u \cup Q$, and define v by $A_v := A_u, \leq_v := R^+$. In this way every smoothing of v will have t as prefix.

Of course we shall at first prove that v is indeed a semiword.
Notice that $\leq_u$ and Q are contained in $R \subseteq \leq_v$. Clearly $SW1$ and $SW2$ holds for v because $u \in SW$ and $A_v = A_u, \leq_u \subseteq \leq_v$. Since $t \sqsubseteq u$ we have $DC_{\leq_u}(A_t) \subseteq A_t$ and by construction Q satisfies b) of lemma 1.3.37 (with $A = A_u$, $\leq = \leq_u$ and $B = A_t$). Hence we conclude that v is a semiword.

Clearly $v \preceq u \preceq s$. By proposition 1.3.5 $\lambda(v) \neq \emptyset$, so chose a $u' \in \lambda(v)$. Then $u' \preceq s$.

To see $t \sqsubseteq u'$ notice that $A_{u'} = A_u$, hence $A_t = A_u|_{A_t}$. Clearly $\leq_u|_{A_t{}^2} = \leq_{u'}|_{A_t{}^2}$ since no more refinements of $\leq_u|_{A_t{}^2}$ were possible, because $\leq_t = \leq_u|_{A_t{}^2}$ and $t \in W$. That means $\leq_t = \leq_{u'}|_{A_t{}^2}$ wherefore t is a subsemiword of u', so we just need to prove $DC_{\leq_{u'}}(A_t) \subseteq A_t$. Suppose this is not the case. Then there is some $\in A_u \setminus A_t$, $b \in A_t$ such that $a \leq_{u'} b$. Now $R^+ \subseteq \leq_v$ implies by lemma 1.3.9 that every v' of $\lambda(v)$ has $R^+ \subseteq \leq_{v'}$. Especially we have $R^+ \subseteq \leq_{u'}$. Hence also $Q \subseteq \leq_{u'}$. Now $a \in A_t, b \in A_u \setminus A_t$ implies $a \; Q \; b$ wherefore $a \leq_{u'} b$. By antisymmetry of $\leq_{u'}$: $a = b$—which contradicts that a, b belongs to disjoint sets.

c) $\upsilon\pi(s) \subseteq \pi\upsilon(s)$: Let $t \in \upsilon\pi(s)$ be given. I.e., there exists t' such that $t' \sqsubseteq s$ and $t' \preceq t$.

The problem is now to find a u such that $s \preceq u$ and $t \sqsubseteq u$. The idea is to define u such that it is the least extension of t to the elements of $A_s (\supseteq A_t)$ such that u is a semiword. Define u by $A_u := A_s$ and $\leq_u := R^+$, where $R = \leq \cup \leq_t$ and $\leq = \{(a,b) \in A_s{}^2 \mid (a,b) = (c^i, c^j)$ for some $c \in \Delta$ and $i \leq j\}$.

At first we want to show that u is a semiword.
Since $A_u = A_s$ and $s \in SW$ we have A_u fulfills $SW1$. Notice that $\leq$ is the least po on A_u which satisfies $SW2$. Because $\leq \subseteq R^+$ we see that R^+ fulfills $SW2$ if R^+ is a po. Since $\leq_s$ satisfies $SW2$ and $\leq$ is the least po that does so we have $\leq \subseteq \leq_s$. Then we see $DC_{\leq}(A_t) \subseteq A_t$ because $DC_{\leq_s}(A_{t'}) \subseteq A_{t'}, A_t = A_{t'}$. Also $\leq|_{A_t{}^2}$ is the least po on A_t which satisfies $SW2$, so $\leq|_{A_t{}^2} \subseteq \leq_t$ and a) of lemma 1.3.37 is satisfied (with $A = A_s$, $Q = \leq_t$ and $B = A_t$). Hence we conclude that u is a semiword and $DC_{\leq_u}(A_t) \subseteq A_t$.

Since $\leq \subseteq \leq_s$ and $\leq_t \subseteq \leq_{t'} = \leq_s |_{A_{t'}{}^2} \subseteq \leq_s$ it follows that $\leq_u \subseteq \leq_s$ wherefore $s \preceq u$.

To see $t \sqsubseteq u$, at first notice $A_t = A_{t'} = A_s|_{A_{t'}} = A_u|_{A_{t'}} = A_u|_{A_t}$. $\leq_t \subseteq \leq_u|_{A_t{}^2}$ by definition of $\leq_u$. And from (1.7) of lemma 1.3.37 $\leq_u |_{A_t{}^2} \subseteq \leq_t$ follows. So t is a subsemiword of u and we already know $DC_{\leq_u}(A_t) \subseteq A_t$. □

It is easy to see that $\pi\upsilon(s) \not\subseteq \upsilon\pi(s)$. Take for instance the semiword $s = a \rightarrow b$. Then for $t' = \begin{matrix} a \\ b \end{matrix}$ we have $t' \in \upsilon(s)$ and $t = \underline{b} \in \pi(t')$. Hence $t \in \pi\upsilon(s)$. If t should belong to $\upsilon\pi(s)$ there should be an s' such that $s' \preceq t$ and $s' \sqsubseteq s$. Now $s' \preceq t$ implies $A_{s'} = \{b^1\}$. But there is no prefix s' of s with $A_{s'} = \{b^1\}$ because $a^1 \in DC_{\leq_s}(b^1)$ and therefore also should be included in $A_{s'}$.

By χ it even gets worse. In general we have neither $\pi\chi S \subseteq \chi\pi S$ nor $\chi\pi S \subseteq \pi\chi S$. The latter can be seen by the example $S = \left\{ \begin{matrix} a \rightarrow b \rightarrow c \\ d \end{matrix}, \begin{matrix} a \rightarrow b \\ c \end{matrix} \right\}$, $a \begin{matrix} \nearrow b \\ \searrow c \end{matrix} \in \chi\pi S$ and the former by $S = \left\{ a \rightarrow b \rightarrow c, \begin{matrix} a \rightarrow b \\ c \end{matrix} \right\}, a \rightarrow c \in \pi\chi S$.

In the next propositions the interrelation between the connected components of semiwords which are in $\preceq$.

Proposition 1.3.39 $s \preceq t \Rightarrow \forall u \in \gamma(s) \exists D \subseteq \gamma(t).\ u \preceq \| D$.

Proof Induction on the number of connected components of s.

$|\gamma(s)| = 1$: Then $s = \varepsilon$ and therefore also $t = \varepsilon$. Chose $D = \{\varepsilon\} = \gamma(t)$.

$|\gamma(s)| > 1$: Then there is an $s' \in \gamma(s)$. $s' \neq \varepsilon$, and we can write $s = s' \parallel s''$, where $s'' = \parallel \gamma(s) \setminus \{s'\}$. By proposition 1.3.17 we have $s = s' \parallel s'' \preceq t$ implies $\exists s' \preceq t', s'' \preceq t''. t = t' \parallel t''$. From proposition 1.2.13.a) we see $\gamma(t) = \gamma(t') \cup \gamma(t'')$. Hence for s' we can chose $D = \gamma(t') \subseteq \gamma(t)$ and get $s' \preceq t' = \parallel \gamma(t') = \parallel D$. This settles the case if $u = s'$.

So it remains to prove $\forall u \in \gamma(s) \setminus \{s'\} \exists D \subseteq \gamma(t). u \preceq \parallel D$. $s' \in \gamma(s)$ implies $\gamma(s') = \{\varepsilon, s'\}$, so from $s' \neq \varepsilon$ and proposition 1.2.13.a) we get $\gamma(s) \setminus \{s'\} = \gamma(s' \| s'') \setminus \{s'\} = ((\gamma(s') \setminus \{\varepsilon\} \uplus \gamma(s'') \setminus \{\varepsilon\}) \cup \{\varepsilon\}) \setminus \{s'\} = ((\{s'\} \uplus \gamma(s'') \setminus \{\varepsilon\}) \setminus \{s'\}) \cup \{\varepsilon\} = (\gamma(s'') \setminus \{\varepsilon\}) \cup \{\varepsilon\} = \gamma(s'')$. Since $t = t' \| t''$ only if $\gamma(t'') \subseteq \gamma(t)$ it follows that it is enough to prove $\forall u \in \gamma(s'') \exists D \subseteq \gamma(t''). u \preceq \parallel D$. We have $s'' \preceq t''$, so we get the wanted directly by hypothesis of induction if we can prove $|\gamma(s'')| < |\gamma(s)|$. Now proposition 1.2.13 gives $|\gamma(s) \setminus \{\varepsilon\}| = |\gamma(s') \setminus \{\varepsilon\}| + |\gamma(s'') \setminus \{\varepsilon\}|$ so $|\gamma(s'') \setminus \{\varepsilon\}| = |\gamma(s) \setminus \{\varepsilon\}| - |\gamma(s') \setminus \{\varepsilon\}| =$ (since $\gamma(s') = \{\varepsilon, s'\}, s' \neq \varepsilon$) $|\gamma(s) \setminus \{\varepsilon\}| - 1$. Because in general $\varepsilon \in \gamma(v)$ for arbitrary v we have $|\gamma(s'')| = |\gamma(s)| - 1 < |\gamma(s)|$. □

In general $\gamma(s) \neq \emptyset$ wherefore we also have $s \preceq t \Rightarrow \exists u \in \gamma(s) \exists D \subseteq \gamma(t).\ u \preceq \parallel D$.

If D is a set of semiwords we let A_D denote $\cup_{s \in D} A_s$ in the following proposition and it's proof.

Proposition 1.3.40 Given s, t such that $A_s = A_t$ and for each $s' \in \gamma(s)$ a $D_{s'} \subseteq \gamma(t)$ with $A_{s'} = A_{D_{s'}}$. Then:

$$\gamma(t) = \bigcup_{s' \in \gamma(s)} D_{s'}$$

Proof Let D denote $\cup_{s' \in \gamma(s)} D_{s'}$. Because each $D_{s'} \subseteq \gamma(t)$ we clearly have $D \subseteq \gamma(t)$.
To see $D \supseteq \gamma(t)$ assume on the contrary that there exists a $t' \in \gamma(t)$ such that $t' \notin D$.
At first notice $t' \in \gamma(t)$ implies $A_{t'} \subseteq A_t$.
Next we prove $t' \neq \varepsilon$. Because $\varepsilon \in \gamma(s)$ we have a $D_\varepsilon \subseteq \gamma(t)$ with $A_\varepsilon = \emptyset = A_{D_\varepsilon}$. Since $u = \varepsilon$ is the only semiword with $A_u = \emptyset$ we must have $D_\varepsilon = \{\varepsilon\}$. Hence $\varepsilon \in D$ and from $t' \notin D$ we then see $t' \neq \varepsilon$.
Because $D \subseteq \gamma(t)$ and $\gamma(t)$ consists of disjoint semiwords $t' \in \gamma(t) \setminus D$ must imply $A_{t'} \cap A_{t''} \neq \emptyset$ for every $t'' \in D$. From $t \neq \varepsilon$ and thereby $A_{t'} \neq \emptyset$ we then conclude $A_{t'} \nsubseteq A_D$ But this implies $A_{t'} \nsubseteq A_D = \bigcup_{s' \in \gamma(s)} A_{D_{s'}} = \bigcup_{s' \in \gamma(s)} A_{s'} = A_s = A_t$ which contradicts $A_{t'} \subseteq A_t$. □

Because $s \preceq t$ only if $A_s = A_t$ we have the following.

Corollary 1.3.41 Given s, t such that $s \preceq t$ and for each $s' \in \gamma(s)$ a $D_{s'} \subseteq \gamma(t)$ with $s' \preceq \parallel D_{s'}$. Then:

$$\gamma(t) = \bigcup_{s' \in \gamma(s)} D_{s'}$$

Proposition 1.3.42 $a.s \preceq t, |\gamma(t)| > 3$ implies $\exists u.\ a.s \prec u \prec t$

Proof Since t is finite and thereby $|\gamma(t)|$ too we will by repeated use of proposition 1.3.30 find a $t_1 \in \gamma(t)$ such that $a.t'_1 \preceq t_1, s \preceq t'_1 \parallel (\parallel \gamma(t) \setminus \{t_1\})$ for some t'_1. Because $t_1 \in \gamma(t)$ and $|\gamma(t)| > 3$ we can write $\|\gamma(t) \setminus \{t_1\}$ as $t_2 \parallel t_3$ for some $t_2, t_3 \neq \varepsilon$. So we have $a.s \preceq t_1 \parallel t_2 \parallel t_3, a.t'_1 \preceq t_1, s \preceq t'_1 \parallel t_2 \parallel t_3$.
Now chose $u = a.(t'_1 \parallel t_2) \parallel t_3$.
From proposition 1.3.29.a) we have $a.((t'_1 \| t_2) \| t_3) \preceq a.(t'_1 \| t_2) \| t_3 = u$. Since $t_3 \neq \varepsilon$ we have $\gamma(a.t'_1 \| t_2 \| t_3) \neq \gamma(u)$, so $a.(t'_1 \| t_2 \| t_3) \prec u$. From $s \preceq t'_1 \| t_2 \| t_3$ we see $a.s \preceq a.(t'_1 \| t_2 \| t_3)$. Hence $a.s \prec u$. Again from proposition 1.3.29 we get $a.(t'_1 \parallel t_2) \preceq a.t'_1 \parallel t_2$ and thereby $u = a.(t'_1 \parallel t_2) \parallel t_3 \preceq (a.t'_1 \parallel t_2) \parallel t_3$. As $t_2, t_3 \neq \varepsilon$ we conclude $u \prec a.t'_1 \parallel t_2 \parallel t_3$. Now $a.t'_1 \preceq t_1$ implies $a.t'_1 \parallel t_2 \parallel t_3 \preceq t_1 \parallel t_2 \parallel t_3 = t$, so also $u \prec t$. □

The definition of the relation $<\!\cdot$ for a po $(A, \leq)$ is:

$$\forall a, b \in A.\ a <\!\cdot\ b \ \textit{iff}\ \ a < b \text{ and } \ \nexists c \in A.\ a < c < b$$

That is $a <\!\cdot\ b$ means a is an *immediate predecessor* of b in the relation $<$. The $<\!\cdot$ might be empty some pos though $\leq$ is not, but for $\preceq$ on SW we in fact have $\prec\!\cdot^+ = \prec$ and $\prec\!\cdot^* = \preceq$. This is seen as follows. Let $s \prec t$. This means $A_s = A_t$ and $s \in \delta(t), t \in \upsilon(s)$. So every semiword u of a $\prec$-path from $s \prec t$ must be in $\delta(t) \cap \upsilon(s)$. As noticed earlier δ and υ are in general finite, so all such path's are finite as well wherefore there exists $0 \leq n$, and some $u_i, i \in \underline{n}$ such that $s \prec\!\cdot\ u_1 \prec\!\cdot\ u_2 \ldots \prec\!\cdot\ u_n \prec\!\cdot\ t$. Clearly then $\prec\!\cdot^+ = \prec$ and $\prec\!\cdot^* = \preceq$

The lately proved properties allows us to show a implication of $s \prec\!\cdot\ t$.

Proposition 1.3.43 $s \prec\!\cdot\ t$ implies $\exists s' \in \gamma(s) \setminus \{\varepsilon\} \exists D \subseteq \gamma(t).\ \gamma(s) \setminus \{s'\} = \gamma(t) \setminus D, s' \prec\!\cdot \parallel D$.

Proof Clearly the proof must find an $s' \in \gamma(s) \setminus \{\varepsilon\}$ such that

$$\exists D_{s'} \subseteq \gamma(t).\ s' \prec \parallel D_{s'} \tag{1.12}$$

Notice that there is no t with $\varepsilon \prec\!\cdot\ t$. For the same reason (1.12) cannot hold for $s' = \varepsilon$ neither.

At first we prove that there is at most one $s' \in \gamma(s) \setminus \{\varepsilon\}$ such that (1.12) holds.
Assume on the contrary that there are (at least) two different nonempty connected components of s for which (1.12) holds. I.e., assume $\exists s', s'' \in \gamma(s) \setminus \{\varepsilon\} \exists D_{s'}, D_{s''} \subseteq \gamma(t).\ s' \neq s'', s' \prec \parallel D_{s'}, s'' \prec \parallel D_{s''}$.
By proposition 1.3.39 we find a $D_u \subseteq \gamma(t).\, u \preceq \parallel D_u$ for every $u \in \gamma(s)$. Let $v = \|\{u \mid u \in \gamma(s) \setminus \{s', s''\}\} = \parallel \gamma(s) \setminus \{s', s''\}$ and $v' = \|\{D_u \mid u \in \gamma(s) \setminus \{s', s''\}\}$. Clearly $s = s' \| s'' \| v$ and $v \preceq v'$ so by corollary 1.3.41 we have $D_{s'} \cup D_{s''} \cup \{D_u \mid u \in \gamma(s) \setminus \{s', s''\}\} = \gamma(t)$ and thereby $(\parallel D_{s'}) \parallel (\parallel D_{s''}) \parallel v' = t$. From $s' \prec \|D_{s'}, s'' \prec \parallel D_{s''}$ and $v \preceq v'$ we now get $s = s' \parallel s'' \parallel v \prec (\parallel D_{s'}) \parallel s'' \parallel v \prec (\parallel D_{s'}) \parallel (\parallel D_{s''}) \parallel v \preceq (\parallel D_{s'}) \parallel (\parallel D_{s''}) \parallel v' = t$—a contradiction to $s \prec\!\cdot\ t$.

Next we prove that there is at least one $s' \in \gamma(s) \setminus \{\varepsilon\}$ such that (1.12) holds.
Assume on the contrary there is no such s'. As noticed (1.12) does not hold for $s' = \varepsilon$, so we can in fact assume (1.12) not to hold for $s' \in \gamma(s)$.
From proposition 1.3.39 we see $\forall s' \in \gamma(s) \exists D_{s'}.\ s' \preceq \parallel D_{s'}$. Since there by assumption

is no $s' \in \gamma(s)$ such that (1.12) holds this implies $\forall s' \in \gamma(s).\ s' = \| D_{s'}$. This has as consequence $\gamma(s') = D_{s'}$ and $A_{s'} = A_{D_{s'}}$ for all $s' \in \gamma(s)$. Then by proposition 1.3.40 we have $\gamma(t) = \bigcup_{s' \in \gamma(s)} D_{s'}$ from which we get: $\gamma(t) = \bigcup_{s' \in \gamma(s)} D_{s'} = \bigcup_{s' \in \gamma(s)} \gamma(s') = \gamma(s)$, so $s = t$ which contradicts $s \prec\!\!\cdot\, t$.

Now let $s' \in \gamma(s) \setminus \{\varepsilon\}$ be the only one for which (1.12) holds and $D_{s'}$ the corresponding subset of $\gamma(t)$.
We know $s' \neq \varepsilon$, so we might define $D = D_{s'} \setminus \{\varepsilon\}$ and still have $D \neq \emptyset, s' \prec \| D$. Using proposition 1.3.39 again we have $\forall u \in \gamma(s) \exists D_u \subseteq \gamma(t).\ u \preceq \| D_u$. Since s' is the only semiword of $\gamma(s)$ with $s' \prec \| D$ we have $u = \| D_u$ for $u \in \gamma(s) \setminus \{s'\}$. From proposition 1.3.40 we now get $\gamma(t) = D \cup \bigcup_{u \in \gamma(s) \setminus \{s'\}} D_u$. D is disjoint to $\cup_{u \in \gamma(s) \setminus \{s'\}} D_u$. If not then we have a $v \in D \cap D_u$ for some $v \in \gamma(s) \setminus \{s'\}$. Because $\varepsilon \notin D$ we have $v \neq \varepsilon$. This together with $v \in D \cap D_u$ implies $A_D \cap A_{D_u} \neq \emptyset$. Since $A_D = A_{s'}$ and $A_u = A_{D_u}$ this also means $A_{s'} \cap A_u \neq \emptyset$ which contradicts $u, s' \in \gamma(s) \setminus \{\varepsilon\}$ and $\gamma(s)$ consisting of disjoint semiwords. Hence $\gamma(t) = D \uplus \cup_{u \in \gamma(s) \setminus \{\varepsilon\}} D_u$. So $\gamma(t) \setminus D = \bigcup_{u \in \gamma(s) \setminus \{s'\}} D_u =$ (because $u = \| D_u$) $\cup_{u \in \gamma(s) \setminus \{s'\}} \gamma(u) =$ (because $s' \neq \varepsilon$) $\gamma(s) \setminus \{s'\}$.

The conclusion of the first three steps of the proof is now:
$s \prec\!\!\cdot\, t$ implies $\exists s' \in \gamma(s) \setminus \{\varepsilon\} \exists D \subseteq \gamma(t).\ \gamma(s) \setminus \{s'\} = \gamma(t) \setminus D, s' \prec \| D$, so the final step is to prove $s' \prec\!\!\cdot\, \| D$.
Assume on the contrary that there exists a u with $s' \prec u \prec \|D$. Then $s = s' \| (\| \gamma(s) \setminus \{s'\}) \prec u \| (\| \gamma(s) \setminus \{s'\}) \prec (\| D) \| (\| \gamma(s) \setminus \{s'\}) = (\| D) \| (\| \gamma(t) \setminus D) = \| \gamma(t) = t$—a contradiction to $s \prec\!\!\cdot\, t$! $\square$

Chapter 2

Tree Semiwords: TSW

2.0 Preliminaries

We are now going to define a particular subclass of semiwords called tree semiwords which can be seen as reflecting non-synchronized behaviour.

Definition 2.0.1 A poset t of $\Delta \times I\!N^+$ is a tree-semiword *iff* t fulfills $SW1$, $SW2$ and:

T: $\forall a, b, c \in A_t.\ a \leq_t c, b \leq_t c \Rightarrow a \leq_t b \vee b \leq_t a$

The class of *tree-semiwords* over Δ is denoted $TSW(\Delta)$ (TSW for short). □

Corollary 2.0.2 $W \subseteq TSW \subseteq SW$

Technically it is convenient to introduce the notion of a rooted tree-semiword.

Definition 2.0.3 r is a rooted tree-semiword *iff* r is a tree-semiword and:

RT: $\exists a \in A_r\ \forall b \in A_r.\ a \leq_r b$

The class of *rooted tree-semiwords* over Δ is denoted $RTSW(\Delta)$ ($RTSW$ for short). □

Corollary 2.0.4 $W \setminus \{\varepsilon\} \subseteq RTSW \subseteq TSW$.

It would be nice if we could carry over all the definitions and results of semiwords to the subclass of tree-semiwords. Unfortunately, this cannot be done entirely, the main reason being that though a construction from some tree-semiwords yields a semiword, it is not ensured to be a tree-semiword. The most conspicuous example is that the concatenation of two tree-semiwords does not necessarily give a tree-semiword.

Therefore, we will briefly repeat the definitions and results of the previous chapter, making a few changes and necessary additions. Whenever a result or definition of this chapter is referred (as e.g., corollary 1.2.14) later on and it is not stated here explicit it is because it carry over directly from chapter 1 (of course with SW changed to TSW). To emphasize that it is a tree-semiword version the reference will be subscribed with a T like: proposition$_T$.

2.1 Basic Definitions

The definition of restriction, subsemiword and complement semiword of semiwords can directly be carried over to tree-semiwords.

Proposition 1.1.4 now says that $s|_{A^2}$ and the complement semiword are tree-semiwords (if $A \subseteq A_s$ and A fulfill $SW1$).

Proof From the corresponding semiword proof we know they are semiwords, so we only have to show that they have the T-property:

At first notice that in general for a poset $(A, \leq)$ having the T-property, any poset $(B, \leq|_{B^2})$ where $B \subseteq A$, has the T-property too. From the corresponding semiword proof we already know that $s|_A$ is a semiword, and since it is a restriction of a tree-semiword s we know that $\leq_s|_{A^2}$ fulfills T and we are done with a).

For b) we also know that t is a semiword. But $\leq_t$ is just $\leq_u|_{(A_t \setminus A_s)^2}$ shifted left according to s so $\leq_t$ must fulfill T too. □

Also the definition of connected components of a semiword and the belonging results can be carried over. Since we already know that a connected component is a subsemiword, we only have to observe that it is a restriction, hence having the T-property too, wherefore it is a tree-semiword.

Having the notion of rooted tree-semiwords we can get a finer view of tree-semiwords. We extend corollary 1.1.8 with:

f) A nonempty connected component (of a tree-semiword) is a rooted tree-semiword.

This is perhaps not totally obvious, so we prove it:

Proof Let s be a connected component of a tree-semiword. We already know that it is a tree-semiword so we shall prove that $\leq_s$ have the RT-property. Define $R := (\leq_s \cup \leq_s^{-1})$. That s is connected means $\forall b, c \in A_s.\ b\ R^+\ c$.

To continue we need an intermediate result:

$$b\ R^+\ c \Rightarrow \exists a (\in A_s).\ a \leq_s b, a \leq_s c \tag{2.1}$$

We prove this by proving $b\ R^n\ c \Rightarrow \exists a.\ a \leq_s b, a \leq_s c$ by induction on n.

$n = 1$: I.e., $b\ R\ c$. This means either $b \leq_s c$ or $c \leq_s b$. Let a equal b in the former case and c in the latter.

$n > 1$: Then there exists a d such that $b\ R\ d\ R^{n-1}\ c$. Using the hypothesis of induction on $d\ R^{n-1}\ c$ we find $a'(\in A_s)$ such that $a' \leq_s d, a' \leq_s c$. We now look at the possibilities of $b\ R\ d$.

$b \leq_s d$: Since $s \in TSW$ we have $a' \leq_s d, b \leq_s d \Rightarrow a' \leq_s b \vee b \leq_s a'$. In the latter case choose $a = b$ and in the former $a = a'$. By reflexivity and transitivity of $\leq_s$ we are then done.

$d \leq_s b$: Then let $a = a'$. We then have $a \leq_s c$ and by transitivity of $\leq_s$ also $a \leq_s b$.

The next is:

$$\exists a \in A_s\ \forall b \in B.\ a \leq_s b \text{ if } \emptyset \neq B \subseteq A_s$$

We prove it by induction on the size of B. Since $B \neq \emptyset$ the induction basis must be:

$|B| = 1$: Then $B = \{b\}$ for some $b \in A_s$. By reflexivity of $\leq_s$ we have $b \leq_s b$. Choose $a = b$. Because $B \subseteq A_s$ we are done.

$|B| > 1$: Pick out some $b \in B$. Use the inductive hypothesis on $B \setminus \{b\}$ to find a $c \in A_s$ such that $\forall d \in B \setminus \{b\}.\ c \leq_s d$. Because s is connected $b\ R^+\ c$. Then by (2.1) there exists an $a \in A_s.\ a \leq_s b, a \leq_s c$. By transitivity of $\leq_s$: $\forall d \in B \setminus \{b\}.\ a \leq_s d$. Hence also $\forall b \in B.\ a \leq_s b$.

With the last result b) now follows directly by noticing that s nonempty implies $A_s \neq \emptyset$ and that A_s is a subset of itself. □

2.2 Operations on TSW

Nullary

We have already noticed that $W \subseteq TSW$, so especially $\varepsilon, \underline{a} \in TSW$.

Binary

We have already noticed that concatenation does not carry over as it is.

In fact we have:

$$\forall s \in SW \setminus W\ \forall t \in SW.\ t \neq \varepsilon \Rightarrow st \notin TSW$$

Proof $s \in SW \setminus W$ implies that there exist $a, b \in A_s$ such that $a \not\leq_s b, b \not\leq_s a$. Since $t \neq \varepsilon$ there exists a $c^i \in A_t$. Then $d = c^{\psi(s,c)+i} \in A_{st} \setminus A_s$. As noticed by definition 1.2.4 we have $a \leq_{st} d, b \leq_{st} d$. Since $\leq_s$ and $\leq_{st}$ agree on A_s, $a \not\leq_s b, b \not\leq_s a$ implies $a \not\leq_{st} b, b \not\leq_{st} a$. Hence st is not a tree-semiword. □

As a consequence of this we must restrict the domain of concatenation from $TSW \times TSW$ to $W \times TSW$.

The properties carry over. The only one we will dwell on is that st in fact is a tree-semiword when $s \in W$. In order not to write $s \in W$ whenever we consider st for $s, t \in TSW$ we take it as a convention from now on.

st is a tree-semiword:

Proof We already know that st is a semiword, so we shall convince ourselves that $\forall a, b, c \in A_{st}.\ a \leq_{st} c, b \leq_{st} c \Rightarrow a \leq_{st} b \vee b \leq_{st} a$ (T-property). Let us consider the membership of c.

$c \in A_s$: Then of course $a, b \in A_s$ and the result follows because s and st agree on A_s.

$c \notin A_s$: I.e., $c \in A_{st} \setminus A_s$. If both $a, b \in A_s$ then $a \leq_s b$ or $b \leq_s a$ because $s \in W$, hence $a \leq_{st} b$ or $b \leq_{st} a$. If both $a, b \in A_{st} \setminus A_s$ we get the result from t being a tree-semiword and the correspondence between $\leq_{st}$ and $\leq_t$. If $a \in A_s, b \in A_{st} \setminus A_s$ we already know $a \leq_{st} b$. Similarly if $a \in A_{st} \setminus A_s, b \in A_s$.

So st is indeed a tree-semiword. □

Whereas we had to restrict the definition of concatenation in order to get the tree-semiwords as results this is not the case for parallel composition. The definition and the results can be carried over.

We conclude this section by a proposition which bring light to the connection between $TSW(/RTSW)$ and its operators.

Proposition 2.2.15

a) $\forall s \in SW.\ s \in RTSW$ *iff* $\exists a \in \Delta, \exists t \in TSW.\ s = a.t$

b) Every $t \in TSW$ can be *generated* from ε, $\|$ and a. $(a \in \Delta)$

Proof
a) *if*: We already have that $a.t \in TSW$ and in general $\forall a \in A_s\ \forall b \in A_{st} \setminus A_s.\ a \leq_{st} b$, so especially for $a^1 \in A_{\underline{a}} \subseteq A_{a.t}$ we have $\forall b \in A_{a.t} \setminus A_{\underline{a}}.\ a^1 \leq_{a.t} b$. Hence $\leq_{a.t}$ fulfills RT and $a.t \in RTSW$.

only if: Given $s \in RTSW$. By definition there exists an $a \in A_s$ such that $\forall b \in A_s . a \leq_s b$. Clearly a must have rank 1, so $\{a^1\}$ fulfills $SW1$. Then $\underline{a} = s|_{\{a^1\}}$ is a subtree-semiword of s. Let t be the complement tree-semiword of $\underline{a}$ w.r.t. s. So we have $t \in TSW$. What remains to prove is that $s = a.t$. Clearly $A_s = A_{a.t}$. $\leq_s = \leq_{a.t}$ is seen by noticing $b^j \neq a^1 \Leftrightarrow b^j \in A_s \setminus \{a^1\} \Leftrightarrow \psi(\underline{a}, b) < j$ and looking at the definition of concatenation and complement tree-semiword.

b) Follows by induction, directly from $t = \varepsilon \parallel (\parallel \gamma(t) \setminus \{\varepsilon\})$, proposition 1.2.13, corollary 1.1.8 and a) above. □

2.3 Partial Orders on TSW

2.3.1 Smoother Than

The definition of smoother than and linearization carries over. However it is worth remarking that the $\preceq$-downwards closure of a tree-semiword t within SW is not contained in TSW. E.g., with

(2.2) $$t = \begin{matrix} c \rightarrow b \\ a \end{matrix} \in TSW \text{ and } s = \begin{matrix} c \searrow \\ a \nearrow \end{matrix} b \in SW$$

we have $s \preceq t$ in SW but $s \notin TSW$.

So it is clear that some care must be taken when using $\preceq$ on TSW, especially when constructing a new (tree-)semiword which is claimed to be smoother than another tree-semiword.

We will now pick out the cases where the difference is significant. One of the most conspicuous cases is in fact the first lemma:

Lemma 2.3.6 $\forall s \in TSW\ \forall a, b \in A_s.\ (a \not\leq_s b, b \not\leq_s a \Rightarrow \exists t \in TSW.\ t \prec s, a \leq_t b)$

Proof Whereas we before just added (a, b) to $\leq_s$ taking the transitive closure we cannot do this any longer, as can be seen from the example above. In general there there can be more least refinements of $\leq_s$ containing (a, b). E.g., in (2.2) above $a \rightarrow c \rightarrow b$ and $c \rightarrow a \rightarrow b$ are two such least refinements of $\leq_s$. So we can just as well choose in what way to refine $\leq_s$. By the new idea (a, b) still is added to $\leq_s$ but not necessarily directly. We consider two cases.

a, b in A_s are *not connected*:
By corollary 1.1.8.f) the connected component which b belongs to is a rooted tree-semiword. So let d denote the root and we have $d \leq_s b$. Now define $A_t = A_s, \leq_t = Q^+$, where $Q = (\leq_s \cup \{(a, d)\})$. Clearly $a \leq_t b$ and $\leq_s \subset \leq_t$. As in the proof for semiwords we see that A_t fulfills $SW1$, $\leq_t$ fulfills $SW2$ and is transitive so as reflexive. Now for the antisymmetry:
We shall show $f\ Q^+\ g, g\ Q^+\ f \Rightarrow f = g$.
Since a, d belongs to two different connected components of s we cannot have $a \leq_s d$ or $d \leq_s a$. Hence $f\ Q^n\ g$ implies $f \leq_s g$ or $f \leq_s (a, d) \leq_s g$. Similar for $g\ Q^n\ f$. So there are four cases to consider. If $f \leq_s g, g \leq_s f$ we get $f = g$ from the antisymmetry of $\leq_s$. The remaining cases can be excluded since they all implies $d \leq_s a$ which as noticed is impossible.
It remains to show the T-property of $\leq_t$. Suppose $f\ Q^+\ h, g\ Q^+\ h$. We shall show $f\ Q^+\ g$ or $g\ Q^+\ f$. Again there are four cases:

$f \leq_s h, g \leq_s h$: Follows form $\leq_s$ having the property.

$f \leq_s h, g \leq_s (a, d) \leq_s h$: Then $f \leq_s d$ or $d \leq_s f$. In the former case we must have $f = d$ by the way d is chosen. But then $g \leq_s (a, d)\ f$ i.e., $g\ Q^+\ f$. In the latter case $g \leq_s (a, d) \leq_s f$.

$f \leq_s (a, d) \leq_s h, g \leq_s h$: Symmetric.

$f \leq_s (a, d) \leq_s h, g \leq_s (a, d) \leq_s h$: Then $f \leq_s a, g \leq_s a$ and the result follows.

Now suppose a, b are *connected.*
I.e., $a\ R^+\ b$, where $R = (\leq_s \cup \leq_s{}^{-1})$. By corollary 1.1.8.f) this component a and b belongs to is a rooted tree-semiword so there exists a $c' \in A_s$ such that $c' \leq_s a, c' \leq_s b$. Let c be the element of A_s such that $c' \leq_s c, c \leq_s a, c \leq_s b$ and there is no c'' with $c <_s c''$ such that $c'' \leq_s a, c'' \leq_s b$, i.e., c is the greatest lower bound of a, b w.r.t. $\leq_s$ (exists since $DC_{\leq_s}(s)$ and $DC_{\leq_s}(b)$ are finite). Since we are dealing with trees, the paths leading from c to a and b must be unique. Let d denote the first element after c on the path to b. The situation is as illustrated:

$$\cdots c \begin{matrix} \nearrow \cdots\cdot a \\ \searrow d \cdots b \end{matrix}$$

We are now ready to define t:

$A_t = A_s$ and $\leq_t = Q^+$, where $Q = \leq_s \cup \{(a,d)\}$. By construction we immediately have $a \leq_t b$ and $\leq_s \subset \leq_t$. As above we immediate see all the propertys of t in order to be a semiword except for antisymmetry. To see this we first need some intermediate results.

$f\ Q^+\ a \Rightarrow f \leq_s a$: Suppose $f \not\leq_s a$. Then it must be possible to write the path of Q establishing $f\ Q^+\ a$ as: $f\ Q^n(a,d) \leq_s a$. This means $d \leq_s a$. By the way d is chosen we also have $d \leq_s b$. But this contradicts the way c is chosen.

$d\ Q^+\ g \Rightarrow d \leq_s g$: Similarly we see that $d \leq_s (a,d) Q^m\ g$, and the contradiction is obtained in the same way.

Now if $f \not\leq_s g$ and $f\ Q^+\ g$ then $f\ Q^n\ a$ and $d\ Q^m\ g$. From the above we see that this implies $f \leq_s a$ and $d \leq_s g$. The antisymmetry can now be seen as in the proof for semiwords.

What remains to prove is that $\leq_t$ fulfills T. Assume $f\ Q^+\ h$ and $g\ Q^+\ h$. We shall prove either $f\ Q^+\ g$ or $g\ Q^+\ f$.

$f \leq_s h, g \leq_s h$: Then we get it from $\leq_s$ having the property.

$f \not\leq_s h, g \not\leq_s h$: Here we by definition know that $f \leq_s a$ and $g \leq_s a$, so the T-property follows again.

$f \leq_s h, g \not\leq_s h$: Then we have $g \leq_s a, d \leq_s h$. From the former case we conclude $d \leq_s f$ or $f \leq_s d$. If $d \leq_s f$ we have $g \leq_s (a,d) \leq_s f$ i.e., $g\ Q^+\ f$. For $f <_s d$ notice that any path of $\leq_s$ from f to d must go through c since c is the immediate predecessor of d, so $f \leq_s c$ (may be $f = c$). By the way c is chosen $c \leq_s a$, so $f \leq_s a$. From this and $g \leq_s a$ we conclude $f \leq_s g$ or $g \leq_s f$. If $f = d$ clearly $g \leq_s (a,d)\ f$ or equally $g\ Q^+\ f$.

$f \not\leq_s h, g \leq_s h$: Is handled symmetrically.

□

The next lemmas and corollaries carry over directly, so proposition 1.3.5 also holds for tree-semiwords.

The propositions concerning concatenation have to be modified a little. If we take over the formulation $s \preceq t \Leftrightarrow sr \preceq tr \Leftrightarrow rs \preceq rt$ in proposition 1.3.10, it is trivial because it only is defined when $s, t \in W$. Instead we have the proposition.

Proposition 2.3.10 a) $s \preceq t \Leftrightarrow rs \preceq rt$

The a) part of the next corollary can be left out since it is just a) of the proposition, because $s_1 \preceq t_1$ and $s_1, t_1 \in w$ implies $s_1 = t_1$. From this we also get that the next proposition is formulated:

Proposition 2.3.12 $u \preceq st \Rightarrow \exists t' \preceq t.\, u := st'$

The proof can be carried over with the addition $s' \preceq s \in W$ implies $s' = s$.

Proposition 2.3.13

a) $u \in \lambda(st) \Leftrightarrow \exists t' \in \lambda(t).\, u = st'$

b) $u \in \lambda(s \parallel t) \Leftrightarrow \exists s' \in \lambda(s)\ \exists t' \in \lambda(t).\, u \preceq s' \parallel t',\ \nexists u'.\, u' \prec u$

Proof

a) The proof of semiwords can be used directly because it only uses previous results which we know hold for tree-semiwords.

b) Also this proof can be carried over.

$\square$

Since $s \in W$ implies $\delta(s) = \{s\}$, c) of proposition 1.3.15 can be read $\delta(st) = s\delta(t)$. The proofs are identical.

We now turn to the $\preceq$-upwards closure υ. proposition 1.3.17 is the same. c) in the next proposition does not carry over because $\upsilon(s)\upsilon(t)$ is not defined for all $s \in W$. Instead it should be

Proposition 2.3.18 c) $\{s\}\upsilon(t) \subseteq \upsilon(st)$.

Proof The same. $\square$

The corollary reads $\upsilon(st) = \upsilon(\{s\}\upsilon(t))$, but for $s = \underline{a}$ we do have $\upsilon(a.t) = \upsilon(\upsilon(\underline{a})\upsilon(t))$!

The definition of χ and the associated results carry over smoothly.

2.3.2 Prefix of

The definition of prefix can be used directly, so as e.g., the proof that $\sqsubseteq$ is a po an TSW.

The propositions concerning $\sqsubseteq/\pi$ alone and the proofs of these all carry over because when constructing new semiwords these are either subsemiwords or complement semiwords and we know that if these constructions derive from tree-semiwords, we will also have subtree-semiwords or complement tree-semiwords respectively. Remember that when we write $u \sqsubseteq st \Rightarrow u \sqsubseteq s$ or $\exists t' \sqsubseteq t.\, u = st'$ in proposition 1.3.34 we still *presume s to be a word*.

The matter is rather different when it comes to the relations between $\lambda, \delta, \upsilon$ and π, i.e., proposition 1.3.38. In the proofs, semiwords are constructed where it is not obvious that the constructions yield tree-semiwords when they are constructed from such.

Proof The only proof which we can take over directly is $\pi\lambda(s) = \lambda\pi(s)$ because $\lambda(s) \subseteq W$ and $s \in W, t \sqsubseteq s$ implies $t \in W$, such that the constructions yield tree-semiwords when $s \in TSW$. We now look at the other proofs one by one.

$\pi\delta(s) \subseteq \delta\pi(s)$: The constructed u is to be a subsemiword of s. Since $s \in TSW$ we know that $u \in TSW$ and the proof can be reused directly.

It is not easy to see that the constructed u in the proof of the other inclusion is a subtree-semiword, but it will turn out that it in fact is a tree-semiword which we will now prove.

$\delta\pi(s) \subseteq \pi\delta(s)$: With $t \preceq t' \sqsubseteq s, A_u = A_s$ and $\leq_u = R^+$, where $R = \leq_s \cup \leq_t$ and under the assumption $t, s \in TSW$. It is already proved to be a semiword so what remains is to prove $a\ R^+\ c, b\ R^+\ c \Rightarrow a\ R^+\ b$ or $b\ R^+\ a$ (T-property). We will prove this by proving:

$$a\ R^n\ c, b\ R^m\ c \Rightarrow l < n+m \text{ and } (a\ R^l\ b \text{ or } b\ R^l\ a)$$

by induction on $n+m$.

$n+m=2$: Then $a\ R\ c, b\ R\ c$. Look at the different possibilities.

$a \leq_s c, b \leq_s c$: Since $s \in TSW$ we have $a \leq_s b$ or $b \leq_s a$, hence also $a\ R\ b$ or $b\ R\ a$.

$a \leq_t c, b \leq_t c$: Similar.

$a \leq_s c, b \leq_t c$: Since $\leq_t$ only is defined on $A_t \subseteq A_s$ we conclude $c \in A_t$ for $b \leq_t c$. From $t \preceq t'$ we get $A_t = A_{t'}$, hence $c \in A_{t'}$ $t' \sqsubseteq s$ gives us that $DC_{\leq_s}(c) \subseteq A_{t'}$, so $a \in A_{t'} = A_t$ and $a \leq_{t'} c$ by definition of $\sqsubseteq$. Again from $t \preceq t'$ we see $\leq_{t'} \subseteq \leq_t$ and therefore $a \leq_t c$. The result now follows from the case above.

$a \leq_t c, b \leq_s c$: Symmetric.

$n+m>2$: Then either $n \geq 2$ or $m \geq 2$. W.l.o.g. assume $n \geq 2$. This implies that there exists d such that $a\ R\ d, d\ R^{n-1}\ c$. Using the hypothesis on $d\ R^{n-1}\ c, b\ R^m\ c$ we get $(d\ R^{l'}\ b$ or $b\ R^{l'}\ d)$ and $l' < n+m-1$. We look at the two cases:

$d\ R^{l'}\ b$: Then from $a\ R\ d$ we get $a\ R^l\ b$, where $l = l'+1 < n+m-1+1 = n+m$.

$b\ R^{l'}\ d$: We have $a\ R\ d$, and since $l'+1 < m+n$ we can use the hypothesis of induction on this to obtain $a\ R^l$ or $b\ R^l\ a$, where $l < l'+1 < m+n$.

$\upsilon\pi(s) \subseteq \pi\upsilon(s)$: The situation is that for a given $t \in \upsilon\pi(s)$ where $s, t \in TSW$ a semiword u with $s \preceq u$ and $t \sqsubseteq u$ is defined by $u = (A_s, (\leq \cup \leq_t)^+)$, where $\leq = \{(a,b) \in {A_s}^2 \mid (a,b) = (c^i, c^j)$ for some $c \in \Delta$ and $i \leq j\}$. So it just remains to prove u has the T-property in order to get $u \in TSW$. Let a, b, c be given such that $a \leq_u c$ and $b \leq_u c$. We shall prove $a \leq_u b$ or $b \leq_u a$. We consider two main cases:

$c \in A_t$: Clearly $a \leq_u c$ and $b \leq_u c$ then implies $a \leq_t c$ and $b \leq_t c$. Since $t \in TSW$ it follows that $a \leq_t b$ or $b \leq_t a$ and so $a \leq_u b$ or $b \leq_u a$.

$c \notin A_t$: There are actually four subcases:

$a, b \notin A_t$: Then a, b, c must be equal labelled and so are ordered by definition of $\leq_u$.

$a \in A_t, b \notin A_t$: By $a \leq_u c$ and construction of $\leq_u$ from $\leq$ it then follows that there is an element c' labelled like c with $a \leq_t c' \leq c$. From $b \notin A_t$ and $b \leq_u c$ follows $c' \leq b \leq c$ so $a \leq_u b$.

$a \notin A_t, b \in A_t$: Symmetricly as in the last case we here see $b \leq_u a$.

$a, b \in A_t$: As above we see there are elements c' and c'' of A_t labelled like c such that $a \leq_t c$ and $b \leq_t c''$. Since c' and c'' are equally labelled either $c' \leq_t c''$ or $c'' \leq_t c'$. W.l.o.g. assume the former. Then $a \leq_t c''$ and $b \leq_t c''$ and the result follows from $t \in TSW$.

□

The remaining of chapter 1 carries over. Now to a proposition special for tree-semiwords.

Proposition 2.3.44 $s \prec\!\cdot\ t$ implies $\exists u \in \gamma(s), D \subseteq \gamma(t).\ \gamma(s) \setminus \{u\} = \gamma(t) \setminus D$ and for some $a, b \in Act$, $s', s'', t' \in TSW$ either

a) $u = a.(s' \parallel b.s''), D = \{a.s', b.s''\}$

or

b) $u = a.s', D = \{a.t'\}, s' \prec\!\cdot\ t'$

Proof We already have $s \prec\!\cdot\ t \Rightarrow \exists u \in \gamma(s) \setminus \{\varepsilon\}, D \subseteq \gamma(t). \gamma(s) \setminus \{u\} = \gamma(t) \setminus D, u \prec\!\cdot \parallel D$ from proposition 1.3.43, so it is enough to prove $u \prec\!\cdot \parallel D$ and $u \in \gamma(s) \setminus \{\varepsilon\}$ implies a) or b).

Now since u is a nonempty connected component of s it is (by corollary 1.1.8.f) a rooted tree-semiword. Hence $u = a.u'$ for some $u' \in TSW$. Since $u \neq \varepsilon$, $\varepsilon \in \gamma(s)$ and $\gamma(s) \setminus \{u\} = \gamma(t) \setminus D$ we have $\varepsilon \notin D$, so $\gamma(\parallel D) = D \uplus \{\varepsilon\}$. Then by proposition 1.3.42 we see $a.u' \prec\!\cdot \parallel D$ implies $|D| \leq 2$. Since $\varepsilon \notin D$, D must consist of nonempty connected components. By corollary 1.1.8.f) then $D = \{c.s', b.s''\}$ or $D = \{c.t'\}$ for some $b, c \in Act$, $s', s'', t' \in TSW$.

$D = \{c.t'\}$: Then $u \prec\!\cdot \parallel D$ reads $a.u' \prec\!\cdot\ c.t'$. Clearly then $a = c$ and $u' \prec\!\cdot\ t'$. Chose $s' = u'$.

$D = \{c.s', b.s''\}$: By proposition 1.3.30 we get w.l.o.g.: $a.u' \prec\!\cdot\ c.s' \parallel b.s'' \Rightarrow \exists v.\ a.v \preceq c.s', u' \preceq v \parallel b.s''$, $(a^1 \notin A_{b.s''})$. We examine the cases of $\preceq$.

$a.v = c.s', u' = v \parallel b.s''$: Then $a = c, v = s, u' = s' \parallel b.s'$ and $D = \{a.s', b.s''\}$.

$a.v \prec c.s', u' = v \parallel b.s''$: $a.v \prec c.s' \Rightarrow a = c, v \prec s'$. By proposition 1.3.29.a) we see $a.u' = a.(v \parallel b.s'') \prec a.v \parallel b.s''$ since $|\gamma(a.u')| = 2$, $|\gamma(a.v \parallel b.s'')| = 3 \Rightarrow \gamma(a.u') \neq \gamma(a.v \parallel b.s'') \Rightarrow a.u' \neq a.v \parallel b.s''$. Now $v \prec s'$ implies $a.v \parallel b.s'' \prec a.s' \parallel b.s'' = c.s' \parallel b.s'' = \parallel D$, so $a.u \prec a.v \parallel b.s'' \prec \parallel D$ which contradicts $u = a.u' \prec\!\cdot \parallel D$ wherefore this case can be ruled out.

$a.v = c.s', u' \prec v \parallel b.s''$: Then $a.u' \prec a.(v \parallel b.s'') \prec a.v \parallel b.s'' = c.s' \parallel b.s'' = \parallel D$. A contradiction.

$a.v \prec c.s', u' \prec v \parallel b.s''$: As the previous case.

□

Chapter 3

Semantics for a Simple Process Language: *PL*

In this chapter we shall give three different semantics to a simple process language, PL, for describing finite nondeterministic processes which in turn is a restricted subset of the basic language, BL, obtained as the term algebra for the signature Σ—essentially the operators (symbols) from the chapters with semiwords/ tree-semiwords. The restriction will be that processes only can be parallel composed when they have no action symbols in common. This restriction is mainly technical motivated, but can also be seen as reflecting the idea that an atomic action cannot be duplicated (however it may reinitiated). The restriction allows us to define the different interpretations of parallel composition of processes on the basis of the corresponding partial defined parallel composition of tree-semiwords.

3.1 Denotational Semantics

The concrete signature, Σ, from which BL is derived as the term algebra is:

Definition 3.1.1 Σ is defined by:

$$\begin{array}{lll} \Sigma_0 &= \{NIL\} & \\ \Sigma_1 &= \{a.\} & \text{where } a \in Act \\ \Sigma_2 &= \{+, \|\} & \\ \Sigma_n &= \emptyset & n > 3 \end{array}$$

□

Act is a set of abstract atomic action symbols fixed throughout the rest of this part.

Writing binary operators as usual as infixes and the unary as prefixes, BL can be considered defined from the following BNF-like schema:

$$p ::= NIL \mid a.p,\ a \in Act \mid p + p \mid p \parallel p$$

To formalize the restriction we shall impose on the processes we for every $p \in BL$ we define it's *sort*, $L(p)$, or label set as follows:

Definition 3.1.2 Let $L : BL \longrightarrow \mathcal{P}(Act)$ $(= L)$ be defined by:

$$\begin{aligned} NIL &\mapsto \emptyset \\ a.p &\mapsto \{a\} \cup L(p) \\ p + q &\mapsto L(p) \cup L(q) \\ p \parallel q &\mapsto L(p) \cup L(q) \end{aligned}$$

□

Whit this in the hand we can define the process language, PL, as those terms of BL where every subterm of the form:

$$p \parallel q$$

satisfies:

$$L(p) \cap L(q) = \emptyset$$

That is parallel composition is only allowed between processes with different sorts.

The next step will be to define the three interpretations of the terms from PL by means of corresponding Σ-po algebras as explained in [Hen85a].

However because of the restriction on PL some modifications are needed. Formally the semantics should be given within a theoretical framework which address the question of giving semantics to terms with certain sorts as e.g., in sorted algebras [GTWW77]. This would to the opinion of the author obscure the presentation unnecessarily, since these questions not are the main concern of this thesis. So under the conviction that the presentation easely (but lengthly) could be given within such a framework, we shall merely on the way state the most important changes which arrise.

Common to the carriers of the three Σ-po algebras is that they consists of closures of prefix-closed sets of tree-semiwords over Act (i.e., $\Delta = Act$). The differences between the carriers derive from the chosen closures which all are based on the smoother than relation $(\preceq)$ between single tree-semiwords. The three closures are δ, υ and χ respectively. We denote the three carriers by C_δ, C_υ and C_χ respectively. Formally:

Definition 3.1.3 For $\star$ in $\{\delta, \upsilon, \chi\}$ we define:

$\star)$ $C_\star := \{S \neq \emptyset \mid \exists T \subseteq TSW(Act).\, T \text{ is finite, } S = \star(\pi T)\}$

and call it the $\star$-carrier. □

It would have been nicer to define $C_\star$ as the finite $\star$ and π closed subsets of $TSW(Act)$ (TSW for short), but from proposition$_T$ 1.3.38 and the comments there we see that this only could be done for $\star = \delta$

In the sequel $\mathcal{P}_f(A) \subseteq \mathcal{P}(A)$ will denote the finite sets of the power set. With this notation we can read $C_\star$ as:

$$\{S \mid \exists T \in \mathcal{P}_f(TSW) \setminus \emptyset.\, S = \star(\pi T)\}$$

Corollary 3.1.4 For $\star$ in $\{\delta, \upsilon, \chi\}$ we have:

$\star$) $\forall T \in C_\star. \star(T) = T$

For each of these carriers we are going to define a interpretation $\Sigma\star$ of the symbols of the signature Σ as a function from ${C_\star}^n$ to $C_\star$ where n is the rank of the symbol in question. Most of the definitions of these functions will lean on the corresponding functions defined on single tree-semiwords and the $\star$-closure properties.

Definition 3.1.5 The sort of a nonempty set of tree-semiwords, S, ambiguously denoted $L(S)$, is defined by:

$$\begin{aligned} L(\{s\}) &\mapsto \{a \mid a^i \in A_s\} \ (= \{a \mid \psi(s,a) > 0\}) \\ L(S \cup T) &\mapsto L(S) \cup L(T) \end{aligned}$$

□

If S is a singleton set $\{s\}$ we often just write $L(s)$ in place of $L(\{s\})$, so L can be considered defined on TSW also. Notice that because tree-semiwords satisfies $SW1$ we have for arbitrary tree-semiwords s and t: $A_s \cap A_t = \emptyset$ *iff* $L(s) \cap L(t) = \emptyset$. I.e., s and t are disjoint *iff* there sorts are disjoint. Also remark that $L(\varepsilon) = \emptyset$.

For each carrier, $C_\star$, the function, $\|_\star$, corresponding to the interpretation of $\|$ will then be partially defined: $S \|_\star T$ is only defined when $L(S) \cap L(T) = \emptyset$. But due to the restriction on terms from PL it will be ensured that the interpretations are defined.

We are now ready to define the interpretations of the operator symbols.

Definition 3.1.6 With S and T considered to be elements of the appropriate $C_\star$-carrier we define:

$$\begin{array}{llcll} \Sigma_\delta : & NIL_\delta & = & \{\varepsilon\} & \\ & a._\delta S & = & a.S \cup \{\varepsilon\} & \\ & S +_\delta T & = & S \cup T & \\ & S \|_\delta T & = & \delta(S \| T) & \text{provided } L(S) \cap L(T) = \emptyset \\ & & & & \\ \Sigma_\upsilon : & NIL_\upsilon & = & \{\varepsilon\} & \\ & a._\upsilon S & = & \upsilon(a.S) \cup \{\varepsilon\} & \\ & S +_\upsilon T & = & S \cup T & \\ & S \|_\upsilon T & = & S \| T & \text{provided } L(S) \cap L(T) = \emptyset \\ & & & & \\ \Sigma_\chi : & NIL_\chi & = & \{\varepsilon\} & \\ & a._\chi S & = & a.S \cup \{\varepsilon\} & \\ & S +_\chi T & = & \chi(S \cup T) & \\ & S \|_\chi T & = & \chi(S \| T) & \text{provided } L(S) \cap L(T) = \emptyset \end{array}$$

The result of $S +_\star T$, S, T in $C_\star$, $\star$ in $\{\delta, \upsilon\}$ is easely seen to be a member of $C_\star$ since δ and υ distributes over $\cup$ for arbitrary sets. But χ does in general not distribute over $\cup$ for arbitrary sets, not even for χ-closed sets, as can be seen from the following example.

Example: Let $S = \{a \rightarrow b \rightarrow c\}$, $T = \begin{Bmatrix} a \\ b \\ c \end{Bmatrix}$. Then $\chi(S) = S$, $\chi(T) = T$ and $s = a \begin{smallmatrix} \nearrow b \\ \searrow c \end{smallmatrix} \in \chi(S \cup T)$, but $s \notin S \cup T$.

Therefore we define $S +_\chi T$ to be $\chi(S \cup T)$.

Proposition 3.1.7 The operators of $\Sigma_\star$, $\star$ in $\{\delta, \upsilon, \chi\}$ are well-defined. I.e., they are functions from ${C_\star}^n$ to $C_\star$ for every $\star$.

Proof Notice at first that for all $op_\star \in \Sigma_{\star n}$ and $\star$ in $\{\delta, \upsilon, \chi\}$ $op_\star$ is defined on ${C_\star}^n$ under proviso. What remains is to find a $U \in \mathcal{P}_f(TSW)$ for every $op_\star \in \Sigma_{\star_n}$ and $\bar{S} \in {C_\star}^n$ such that $op_\star(\bar{S}) = \star(\pi U)$, because then $op_\star(\bar{S}) \in C_\star$.

$NIL_\star$: Let $U = \{\varepsilon\}$. For every $\star$-closure we have: $\star(\pi U) = \star(\pi\{\varepsilon\}) = \star(\{\varepsilon\}) = \{\varepsilon\} = NIL_\star$.

$a._\star S$: $S \in C_\star$ implies there exists a $S' \in \mathcal{P}_f(TSW)$ such that $S = \star(\pi S')$. Let $U = a.S' \in \mathcal{P}_f(TSW)$.

- $\star = \delta$: $\delta\pi U = \delta\pi a.S' =$ (corollary$_T$ 1.3.36) $\delta(a.\pi S' \cup \{\varepsilon\}) = \delta a.\pi S' \cup \{\varepsilon\} =$ (corollary$_T$ 1.3.16) $a.\delta\pi S' \cup \{\varepsilon\} = a.S \cup \{\varepsilon\} = a._\delta S$.
- $\star = \upsilon$: $\upsilon\pi U = \ldots = \upsilon(a.\pi S' \cup \{\varepsilon\})$ (corollary$_T$ 1.3.19) $\upsilon a.\upsilon\pi S' \cup \{\varepsilon\} = \upsilon a.S \cup \{\varepsilon\} = a._\upsilon S$.
- $\star = \chi$: $\chi\pi U = \ldots = \chi(a.\pi S' \cup \{\varepsilon\}) =$ (proposition$_T$ 1.3.24) $\chi a.\pi S' \cup \{\varepsilon\} =$ (corollary$_T$ 1.3.25) $a.\chi\pi S' \cup \{\varepsilon\} = a.S \cup \{\varepsilon\} = a._\chi S$.

$S +_\star T$: $S, T \in C_\star$ implies $\exists S', T' \in \mathcal{P}_f(TSW).\, S = \star(\pi S'), T = \star(\pi T')$. Let $U = S' \cup T'$.

- $\star$ in $\{\delta, \upsilon\}$: Since π, δ and υ distributes over $\cup$ the result is immediate.
- $\star = \chi$: $\chi\pi(S' \cup T') = \chi(\pi S' \cup \pi T') =$ (corollary$_T$ 1.3.23) $\chi(\chi\pi S' \cup \chi\pi T') = \chi(S \cup T) = S +_\chi T$.

$S \parallel_\star T$: Suppose S and T are disjoint. Furthermore let S' and T' be as in the case $S +_\star T$ and let $U = S' \parallel T'$.

- $\star = \delta$: $\delta\pi U = \delta\pi(S' \parallel T') =$ (proposition$_T$ 1.3.35) $\delta(\pi S' \parallel \pi T') =$ (corollary$_T$ 1.3.16) $\delta(\delta\pi S' \parallel \delta\pi T') = \delta(S \parallel T) = S \parallel_\delta T$.
- $\star = \upsilon$: $\upsilon\pi U = \ldots = \upsilon(\pi S' \parallel \pi T') =$ (proposition$_T$ 1.3.18) $\upsilon\pi S' \parallel \upsilon\pi T' = S \parallel T = S \parallel_\upsilon T$.
- $\star = \chi$: $\chi\pi U = \ldots = \chi(\pi S' \parallel \pi T') =$ (corollary$_T$ 1.3.23) $\chi(\chi\pi S' \parallel \chi\pi T') = \chi(S \parallel T) = S \parallel_\chi T$.

□

We now introduce a very simple Σ-po algebra A_π which in this and later chapters will prove useful in establishing properties of the $A_\star$-algebras (based $C_\star$) we are going to introduce in a moment..

Definition 3.1.8 Let $C_\pi := \mathcal{P}_f(TSW) \setminus \emptyset$ and for $S, T \in C_\pi$ define:

$$\Sigma_\pi : \quad \begin{aligned} NIL_\pi &= \{\varepsilon\} \\ a._\pi S &= a.S \cup \{\varepsilon\} \\ S +_\pi T &= S \cup T \\ S \parallel_\pi T &= S \parallel T \qquad \text{provided } L(S) \cap L(T) = \emptyset \end{aligned}$$

□

Clearly the operators of Σ_π are well-defined and monotone w.r.t. $\subseteq$, so $A_\pi = (C_\pi, \trianglelefteq_\pi, \Sigma_\pi)$, where $\trianglelefteq_\pi = \subseteq$, is indeed a Σ-po algebra. Of course $\trianglelefteq_\pi$-monotonicity of $\parallel_\pi$ is relative to the carrier upon which $\parallel_\pi$ is defined. That is $\parallel_\pi$ is e.g., left $\trianglelefteq_\pi$-monotone in the sense that for $S, S', T \in C_\pi$ and $L(S) \cap L(T) = \emptyset = L(S') \cap L(T)$ we have:

$$S \trianglelefteq_\pi S' \text{ implies } S \parallel_\pi T \trianglelefteq_\pi S' \parallel_\pi T$$

Monotonicity for parallel composition under this proviso will be indicated by writing: (relative) monotone.

Also $C_\delta, C_\upsilon, C_\chi \subseteq C_\pi$ wherefore we can formulate the following proposition which displays the close connection between operators of Σ_π and $\Sigma_\star$.

Proposition 3.1.9 Let $\star$ be in $\{\delta, \upsilon, \chi\}$. For all $op_\star \in \Sigma_{\star_n}$, $\bar{S} \in {C_\star}^n$ we have:

$$op_\star(\bar{S}) = \star op_\pi(\bar{S})$$

Proof In most of the operator cases we use corollary 3.1.4.

$NIL_\star$: Evident since $NIL_\pi = \{\varepsilon\}$ and $\delta(\varepsilon) = \upsilon(\varepsilon) = \chi(\varepsilon) = \{\varepsilon\}$.

$a._\delta$: $a._\delta S = a.S \cup \{\varepsilon\} = a.\delta S \cup \{\varepsilon\} = \delta a.S \cup \{\varepsilon\} = \delta(a.S \cup \{\varepsilon\}) = \delta a._\pi S$.

$a._\upsilon$: $a._\upsilon S = \upsilon a.S \cup \{\varepsilon\} = \upsilon(a.S \cup \{\varepsilon\}) = \upsilon a._\pi S$.

$a._\chi$: $a._\chi S = a.S \cup \{\varepsilon\} = a.\chi S \cup \{\varepsilon\} =$ (corollary$_T$ 1.3.25) $\chi a.S \cup \{\varepsilon\} =$ (proposition$_T$ 1.3.24) $\chi(a.S \cup \{\varepsilon\}) = \chi a._\pi S$.

$+_\delta$: $S +_\delta T = S \cup T = \delta S \cup \delta T = \delta(S \cup T) = \delta(S +_\pi T)$.

$+_\upsilon$: Similar.

$+_\chi$: $S +_\chi T = \chi(S \cup T) = \chi(S +_\pi T)$.

$\parallel_\delta$: $S \parallel_\delta T = \delta(S \parallel T) = \delta(S \parallel_\pi T)$.

$\parallel_\upsilon$: $S \parallel_\upsilon T = S \parallel T = \upsilon S \parallel \upsilon T = \upsilon(S \parallel T) = \upsilon(S \parallel_\pi T)$.

$\parallel_\chi$: As $S \parallel_\delta T$.

□

The next to define is the partial order $\trianglelefteq_\star$ on $C_\star$.

Definition 3.1.10 For every $\star$ in $\{\delta, \upsilon, \chi\}$ define the $\trianglelefteq_\star$—the partial order over $C_\star$—to be the set inclusion ($\subseteq$). I.e.,

$$\forall S, T \in C_\star.\, S \trianglelefteq_\star T \textit{ iff } S \subseteq T$$

$\square$

Clearly it is a partial order and $\{\varepsilon\}$ is a least element in every $C_\star$.

Since the δ-, υ- and χ-closures in general are monotone w.r.t. $\subseteq$ we immediately from proposition 3.1.9 and op_π being monotone get:

Corollary 3.1.11 All $op_\star \in \Sigma_\star$ are (relative) monotone on $C_\star$ (w.r.t. $\trianglelefteq_\star$) for all $\star$ in $\{\delta, \upsilon, \chi\}$ (with the modification that $S \parallel_\star T$ only is defined when $L(S) \cap L(T) = \emptyset$).

From the preceding and this corollary we then also have:

Corollary 3.1.12 For every $\star$ in $\{\delta, \upsilon, \chi\}$ $A_\star = (C_\star, \trianglelefteq_\star, \Sigma_\star)$ is a Σ-po algebra.

Our different models, $M_\star$, then consists of these Σ-po algebras and denotational maps, $[\![\_]\!]_\star$ given below:

Definition 3.1.13 The interpretation, $[\![\_]\!]_\star$, in the $M_\star$ model of terms from PL is defined compositionally (on the basis of $A_\star$) as follows:

$$\begin{aligned}
[\![NIL]\!]_\star &= NIL_\star \\
[\![a.p]\!]_\star &= a._\star [\![p]\!]_\star \\
[\![p + q]\!]_\star &= [\![p]\!]_\star +_\star [\![q]\!]_\star \\
[\![p \parallel q]\!]_\star &= [\![p]\!]_\star \parallel_\star [\![q]\!]_\star
\end{aligned}$$

$\square$

From $L(p) = L([\![p]\!]_\star)$ and $p \parallel q \in PL$ only if $L(p) \cap L(q) = \emptyset$ it is seen that the definition is well-defined.

3.2 Operational Semantics

The operational semantics we are going to define are based on a labelled transition system (lts for short) which determines a process's ability to develop from one configuration to another. For the purpose of this we define the set of possible configurations. In the definition, actions of a set of atomic complementary action symbols, $\overline{Act}$, disjoint but equipotent to Act is used. Furthermore a bijective map $\bar{\ } : Act \longrightarrow \overline{Act}$.

Definition 3.2.1 $\overline{BL}$ is defined to be the least set C which satisfies:

$$\begin{array}{rl} BL \subseteq C & \\ \bar{a}.p \in C & \text{if } p \in C \text{ and } a \in Act \\ p_1 \parallel p_2 \in C & \text{if } p_1, p_2 \in C \end{array}$$

L is extended to $\overline{BL}$ by: $L(\bar{a}.q) = \{a\} \cup L(q)$.

The *configuration language*, CL, is defined to be the subset of $\overline{BL}$ where every subterm of the form $p \parallel q$ has $L(p) \cap L(q) = \emptyset$. □

Notice

i) L "forgets" whether a label belongs to Act or $\overline{Act}$. So $L(p)$ for $p \in CL$ could be defined as taking L on $p' \in PL$, where p' is p with all $^-$'s striped of.

ii) $PL \subseteq CL$ and $p + q \in CL$ only if $p, q \in PL$.

□

In the sequel we will have the implicit requirements $L(p) \cap L(q) = \emptyset$ whenever writing $p \parallel q$.

What remains to define for the lts over CL and $Act \cup \overline{Act}$ is the action relation.

Definition 3.2.2 Let $\longrightarrow \subseteq CL \times (Act \cup \overline{Act}) \times CL$ (writing $p \stackrel{y}{\longrightarrow} q$ for $(p, y, q) \in \longrightarrow$) be the least relation over CL which satisfies:

1) $a.p \stackrel{a}{\longrightarrow} \bar{a}.p$ 2) $\bar{a}.p \stackrel{\bar{a}}{\longrightarrow} p$

3) $$\frac{p \stackrel{b}{\longrightarrow} p'}{\bar{a}.p \stackrel{b}{\longrightarrow} \bar{a}.p'}$$

4) $$\frac{p \stackrel{a}{\longrightarrow} p'}{\begin{array}{c} p + q \stackrel{a}{\longrightarrow} p' \\ q + p \stackrel{a}{\longrightarrow} p' \end{array}}$$ 5) $$\frac{p \stackrel{y}{\longrightarrow} p',\ y \in Act \cup \overline{Act}}{\begin{array}{c} p \parallel q \stackrel{y}{\longrightarrow} p' \parallel q \\ q \parallel p \stackrel{y}{\longrightarrow} q \parallel p' \end{array}}$$

where $y \in Act \cup \overline{Act}$ and $a, b, \ldots$ range over Act.

Corollary 3.2.3 $p \stackrel{y}{\longrightarrow} p' \Rightarrow L(y) \subseteq L(p),\ L(p') \subseteq L(p)$.

The fact that $p \stackrel{y}{\longrightarrow} p'$ implies $L(p') \subseteq L(p)$ gives the well-definedness of the relation since then p, q disjoint implies p', q disjoint too in 5).

Proposition 3.2.4 $\forall a \in Act \forall p \in CL.\ |\{q \mid p \stackrel{\bar{a}}{\longrightarrow} q\}| \leq 1$

Proof Induction on the structure of p.

$p \in PL$: Then no subterm r of p is of the form $\bar{b}.r'$, $b \in Act$. By inspection of definition 3.2.2 we clearly have $\{q \mid p \xrightarrow{\bar{a}} q\} = \emptyset$.

$p = \bar{b}.p'$: Two cases: $a \neq b$: Again by inspection of the definition we see $\bar{b}.p' \stackrel{\bar{a}}{\not\longrightarrow}$ or equivalently $\{q \mid p = \bar{b}.p' \xrightarrow{\bar{a}} q\} = \emptyset$. $a = b$: We see $\bar{a}.p' \xrightarrow{\bar{a}} q$ implies $q = p'$.

$p = p_1 \parallel p_2$: By inspection and the disjointness of p_1 and p_2 we see $p = p_1 \| p_2 \xrightarrow{\bar{a}} q$ implies that exactly one of the two cases $p_1 \xrightarrow{\bar{a}} p_1', q = p_1' \parallel p_2$ or $p_2 \xrightarrow{\bar{a}} p_2', q = p_1 \parallel p_2'$ hold, so the cardinality of $\{q \mid p \xrightarrow{\bar{a}} q\}$ is equal to the cardinality of $\{p' \mid p_1 \xrightarrow{\bar{a}} p'\}$ in the former case and $\{p' \mid p_2 \xrightarrow{\bar{a}} p'\}$ in the latter. By the inductive hypothesis these will be less than or equal to one.

□

Intuitively one can think of the $a.p$ as the process which can be signaled to initiate action a and thereby transforming to $\bar{a}.p$. This term again represents a process which contains an action a signaled to initiate and which can signal it's completion by transforming by $\bar{a}$ into p. The inference rule 3) says that more actions can be signaled to initiate before earlier signaled actions them selfs signal there completion. The term $p + q$ represents the process which can act either as p or q, and $p \parallel q$ represents the process which can act both as p and as q, so actions of one subprocess can be signaled to initiate or complete independent of the other.

Example: $$\begin{aligned} a.NIL + b.(a.NIL \parallel b.NIL) &\xrightarrow{b} \bar{b}.(a.NIL \parallel b.NIL) \\ &\xrightarrow{a} \bar{b}.(\bar{a}.NIL \parallel b.NIL) \\ &\xrightarrow{b} \bar{b}.(\bar{a}.NIL \parallel \bar{b}.NIL) \\ &\xrightarrow{\bar{b}} \bar{a}.NIL \parallel \bar{b}.NIL \\ &\xrightarrow{\bar{a}} NIL \parallel \bar{b}.NIL \\ &\xrightarrow{\bar{b}} NIL \parallel NIL \end{aligned}$$

We extend the (atomic) action relation to strings over $Act \cup \overline{Act}$ by:

$$p \xrightarrow{z} p' \textit{ iff } \begin{cases} z = \varepsilon,\ p' = p \\ \text{or} \\ z = az',\ p \xrightarrow{a} p'',\ p'' \xrightarrow{z'} p' \end{cases}$$

where $z \in (Act \cup \overline{Act})^*$.
On the basis of this and the notion of experiment we define how two processes are operational semantically related.

We consider two statements about a process p and an experiment e:

- p may accept e
- p may reject e

The relations can then be defined as follows:

Definition 3.2.5 For processes $p, q \in PL$ we define

$p \sqsubseteq_{\mathrm{a}} q$ *iff* for all experiments e: p may accept e implies q may accept e
$p \sqsubseteq_{\mathrm{r}} q$ *iff* for all experiments e: p may reject e implies q may reject e
$p \sqsubseteq q$ *iff* $p \sqsubseteq_{\mathrm{a}} q$ and $p \sqsubseteq_{\mathrm{r}} q$

□

The next thing to consider is which experiments we will allow and when a process may accept/ reject an experiment.
An *experiment*, e, will be split out into two. First a set of actions A are signaled to initiate and second a test t is done on these. So

an experiment, e, is a pair: (A, t).

In fact the signaled set of actions can be considered as a *multiset* over Act because we want to be able to signal the same action more than once. If A is a finite multiset over Act and $a \in Act$ let $|A|_a$ denote the number of a's in A. For a tree-semiword s and such a multiset A we write:

$$A_s \cong A \textit{ iff } \forall a \in Act.\, |A|_a = \psi(s, a)$$

For a multiset A and a process $p \in PL$ the set of possible configurations we can obtain by signaling A is $D(A, p)$ defined as follows:

Definition 3.2.6 $D(A, p) := \{p' \in CL \mid \exists w \in W.\, A_w \cong A,\ p \xrightarrow{w} p'\}$ □

Recall that W and Act^* are isomorphic, so it gives sense to write $p \xrightarrow{w} p'$ for a $w \in W$. Notice that nondeterministic choices are made when signaling A to initiate.

The next to decide is the test language TL. We will only allow tests on the actions which are signaled to initiate, so the language must be based on $\overline{Act}$. It shall be possible to test the order in which the process can signal completions of the actions previously signaled to initiate, so if t is a test $\bar{a}.t$ is a test too. If p is a configuration the test $t \,\&\, t'$ denotes the test whether *both* the test t *and* t' are possible on p. Similar $t \,\nabla\, t'$ is the test whether *either* t *or* t' are possible. A test is ended with $\top$ to notify that the test was possible.

Definition 3.2.7 The *test language*, TL, is defined by the schema

$$t ::= \top \mid \bar{a}.t,\ a \in Act \mid t \,\&\, t \mid t \,\nabla\, t$$

□

Observe that for a test like $t \,\&\, t'$ or $t \,\nabla\, t'$ there is no restrictions on the sorts of t and t'.

$\top$—the successful test—is one of the two possible outcomes of a test. The other—the unsuccessful test—is denoted $\bot$. When having a test like $t \,\&\, t'$ one subtest can turn out to be successful and the other unsuccessful, so during the total test, subconfigurations like $\top \,\&\, \bot$ are possible.

Definition 3.2.8 The *test configurations*, TC, are defined by the schema:

$$o ::= (t,p),\ t \in TL,\ p \in CL \mid \top \mid \bot \mid o \,\&\, o \mid o \,\nabla\, o$$

□

A test is finished when it is known to be successful or unsuccessful, i.e., when one of the test configurations $\top$ or $\bot$ are reached. The relation between the different test configurations is determined by the *test relation* $\rightarrow\ \subseteq TC \times TC$ defined below.

Definition 3.2.9 Let $p, p' \in CL$, $t \in TL$, $o, o', o'' \in TC$ in the following.

Axioms:

1) $\Box$: $(t \,\Box\, t', p) \rightarrow (t,p) \,\Box\, (t',p)$ for $\Box \in \{\&, \nabla\}$

2) $(\top, p) \rightarrow \top$

3)

$$\begin{array}{llll}
\&\top: & o \,\&\, \top \rightarrow o & \top\&: & \top \,\&\, o \rightarrow o \\
\&\bot: & o \,\&\, \bot \rightarrow \bot & \bot\&: & \bot \,\&\, o \rightarrow \bot \\
\nabla\top: & o \,\nabla\, \top \rightarrow \top & \top\nabla: & \top \,\nabla\, o \rightarrow \top \\
\nabla\bot: & o \,\nabla\, \bot \rightarrow o & \bot\nabla: & \bot \,\nabla\, o \rightarrow o
\end{array}$$

Inferences:

4) $\Box$: $$\frac{o \rightarrow o'}{\begin{array}{c} o \,\Box\, o'' \rightarrow o' \,\Box\, o'' \\ o'' \,\Box\, o \rightarrow o'' \,\Box\, o' \end{array}}$$ for $\Box \in \{\&, \nabla\}$

5) $$\frac{p \xrightarrow{\bar{a}} p',\ a \in Act}{(\bar{a}.t, p) \rightarrow (t, p')}$$

6) $$\frac{p \stackrel{\bar{a}}{\not\longrightarrow},\ a \in Act}{(\bar{a}.t, p) \rightarrow \bot}$$

$p \stackrel{\bar{a}}{\not\longrightarrow}$ is just a shorthand notation for $\nexists p' : p \xrightarrow{\bar{a}} p'$.

Example:

$(\bar{a}.\bar{b}.\top \,\&\, \bar{b}.\bar{a}.\top, \bar{a}.NIL \parallel \bar{b}.(c.NIL + d.NIL))$
$\rightarrow (\bar{a}.\bar{b}.\top, \bar{a}.NIL \parallel \bar{b}.(c.NIL + d.NIL)) \,\&\, (\bar{b}.\bar{a}.\top, \bar{a}.NIL \parallel \bar{b}.(c.NIL + d.NIL))$
$\rightarrow (\bar{b}.\top, NIL \parallel \bar{b}.(c.NIL + d.NIL)) \,\&\, (\bar{b}.\bar{a}.\top, \bar{a}.NIL \parallel \bar{b}.(c.NIL + d.NIL))$
$\rightarrow (\bar{b}.\top, NIL \parallel \bar{b}.(c.NIL + d.NIL)) \,\&\, (\bar{a}.\top, \bar{a}.NIL \parallel (c.NIL + d.NIL))$
$\rightarrow (\top, NIL \parallel (c.NIL + d.NIL)) \,\&\, (\bar{a}.\top, \bar{a}.NIL \parallel (c.NIL + d.NIL))$
$\rightarrow \top \,\&\, (\bar{a}.\top, \bar{a}.NIL \parallel (c.NIL + d.NIL))$
$\rightarrow (\bar{a}.\top, \bar{a}.NIL \parallel (c.NIL + d.NIL))$
$\rightarrow (\top, NIL \parallel (c.NIL + d.NIL))$
$\rightarrow \top$

Notice that this only is one of many possible derivation that leads $\top$.

A test configuration o is called *terminal iff* $o \not\rightarrow$ (i.e., $\nexists o' \in TC.\, o \rightarrow o'$). In a moment we will show that the only possible terminal test configuration is exactly one of $\top$ and $\bot$, such that a test is either successful or unsuccessful, and cannot be both. In this sense our notion of test is well-defined.

The fact that the terminal configurations are $\{\top, \bot\}$ and that one and only one of these can be reached from a test configuration, o, has as consequence:

$$\forall p \in CL \forall t \in TL.\, (t, p) \rightarrow^* \top \Leftrightarrow (t, p) \not\rightarrow^* \bot$$

An experiment e can now be considered as (A, t), where A is the multiset over Act, which determines the actions that should be signaled to initiate, such that a test can be run on them, and t is the actual test to run.

Informally a process p may accept the experiment $e = (A, t)$ if

a) It gives sense to run the test, i.e., the actions of A can be signaled.

b) One of the processes p' obtainable from p by signaling A to initiate, pass the test t successfully.

Similar p may reject (A, t) if under the same conditions as above one of the obtainable processes p' pass the the test t unsuccessfully. Notice that we may have a process p and experiment e such that p may accept e and p may reject e! Also notice that the two statements are not dual. I.e., we do not have p ~~may~~ accept e implies p may reject e (where p ~~may~~ accept e means it is not the case that p may accept e). This is because the reason why p ~~may~~ accept e can be that it does not make sense to run the test t, in which case we have p ~~may~~ reject e too. Formally:

Definition 3.2.10 Denote the set of experiments by E. I.e., $e \in E$ *iff* $e = (A, t)$, A is a finite multiset over Act and $t \in TL$.

Let $p \in PL$ and $e = (A, t)$ be an experiment. Then:

a) p may accept e *iff* $\exists q \in D(A, p).\, (t, q) \rightarrow^* \top$

b) p may reject e *iff* $\exists q \in D(A, p).\, (t, q) \rightarrow^* \bot$

□

Example:
$a.b.NIL + b.a.NIL + a.NIL \parallel b.NIL$ may accept $(\{a, b\}, \bar{a}.\bar{b}.\top \,\&\, \bar{b}.\bar{a}.\top)$

$a.b.NIL + b.a.NIL$ ~~may~~ accept $(\{a, b\}, \bar{a}.\bar{b}.\top \,\&\, \bar{b}.\bar{a}.\top)$

$a.b.NIL + b.a.NIL + a.NIL \parallel b.NIL$ may reject $(\{a, b\}, \bar{a}.\bar{b}.\top \,\nabla\, (\bar{b}.\bar{a}.\top \,\&\, \bar{a}.\bar{b}.\top))$

$a.NIL \parallel b.NIL$ ~~may~~ reject $(\{a, b\}, \bar{a}.\bar{b}.\top \,\nabla\, (\bar{b}.\bar{a}.\top \,\&\, \bar{a}.\bar{b}.\top))$

So we have now formally defined what was used in the definition of the three testing preorders $\sqsubseteq_{\mathrm{a}}$, $\sqsubseteq_{\mathrm{r}}$ and $\sqsubseteq$ on PL. In the following $\mathrel{\underset{\sim}{\sqsupset\!\sqsubset}}$ shall denote the equivalence of $\sqsubseteq$. Similar for the other preorders.

Example:

$$a.b.NIL + b.a.NIL \begin{Bmatrix} \sqsubseteq_{\mathrm{a}} \\ \sqsupseteq_{\mathrm{r}} \end{Bmatrix} a.b.NIL + b.a.NIL + a.NIL \parallel b.NIL \begin{Bmatrix} \sqsupseteq\sqsubseteq_{\mathrm{a}} \\ \sqsupseteq_{\mathrm{r}} \end{Bmatrix} a.NIL \parallel b.NIL$$

$$a.NIL \begin{Bmatrix} \sqsubseteq_{\mathrm{a}} \\ \sqsubseteq_{\mathrm{r}} \\ \sqsubseteq \end{Bmatrix} a.b.NIL$$

$$a.b.c.NIL + a.(b.NIL \parallel c.NIL) + a.NIL \parallel b.c.NIL \ \sqsupseteq\sqsubseteq \ a.b.c.NIL + a.NIL \parallel b.c.NIL$$

As indicated by:

$$a.(b.NIL + c.NIL) \begin{Bmatrix} \sqsupseteq\sqsubseteq_{\mathrm{a}} \\ \sqsupseteq\sqsubseteq_{\mathrm{r}} \\ \sqsupseteq\sqsubseteq \end{Bmatrix} a.b.NIL + a.c.NIL$$

non of the equivalences are able to distinguish nondeterminism.

That p may accept e and p may reject e are not dual can now also formally easely be seen:

$$\begin{aligned} \neg(\exists q \in D(A,p).\,(t,q) \rightarrow^* \top) \quad &\textit{iff} \quad \forall q \in D(A,p)(t,q) \not\rightarrow^* \top \\ &\textit{iff} \quad \forall q \in D(A,p).\,(t,q) \rightarrow^* \bot \\ &\not{\textit{iff}} \quad \exists q \in D(A,p).\,(t,q) \rightarrow^* \bot \end{aligned}$$

Before we give the promised proof of one and only one terminal configuration for every test configuration we prove the following lemma, which also clears the rôle of & and ∇.

Lemma 3.2.11

a) $o_1 \,\&\, o_2 \rightarrow^* \top$ *iff* $o_1 \rightarrow^* \top$ and $o_2 \rightarrow^* \top$

b) $o_1 \,\&\, o_2 \rightarrow^* \bot$ *iff* $o_1 \rightarrow^* \bot$ or $o_2 \rightarrow^* \bot$

c) $o_1 \nabla o_2 \rightarrow^* \top$ *iff* $o_1 \rightarrow^* \top$ or $o_2 \rightarrow^* \top$

d) $o_1 \nabla o_2 \rightarrow^* \bot$ *iff* $o_1 \rightarrow^* \bot$ and $o_2 \rightarrow^* \bot$

Proof

a) *only if*: We prove it by proving $o_1 \,\&\, o_2 \rightarrow^n \top$ only if $o_1 \rightarrow^* \top \wedge o_2 \rightarrow^* \top$ by induction on n ($n = 0$ impossible because $\top$ is not of the form $t \,\&\, t'$).

$n = 1$: By inspection of definition 3.2.9 we see $o_1 \,\&\, o_2 \rightarrow \top$ implies $o_1 = \top = o_2$. Then of course also $o_1 \rightarrow^* \top$ and $o_2 \rightarrow^* \top$.

$n > 1$: $o_1 \& o_2 \rightarrow o, o \rightarrow^{n-1} \top$. Looking at the definition again we see that only 3) $\&\top, \top\&$ or 4) & can come under discussion for the move $o_1 \,\&\, o_2 \rightarrow o$.

3) $\& \top, \top\&$: W.l.o.g. assume $o_1 \,\&\, o_2 \rightarrow o$ is $o_1 \,\&\, \top \rightarrow o_1$. Then clearly $o_2 \rightarrow^* \top$ and $o_1 \rightarrow^{n-1} \top$ which implies $o_1 \rightarrow^* \top$.

4) $\&$: Two cases: $o_1 \rightarrow o_1', o = o_1' \,\&\, o_2$ and $o_2 \rightarrow o_2', o = o_1 \,\&\, o_2'$.
For the former then $o \rightarrow^{n-1} \top$ means $o_1' \,\&\, o_2 \rightarrow^{n-1} \top$. By hypothesis of induction $o_1' \rightarrow^* \top \wedge o_2 \rightarrow^* \top$. $o_1 \rightarrow o_1', o_1' \rightarrow^* \top$ implies $o_1 \rightarrow^* \top$, so we are done for this case.
The latter is handled symmetric. This also completes the inductive step.

if: It is enough to prove $o_1 \rightarrow^n \top \wedge o_2 \rightarrow^m \top$ implies $o_1 \,\&\, o_2 \rightarrow^* \top$ by induction on $n+m$.

$n+m=0$: Then $o_1 = \top = o_2$. No matter whether we use 3) $\& \top$ or 3) $\top\&$ we get $o_1 \,\&\, o_2 \rightarrow \top$ and thereby $o_1 \,\&\, o_2 \rightarrow^* \top$.

$n+m>0$: We split out in two subcases:
$m>0$: This implies $o_2 \rightarrow o_2' \rightarrow^{m-1} \top$. By hypothesis of induction $o_1 \,\&\, o_2' \rightarrow^* \top$. Using 4)& we now get $o_1 \,\&\, o_2 \rightarrow o_1 \,\&\, o_2'$, hence $o_1 \,\&\, o_2 \rightarrow^* \top$.
$n>0$: Then $o_1 \rightarrow o_1' \rightarrow^{n-1} \top$. Similar to the case $n=0$.

b) -d): Similar to a). □

Now for an $o \in TC$ let $B(o)$ denote the set of terminal configurations i.e.,

$$B(o) := \{o' \in TC \mid o \rightarrow^* o', o' \not\rightarrow\}$$

So what we shall prove is that $\{\top, \bot\}$ equals the terminal configurations and $|B(o)| = 1$ which follows from the proposition below.

Proposition 3.2.12 Let $o \in TC$. Then

a) $\{\top, \bot\}$ = terminal configurations   b) $o \rightarrow^* \top \,\dot{\vee}\, o \rightarrow^* \bot$

where $\dot{\vee}$ means that exactly one of the possibilities are true.

Proof
a) By inspection of definition 3.2.9 we easely see that $\top$ and $\bot$ are the only possible terminal configurations.

b) We split the proof out in two, i) $o \rightarrow^* \top \vee o \rightarrow^* \bot$ and ii) $o \rightarrow^* o', o' \in \{\top, \bot\} \Rightarrow B(o) = \{o'\}$, from which b) can be seen.

i) We prove $o \rightarrow^* \top \vee o \rightarrow^* \bot$ by induction on the structure of o.

$o = \top$ or $o = \bot$ (basis): Immediate.

$o = o_1 \,\&\, o_2$: By hypothesis of induction $o_i \rightarrow^* \top \vee o_i \rightarrow^* \bot$ for $i \in \underline{2}$. Four cases:
$o_1 \rightarrow^* \top, o_2 \rightarrow^* \top$: Using the *if* part of lemma 3.2.11 a) we get $o = o_1 \,\&\, o_2 \rightarrow^* \top$.
In the three other cases we get $o \rightarrow^* \bot$ using the *if* part of b) in the lemma.

$o = o_1 \nabla o_2$: Similar.

$o = (t, p), t \in TL, P \in CL$: To prove this part of the inductive step we use induction on the structure of t.

$t = \top$: 2) of definition 3.2.9 gives us $o = (t, p) = (\top, p) \to \top$ hence $o \to^* \top$.

$t = \bar{a}.t'$: Clearly either 5) and 6) can be used. In the latter case we directly get $o \to^* \bot$. In the former we have $(\bar{a}.t', p) \to (t', p')$. By hypothesis of induction either $(t', p') \to^* \top$ or $(t', p') \to^* \bot$ and the result follows.

$t = t_1 \,\&\, t_2$: Using 1)& we get $o = (t_1 \,\&\, t_2, p) \to (t_1, p) \,\&\, (t_2, p)$. By the hypothesis of induction $(t_i, p) \to^* \top \vee (t_i, p) \to^* \bot$ for $i \in \underline{2}$. Similar as in the case $o = o_1 \,\&\, o_2$ we get $(t_1, p) \,\&\, (t_2, p) \to^* \top \vee (t_1, p) \,\&\, (t_2, p) \to^* \bot$. Hence also $o \to^* \top \vee o \to^* \bot$.

$t = t_1 \nabla t_2$: Similar.

ii) Since $o' \in \{\top, \bot\}$ we can part the proof in two:

$o' = \top$: Then we have to prove $o \to^* \top \Rightarrow B(o) = \{\top\}$. We will do this by induction on the structure of o. Since $\top \not\to$ and $o \to^* \top$ implies $\top \in B(o)$ we only need to prove $o \to^* \top \Rightarrow B(o) \subseteq \{\top\}$.

$o = \top$ or $o = \bot$ (basis): Looking at definition 3.2.9 we see $\bot \not\to^* \top$ and $B(\top) = \{\top\}$.

$o = o_1 \,\&\, o_2$: By the *only if* part of lemma 3.2.11 we have $o_1 \to^* \top$ and $o_2 \to^* \top$. By hypothesis of induction we then have $B(o_1) = \{\top\}$ and $B(o_2) = \{\top\}$. Now assume $B(o) \not\subseteq \{\top\}$. Since $\top$ and $\bot$ are the only terminal configurations (by a)) this implies $o = o_1 \,\&\, o_2 \to^* \bot$ and by the *only if* part of lemma 3.2.11.b) we have $o_1 \to^* \bot$ or $o_2 \to^* \bot$, which contradicts $B(o_1) = \{\top\}$ and $B(o_2) = \{\top\}$. So $B(o) \subseteq \{\top\}$.

$o = o_1 \nabla o_2$: Symmetric.

$o = (t, p), t \in TL, p \in CL$: This part of the inductive step is also proved by structural induction, but this time on the structure of t.

$t = \top$: Looking at definition 3.2.9 we see that only 2) can be used, hence $B(o) = B(\top) = \{\top\}$.

$t = \bar{a}.t'$: Clearly only 5) or 6) can come under discussion. Two cases depending on p.

$\exists q.\, p \xrightarrow{\bar{a}} q$: By proposition 3.2.4 there is at most one such q. Hence $B(o) = B((t', q))$. By hypothesis of induction $B((t', q)) \subseteq \{\top\}$.

$\nexists q.\, p \xrightarrow{\bar{a}} q$: This case can be excluded since $o \to \bot$ is the only possibility, $\bot \not\to$ and we assume $o \to^* \top$.

$t = t_1 \,\&\, t_2$: The only possibility is $o = (t_1 \,\&\, t_2, p) \to (t_1, p) \,\&\, (t_2, p) = o'' \to^* \top$ and $B(o) = B(o'')$. Similar as in the case $o = o_1 \,\&\, o_2$ we get $B(o'') \subseteq \{\top\}$.

$t = t_1 \nabla t_2$: Symmetric.

$o' = \bot$: Similar as the case $o' = \top$, but now $t = \top$ can be excluded and in $t = \bar{a}.t'$, $\nexists q.\, p \xrightarrow{\bar{a}} q$ no longer can be ignored. In this last case we get $o = (\bar{a}.t, p) \to \bot$ and $B(o) = B(\bot) = \{\bot\}$, which is wanted.

□

In the following we will investigate other properties of our test language which will be useful in the following sections.

The most important property is that every test t has a normal form which we define in a moment. On the way to show this we introduce some notation.

Definition 3.2.13 For $t, t' \in TL$ we let $t \cong t'$ denote:

$$\forall p \in CL.\ B((t,p)) = B((t',p))$$

or equivalently by the last proposition:

$$\forall p \in CL.\ (t,p) \rightarrow^* \top \mathit{iff}\ (t',p) \rightarrow^* \top$$

□

Proposition 3.2.14 $\cong$ is an equivalence relation on TL such that for all $t, t', t'' \in TL$:

a) $t \,\&\, t' \cong t' \,\&\, t$

b) $t \nabla t' \cong t' \nabla t$

c) $t \,\&\, (t' \,\&\, t'') \cong (t \,\&\, t') \,\&\, t''$

d) $t \nabla (t' \nabla t'') \cong (t \nabla t') \nabla t''$

e) $t \nabla (t' \,\&\, t'') \cong (t \nabla t') \,\&\, (t \nabla t'')$

f) $\bar{a}.(t \,\&\, t') \cong \bar{a}.t \,\&\, \bar{a}.t'$

g) $\bar{a}.(t \nabla t') \cong \bar{a}.t \nabla \bar{a}.t'$

h) $t \cong t' \Rightarrow \begin{cases} \bar{a}.) & \bar{a}.t \cong \bar{a}.t' \\ \&) & t \,\&\, t'' \cong t' \,\&\, t'' \\ \nabla) & t \nabla t'' \cong t' \nabla t'' \end{cases}$

Proof That $\cong$ is an equivalence relation is immediate from the definition.

a) – e) follows by lemma 3.2.11 from the similar properties of $\wedge$ and $\vee$. This is not the case with f) – h).

f) We shall prove $(\bar{a}.(t \,\&\, t'), p) \rightarrow^* \top$ *iff* $(\bar{a}.t \,\&\, \bar{a}.t', p) \rightarrow^* \top$.
if: By definition 3.2.9 $(\bar{a}.t \,\&\, \bar{a}.t', p) \rightarrow^* \top$ implies $(\bar{a}.t, p) \,\&\, (\bar{a}.t', p) \rightarrow^* \top$ which by lemma 3.2.11 implies $(\bar{a}.t, p) \rightarrow^* \top$ and $(\bar{a}.t', p) \rightarrow^* \top$. Now $(\bar{a}.t, p) \rightarrow^* \top$ implies $(\bar{a}.t', p) \rightarrow (t', p'') \rightarrow^* \top$ where $p \xrightarrow{\bar{a}} p'$. Similar $(\bar{a}.t', p) \rightarrow (t', p'') \rightarrow^* \top$, where $p \xrightarrow{\bar{a}} p''$. By proposition 3.2.4 we must have $p' = p''$. Hence $(t, p') \rightarrow^* \top$, $(t', p') \rightarrow^* \top$ and $p \xrightarrow{\bar{a}} p'$. Using lemma 3.2.11 again we get $(t, p') \,\&\, (t, p') \rightarrow^* \top$. From this and $p \xrightarrow{\bar{a}} p'$, 5), 1)& of definition 3.2.9 we see $(\bar{a}.(t \,\&\, t'), p) \rightarrow (t \,\&\, t', p') \rightarrow (t, p') \,\&\, (t', p') \rightarrow^* \top$.
only if: By inspection of definition 3.2.9 we see $(\bar{a}.(t \& t'), p) \rightarrow^* \top$ implies $(\bar{a}.(t \& t'), p) \rightarrow (t \& t', p') \rightarrow (t, p') \& (t', p') \rightarrow^* \top$, where $p \xrightarrow{\bar{a}} p'$. From this definition 3.2.9 directly gives us: $(\bar{a}.t \,\&\, \bar{a}.t', p') \rightarrow (\bar{a}.t, p) \,\&\, (\bar{a}.t', p) \rightarrow (t, p') \,\&\, (\bar{a}.t', p) \rightarrow (t, p') \,\&\, (t', p') \rightarrow^* \top$.

g) Similar, but here proposition 3.2.4 is not necessary!

h) Assume $t \cong t'$.

$\bar{a}.$): Let $p \in PL$, $\bar{a} \in \overline{Act}$ be given. Shall show that $(\bar{a}.t, p) \rightarrow^* \top$ *iff* $(\bar{a}.t', p) \rightarrow^* \top$. This is evident from proposition 3.2.4.

&): Let a $p \in CL$ and $t'' \in TL$ be given. Shall show $(t \,\&\, t'', p) \rightarrow^* \top$ *iff* $(t' \,\&\, t'') \rightarrow^* \top$. *if*: $(t' \,\&\, t'', p) \rightarrow^* \top$ implies $(t' \,\&\, t'', p) \rightarrow (t', p) \,\&\, (t'', p) \rightarrow^* \top$. By lemma 3.2.11

this implies $(t', p) \rightarrow^* \top$ and $(t'', p) \rightarrow^* \top$. From $t \cong t'$ we then have $(t, p) \rightarrow^* \top$. Reversing the arguments we obtain $(t \,\&\, t'', p) \rightarrow^* \top$.
only if: Symmetric.

∇): Similar.

□

Definition 3.2.15 $t \in TL$ is a *test normal form iff* t is of the form

$$\underset{j\in\underline{n}}{\&}\,(\underset{k\in\underline{n_j}}{\nabla}\,\bar{w}_{jk}\top), w_{jk} \in W \text{ for } k \in \underline{n_j}, 1 \leq n_j \text{ with } j \in \underline{n}, 1 \leq n$$

where $\bar{w}$ simply is the string w with every a in w exchanged with $\bar{a}$. □

By a) – d) of proposition 3.2.14 it gives sense to use the notational convenience of & and ∇ in the definition of a normal form.

Proposition 3.2.16 For every $t \in TL$ there is a normal form $t' \in TL$ such that $t \cong t'$.

Proof We can transform t into a normal form t' by using a) – h) of proposition 3.2.14. At first we use e) and f) to get all &'s of t out on the outmost level such that we obtain a $t'' = \&_{j\in\underline{n}}\, t_j$, where t_j is built from ∇, $\bar{a} \in \overline{Act}$ and $\top$ for $j \in \underline{n}$.
Then for every $j \in \underline{n}$ transform t_j by g) into $t'_j = \nabla_{k\in\underline{n_j}}\, w_{jk}\top$. By the congruence h) it should be clear that $t' = \&_{j\in\underline{n}}\, t'_j \cong t$. □

By inspection of definition 3.2.9 the following corollary is evident.

Corollary 3.2.17 For all $p \in CL$, $w \in W$ we have:

$$\exists q.\, p \xrightarrow{\bar{w}} q \textit{ iff } \bar{w}\top \in TL, (\bar{w}\top, p) \rightarrow^* \top$$

Proposition 3.2.18 Let t be on the normal form: $\&_{j\in\underline{n}}(\nabla_{k\in\underline{n_j}}\, \bar{w}_{jk}\top)$. Then for all $p \in CL$:

$$(t, p) \rightarrow^* \top \textit{ iff } \forall j \in \underline{n} \exists k \in \underline{n_j}.\, (\bar{w}_{jk}\top, p) \rightarrow^* \top$$

Proof Follows immediately from lemma 3.2.11. □

3.3 Full Abstractness

The aim of this section is to show that the denotational and operational semantics corresponds or rather that $\trianglelefteq_\delta$, $\trianglelefteq_\upsilon$ and $\trianglelefteq_\chi$ are fully abstract w.r.t. $\lessapprox_{\mathrm{a}}$, $\lessapprox_{\mathrm{r}}$ and $\lessapprox$ respectively. Formally we want to prove:

Theorem 3.3.1 *Operational Characterization Theorem* If $p, q \in PL$ then $\sqsubseteq\!\!\!\sim_{\mathrm{a}}$, $\sqsubseteq\!\!\!\sim_{\mathrm{r}}$ and $\sqsubseteq\!\!\!\sim$ are (relative) precongruences and

δ) $[\![p]\!]_\delta \trianglelefteq_\delta [\![q]\!]_\delta$ *iff* $p \sqsubseteq\!\!\!\sim_{\mathrm{a}} q$

υ) $[\![p]\!]_\upsilon \trianglelefteq_\upsilon [\![q]\!]_\upsilon$ *iff* $p \sqsubseteq\!\!\!\sim_{\mathrm{r}} q$

χ) $[\![p]\!]_\chi \trianglelefteq_\chi [\![q]\!]_\chi$ *iff* $p \sqsubseteq\!\!\!\sim q$

As for Hennessy [Hen85a] it will prove convenient to introduce some more plain relations, $\ll_*$, on processes from PL, which are entirely defined on basis of the lts, and which coincide with the three testing preorders.

The rest of this section is devoted to definitions and intermediate results necessary to prove the *Semantic Characterization Theorem* of these relations and the *Operational Characterization Theorem*.

At first we define a map, $\bar{\theta}$, on configurations which associates in a natural way a tree-semiword with the "barred" part of a configuration. I.e., for a configuration, p, $\bar{\theta}(p)$ gives a tree-semiword which reflects the causal order in which initiated actions can signal there completion.

Definition 3.3.2

a) Let $\bar{\theta} : CL \longrightarrow TSW$ be defined inductively as follows:

$$\begin{array}{rll} p & \mapsto \varepsilon & \text{if } p \in PL \\ \bar{a}.p & \mapsto a.\bar{\theta}(p) & \\ p \parallel q & \mapsto \bar{\theta}(p) \parallel \bar{\theta}(q) & \text{if either } p \notin PL \text{ or } q \notin PL \end{array}$$

b) Let $\bar{\Theta} : PL \longrightarrow \mathcal{P}(TSW)$ be defined by: $\bar{\Theta}(p) := \{\bar{\theta}(q) \mid \exists w \in W.\, p \stackrel{w}{\longrightarrow} q\}$

□

For arbitrary sets of semiwords we use the following notation:

$$\begin{array}{rl} S <_{\mathrm{a}} T \; \textit{iff} & \forall s \in S \exists t \in T.\, s \preceq t \\ S <_{\mathrm{r}} T \; \textit{iff} & \forall s \in S \exists t \in T.\, t \preceq s \\ S < T \; \textit{iff} & S <_{\mathrm{a}} T \text{ and } S <_{\mathrm{r}} T \end{array}$$

We can now formulate the three alternative preorders.

Definition 3.3.3 Let $p, q \in PL$. Then $\ll_{\mathrm{a}}$, $\ll_{\mathrm{r}}$ and $\ll$ are defined as follows:

$$p \ll_* q \; \textit{iff} \; \bar{\Theta}(p) <_* \bar{\Theta}(q)$$

where $*$ as usual is either left out or one of a, r. □

In the future we will mostly omit the comment about $*$.

Theorem 3.3.4 *Semantic Characterization Theorem* For all p, q in PL:

$$p \precsim_* q \textit{ iff } p \ll_* q$$

The first step in the prove of this theorem is a rewriting of $p \precsim_* q$.

Lemma 3.3.5 For $* = \mathrm{a}$ ($* = \mathrm{r}$) and $o_{\mathrm{a}} = \top$ ($o_{\mathrm{r}} = \bot$) we have:

$$p \precsim_* q$$

iff

$$*)\ \forall(A,t)\forall p' \in D(A,p)\exists q' \in D(A,q).\, (t,p') \rightarrow^* o_* \Rightarrow (t,q') \rightarrow^* o_*$$

Proof We only prove the case $* = \mathrm{a}$, since the case $* = \mathrm{r}$ follows in exactly the same way.
We prove $p \precsim_{\mathrm{a}} q$ *iff* a) by proving $p \not\precsim_{\mathrm{a}} q$ *iff* $\neg$a).
$p \not\precsim_{\mathrm{a}} q$ *iff* (by definition)

$\neg(\forall(A,t)(\exists p' \in D(A,p).\, (t,p') \rightarrow^* \top) \Rightarrow (\exists q' \in D(A,q).\, (t,q') \rightarrow^* \top))$ *iff*

$\exists(A,t)(\exists p' \in D(A,p).\, (t,p') \rightarrow^* \top) \wedge (\forall q' \in D(A,q).\, (t,q') \rightarrow^* \bot)$ *iff*

$\exists(A,t)\exists p' \in D(A,p).\, ((t,p') \rightarrow^* \top \wedge \forall q' \in D(A,q).\, (t,q') \rightarrow^* \bot)$ *iff*

$\exists(A,t)\exists p' \in D(A,p)\forall q' \in D(A,q).\, (t,p') \rightarrow^* \top \wedge (t,q') \rightarrow^* \bot$ *iff*

$\neg$a).
In these derivations we used $(t,p) \rightarrow^* \top$ *iff* $(t,p) \not\rightarrow^* \bot$ which was a consequence of proposition 3.2.12. □

The next step is to prove that the $\forall t$-quantifier can be moved past $\exists q' \in D(A,q)$ in $*$) of the last lemma.

Lemma 3.3.6 For $* = \mathrm{a}$ ($* = \mathrm{r}$) and $o_{\mathrm{a}} = \top$ ($o_{\mathrm{r}} = \bot$) we have:

i)$_*$ $\forall(A,t)\forall p' \in D(A,p)\exists q' \in D(A,q).\, (t,p') \rightarrow^* o_* \Rightarrow (t,q') \rightarrow^* o_*$

iff

ii)$_*$ $\forall A\forall p' \in D(A,p)\exists q' \in D(A,q)\forall t'.\, (t',p') \rightarrow^* o_* \Rightarrow (t',q') \rightarrow^* o_*$

Proof We prove $\neg$i)$_*$ *iff* $\neg$ii)$_*$ for the two cases of $*$. I.e.,

$\neg$i)$_*$ $\exists(A,t)\exists p' \in D(A,p)\forall q' \in D(A,q).\, (t,p') \rightarrow^* o_* \wedge (t,q') \not\rightarrow^* o_*$

iff

$\neg$ii)$_*$ $\exists A\exists p' \in D(A,p)\forall q' \in D(A,q)\exists t'.\, (t',p') \rightarrow^* o_* \wedge (t',q') \not\rightarrow^* o_*$

If $D(A,q) = \emptyset$ the result is trivial, so assume $D(A,q) \neq \emptyset$ in the following.

$* = \mathrm{a}$: The *only if* part is trivial since one just have to chose A, p' as in $\neg\mathrm{i})_\mathrm{a}$ and $t' = t$ for every $q' \in D(A, q)$.
if: Given is $\neg\mathrm{ii})_\mathrm{a} = \exists A \exists p' \in D(A, p) \forall q' \in D(A, q) \exists t'. (t', p') \to^* \top \wedge (t', q') \to^* \bot$. From here we use the A and p' in $\neg\,\mathrm{i})_\mathrm{a}$. Now let $t'_{q'}$ be the t' which $\neg\mathrm{ii})_\mathrm{a}$ ensures exists for $q' \in D(A, q)$ such that $(t', p') \to^* \top \wedge (t', q') \to^* \bot$. Then for all $q' \in D(A, q)$ we have:

$$(t'_{q'}, p') \to^* \top \text{ and } (t'_{q'}, q') \to^* \bot \tag{3.1}$$

Since $t'_{q'} \in TL$ for every $q' \in D(A, q)$ and $D(A, q)$ is finite, we have $t = \&_{q' \in D(A,q)} t'_{q'} \in TL$ too. By lemma 3.2.11.a),c) and (3.1) we have $(t, p') \to^* \top$ and for every $q' \in D(A, q)$. $(t, q') \to^* \bot$, thereby establishing $\neg\mathrm{i})_\mathrm{a}$.

$* = \mathrm{r}$: Proved similar as the $* = \mathrm{a}$ case, just with the difference in the *if*-part that t is constructed as $\nabla_{q' \in D(A,q)} t'_{q'}$ and lemma 3.2.11.b), d) is used in proving $(t, p') \to^* \bot$, and for every $q' \in D(A, q)$. $(t, q') \to^* \top$.

□

As seen from the proof we could not have moved the $\forall t$-quantifier if our test language did not contain & and ∇.

Lemma 3.3.7 Let $Z = (Act \cup \overline{Act})^*$. For $p_1, p_2 \in CL$ we have for $z \in Z$:

$$p_1 \parallel p_2 \xrightarrow{z} q$$

iff

$$\exists z_i \in Z, q_i.\, p_i \xrightarrow{z_i} q_i \text{ for } i \in \underline{2} \text{ and } q = q_1 \parallel q_2, z \preceq z_1 \parallel z_2$$

Notice that $W \subseteq Z$ and $\overline{W} \subseteq Z$, but $Z \not\subseteq W \cup \overline{W}$.

Proof Both of the implications in this lemma is proved by induction on the length of z.

if:

$z = \varepsilon$: $z = \varepsilon \preceq z_1 \parallel z_2$ implies $z_1 = \varepsilon = z_2$, so $q_i = p_i$ for $i \in \underline{2}$ and $q = p_1 \parallel p_2$. By definition $p_1 \parallel p_2 \xrightarrow{\varepsilon} p_1 \parallel p_2 = q$, so ok.

$z \neq \varepsilon$: Then $z = a.z'$ for some $a \in Act \cup \overline{Act}$ and $z' \in Z$. By proposition$_T$ 1.3.31 this implies $\exists z'_1.\, a.z'_1 = z_1, z' \preceq z'_1 \parallel z_2$ or $\exists z'_2.\, a.z'_2 = z_2, z' \preceq z_1 \parallel z'_2$. W.l.o.g. assume the former is true. Then $p_1 \xrightarrow{z_1} q_1$ is the same as $p_1 \xrightarrow{a.z'_1} q_1$, so there exists some p'_1 such that $p_1 \xrightarrow{a} p'_1 \xrightarrow{z'_1} q_1$. Since $|z'| < |z|$ we can use the inductive hypothesis to get $p'_1 \parallel p_2 \xrightarrow{z'} q_1 \parallel q_2 = q$. From the inference rule 5) of definition 3.2.2 we obtain $p_1 \parallel p_2 \xrightarrow{a} p'_1 \parallel p_2$ from $p_1 \xrightarrow{a} p'_1$, so $p_1 \parallel p_2 \xrightarrow{a.z'} q$ or equivalently $p_1 \parallel p_2 \xrightarrow{z} q$.

only if:

$z = \varepsilon$: Then $q = p_1 \parallel p_2$ and the result should be clear.

$z \neq \varepsilon$: Then $z = a.z'$ for some $a \in Act \cup \overline{Act}$ and $p_1 \parallel p_2 \xrightarrow{a} q' \xrightarrow{z'} q$ for some $q' \in CL$. Looking at definition 3.2.2 we see that 5) must have been used to ensure $p_1 \parallel p_2 \xrightarrow{a} q'$. W.l.o.g. assume this is obtained from $p_1 \xrightarrow{a} p'_1$ such that $q' = p'_1 \parallel p_2$. By hypothesis of induction we have that there exists $z_1', z_2 \in Z$ and q_1, q_2 such that $p_1' \xrightarrow{z_1'} q_1, p_2 \xrightarrow{z_2} q_2, q = q_1 \parallel q_2, z' \preceq z_1' \parallel z_2$. Let $z_1 = a.z_1'$. From $p_1 \xrightarrow{a} p_1'$ we then see $p_1 \xrightarrow{z_1} q_1$ and from proposition$_T$ 1.3.31 we get $z \preceq z_1 \parallel z_2$. This completes the inductive step.

□

Lemma 3.3.8 For all $p \in CL$ and $w \in W$ we have:

$$w \in \pi\lambda\bar{\theta}(p) \text{ iff } \exists p'.\, p \xrightarrow{\bar{w}} p'$$

Proof We will prove $w \in \pi\lambda\bar{\theta}(p) \Leftrightarrow \exists p'.\, p \xrightarrow{\bar{w}} p'$ by induction on the structure of p.

$p \in PL$: We split this case out in two depending of whether $w = \varepsilon$ or not.

- $w = \varepsilon$: Since $p \in PL$ we have $\bar{\theta}(p) = \varepsilon$, so $w = \varepsilon \in \{\varepsilon\} = \lambda(\varepsilon) = \pi\lambda(\varepsilon) = \pi\lambda\bar{\theta}(p)$. Also $\exists p'.\, p \xrightarrow{\bar{\varepsilon}} p'$, since $\varepsilon = \bar{\varepsilon}$ and $p \xrightarrow{\varepsilon} p$ by definition.
- $w \neq \varepsilon$: We neither have a $w' \in \pi\lambda\bar{\theta}(p)$ with $w' \neq \varepsilon$ nor any p' such that $p \xrightarrow{\bar{w}} p'$, when $p \in PL$ and $w \neq \varepsilon$.
  To see the latter assume on the contrary $\exists p'.\, p \xrightarrow{\bar{w}} p'$. $w \neq \varepsilon$ implies $w = a.w'$ for some $w' \in W$ and $a \in Act$. So $\bar{w} = \bar{a}.\bar{w}'$. Then $p \xrightarrow{\bar{w}} p'$ can be written $p \xrightarrow{\bar{a}.\bar{w}'} p'$ and implies $p \xrightarrow{\bar{a}} p'' \xrightarrow{\bar{w}'} p'$ for some $p'' \in CL$. But it is easely seen that $p \in PL, p \xrightarrow{y}$ implies $y \in Act$, which contradicts $p \xrightarrow{\bar{a}} p'', \bar{a} \in \overline{Act}$.
  The former is seen from $\pi\lambda\bar{\theta}(p) = \{\varepsilon\}$ as shown in the case $w = \varepsilon$.

$p = \bar{a}.p'', a \in Act$: At first notice $\pi\lambda\bar{\theta}(p) = \pi\lambda\bar{\theta}(\bar{a}.p'') =$ (by definition of $\bar{\theta}$) $\pi\lambda a.\bar{\theta}(p'') =$ (deduced from proposition$_T$ 1.3.13) $\pi a.\lambda\bar{\theta}(p'') =$ (corollary$_T$ 1.3.36) $a.\pi\lambda\bar{\theta}(p'') \cup \{\varepsilon\}$. We show the implications separately.

- $\Rightarrow$: $w \in \pi\lambda\bar{\theta}(p)$ implies $w \in a.\pi\lambda(p'')$ or $w = \varepsilon$ which again implies $w = a.w'$ or $w = \varepsilon$, where $w' \in \pi\lambda\bar{\theta}(p'')$. By definition $p \xrightarrow{\varepsilon} p = p'$ which handles the former case. In the latter we use the hypothesis of induction to get $\exists p'.\, p'' \xrightarrow{\bar{w}'} p'$. Using 3) of definition 3.2.2 we have $p = a.p'' \xrightarrow{\bar{a}} p''$. Hence $p \xrightarrow{\bar{w}} p'$.
- $\Leftarrow$: Looking at definition 3.2.2 we see that $p = \bar{a}.p'' \xrightarrow{\bar{w}} p'$ implies $\bar{w} = \varepsilon$ or $\bar{w} = \bar{a}.\bar{w}'$ for some $\bar{w}'$ such that $p'' \xrightarrow{\bar{w}'} p'$. If $\bar{w} = \varepsilon$ we have $w = \bar{\varepsilon} = \varepsilon$, so $w \in a.\pi\lambda\bar{\theta}(p'') \cup \{\varepsilon\} = \pi\lambda\bar{\theta}(p)$. In the other case $\bar{w} = \bar{a}.\bar{w}'$, the hypothesis of induction gives us $w' \in \pi\lambda\bar{\theta}(p'')$, so $w = a.w' \in a.\pi\lambda\bar{\theta}(p'') \cup \{\varepsilon\} = \pi\lambda\bar{\theta}(p)$.

$p = p_1 \parallel p_2$: At first notice that by lemma 3.3.7 we have:
$p_1 \parallel p_2 \xrightarrow{\bar{w}} p' \Leftrightarrow \exists p_1', p_2' \in CL, \bar{w}_1, \bar{w}_2 \in \overline{W}$ such that $p_1 \xrightarrow{\bar{w}_i} p_i'$ for $i \in \underline{2}, \bar{w} \preceq \bar{w}_1 \parallel \bar{w}_2, p = p_1' \parallel p_2'$.
Clearly we also have $\bar{w} \preceq \bar{w}_1 \parallel \bar{w}_2$ *iff* $w \preceq w_1 \parallel w_2$. The two implications:

- $\Rightarrow$: We have $\pi\lambda\bar{\theta}(p_1 \parallel p_2) =$ (by proposition$_T$ 1.3.38 and definition of $\bar{\theta}$) $\lambda\pi(\bar{\theta}(p_1) \parallel \bar{\theta}(p_2)) = \lambda(\pi\bar{\theta}(p_1) \parallel \pi\bar{\theta}(p_2))$, so using proposition$_T$ 1.3.13 we get $w \in \pi\lambda\bar{\theta}(p)$ implies $\exists w_i \in \lambda\pi\bar{\theta}(p_i), i \in \underline{2}$ such that $w \preceq w_1 \parallel w_2$. Since $\lambda\pi\bar{\theta}(p_i) = \pi\lambda\bar{\theta}(p_i)$ for $i \in \underline{2}$ we can use the hypothesis to get $\exists p_i'.\, p_i \xrightarrow{\bar{w}} p_i'$ for $i \in \underline{2}$. Then, as noticed, $p = p_1 \parallel p_2 \xrightarrow{\bar{w}} p_1' \parallel p_2' = p'$.

$\Leftarrow$: Using the noticed again $p = p_1 \| p_2 \xrightarrow{\bar{w}} p'$ implies $p_i \xrightarrow{\bar{w}_i} p'_i$, $i \in \underline{2}$. By hypothesis of induction $w_i \in \pi\lambda\bar{\theta}(p_i) = \lambda\pi\bar{\theta}(p_i)$. So since $w \in W$ implies $\nexists w'. w' \prec w$ and $w \preceq w_1 \parallel w_2$ we get from proposition$_T$ 1.3.13 $w \in \lambda(\pi\bar{\theta}(p_1) \parallel \pi\bar{\theta}(p_2))$ which, as seen above, is the same as $w \in \pi\lambda\bar{\theta}(p)$.

□

Lemma 3.3.9 For $p, q \in CL$ we have

$$\bar{\theta}(p) \preceq \bar{\theta}(q) \text{ iff } A_{\bar{\theta}(p)} = A_{\bar{\theta}(q)} \text{ and } \forall t. (t,p) \to^* \top \Rightarrow (t,q) \to^* \top$$

Proof
if: Assume $A_{\bar{\theta}(p)} = A_{\bar{\theta}(q)}$ and $\forall t. (t,p) \to^* \top \Rightarrow (t,q) \to^* \top$. We shall prove $\bar{\theta}(p) \preceq \bar{\theta}(q)$. By proposition$_T$ 1.3.5 it is enough to prove $\lambda\bar{\theta}(p) \subseteq \lambda\bar{\theta}(q)$. Let $w \in \lambda\bar{\theta}(p)$ be given. Then also $A_w = A_{\bar{\theta}(p)}$ and $w \in \pi\lambda\bar{\theta}(p)$, $w \in W$, so by lemma 3.3.8 $\exists p'. p \xrightarrow{\bar{w}} p'$ and from corollary 3.2.17 $\bar{w}\top \in TL, (\bar{w}\top, p) \to^* \top$. By the assumption then $(\bar{w}\top, q) \to^* \top$. Using the same lemmas in the opposite direction we get $w \in \pi\lambda\bar{\theta}(q)$. We cannot conclude $w \in \lambda\bar{\theta}(q)$ directly. Assume on the contrary w is a proper prefix of some $w' \in \lambda\bar{\theta}(q)$. Then $A_w \subset A_{w'}$. In general $\forall s \in \lambda(t). A_s = A_t$, so $A_{\bar{\theta}(p)} = A_w \subset A_{w'} = A_{\bar{\theta}(q)}$ which contradicts the assumption $A_{\bar{\theta}(p)} = A_{\bar{\theta}(q)}$. Hence $w \in \lambda\bar{\theta}(q)$.

only if: Assume $\bar{\theta}(p) \preceq \bar{\theta}(q)$. By definition $A_{\bar{\theta}(p)} = A_{\bar{\theta}(q)}$. Let $t \in TL$ be given such that $(t,p) \to^* \top$. We shall prove $(t,q) \to^* \top$. By proposition 3.2.16 t can be chosen to be on normal form. I.e.,

$$t = \underset{j \in \underline{n}}{\&} \Big(\underset{k \in \underline{n_j}}{\nabla} \bar{w}_{jk}\top \Big), w_{jk} \in W \text{ for } k \in \underline{n_j} \text{ and } j \in \underline{n}$$

Then by proposition 3.2.18 $(t,p) \to^* \top$ implies $\forall j \in \underline{n} \exists k \in \underline{n_j}. (\bar{w}_{jk}\top, p) \to^* \top$ and by corollary 3.2.17 $\forall j \in \underline{n} \exists k \in \underline{n_j} \exists p_{jk}. p \xrightarrow{\bar{w}_{jk}} p_{jk}$. By lemma 3.3.8 $w_{jk} \in \pi\lambda\bar{\theta}(p)$ for $j \in \underline{n}, k \in \underline{n_j}$. Now $\bar{\theta}(p) \preceq \bar{\theta}(q) \Rightarrow$ (by proposition$_T$ 1.3.5) $\lambda\bar{\theta}(p) \subseteq \lambda\bar{\theta}(q) \Rightarrow \pi\lambda\bar{\theta}(p) \subseteq \pi\lambda\bar{\theta}(q)$, so $w_{jk} \in \pi\lambda\bar{\theta}(q)$ and using lemma 3.3.8 again $\exists q_{jk}. q \xrightarrow{\bar{w}_{jk}} q_{jk}$. Reversing the arguments from above we get $(t,q) \to^* \top$. □

Lemma 3.3.10 For all $p \in PL$ and $w \in W$ we have $p \xrightarrow{w} q$ implies $w \in \lambda\bar{\theta}(q)$.

For the proof of the lemma we shall temporarily assume to work with semiwords and not just tree-semiwords.

Proof Actually we prove the stronger result:

(3.2) $$\text{for all } p \in CL \text{ and } w \in W. p \xrightarrow{w} q \Rightarrow \bar{\theta}(p)w \preceq \bar{\theta}(q)$$

from which the lemma follows since $p \in PL \Rightarrow \bar{\theta}(p) = \varepsilon$ and $w \preceq \bar{\theta}(q), w \in W \Rightarrow w \in \lambda\bar{\theta}(q)$. Notice that though $\bar{\theta}(p) \in TSW$ we do not necessarily have $\bar{\theta}(p)w \in TSW$. However this does not change the truth of (3.2).

To prove (3.2) we first prove:

(3.3) $$\text{for all } p \in CL, a \in Act. p \xrightarrow{a} q \Rightarrow \bar{\theta}(p)\underline{a} \preceq \bar{\theta}(q)$$

by induction on the structure of p considered as a member of PL.

$p \in PL$: Then $\bar{\theta}(p) = \varepsilon$, so we shall prove $\underline{a} \preceq \bar{\theta}(q)$. This again will be proved by induction on the structure of p.

$p = NIL$: Looking at definition 3.2.2 we see $p \stackrel{b}{\not\longrightarrow}$ for all b, so we are done for this case.

$p = b.p'$: Recalling $p \in PL$ when inspecting definition 3.2.2 we see that only 1) can could have been used to obtain $p \stackrel{a}{\longrightarrow} q$ and $b = a$, so $q = \bar{a}.p'$. Since $p = a.p' \in PL \Rightarrow p' \in PL$ we have $\bar{\theta}(q) = \bar{\theta}(\bar{a}.p') = a.\varepsilon = \underline{a}$.

$p = p_1 + p_2$: Only 4) could have been used. W.l.o.g. assume $p_1 \stackrel{a}{\longrightarrow} q$. By hypothesis of induction $\underline{a} \preceq \bar{\theta}(q)$.

$p = p_1 \parallel p_2$: Here only 5) can come under discussion. W.l.o.g. assume $p_1 \stackrel{a}{\longrightarrow} p_1'$, $q = p_1' \parallel p_2$. By hypothesis of induction $\underline{a} \preceq \bar{\theta}(p_1')$. Since $p_2 \in PL$ we have $\underline{a} \preceq \bar{\theta}(p_1') = \bar{\theta}(p_1') \parallel \varepsilon = \bar{\theta}(p_1') \parallel \bar{\theta}(p_2) = \bar{\theta}(p_1' \parallel p_2) = \bar{\theta}(q)$. This concludes the inductive step in the proof of $\bar{\theta}(p)\underline{a} \preceq \bar{\theta}(q)$ for $p \in PL$.

$p = \bar{b}.p'$: Inspecting definition 3.2.2 we see $p = \bar{b}.p' \stackrel{a}{\longrightarrow} q$, $a \in Act$ implies $p' \stackrel{a}{\longrightarrow} q'$, where $q = \bar{b}.q'$. By induction $\bar{\theta}(p')\underline{a} \preceq \bar{\theta}(q')$. By congruence of $\preceq$ we then have $\bar{\theta}(p)\underline{a} = \bar{\theta}(\bar{b}.p')\underline{a} = b.\bar{\theta}(p')\underline{a} \preceq b.\bar{\theta}(q') = \bar{\theta}(\bar{b}.q') = \bar{\theta}(q)$.

$p = p_1 \parallel p_2$: Only 5) could have been used. W.l.o.g. assume $p_1 \stackrel{a}{\longrightarrow} p_1'$, $q = p_1' \parallel p_2$. Then by hypothesis $\bar{\theta}(p_1)\underline{a} \preceq \bar{\theta}(p_1')$. Since $L(p_1) \cap L(p_2) = \emptyset$ and we in general for $r \in CL$ have: $L(\bar{\theta}(r)) \subseteq L(r)$ and $r \stackrel{b}{\longrightarrow} r' \Rightarrow b \in L(r), L(r') \subseteq L(r)$ it follows that $\bar{\theta}(p_1)\underline{a}$ and $\bar{\theta}(p_2)$ are disjoint so as $\bar{\theta}(p_1')$ and $\bar{\theta}(p_2)$. Therefore $(\bar{\theta}(p_1) \parallel \bar{\theta}(p_2))\underline{a}$, $\bar{\theta}(p_1') \parallel \bar{\theta}(p_2)$ and $\bar{\theta}(p_1)\underline{a} \parallel \bar{\theta}(p_2)$ are well-defined and members of TSW. So we can use proposition$_T$ 1.3.29 to get $(\bar{\theta}(p_1) \parallel \bar{\theta}(p_2))\underline{a} \preceq \bar{\theta}(p_1)\underline{a} \parallel \bar{\theta}(p_2)$. From the congruence of $\preceq$ and $\bar{\theta}(p_1)\underline{a} \preceq \bar{\theta}(p_1')$ we get $\bar{\theta}(p_1)\underline{a} \parallel \bar{\theta}(p_2) \preceq \bar{\theta}(p_1') \parallel \bar{\theta}(p_2)$. By transitivity of $\preceq$: $(\bar{\theta}(p_1) \parallel \bar{\theta}(p_2))\underline{a} \preceq \bar{\theta}(p_1') \parallel \bar{\theta}(p_2)$. So $\bar{\theta}(p)\underline{a} = \bar{\theta}(p_1 \parallel p_2)\underline{a} = (\bar{\theta}(p_1) \parallel \bar{\theta}(p_2))\underline{a} \preceq \bar{\theta}(p_1') \parallel \bar{\theta}(p_2) = \bar{\theta}(p_1' \parallel p_2) = \bar{\theta}(q)$ thereby finishing the inductive step in the proof of (3.3).

Having established (3.3) it is easy to prove (3.2) by induction on the length of w.

$w = \varepsilon$: Then $q = p$. Clearly $\bar{\theta}(p)w = \bar{\theta}(q)\varepsilon \preceq \bar{\theta}(q)$.

$w = a.w'$ for some $a \in Act$, $w' \in W$: Then $p \stackrel{a}{\longrightarrow} p' \stackrel{w'}{\longrightarrow} q$ for some $p' \in CL$. By hypothesis of induction $p' \stackrel{w'}{\longrightarrow} q \Rightarrow \bar{\theta}(p')w' \preceq \bar{\theta}(q)$. From (3.3) we have $p \stackrel{a}{\longrightarrow} p' \Rightarrow \bar{\theta}(p)\underline{a} \preceq \bar{\theta}(p')$. Hence $\bar{\theta}(p)a.w' \preceq \bar{\theta}(p')w'$ and by transitivity $\bar{\theta}(p)w = \bar{\theta}(q)$

□

Lemma 3.3.11 For $* = \mathrm{a}$ ($* = \mathrm{r}$) and $o_\mathrm{a} = \top$ ($o_\mathrm{r} = \bot$) and all $p, q \in PL$ we have:

$$*)\ \forall A \forall p' \in D(A,p) \exists q' \in D(A,q) \forall t.\ (t,p') \rightarrow^* o_* \Rightarrow (t,q') \rightarrow^* o_*$$

iff

$$\bar{\Theta}(p) <_* \bar{\Theta}(q)$$

Proof

$* = \mathrm{a}$: *if*: Let a multiset A over Act and a $p' \in D(A,p)$ be given. We shall find a $q' \in D(A,q)$ such that

(3.4) $$\forall t.\ (t,p') \rightarrow^* \top \Rightarrow (t,q') \rightarrow^* \top$$

By definition $p' \in D(A,p)$ implies $\exists w \in W.\, p \xrightarrow{w} p', A_w \cong A$ and then from the definition of $\bar{\Theta}$ we have $\bar{\theta}(p') \in \bar{\Theta}(p)$. The assumed premise is $\bar{\Theta}(p) <_{\mathrm{a}} \bar{\Theta}(q)$, so $\exists s_q \in \bar{\Theta}(q).\, \bar{\theta}(p') \preceq s_q$. Again by definition of $\bar{\Theta}$ we know that there exists $w' \in W$ and a q' such that $q \xrightarrow{w'} q'$, $\bar{\theta}(q') = s_q$. By lemma 3.3.9 $\bar{\theta}(p') \preceq \bar{\theta}(q')$ implies (3.4) and $A_{\bar{\theta}(p')} = A_{\bar{\theta}(q')}$. So if we can prove $q' \in D(A,q)$ we have proved this implication. From lemma 3.3.10 we see $w' \in W$, $q \xrightarrow{w'} q'$ implies $w' \in \lambda\bar{\theta}(q')$. Hence $A_{w'} = A_{\bar{\theta}(q')} = A_{\bar{\theta}(p)} \cong A$. All in all we have $w' \in W$, $A_{w'} \cong A$ and $q \xrightarrow{w'} q'$, so $q' \in D(A,q)$.

only if: Let $s_p \in \bar{\Theta}(p)$ be given. By definition of $<_{\mathrm{a}}$ we shall find an $s_q \in \bar{\Theta}(q)$ such that $s_p \preceq s_q$. $s_p \in \bar{\Theta}(p)$ means $\exists w \in W \exists p'.\, p \xrightarrow{w} p'$, $\bar{\theta}(p') = s_p$. By definition of $\cong$ we have $A \cong A_w$ for the multiset A over Act defined by $|A|_a = \psi(w,a)$ for all $a \in Act$. So $p' \in D(A,p)$. Assuming the premise of the implication to be true there exists a $q' \in D(A,q)$ such that (3.4) holds. $q' \in D(A,q)$ implies $\exists w' \in W.\, q \xrightarrow{w'} q'$, $A_{w'} \cong A$, so for all $a \in Act.\, \psi(w',a) = |A|_a$, wherefore for all $a \in Act.\, \psi(w',a) = \psi(w,a)$ and thereby $A_{w'} = A_w$. By lemma 3.3.10 we see $w' \in \lambda\bar{\theta}(q')$ and $w \in \lambda\bar{\theta}(p')$, so $A_{\bar{\theta}(q')} = A_{w'} = A_w = A_{\bar{\theta}(p')}$. This and (3.4) together with lemma 3.3.9 gives $\bar{\theta}(p') \preceq \bar{\theta}(q')$. Defining $s_q := \bar{\theta}(q')$ this reads $s_p = \bar{\theta}(p') \preceq \bar{\theta}(q') = s_q$. Now $s_q \in \bar{\Theta}(q)$ since $q \xrightarrow{w'} q'$, $w' \in W$ by definition of $\bar{\Theta}$ implies $\bar{\theta}(q') \in \bar{\Theta}(q)$.

$* = \mathrm{r}$: The proof is similar as for the case $* = \mathrm{a}$, with the difference that another version of lemma 3.3.9 is used:
$\bar{\theta}(q) \preceq \bar{\theta}(p)$ *iff* $A_{\bar{\theta}(q)} = A_{\bar{\theta}(p)}$ and

$$\forall t.\, (t,p) \rightarrow^* \bot \Rightarrow (t,q) \rightarrow^* \bot \tag{3.5}$$

To see this from lemma 3.3.9 notice that by proposition 3.2.12 we in general have

$$\neg((t,r) \rightarrow^* \top) \;\textit{iff}\; (t,r) \rightarrow^* \bot$$

wherefore (3.5) is equivalent to $\forall t.\, (t,q) \rightarrow^* \top \Rightarrow (t,p) \rightarrow^* \top$. □

Proof of *Semantic Characterization Theorem*
Since for $* = \mathrm{a}$ and $* = \mathrm{r}$ we have $\bar{\Theta}(p) <_* \bar{\Theta}(q)$ *iff* $p \ll_* q$ we get the theorem from lemma 3.3.5, lemma 3.3.6 and lemma 3.3.11 in the cases $* = \mathrm{a}$ and $* = \mathrm{r}$.
Finally: $p \sqsubseteq\!\!\!\sim q$ *iff* $p \sqsubseteq\!\!\!\sim_{\mathrm{a}} q$ and $p \sqsubseteq\!\!\!\sim_{\mathrm{r}} q$ *iff* $p \ll_{\mathrm{a}} q$ and $p \ll_{\mathrm{r}} q$ *iff* $p \ll q$. □

We have already seen the close connection between the operators of A_π and $A_\star$ for $\star$ in $\{\delta, \upsilon, \chi\}$ and soon we shall investigate how the interpretations of PL in $A_\star$ are related to the interpretation of PL in A_π which we now define in exactly the same way as we did in definition 3.1.13 for the $A_\star$ algebras.

Definition 3.3.12 $[\![\_]\!]_\pi : PL \longrightarrow C_\pi$ is recursively defined as follows:

$$\begin{aligned}
[\![NIL]\!]_\pi &= NIL_\pi \\
[\![a.p]\!]_\pi &= a._\pi [\![p]\!]_\pi \\
[\![p+q]\!]_\pi &= [\![p]\!]_\pi +_\pi [\![q]\!]_\pi \\
[\![p \parallel q]\!]_\pi &= [\![p]\!]_\pi \parallel_\pi [\![q]\!]_\pi
\end{aligned}$$

□

Recalling definition 3.1.8 where the interpretations in the Σ-po algebra A_π of the operator symbols where defined, we see that definition 3.3.12 can be read as:

$$\begin{aligned}
[\![NIL]\!]_\pi &= \{\varepsilon\}\\
[\![a.p]\!]_\pi &= a.[\![p]\!]_\pi \cup \{\varepsilon\}\\
[\![p+q]\!]_\pi &= [\![p]\!]_\pi \cup [\![q]\!]_\pi\\
[\![p \parallel q]\!]_\pi &= [\![p]\!]_\pi \parallel [\![q]\!]_\pi
\end{aligned}$$

Lemma 3.3.13 For all $p \in PL : \bar{\Theta}(p) = [\![p]\!]_\pi$

Proof Induction on the structure of p.

$p = NIL$: Since $NIL \not\rightarrow$ we have $\bar{\Theta}(NIL) = \{\bar{\theta}(NIL)\} = \{\varepsilon\} = [\![NIL]\!]_\pi$.

$p = a.p'$: $\bar{\Theta}(p) = \bar{\Theta}(a.p') = \{\bar{\theta}(q) \mid \exists w \in W.\, a.p' \xrightarrow{w} q\} = \{\bar{\theta}(q) \mid \exists w \in W.\, w \neq \varepsilon, a.p' \xrightarrow{w} q\} \cup \{\bar{\theta}(p)\} =$ (since $p \in PL$)

$$\{\bar{\theta}(q) \mid \exists w \in W.\, w \neq \varepsilon, a.p' \xrightarrow{w} q\} \cup \{\varepsilon\} \tag{3.6}$$

Inspecting definition 3.2.2 we see $a.p' \xrightarrow{w} q$, $w \neq \varepsilon$ *iff* $w = a.w'$, $p' \xrightarrow{w'} q'$ and $q = \bar{a}.q'$ for some $w' \in W$. So (3.6) $= \{\bar{\theta}(\bar{a}.q') \mid \exists w' \in W.\, p' \xrightarrow{w'} q'\} \cup \{\varepsilon\} =$ (by definition of $\bar{\theta}$) $a.\{\bar{\theta}(q') \mid \exists w' \in W.\, p' \xrightarrow{w'} q'\} \cup \{\varepsilon\} =$

$$a.\bar{\Theta}(p') \cup \{\varepsilon\} \tag{3.7}$$

By hypothesis of induction $\bar{\Theta}(p') = [\![p]\!]_\pi$, so (3.7) equals $[\![a.p']\!]_\pi$.

$p = p_1 + p_2$: $\bar{\Theta}(p) = \bar{\Theta}(p_1 + p_2) =$

$$\{\bar{\theta}(q) \mid \exists w \in W.\, p_1 + p_2 \xrightarrow{w} q\} \tag{3.8}$$

Again from definition 3.2.2 we see $p_1 + p_2 \xrightarrow{w} q$ *iff* either $p_1 \xrightarrow{w} q$ or $p_2 \xrightarrow{w} q$, so (3.8) equals $\{\bar{\theta}(q) \mid \exists w \in W.\, p_1 \xrightarrow{w} q\} \cup \{\bar{\theta}(q) \mid \exists w \in W.\, p_2 \xrightarrow{w} q\}$. The result then follows directly from the hypothesis.

$p = p_1 \parallel p_2$: $\bar{\Theta}(p) = \bar{\Theta}(p_1 \parallel p_2) =$

$$\{\bar{\theta}(q) \mid \exists w \in W.\, p_1 \parallel p_2 \xrightarrow{w} q\} \tag{3.9}$$

The next to see is that (3.9) equals

$$\{\bar{\theta}(q_1 \parallel q_2) \mid \exists w_1, w_2 \in W \exists q_1, q_2 \in CL.\, p_1 \xrightarrow{w_1} q_1, p_2 \xrightarrow{w_2} q_2\} \tag{3.10}$$

$\subseteq$: Follows directly from lemma 3.3.7.

$\supseteq$: Let s in (3.10) be given. Then there exists $w_i \in W, q_i \in CL.\, p_i \xrightarrow{w_i} q_i$ for $i \in \underline{2}$. Since $p_1 \parallel p_2$ is well-defined we have $L(p_1) \cap L(p_2) = \emptyset$. By corollary 3.2.3 then $q_1 \parallel q_2$ and $w_1 \parallel w_2$ are well-defined too, so we can define $q := q_1 \parallel q_2$. From proposition$_T$ 1.3.5.b) we know $\lambda(w_1 \parallel w_2) \neq \emptyset$, so there exists a $w \in W.\, w \preceq w_1 \parallel w_2$. Then we can use lemma 3.3.7 to get $p_1 \parallel p_2 \xrightarrow{w} q$. Hence $s = \bar{\theta}(q_1 \parallel q_2) = \bar{\theta}(q)$ in (3.9). Clearly (3.10) equals $\{\bar{\theta}(q_1) \mid \exists w_1 \in W.\, p_1 \xrightarrow{w_1} q_1\} \parallel \{\bar{\theta}(q_2) \mid \exists w_2 \in W.\, p_2 \xrightarrow{w_2} q_2\} = \bar{\Theta}(p_1) \parallel \bar{\Theta}(p_2)$ from which the result follows directly from the hypothesis of induction.

$\square$

Lemma 3.3.14 For all $S \subseteq SW$ (hence also for $S \subseteq TSW$) we have

$$\delta S <_{\mathrm{a}} S <_{\mathrm{a}} \delta S$$
$$\upsilon S <_{\mathrm{r}} S <_{\mathrm{r}} \upsilon S$$
$$\chi S < S < \chi S$$

Proof $\delta S <_{\mathrm{a}} S$ and $\upsilon S <_{\mathrm{r}} S$ follows directly from the definition of δ and υ. $S <_{\mathrm{a}} \delta S$, $S <_{\mathrm{r}} \upsilon S$ and $S < \chi S$ follows from the reflexivity of $\preceq$ and $S \subseteq \delta S, \upsilon S, \chi S$.
$\chi S < S$: We shall prove $\chi S <_{\mathrm{a}} S$ and $\chi S <_{\mathrm{r}} S$. But this is evident since by definition $s \in \chi S$ implies $t \preceq s \preceq t'$ for some $t, t' \in S$. $\square$

Combining the last two lemmas we immediately have:

Corollary 3.3.15 For all $p, q \in PL$:

$$p \ll_{\mathrm{a}} q \textit{ iff } \delta[\![p]\!]_\pi <_{\mathrm{a}} \delta[\![q]\!]_\pi$$
$$p \ll_{\mathrm{r}} q \textit{ iff } \upsilon[\![p]\!]_\pi <_{\mathrm{r}} \upsilon[\![q]\!]_\pi$$
$$p \ll q \textit{ iff } \chi[\![p]\!]_\pi < \chi[\![q]\!]_\pi$$

Lemma 3.3.16 For $\star$ in $\{\delta, \upsilon, \chi\}$ and all $p \in PL$:

$\star$) $[\![p]\!]_\star = \star[\![p]\!]_\pi$

Proof If we for $\overline{\star S} \in \overline{C_\star}$ have:

$$\star op_\pi(\overline{\star S}) = \star op_\pi(\overline{S}) \tag{3.11}$$

then $\star$) is easely proved by induction on the structure of p as indicated here: $[\![op(\overline{p})]\!]_\star = op_\star(\overline{[\![p]\!]_\star})$ = (by proposition 3.1.9) $\star op_\pi(\overline{[\![p]\!]_\star})$ = (hypothesis of induction) $\star op_\pi(\overline{\star[\![p]\!]_\pi})$ = (by (3.11)) $\star op_\pi(\overline{[\![p]\!]_\pi}) = \star[\![op(\overline{p})]\!]_\pi$.
In proving (3.11) we use the properties of (sets of) tree semiwords, the fact that δ and υ distributes over $\cup$ and the closure properties of $\star$:

$$\star\star S = \star S \tag{3.12}$$

for arbitrary sets of semiwords S.
The proof of (3.11):

$op_\pi = NIL_\pi$: Trivial.

$op_\pi = a._\pi$: $\star a._\pi \star S = \star(a.\star S \cup \{\varepsilon\})$, which by the $\cup$-distributivity of δ, υ and for $\star = \chi$: Proposition$_T$ 1.3.24.a) equals $\star a.\star S \cup \star\{\varepsilon\}$. By corollary$_T$ 1.3.19.b) in the case of $\star = \upsilon$ and (3.12), corollary$_T$ 1.3.16.a), corollary$_T$ 1.3.25 in the other cases we see that this quantity is the same as $\star a.S \cup \star\{\varepsilon\}$. With the same arguments as above this equals $\star(a.S \cup \{\varepsilon\}) = \star a._\pi S$.

$op_\pi = +_\pi$: $\star(\star S +_\pi \star T) = \star(\star S \cup \star T) =$ (by (3.12), $\cup$-distributivity of δ, υ and in the case $\star = \chi$: corollary$_T$ 1.3.23.c)) $\star(S \cup T) = \star(S +_\pi T)$.

$op_\pi = \|_\pi$: $\star(\star S \|_\pi \star T) = \star(\star S \| \star T)$. Using corollary$_T$ 1.3.16.b) when $\star = \delta$, (3.12) and proposition$_T$ 1.3.18.d) in the case $\star = \upsilon$ and finally for $\star = \chi$: corollary$_T$ 1.3.23.b), we see $\star(\star S \| \star T) = \star(S \| T) = \star(S \|_\pi T)$.

$\square$

Lemma 3.3.17

δ) $\forall S, T \in C_\delta.\ S <_{\mathrm{a}} T$ *iff* $S \trianglelefteq_\delta T$

υ) $\forall S, T \in C_\upsilon.\ S <_{\mathrm{r}} T$ *iff* $S \trianglelefteq_\upsilon T$

χ) $\forall S, T \in C_\chi.\ S < T$ *iff* $S \trianglelefteq_\chi T$

Proof Recall at first corollary 3.1.4 that for $\star$ in $\{\delta, \upsilon, \chi\}$: $\star$) $\forall T \in C_\star.\ \star(T) = T$.

δ) *if*: Assume $S \trianglelefteq_\delta T$ or equivalently $S \subseteq T$. We shall show $\forall s \in S \exists t \in T.\ s \preceq t$. Let $s \in S$ be given. Since $S \subseteq T$ and $\preceq$ is reflexive we can chose $t = s$.
only if: Assume $S <_{\mathrm{a}} T$. We shall show $S \subseteq T$. Let a $s \in S$ be given. By assumption there exists a $t \in T$ such that $s \preceq t$. Since $\delta T = T$ we have $s \in T$ too.

υ) Similar.

χ) *if*: Assume $S \trianglelefteq_\chi T$. We shall prove $S <_{\mathrm{a}} T$ and $S <_{\mathrm{r}} T$. This follows as for δ) and υ).
only if: Assume $S < T$. We shall prove $S \subseteq T$. Let $s \in S$. From $S <_{\mathrm{a}} T$ we see $\exists t' \in T.\ s \preceq t'$ and from $S <_{\mathrm{r}} T$: $\exists t \in T.\ t \preceq s$. Hence $\exists t, t' \in T.\ t \preceq s \preceq t'$ and thereby $s \in \chi T$. Since for $\chi T = T$ the result follows.

$\square$

From the last two lemmas and corollary 3.3.15 it follows:

Corollary 3.3.18 For all $p, q \in PL$:

δ) $p \ll_{\mathrm{a}} q$ *iff* $[\![p]\!]_\delta \trianglelefteq_\delta [\![q]\!]_\delta$

υ) $p \ll_{\mathrm{r}} q$ *iff* $[\![p]\!]_\upsilon \trianglelefteq_\upsilon [\![q]\!]_\upsilon$

χ) $p \ll q$ *iff* $[\![p]\!]_\chi \trianglelefteq_\chi [\![q]\!]_\chi$

We are now in a position to prove the operational characterization theorem on page 75.

Proof (of *Operational Characterization Theorem*)
From the *Semantic Characterisation Theorem* and corollary 3.3.18 it follows that e.g., $\lesssim_a$ and $\trianglelefteq_\delta$ agrees on PL. By corollary 3.1.11 the different operators of Σ_δ are (relative) $\trianglelefteq_\delta$-monotone, so from the compositional definition of $[\![\_]\!]_\delta$ we deduce that $\trianglelefteq_\delta$ is a (relative) precongruence. Because of the agreement between $\lesssim_a$ and $\trianglelefteq_\delta$ this must be the case for $\lesssim_a$ too. Similar for the other preorders. □

Before ending the section we shall prove that the denotational maps can denote any element of the relevant domain—a fact which will be used in the next chapter.

Proposition 3.3.19 $[\![\_]\!]_\star : PL \longrightarrow C_\star$ is surjective for $\star$ in $\{\delta, \upsilon, \chi\}$.

Proof Given a $S \in C_\star$. We shall find a $p \in PL$ with $[\![p]\!]_\star = S$. $S \in C_\star$ implies that there exists a $T \in \mathcal{P}_f(TSW) \setminus \emptyset$ such that $\star\pi T = S$. Because by lemma 3.3.16 $[\![p]\!]_\star = \star[\![p]\!]_\pi$ we see that it is enough to find a p such that $[\![p]\!]_\pi = \pi T$ since then $[\![p]\!]_\star = \star[\![p]\!]_\pi = \star\pi T = S$. Now π is $\cup$-distributive, so $\pi T = \cup_{t \in T}\pi(t)$. Hence we are done if we for every $t \in T$ can find a $p_t \in PL$ such that $[\![p_t]\!]_\pi = \pi(t)$, because then we can chose p to be $\Sigma_{t\in T} p_t$ (T is finite) and get $[\![p]\!]_\pi =$ (by definition of $+_\pi$ and proposition$_T$ 1.3.35.c)) $\cup_{t\in T}[\![p_t]\!]_\pi = \cup_{t\in T}\pi(t) = \pi T$.

We will now find such a p_t for a given t by induction on the size of t.

Basis: Clearly $t = \varepsilon$. Let $p_t = NIL$. Then $[\![p]\!]_\pi = NIL_\pi = \{\varepsilon\} = \pi(\varepsilon) = \pi(t)$.

Inductive step: Then $\gamma(t) \neq \{\varepsilon\}$ and $t = \varepsilon \parallel (\|\gamma(t) \setminus \{\varepsilon\}) = \|\gamma(t) \setminus \{\varepsilon\}$. By corollary$_T$ 1.1.8.f) $t' \in RTSW$ for every $t' \in \gamma(t) \setminus \{\varepsilon\}$.

If $\gamma(t) \setminus \{\varepsilon\}$ only consists of one rooted tree semiword, t', proposition$_T$ 2.2.15.a) gives us $\exists t'' \in TSW.\, t' = a.t''$. By hypothesis of induction we can find a $p'' \in PL$ such that $[\![p'']\!]_\pi = \pi(t'')$. Let $p = a.p''$. Then $[\![p]\!]_\pi = a._\pi[\![p'']\!]_\pi = a.\pi(t'') \cup \{\varepsilon\} =$ (by corollary$_T$ 1.3.36) $\pi(a.t'') = \pi(t') = \pi(\|\gamma(t) \setminus \{\varepsilon\}) = \pi(t)$.

If $\gamma(t) \setminus \{\varepsilon\}$ consists of more than one rooted tree-semiword we clearly can write t as $t_1 \parallel t_2$, where t_1, t_2 are nonempty tree-semiwords of size less than t. By hypothesis we find $p_i \in PL.\, [\![p_i]\!]_\pi = t_i$ for $i \in \underline{2}$. Let $p = p_1 \parallel p_2$. Then $[\![p]\!]_\pi = [\![p_1]\!]_\pi \parallel_\pi [\![p_2]\!]_\pi = \pi(t_1) \parallel \pi(t_2) =$ (by proposition$_T$ 1.3.35) $\pi(t_1 \parallel t_2) = \pi(t)$.

□

Finally we will briefly compare the equivalences. Since $\lesssim$ is defined as the intersection of $\lesssim_a$ and $\lesssim_r$ it is immediate from the full abstraction results of this section, that both $[\![\_]\!]_\delta$ and $[\![\_]\!]_\upsilon$ is as abstract as $[\![\_]\!]_\chi$. By the two process terms:

$$p_1 = a.b.NIL + a.NIL \parallel b.NIL \quad \text{and} \quad p_2 = a.NIL \parallel b.NIL$$

it follows that $[\![\_]\!]_\delta$ is strictly more abstract than $[\![\_]\!]_\chi$ (identified by $[\![\_]\!]_\delta$ but not by $[\![\_]\!]_\chi$). That $[\![\_]\!]_\upsilon$ also is strictly more abstract than $[\![\_]\!]_\chi$ is seen from p_1 and

$$p_3 = a.b.NIL$$

The same examples can be used to see that in general $[\![\_]\!]_\delta$ is not as abstract as $[\![\_]\!]_\upsilon$ and vice versa; p_1 and p_2 are identified by $[\![\_]\!]_\delta$ but not by $[\![\_]\!]_\upsilon$ and conversely with p_1 and p_3.

Chapter 4

Algebraic Characterization

The purpose of this chapter is to introduce two proof systems which on a purely syntactical level enables us to reason about how processes of PL can be interrelated via the different operational precongruences. The idea will be that if a certain relation between two process terms can be shown in the proof system then they will be related via the corresponding test preorder in the same way.

As for the previous chapter most of the concepts are as described in [Hen85a] and we shall also use some of the results.

Given a set of variables, X, and an arbitrary Σ-po algebra, $A = (C_A, \leq_A, \Sigma_A)$, an A-assignment is a mapping $\rho_A : X \longrightarrow C_A$, and from the proof of the freeness theorem we know a structural defined unique extension of ρ_A to the term algebra for $\Sigma(X)$. If BL is extended to the term algebra for the signature with variables, the corresponding extensions of the different $A_\star$-assignments would not always be well-defined. Some modifications are therefore necessary.

For a set of variables, X, BL is extended to include terms with variables from X simply by extending the signature Σ to $\Sigma(X)$ by augmenting Σ_0 with X. The so obtained term algebra is denoted $BL(X)$. We shall assume that each variable, $x \in X$, has an associated sort/ label set, L_x and furthermore that there is an infinite number variables for every possible sort (finite subset of *Act*). Extending the map L from BL to $BL(X)$ by letting $L(x) = L_x$ for every $x \in X$ we can similarly as PL was extracted from BL define $PL(X)$—the *open process terms*—to be those terms of $BL(X)$ where every subterm of form $t \parallel t'$ satisfies $L(t) \cap L(t') = \emptyset$. To emphasize the possibility of variables we shall often use $t, t', \ldots$ to denote terms from $PL(X)$.

An $A_\star$-assignment, $\rho_{A_\star}$, is now defined to be a mapping $X \longrightarrow C_\star$ such that for all $x \in X$ we have:

$$\rho_{A_\star}(x) = S \Rightarrow L(x) = L(S)$$

If $\rho_{A_\star}$ is extended to $PL(X)$ in the same way as in the freeness theorem, $\rho_{A_\star}$ is in this way ensured to be well-defined (and unique). Notice that $[\![p]\!]_\star = \rho_{A_\star}(p)$ for all $A_\star$-assignments, $\rho_{A_\star}$, if $p \in PL$.

The same goes for *syntactic substitutions*, i.e., $PL(X)$-assignments. That is $\rho : X \longrightarrow$

$PL(X)$ is a $PL(X)$-assignment if for every $x \in X$:

$$\rho(x) = t \Rightarrow L(x) = L(t)$$

The extension of a $PL(X)$-assignment to syntactic substitution will be written postfix.

4.1 Proof Systems

In this section we are going to formulate two proof systems DED_δ and DED_π respectively. These proof systems DED_δ and DED_π will contain the (usual) inference rules for (relative) precongruence, instantiation, transitivity, reflexivity as well as the inference rule for basic inequations:

Reflexivity: $$\frac{}{t \leq t}$$

Transitivity: $$\frac{t \leq t', t' \leq t''}{t \leq t''}$$

Substitutivity: $$\frac{t \leq t'}{a.t \leq a.t'} \qquad \frac{t_1 \leq t_1', t_2 \leq t_2'}{t_1 + t_2 \leq t_1' + t_2'}$$

$$\frac{t_1 \leq t_1', t_2 \leq t_2'}{t_1 \parallel t_2 \leq t_1' \parallel t_2'} \text{ provided } L(t_1) \cap L(t_2) = \emptyset = L(t_1') \cap L(t_2')$$

Instantiation: $$\frac{t \leq t'}{t\rho \leq t'\rho} \text{ for every } PL(X)\text{-assignment } \rho$$

Inequations: $$\frac{}{t \leq t'} \text{ for every } (\pi\text{-}) \ \delta\text{-inequation } t \leq t'$$

The δ-inequations and π-inequations respectively are as displayed below:

π-inequations:

+1	$x + (y + z) = (x + y) + z$
+2	$x + y = y + x$
+3	$x = x + NIL$
+4	$x = x + x$
$\|1$	$x \parallel (y \parallel z) = (x \parallel y) \parallel z$
$\|2$	$x \parallel y = y \parallel x$
$\|3$	$x = x \parallel NIL$
.+	$a.(x + y) = a.x + a.y$
$\|+$	$x \parallel (y + z) = x \parallel y + x \parallel z$
+5	$x \leq x + y$

δ-inequations: π-inequations and

$\delta.\|$	$a.(x \parallel y) \leq a.x \parallel y$

So the inequations are just relations between terms of $PL(X)$.

More generaly for a proof system $DED(E)$ of inequations as described by Hennessy [Hen85a], where E is the set of basic inequations we have the following notions.

The inequations inference rule gives a statement $t \leq t'$ for every $t \leq t'$ in E. These statements together with $t \leq t$ obtained by the reflexivity inference rule (with no premise) will be denoted the axioms.

A proof, P, is a sequence of statements

$$t_1 \leq t'_1, t_2 \leq t'_2, \ldots, t_n \leq t'_n$$

where each statement $t_i \leq t'_i, i \in \underline{n}$, is derived by applying the inference rules to statements earlier in the sequence. Clearly each $t_i \leq t'_i, i \in \underline{n}$ has it's own proof which is a part of P. We denote it by $P_{t_i \leq t'_i}$.

We will say that a proof has the *simple instantiation property* if instantiation only is used on the axioms.

Later in section 4.3 we shall see that if P is a proof of $t \leq t'$ then there is another proof P' of $r \leq t'$ with this property.

Notice that DED_δ and DED_π just are special cases of $DED(E)$ with $E = \delta$-inequations and $E = \upsilon$-inequations respectively.

If the statement $t \leq t'$ can be proved in DED_π we shall write this as $\vdash_\pi t \leq t'$. Similar for the statements of DED_δ. Since the π-inequations are contained in the δ-inequations it follows that $\vdash_\pi t \leq t'$ implies $\vdash_\delta t \leq t'$. As mentioned in the beginning to the chapter the proof systems can be used to deduce how processes can be operationally related. This will be more accurately addressed in the next two sections.

4.2 Soundness

In this section we shall see that all statements proved in DED_π will hold for the different operational (relative) precongruences. More precisely DED_π is sound w.r.t. $\lesssim_*$ over $PL(X)$ in the sense:

$$\vdash_\pi t \leq t' \text{ implies } t \lesssim_* t'$$

where $\lesssim_*$ is extended from PL to $PL(X)$ in the usual way by letting:

$$t \lesssim_* t' \textit{ iff } t\rho_{PL} \lesssim_* t'\rho_{PL} \text{ for all } PL\text{-assignments } \rho_{PL}$$

Furthermore the larger proof system, DED_δ, will also be sound w.r.t. $\lesssim_{\mathrm{a}}$.

Theorem 4.2.1 (*Soundness*)

> DED_δ is sound w.r.t. $\lesssim_{\mathrm{a}}$ over $PL(X)$.
>
> DED_π is sound w.r.t. $\lesssim_{\mathrm{r}}$ over $PL(X)$.
>
> DED_π is sound w.r.t. $\lesssim$ over $PL(X)$.

Proof For an arbitrary PL-assignment, ρ_{PL}, it is easy to see from the substitution lemma (see [Hen85a]) that the $A_\star$-assignment, $\rho_{A_\star}$, given by $\rho_{A_\star}(x) = [\![\rho_{PL}(x)]\!]_\star$ fulfills

$$\forall t \in PL(X).\ \rho_{A_\star}(t) = [\![t\rho_{PL}]\!]_\star \tag{4.1}$$

Conversely it is, due to the surjectivity of $[\![\_]\!]_\star$ as seen in proposition 3.3.19, also possible for any given $A_\star$-assignment, $\rho_{A_\star}$, to find a PL-assignment such that $[\![\rho_{PL}(x)]\!]_\star = \rho_{A_\star}$. From the substitution lemma, (4.1) then holds. Consequently for $t, t' \in PL(X)$ we have

$$[\![t\rho_{PL}]\!]_\star \subseteq [\![t'\rho_{PL}]\!]_\star \text{ for all } PL\text{-assignments } \rho_{PL}$$

iff

$$\rho_{A_\star}(t) \subseteq \rho_{A_\star}(t') \text{ for all } A_\star\text{-assignments } \rho_{A_\star}$$

The theorem is then immediate from the following proposition and the full abstractness results of the preceding chapter. The denotational preorders, $\trianglelefteq_\star$, are extended to $PL(X)$ by:

$$t \trianglelefteq_\star t' \textit{ iff } \rho_{A_\star}(t) \subseteq \rho_{A_\star}(t') \text{ for all } A_\star\text{-assignments } \rho_{A_\star}$$

□

Proposition 4.2.2

δ) DED_δ is sound w.r.t. $\trianglelefteq_\delta$ over $PL(X)$.

υ) DED_π is sound w.r.t. $\trianglelefteq_\upsilon$ over $PL(X)$.

χ) DED_π is sound w.r.t. $\trianglelefteq_\chi$ over $PL(X)$.

Proof In the δ)-case $t \trianglelefteq_\delta t'$ (for $t, t' \in PL(X)$) means that $\rho_{A_\delta}(t) \trianglelefteq_\delta \rho_{A_\delta}(t')$ for every A_δ-assignment ρ_{A_δ}. So to show DED_δ sound w.r.t. $\trianglelefteq_\delta$ over $PL(X)$ is the same as to show $\vdash t \leq t'$ implies $\rho_{A_\delta}(t) \trianglelefteq_\delta \rho_{A_\delta}(t')$ for every A_δ-assignment ρ_{A_δ}. To do this it is according to Hennessy enough to show A_δ satisfies the set of δ-inequations. I.e., we shall show for every $t \leq t'$ in the δ-inequations and every A_δ-assignment, ρ_{A_δ}, that $\rho_{A_\delta}(t) \trianglelefteq_\delta \rho_{A_\delta}(t')$ or equivalently by the definition of $\trianglelefteq_\delta$ that $\rho_{A_\delta}(t) \subseteq \rho_{A_\delta}(t')$. Since $[\![\_]\!]_\delta$ by proposition 3.3.19 is surjective and agrees with A_δ on closed terms this follows from the substitution lemma if for all PL-assignments, ρ_{PL}, and all $t \leq t'$ in the δ-inequations: $[\![t\rho_{PL}]\!]_\delta \subseteq [\![t\rho_{PL}]\!]_\delta$.

Similar considerations for the υ) and χ) case.

δ): We look at the (in)equations one by one.

+1: We shall show that for all $p, q, r \in PL$. $[\![p + (q + r)]\!]_\delta = [\![(p + q) + r]\!]_\delta$.
This is seen as follows: $[\![p+(q+r)]\!]_\delta = [\![p]\!]_\delta +_\delta [\![q+r]\!]_\delta = \ldots = [\![p]\!]_\delta \cup ([\![q]\!]_\delta \cup [\![r]\!]_\delta) = ([\![p]\!]_\delta \cup [\![q]\!]_\delta) \cup [\![r]\!]_\delta = \ldots = [\![(p + q) + r]\!]_\delta$

+2: Similar.

+3: We prove $[\![p]\!]_\delta = [\![p+NIL]\!]_\delta$ for a given $p \in PL$. Now $[\![p+NIL]\!]_\delta = [\![p]\!]_\delta \cup [\![NIL]\!]_\delta = [\![p]\!]_\delta \cup \{\varepsilon\}$, so $\subseteq$ is evident.
For all $r \in PL$ we have $\varepsilon \in [\![r]\!]_\pi$ and thereby also $\{\varepsilon\} \subseteq \delta([\![r]\!]_\pi)$. Hence by lemma 3.3.16

$$\{\varepsilon\} \subseteq [\![r]\!]_\delta \tag{4.2}$$

Using $r = p$ in (4.2) we get $\supseteq$ too.

+4: Evident.

$\|$1: The proof of this case is not as obvious as for +1. Let $p, q, r \in PL$ be given.
$[\![p \| (q \| r)]\!]_\delta = \ldots = \delta([\![p]\!]_\delta \| \delta([\![q]\!]_\delta \| [\![r]\!]_\delta)) =$ (corollary 3.1.4) $\delta(\delta[\![p]\!]_\delta \| \delta([\![q]\!]_\delta \| [\![r]\!]_\delta)) =$ (corollary$_T$ 1.3.16) $\delta([\![p]\!]_\delta \| ([\![q]\!]_\delta \| [\![r]\!]_\delta)) =$ (corollary$_T$ 1.2.11) $\delta(([\![p]\!]_\delta \| [\![q]\!]_\delta) \| [\![r]\!]_\delta) = \ldots = [\![(p \| q) \| r]\!]_\delta$.

$\|$2: By the commutativity of $\|$.

$\|$3: $[\![p \| NIL]\!]_\delta = \ldots =$

$$\delta([\![p]\!]_\delta \| \{\varepsilon\}) = \delta[\![p]\!]_\delta \tag{4.3}$$

$=$ (corollary 3.1.4)$[\![p]\!]_\delta$. Equation (4.3) follows from $\{\varepsilon\}$ being neutral to $\|$ on $\mathcal{P}(TSW)$ which again is inherited from (corollary$_T$ 1.2.11) ε being neutral to $\|$ on TSW.

.+: $[\![a.(p + q)]\!]_\delta = \ldots = a.([\![p]\!]_\delta \cup [\![q]\!]_\delta) \cup \{\varepsilon\} = a.[\![p]\!]_\delta \cup a.[\![q]\!]_\delta \cup \{\varepsilon\} = (a.[\![p]\!]_\delta \cup \{\varepsilon\}) \cup (a.[\![q]\!]_\delta \cup \{\varepsilon\}) = \ldots = [\![a.p + a.q]\!]_\delta$.

$\|$+: Similar, but with the additional use of the $\cup$-distributivity of δ.

+5: We shall show $[\![p]\!]_\delta \subseteq [\![p + q]\!]_\delta$ which evidently is true since $[\![p + q]\!]_\delta = [\![p]\!]_\delta \cup [\![q]\!]_\delta$.

$\delta.\|$: At first we show that in general for $S, T \subseteq TSW$ we have

$$a.\delta(S \| T) \subseteq \delta((a.S \cup \{\varepsilon\}) \| T) \tag{4.4}$$

Clearly $\delta(a.S \| T) \subseteq \delta((a.S \cup \{\varepsilon\}) \| T)$, so it is enough to show $a.\delta(S \| T) \subseteq \delta(a.S \| T)$. Let $u \in a.\delta(S \parallel T)$. Then $u = a.u'$ where $u' \in \delta(S \parallel T)$ so there is exists some $s \in S, t \in T$ such that $u' \preceq s \parallel t$. By congruence of $\preceq$, $u = a.u' \preceq a.(s \parallel t)$. From proposition$_T$ 1.3.29 we have $a.(s \parallel t) \preceq a.s \parallel t$ wherefore $u \preceq a.s \parallel t$. Hence $u \in \delta(a.S \parallel T)$.

Letting $S = [\![p]\!]_\delta, T = [\![q]\!]_\delta$ in (4.4) it reads $a.\delta([\![p]\!]_\delta \parallel [\![q]\!]_\delta) \subseteq \delta((a.[\![p]\!]_\delta \cup \{\varepsilon\}) \parallel [\![q]\!]_\delta)$. Now $\delta((a.[\![p]\!]_\delta \cup \{\varepsilon\}) \| [\![q]\!]_\delta) = \delta([\![a.p]\!]_\delta \| [\![q]\!]_\delta) = [\![a.p \| q]\!]_\delta$ and $a.\delta([\![p]\!]_\delta \| [\![q]\!]_\delta) = a.[\![p \| q]\!]_\delta$, so we have $a.[\![p \parallel q]\!]_\delta \subseteq [\![a.p \parallel q]\!]_\delta$.
By (4.2) we have $\{\varepsilon\} \subseteq [\![a.p \parallel q]\!]_\delta$ too, so $[\![a.(p \parallel q)]\!]_\delta = a.[\![p \parallel q]\!]_\delta \cup \{\varepsilon\} \subseteq [\![a.p \parallel q]\!]_\delta$.

υ): The arguments are almost as for the δ-case just using the properties of υ instead.

$+1 - +4$: As in the case δ)

$\|1 - \|2$: Similar to $+1$ and $+2$ because $\|_\upsilon$ does not have explicit υ-closure.

$\|3$: Follows with the same arguments as in δ).

$.+$: This case is a little different from the δ)-case because we have explicit υ-closure in the definition of $a._\upsilon$, but it is just as easy though.
$[\![a.(p+q)]\!]_\upsilon = \ldots = \upsilon a.([\![p]\!]_\upsilon \cup [\![q]\!]_\upsilon) \cup \{\varepsilon\} = \upsilon(a.[\![p]\!]_\upsilon \cup a.[\![q]\!]_\upsilon) \cup \{\varepsilon\} =$ ($\cup$-distributivity of υ) $\upsilon a.[\![p]\!]_\upsilon \cup \upsilon a.[\![q]\!]_\upsilon \cup \{\varepsilon\} = (\upsilon a.[\![p]\!]_\upsilon \cup \{\varepsilon\}) \cup (\upsilon a.[\![q]\!]_\upsilon \cup \{\varepsilon\}) = \ldots = [\![a.p + a.q]\!]_\upsilon$.

$\|+$: As the $.+$-case.

$+5$: Similar to the δ-case.

χ): Here we deduce:

$+1$: Suppose $p, q, r \in PL$. Then $[\![p + (q + r)]\!]_\chi = \chi([\![p]\!]_\chi \cup \chi([\![q]\!]_\chi \cup [\![r]\!]_\chi)) =$ (corollary$_T$ 1.3.23) $\chi([\![p]\!]_\chi \cup [\![q]\!]_\chi \cup [\![r]\!]_\chi) = \chi(\chi([\![p]\!]_\chi \cup [\![q]\!]_\chi) \cup [\![r]\!]_\chi) = \ldots = [\![(p + q) + r]\!]_\chi$

$+2$: Direct from definition of $[\![\_]\!]_\chi$ and commutativity of $\cup$.

$+3$: Similar arguments as for $+3$ of δ) but without χ.

$+4$: Evident.

$\|1 - \|3$: Similar to the corresponding cases of δ) but corollary$_T$ 1.3.16 is used in stead of corollary$_T$ 1.3.23.

$.+$: $[\![a.(p + q)]\!]_\chi = \ldots = a.\chi([\![p]\!]_\chi \cup [\![q]\!]_\chi) \cup \{\varepsilon\} =$ (proposition$_T$ 1.3.24.a)) $\chi(a.[\![p]\!]_\chi \cup a.[\![q]\!]_\chi \cup \{\varepsilon\}) = \chi([\![a.p]\!]_\chi \cup [\![a.q]\!]_\chi) = [\![a.p + a.q]\!]_\chi$.

$+5$: From a) of proposition$_T$ 1.3.22 follows $[\![p]\!]_\chi \subseteq \chi([\![p]\!]_\chi)$ so evidently $[\![p]\!]_\chi \subseteq \chi([\![p]\!]_\chi \cup [\![q]\!]_\chi) = [\![p + q]\!]_\chi$ for any $p, q \in PL$.

□

4.3 Completeness

We shall now see that DED_δ is powerful enough to derive any $\sqsubseteq_a$-relationship between two process.

Theorem 4.3.1 (*Completeness*)
DED_δ is complete w.r.t. $\underset{\sim}{\sqsubset}_{\mathrm{a}}$ over PL. I.e.,

$$\forall p, q \in PL.\, p \underset{\sim}{\sqsubset}_{\mathrm{a}} q \text{ implies } \vdash_\delta p \leq q$$

Proof Follows from $[\![\_]\!]_\delta$ being fully abstract w.r.t. $\underset{\sim}{\sqsubset}_{\mathrm{a}}$ and the proposition below. □

From the π-inequations it appears that in DED_π statements concerning prefix (-closures) as well as more ordinary algebraic properties such as commutativity and associativity can be proved. With the extra inequation, $\delta.\|$ it becomes possible to deal with δ-closures. Looking at the inequation $\delta.\|$ it is then tempting to replace it with

$$\upsilon.\| \quad a.x \parallel y \leq a.(x \parallel y)$$

in order to obtain a complete proof system, DED_υ, for $\underset{\sim}{\sqsubset}_{\mathrm{r}}$. However this inequation would not be sound as can be seen by considering the instantiation of $\upsilon.\|$:

$$a.b.NIL \parallel c.NIL \leq a.(b.NIL \parallel c.NIL)$$

Then we would have $a.b.NIL \parallel c.NIL$ may reject $(\{c\}, a.\top)$, but $a.(b.NIL \parallel c.NIL)$ ~~may~~ reject $(\{c\}, a.\top)$. This can just as easy be seen denotationally: $c \in [\![a.b.NIL \parallel c.NIL]\!]_\upsilon$, but $c \notin [\![a.(b.NIL \parallel c.NIL)]\!]_\upsilon$.

We could obtain a more powerful proof system for $\underset{\sim}{\sqsubset}_{\mathrm{r}}$ than DED_π by adding the sound inequations:

$$a.x \parallel y \leq a.(x \parallel y) + a.NIL \parallel y$$
$$a.x \parallel y \leq a.(x \parallel y) + y$$

Still we would not be able to prove e.g., $a.b.NIL \parallel c.NIL \leq a.b.c.NIL + a.NIL \parallel c.NIL$. Of course still more inequations could be added, but we have not been able to find a complete set, wherefore we stick to DED_π which is sound for all three preorders.

Proposition 4.3.2 DED_δ is complete w.r.t. $\trianglelefteq_\delta$ over PL.

The proof can of course not be done so directly as the soundness proof and some auxiliary propositions are needed. To motivate these and the necessary extra definitions below we will at first outline the proof—the full proof is on page 108.

Proof (sketch)
The main idea for proving

(4.5) $$p \trianglelefteq_\delta q \Rightarrow \ \vdash_\delta p \leq q$$

is to reduce p (via $\vdash_\delta$) to a sum, p', of composition forms (terms without $+$) with the property that there is exactly óne summand for each tree semiword in the denotation of p in the M_δ model. If the same is done for q thereby obtaining q' then the premise of (4.5) and definition of $\trianglelefteq_\delta$ ensures that q' can be proved equal to a term of the form $p' + q''$ and so the consequence of (4.5) follows by applying +5. The sum of composition forms with the desired property is obtained through more stages. At first a sum of composition forms is obtained essentially using the axioms for distributivity, .+ and $\|$+. Then all prefixes of the composition forms are added by use of +3 whereupon the composition forms corresponding to the downwards closure of the sum are included via $\delta.\|$. Finally all duplicates (up to commutativity and associativity of $\|$) of the composition forms are removed by means of +4 (idempotent) in order to get the óne to óne correspondence with the denotation. $\square$

Definition 4.3.3
Let p be a process from our process language ($p \in PL$). At first we define two fundamental sublanguages of PL.

p is a *composition form* ($p \in \mathsf{cf}$) is inductively defined by:

$$\begin{aligned} NIL &\in \mathsf{cf} \\ a.p &\in \mathsf{cf} \quad \text{if } p \in \mathsf{cf} \\ p \parallel q &\in \mathsf{cf} \quad \text{if } p, q \in \mathsf{cf} \end{aligned}$$

p is a *sumnormal form* ($p \in \mathsf{snf}$) is defined by:

$$\begin{aligned} \mathsf{cf} &\subseteq \mathsf{snf} \\ p + q &\in \mathsf{snf} \quad \text{if } p, q \in \mathsf{snf} \end{aligned}$$

We can now define the set of *summands*, $\mathsf{S}(p)$, of a sumnormal form p.
Let $\mathsf{S} : \mathsf{snf} \longrightarrow \mathcal{P}(\mathsf{cf})$ be defined by:

$$\begin{aligned} p &\mapsto \{p\} \qquad\qquad \text{if } p \in \mathsf{cf} \\ p_1 + p_2 &\mapsto \mathsf{S}(p_1) \cup \mathsf{S}(p_2) \end{aligned}$$

p is a *minimal sumnormal form* ($p \in \mathsf{msnf}$) is defined by:

$$\begin{aligned} \mathsf{cf} &\subseteq \mathsf{msnf} \\ p_1 + p_2 &\in \mathsf{msnf} \quad \text{if } p_1, p_2 \in \mathsf{msnf} \text{ and } \mathsf{S}(p_1) \cap \mathsf{S}(p_2) = \emptyset \end{aligned}$$

We denote the set of *syntactic "deterministic" prefixes* of a term p by $\mathsf{P}(p)$. Formally:
$\mathsf{P} : PL \longrightarrow \mathcal{P}(\mathsf{cf})$ is defined by:

$$\begin{aligned} NIL &\mapsto \{NIL\} \\ a.p &\mapsto a.\mathsf{P}(p) \cup \{NIL\} \\ p_1 + p_2 &\mapsto \mathsf{P}(p_1) \cup \mathsf{P}(p_2) \\ p_1 \parallel p_2 &\mapsto \mathsf{P}(p_1) \parallel \mathsf{P}(p_2) \end{aligned}$$

We define p is a *prefix form* ($p \in \mathsf{pf}$) by

$$p \in \mathsf{pf}$$
iff
$$p \in \mathsf{snf} \text{ and if this is the case } \mathsf{P}(p) = \mathsf{S}(p).$$

□

Notice

i) $\forall p \in \mathsf{snf}.\, \mathsf{S}(p) \neq \emptyset$.

ii) $p \in \mathsf{msnf} \Rightarrow p \in \mathsf{snf}$.

iii) $(p \in \mathsf{snf},\, (p \equiv a.p' \text{ or } p \equiv p_1 \parallel p_2)) \Rightarrow p, p', p_1, p_2 \in \mathsf{cf}$.

iv) $a.\mathsf{P}(p)$ in the definition of P shall be considered as the natural extension of the operator symbol $a.$ to cover sets as well. I.e., $a.\mathsf{P}(p) = \{a.q \mid q \in \mathsf{P}(p)\}$.

□

Whereas the functions in the last definition mapped to sets of terms they will map to tree-semiwords and sets of these in the following.

Definition 4.3.4
We define $\theta : \mathsf{cf} \longrightarrow TSW$ by:

$$\begin{aligned} NIL &\mapsto \varepsilon \\ a.p &\mapsto a.\theta(p) \\ p_1 \parallel p_2 &\mapsto \theta(p_1) \parallel \theta(p_2) \end{aligned}$$

and $\Theta : \mathsf{snf} \longrightarrow \mathcal{P}(TSW)$ by:

$$\Theta(p) := \{\theta(q) \mid q \in \mathsf{S}(p)\} = \theta\mathsf{S}(p),$$

where θ is extended in the natural way to sets. □

Notice that the ambiguity arising in using $a.$ both for terms and for semiwords is solved in the definition of θ when fixing θ's domain and codomain.

We introduce some notational convenience. We will say that two terms p, q are *sum congruent* written $\vdash p =_s q$ if we can show $\vdash p = q$ by $+1$, $+2$ and the other inference rules. Similar $\vdash p =_c q$ means that $\vdash p = q$ can be shown using $\parallel 1$ and $\parallel 2$ and we say that p and q are *composition congruent*. Often we will omit $\vdash$ and just write $p = q$ instead of $\vdash p = q$. This has as consequence that we also write e.g., $\vdash p =_s q$ as $p =_s q$. To avoid confusion we use $\equiv$ for syntactic equality between terms instead of $=$. Furthermore $p =_i q$ (i for idempotent) means only $+4$ together with the other inference rules are used in the proof of $p = q$ and $p =_n q$ (n for neutrality w.r.t. $\parallel$) that only $\parallel 3$ is used. We will also use

combinations of these as e.g., $p =_{si} q$.
Finally for $A, B \in \mathcal{P}(\mathsf{cf})$ we let $A \subseteq_{nc} B$ denote $\forall p \in A \exists q \in B.\ \vdash p =_{nc} q$.

Most of the following proofs will be induction on the structure of some closed term p considered as a member of PL or as being a composition-, sumnormal- or minimal sumnormal form. We will then often just list the different cases to consider—of course starting with the basic cases.
But we will have some proofs which are by induction on the proof of some statement $p \leq q$—actually on the length of the proof. To this end the following general lemma is useful.

Lemma 4.3.5 Let $DED(E)$ be a proof system of inequations E. Furthermore let P be a proof of $t \leq t'$. Then there exists a proof P' of $t \leq t'$ with the simple instantiation property.

Proof We will use induction on the length, $|P|$, of the proof P of $t \leq t'$.

$|P| = 1$: Then $t \leq t'$ is an inequation of E or $t = t'$ and we cannot have used instantiation, so we can let $P' := P$.

$|P| > 1$: Assume $|P| = n$ and the proof P is

$$t_1 \leq t'_1, t_2 \leq t'_2, \ldots, t_n \leq t'_n$$

Now look at the last inference rule used to obtain $t_n \leq t'_n$. Two cases depending on whether $t_n \leq t'_n$ is obtained by instantiation or not.

No instantiation used:

Assume $t_n \leq t'_n$ is obtained from $t_{i_1} \leq t'_{i_1}, \ldots, t_{i_k} \leq t'_{i_k}$, $i_l \in \underline{n-1}$, $l \in \underline{k}$ by some inference rule. Since $|P_{t_{i_l} \leq t'_{i_l}}| < |P| = n$ for $l \in \underline{k}$ we can use the hypothesis of induction to find proofs $P'_{t_{i_l} \leq t'_{i_l}}$ of $t_{i_l} \leq t'_{i_l}$ with the simple instantiation property. Let

$$P' := P'_{t_{i_1} \leq t'_{i_1}}, \ldots, P'_{t_{i_k} \leq t'_{i_k}}, t_n \leq t'_n$$

where the last statement is obtained by the inference rule. Clearly P' is a proof of $t_n \leq t'_n$ with the desired property.

Instantiation used:

Here we have two subcases.
$t_n \leq t'_n$ is an instantiation of an axiom.

Assume this axiom is $t_j \leq t'_j$, $j \in \underline{n-1}$. We see that

$$P' := t_j \leq t'_j, t_n \leq t'_n$$

is a proof of $t_n \leq t'_n$ with the simple instantiation property.

$t_n \leq t'_n$ is an instantiation, but not of an axiom.

Assume it is a ρ-instantiation of $t_j \leq t'_j$, $j \in \underline{n-1}$, i.e., $t_n \equiv t_j\rho$, $t'_n \equiv t'_j\rho$. Since $t_j \leq t'_j$ is not an axiom some inference rule must have been used to derive $t_j \leq t'_j$. We look at the different possibilities.

transitivity: Assume $t_j \leq t'_j$ is obtained from $t_k \leq t'_k$ and $t_l \leq t'_l$ where $t_j \equiv t_k$, $t'_l \equiv t'_j$, $t'_k \equiv t_l$ and $k, l \in \underline{j-1}$. Then $P_{t_j \leq t'_k}, t_j\rho \leq t'_k\rho$ and $P_{t'_l \leq t'_j}, t'_l\rho \leq t'_j\rho$, where the last statement is obtained by a ρ-instantiation, are proofs of $t_j\rho \leq t'_k\rho$ and $t'_l\rho \leq t'_j\rho$. Now $k, l < j < n$ implies $k+1, l+1 < n$ so the length of these proofs are less than n wherefore we can use the hypothesis to get proofs $P'_{t_j\rho \leq t'_k\rho}$ and $P'_{t_l\rho \leq t'_j\rho}$ with the property. Then since $t'_k \equiv t_l$ implies $t'_k\rho \equiv t_l\rho$

$$P' := P'_{t_j\rho \leq t'_k\rho}, P'_{t_l\rho \leq t'_j\rho}, t_j\rho \leq t'_j\rho$$

is a proof of $t_n \leq t'_n$ with the property.

substitutivity (congruence): Suppose $t_j \equiv f(t_{i_1}, \ldots, t_{i_k}), t'_j \equiv f(t'_{i_1}, \ldots, t'_{i_k})$ for some f in Σ of rank k and that $t_j \leq t'_j$ is obtained from $t_{i_1} \leq t'_{i_1}, \ldots, t_{i_k} \leq t'_{i_k}$. Similar as in the case of transitivity we by substitutivity find a proof

$$P' := P'_{t_{i_1}\rho \leq t'_{i_1}\rho}, \ldots, P'_{t_{i_k}\rho \leq t'_{i_k}\rho}, f(t_{i_1}\rho, \ldots, t_{i_k}\rho) \leq f(t'_{i_1}\rho, \ldots, t'_{i_k}\rho)$$

of $f(t_{i_1}\rho, \ldots, t_{i_k}\rho) \leq f(t'_{i_1}\rho, \ldots, t'_{i_k}\rho)$ with the simple instantiation property. Now since substitution ρ is a homomorphism we see $f(t_{i_1}\rho, \ldots, t_{i_k}\rho) \equiv (f(t_{i_1}, \ldots, t_{i_k}))\rho \equiv t_j\rho \equiv t_n$ and similar $f(t'_{i_1}\rho, \ldots, t'_{i_k}\rho) \equiv t'_n$. So P' is actually a proof of $t_n \leq t'_n$.

Instantiation: Assume $t_j \leq t'_j$ is obtained by ρ'-instantiation of $t_k \leq t'_k$, $k \in \underline{j-1}$. I.e., $t_j \equiv t_k\rho'$ and $t'_j \equiv t'_k\rho'$.

We have a proof $P'_{t_k \leq t'_k}$ for $t_k \leq t'_k$. The length of the proof

$$P_{t_k \leq t'_k}, t_k\rho \circ \rho' \leq t'_k\rho \circ \rho'$$

(where the last statement is a $\rho \circ \rho'$-instantiation of $t_k \leq t'_k$) is less than or equal to $k+1$. Since $k+1 \leq j < n$ we can use the hypothesis to find a proof $P'_{t_k\rho\circ\rho' \leq t'_k\rho\circ\rho'}$ with the simple instantiation property. By the Substitution lemma (of Hennessy [Hen85a]) $t_k\rho \circ \rho' \equiv (t_k\rho')\rho \equiv t_j\rho \equiv t_n$ and similar $t'_k\rho \circ \rho' \equiv t'_n$. So

$$P' := P'_{t_k\rho\circ\rho' \leq t'_k\rho\circ\rho'}$$

is a proof of $t_n \leq t'_n$ with the desired property.

□

The advantage of this lemma is that when proving some property on the basis of the length of a proof of a statement $p \leq q$ where $p, q \in PL$ we can assume that the proof has the simple instantiation property. Since p and q are closed terms the instantiation must be closed too. This means that we can leave out instantiation in our considerations if we instead consider closed instantiations of the axioms when dealing with these.

The first lemma shows that an action prefix of a sumnormal form within the proof system can be distributed over the summands thereby obtaining a sumnormal form.

Lemma 4.3.6 $p' \in \mathsf{snf} \Rightarrow \exists p \in \mathsf{snf}.\ \vdash p = a.p', \mathsf{S}(p) = a.\mathsf{S}(p')$

Proof Induction on the structure of p considered as a sumnormal form.

$p' \in \mathsf{cf}$: Let $p \equiv a.p'$. Reflexivity gives $\vdash p \equiv a.p' = a.p'$. We also have $p' \in \mathsf{cf} \Rightarrow p \equiv a.p' \in \mathsf{cf}$ and thereby $p \in \mathsf{snf}$. So since $\mathsf{S}(p') = \{p'\}$ we have $\mathsf{S}(p) = \{p\} = \{a.p'\} = a.\{p'\} = a.\mathsf{S}(p')$.

$p' \equiv p'_1 + p'_2$, $p'_1, p'_2 \in \mathsf{snf}$: By hypothesis of induction $\exists p_i \in \mathsf{snf}.\ \vdash p_i = a.p'_i, \mathsf{S}(p_i) = a.\mathsf{S}(p'_i)$ for $i \in \underline{2}$. Let $p \equiv p_1 + p_2$. We have:

$$\begin{aligned} \vdash p \equiv p_1 + p_2 &= a.p'_1 + a.p'_2 && \text{by congruence} \\ &= a.(p'_1 + p'_2) && \text{by } .+ \\ &= a.p' \end{aligned}$$

It only remains to show $\mathsf{S}(p) = a.\mathsf{S}(p')$. Notice $p \in \mathsf{snf}$ because $p_1, p_2 \in \mathsf{snf}$. So $\mathsf{S}(p) = \mathsf{S}(p_1 + p_2) = \mathsf{S}(p_1) \cup \mathsf{S}(p_2) = a.\mathsf{S}(p'_1) \cup a.\mathsf{S}(p'_2) = a.(\mathsf{S}(p'_1) \cup \mathsf{S}(p'_2)) = a.\mathsf{S}(p'_1 + p'_2) = a.\mathsf{S}(p')$.

□

In the next lemma a composition form parallel composed with a sumnormal form is distributed in over the summands.

Lemma 4.3.7 $p_1 \in \mathsf{cf}, p_2 \in \mathsf{snf} \Rightarrow \exists p \in \mathsf{snf}.\ \vdash p = p_1 \parallel p_2, \mathsf{S}(p) = \{p_1\} \parallel \mathsf{S}(p_2)$.

Proof The proof is here by induction on the structure of p_2

$p_2 \in \mathsf{cf}$: Let $p \equiv p_1 \parallel p_2$. We have $\vdash p \equiv p_1 \parallel p_2 = p_1 \parallel p_2$. $p_1, p_2 \in \mathsf{cf} \Rightarrow p_1 \parallel p_2 \in \mathsf{cf} \subseteq \mathsf{snf}$ and thereby also $p \in \mathsf{cf}$. As $p, p_2 \in \mathsf{cf}$ we have $\mathsf{S}(p) = \{p\}$ and $\mathsf{S}(p_2) = \{p_2\}$. Then $\mathsf{S}(p) = \{p\} = \{p_1 \parallel p_2\} = \{p_1\} \parallel \{p_2\} = \{p_1\} \parallel \mathsf{S}(p_2)$.

$p_2 \equiv q_1 + q_2$, $q_1, q_2 \in \mathsf{snf}$: By hypothesis of induction:

$\exists p'_i \in \mathsf{snf}.\ \vdash p'_i = p_1 \parallel q_i$, $\mathsf{S}(p'_i) = \{p_1\} \parallel \mathsf{S}(q_i)$ for $i \in \underline{2}$.

Let $p \equiv p'_1 + p'_2$. We have:

$$\begin{aligned} \vdash p \equiv p'_1 + p'_2 &= p_1 \parallel q_1 + p_1 \parallel q_2 && \text{by congruence} \\ &= p_1 \parallel (q_1 + q_2) && \text{by } \parallel + \\ &\equiv p_1 \parallel p_2 \end{aligned}$$

Notice $p \in \mathsf{snf}$, because $p'_1, p'_2 \in \mathsf{snf}$, so S is defined on p. Then $\mathsf{S}(p) = \mathsf{S}(p'_1) \cup \mathsf{S}(p'_2) = (\{p_1\} \parallel \mathsf{S}(q_1)) \cup (\{p_1\} \parallel \mathsf{S}(q_2)) = \{p_1\} \parallel (\mathsf{S}(q_1) \cup \mathsf{S}(q_2)) = \{p_1\} \parallel \mathsf{S}(p_2)$.

□

The last lemma easely generalize to sumnormal forms.

Lemma 4.3.8 $p_1, p_2 \in \mathsf{snf} \Rightarrow \exists p \in \mathsf{snf}.\ \vdash p = p_1 \parallel p_2, \mathsf{S}(p) = \mathsf{S}(p_1) \parallel \mathsf{S}(p_2)$.

Proof

$p_1 \in \mathsf{cf}$: Use the last lemma to find p. From $p_1 \in \mathsf{cf}$ it follows $\mathsf{S}(p_1) = \{p_1\}$, so $\mathsf{S}(p) = \{p_1\} \parallel \mathsf{S}(p_2) = \mathsf{S}(p_1) \parallel \mathsf{S}(p_2)$.

$p_1 \equiv q_1 + q_2$, $q_1, q_2 \in \mathsf{snf}$: We can use the hypothesis of induction on q_1, q_2 to get $q'_1, q'_2 \in \mathsf{snf}$ such that $\vdash q'_i = q_i \parallel p_2$ and $\mathsf{S}(q'_i) = \mathsf{S}(q_i) \parallel \mathsf{S}(p_2)$ for $i \in \underline{2}$. Let $p \equiv q'_1 + q'_2$. Then:

$$\begin{aligned}\vdash p \equiv q_1' + q_2' = {} & q_1 \parallel p_2 + q_2 \parallel p_2 && \text{by congruence}\\ =_c {} & p_2 \parallel q_1 + p_2 \parallel q_2 \\ = {} & p_2 \parallel (q_1 + q_2) && \text{by } \parallel + \\ \equiv {} & p_2 \parallel p_1 =_c p_1 \parallel p_2\end{aligned}$$

Furthermore $\mathsf{S}(p) = \mathsf{S}(q_1') \cup \mathsf{S}(q_2') = (\mathsf{S}(q_1) \parallel \mathsf{S}(p_2)) \cup (\mathsf{S}(q_2) \parallel \mathsf{S}(p_2)) = (\mathsf{S}(q_1) \cup \mathsf{S}(q_2)) \parallel \mathsf{S}(p_2) = \mathsf{S}(p_1) \parallel \mathsf{S}(p_2)$.

□

We are now in a position to show that any term can be reduced to as sumnormal form.

Proposition 4.3.9 $p \in PL \Rightarrow \exists q \in \mathsf{snf}.\ \vdash p = q$.

Proof We use induction on the structure of p considered as a member of PL.

$p \equiv NIL$: $NIL \in \mathsf{snf}$ by definition so chose $q \equiv NIL$ and we have the result by reflexivity.

$p \equiv a.p'$: By hypothesis of induction $\exists q' \in \mathsf{snf}.\ \vdash p' = q'$. By congruence $\vdash p \equiv a.p' = a.q'$. From lemma 4.3.6 we find a $q \in \mathsf{snf}$ such that $\vdash q = a.q'$ and the result follows by transitivity.

$p \equiv p_1 + p_2$: $\exists q_i \in \mathsf{snf}.\ \vdash q_i = p_i$ for $i \in \underline{2}$ by hypothesis of induction. Let $q \equiv q_1 + q_2$. By congruence then $\vdash p \equiv p_1 + p_2 = q_1 + q_2 \equiv q$.

$p \equiv p_1 \parallel p_2$: From the hypothesis we get $\exists q_i \in \mathsf{snf}.\ \vdash q_i = p_i$ for $i \in \underline{2}$. By congruence $\vdash q_1 \parallel q_2 = p_1 \parallel p_2 \equiv p$. The result then follows from lemma 4.3.8.

□

The next lemma merely states that no extra terms are gained by applying P more than once.

Lemma 4.3.10 For $p \in PL$ we have $\mathsf{PP}(p) = \mathsf{P}(p)$.

If D is a set of terms, $\mathsf{P}D$, is as usual to understand as the natural extension of P (defined on single terms) to sets of terms. We will write $\mathsf{P}(p)$ for a single term and e.g., $\mathsf{P}\{p\}$ when considering sets of terms.

Proof By induction on the structure of $p \in PL$.

$p \equiv NIL$: $\mathsf{P}(p) = \mathsf{P}(NIL) = \{NIL\} = \mathsf{P}\{NIL\} = \mathsf{PP}(p)$.

$p \equiv a.p'$: Here we have:

$$\begin{aligned}\mathsf{P}(p) = \mathsf{P}(a.p') &= \{NIL\} \cup a.\mathsf{P}(p') \\ &= \{NIL\} \cup a.\mathsf{PP}(p') \qquad \text{(by hypothesis of induction)} \\ &= \{NIL\} \cup \textstyle\bigcup_{q \in \mathsf{P}(p')} a.\mathsf{P}(q) \\ &= \{NIL\} \cup \textstyle\bigcup_{q \in \mathsf{P}(p')} (a.\mathsf{P}(q) \cup \{NIL\}) \\ &= \mathsf{P}\{NIL\} \cup \textstyle\bigcup_{q \in \mathsf{P}(p')} \mathsf{P}(a.q) \\ &= \textstyle\bigcup_{q \in a.\mathsf{P}(p') \cup \{NIL\}} \mathsf{P}(q) \\ &= \textstyle\bigcup_{q \in \mathsf{P}(a.p')} \mathsf{P}(q) \\ &= \mathsf{PP}(a.p') = \mathsf{PP}(p).\end{aligned}$$

$p \equiv p_1 + p_2$: $\mathsf{P}(p) = \mathsf{P}(p_1 + p_2) = \mathsf{P}(p_1) \cup \mathsf{P}(p_2) =$ (by hypothesis) $\mathsf{PP}(p_1) \cup \mathsf{PP}(p_2) = \mathsf{P}(\mathsf{P}(p_1) \cup \mathsf{P}(p_2)) = \mathsf{PP}(p_1 + p_2) = \mathsf{PP}(p)$.

$p \equiv p_1 \parallel p_2$: We deduce:

$$\begin{aligned}
\mathsf{P}(p) &= \mathsf{P}(p_1) \parallel \mathsf{P}(p_2) \\
&= \mathsf{PP}(p_1) \parallel \mathsf{PP}(p_2) \qquad \text{(by hypothesis)} \\
&= \{p'_1 \parallel p'_2 \mid (p'_1, p'_2) \in \mathsf{PP}(p_1) \times \mathsf{PP}(p_2)\} \\
&= \{p'_1 \parallel p'_2 \mid (p'_1, p'_2) \in \textstyle\bigcup_{(q_1,q_2)\in \mathsf{P}(p_1)\times \mathsf{P}(p_2)} \mathsf{P}(q_1) \times \mathsf{P}(q_2)\} \\
&= \textstyle\bigcup_{(q_1,q_2)\in \mathsf{P}(p_1)\times \mathsf{P}(p_2)} \{p'_1 \parallel p'_2 \mid (p'_1, p'_2) \in \mathsf{P}(q_1) \times \mathsf{P}(q_2)\} \\
&= \textstyle\bigcup_{(q_1,q_2)\in \mathsf{P}(p_1)\times \mathsf{P}(p_2)} \mathsf{P}(q_1) \parallel \mathsf{P}(q_2) \\
&= \textstyle\bigcup_{(q_1,q_2)\in \mathsf{P}(p_1)\times \mathsf{P}(p_2)} \mathsf{P}(q_1 \parallel q_2) \\
&= \textstyle\bigcup_{q\in \mathsf{P}(p_1)\parallel \mathsf{P}(p_2)} \mathsf{P}(q) \\
&= \textstyle\bigcup_{q\in \mathsf{P}(p_1\parallel p_2)} \mathsf{P}(q) = \mathsf{PP}(p).
\end{aligned}$$

□

Lemma 4.3.11 $p \in \mathsf{snf} \Rightarrow \mathsf{P}(p) = \mathsf{PS}(p)$.

Proof Induction on the structure of p considered as a sumnormal form.

$p \in \mathsf{cf}$: Then $\mathsf{S}(p) = \{p\}$ wherefore $\mathsf{P}(p) = \mathsf{P}\{p\} = \mathsf{PS}(p)$.

$p \equiv p_1 + p_2$, $p_1, p_2 \in \mathsf{snf}$: We see $\mathsf{P}(p) = \mathsf{P}(p_1) \cup \mathsf{P}(p_2) =$ (by hypothesis) $\mathsf{PS}(p_1) \cup \mathsf{PS}(p_2) = \mathsf{P}(\mathsf{S}(p_1) \cup \mathsf{S}(p_2)) = \mathsf{PS}(p_1 + p_2) = \mathsf{PS}(p)$.

□

From the last to lemmas we get:

Lemma 4.3.12 If $p \in \mathsf{snf}$ and there exists a $q \in PL$ such that $\mathsf{S}(p) = \mathsf{P}(q)$ then $p \in \mathsf{pf}$.

That is if p is a sumnormal form with summands equal to the prefixes of some other term then the summands of p are already closed under prefix.

Proof $p \in \mathsf{snf}$ so we only have to show $\mathsf{S}(p) = \mathsf{P}(p)$. But this is easely seen:

$$\begin{aligned}
\mathsf{S}(p) &= \mathsf{P}(q) && \text{by assumption} \\
&= \mathsf{PP}(q) && \text{by lemma 4.3.10} \\
&= \mathsf{PS}(p) && \text{by assumption again} \\
&= \mathsf{P}(p) && \text{by lemma 4.3.11 and the fact that } p \in \mathsf{snf}.
\end{aligned}$$

□

Lemma 4.3.13 $p' \in \mathsf{pf} \Rightarrow \exists p \in \mathsf{pf}.\ \vdash p = a.p'$.

Similar as the lemmas leading to proposition 4.3.9 we shall now see how different operators can be distributed in over prefix forms to obtain new prefix forms.

Proof As $p' \in \mathsf{pf}$ implies $p' \in \mathsf{snf}$ we can use lemma 4.3.6 to find a $q \in \mathsf{snf}$ such that $\vdash q = a.p'$ and

$$\mathsf{S}(q) = a.\mathsf{S}(p') \tag{4.6}$$

Now let $p \equiv q + NIL$. By +3 we have $\vdash p \equiv q + NIL = q = a.p'$. It just remains to show $p \in \mathsf{pf}$. Notice that because $q \in \mathsf{snf}$ we have $p \equiv q + NIL \in \mathsf{snf}$. We show $\mathsf{S}(p) = \mathsf{P}(a.p')$.

$\mathsf{S}(p) = \mathsf{S}(q) \cup \mathsf{S}(NIL)$
$\quad = a.\mathsf{S}(p') \cup \{NIL\}$ by (4.6)
$\quad = a.\mathsf{P}(p') \cup \{NIL\}$ because $p' \in \mathsf{pf}$
$\quad = \mathsf{P}(a.p')$

Since $p \in \mathsf{snf}$ we can deduce $p \in \mathsf{pf}$ from $\mathsf{S}(p) = \mathsf{P}(a.p')$ and lemma 4.3.12. □

Lemma 4.3.14 $p_1, p_2 \in \mathsf{pf}$ implies $\exists p \in \mathsf{pf}.\ \vdash p = p_1 \parallel p_2$.

Proof $p_1, p_2 \in \mathsf{pf}$ implies $p_1, p_2 \in \mathsf{snf}$ so we can use lemma 4.3.8 to find a $p \in \mathsf{snf}$ such that $\vdash p = p_1 \parallel p_2$ and

(4.7) $$\mathsf{S}(p) = \mathsf{S}(p_1) \parallel \mathsf{S}(p_2)$$

To get the result it remains to show $p \in \mathsf{pf}$. Since $\mathsf{S}(p) =$ (by (4.7)) $\mathsf{S}(p_1) \parallel \mathsf{S}(p_2) =$ (because $p_1, p_2 \in \mathsf{pf}$) $\mathsf{P}(p_1) \parallel \mathsf{P}(p_2) = \mathsf{P}(p_1 \parallel p_2)$ and $p \in \mathsf{snf}$ we obtain $p \in \mathsf{pf}$ by lemma 4.3.12. □

Lemma 4.3.15 $p \in \mathsf{cf}$ implies $\exists q \in \mathsf{pf}.\ \vdash p = q$.

Proof Induction on the structure of p considered as a composition form.

$p \equiv NIL$: Let $q \equiv NIL$. $\mathsf{P}(NIL) = \{NIL\} = \mathsf{S}(NIL)$ and by reflexivity $\vdash p \equiv NIL = NIL \equiv q$.

$p \equiv a.p'$, $p' \in \mathsf{cf}$: By hypothesis there exists a $q' \in \mathsf{pf}.\ \vdash p' = q'$. Using congruence we get $\vdash p \equiv a.p' = a.q'$. As $q' \in \mathsf{pf}$ lemma 4.3.13 gives us a $q \in \mathsf{pf}$ such that $\vdash q = a.q'$. We see $\vdash p = q$ and $q \in \mathsf{pf}$,

$p \equiv p_1 \parallel p_2$, $p_1, p_2 \in \mathsf{cf}$: There exists $p_1', p_2' \in \mathsf{pf}$ such that $\vdash p_1 = p_1'$ and $\vdash p_2 = p_2'$ by hypothesis of induction. By congruence then $\vdash p \equiv p_1 \parallel p_2 = p_1' \parallel p_2'$. As $p_1', p_2' \in \mathsf{pf}$ lemma 4.3.14 gives us a $q \in \mathsf{pf}$ with $\vdash q = p_1' \parallel p_2'$ from which the result follows.

□

It now easely follows that any sumnormal form can be reduced to a prefix form.

Proposition 4.3.16 $p \in \mathsf{snf}$ implies $\exists q \in \mathsf{pf}.\ \vdash p = q$.

Proof Induction on the structure of p considered as a sumnormal form.

$p \in \mathsf{cf}$: By the last lemma.

$p \equiv p_1 + p_2$, $p_1, p_2 \in \mathsf{snf}$: By hypothesis and congruence we find $q_1, q_2 \in \mathsf{pf}$ such that $\vdash p \equiv p_1 + p_2 = q_1 + q_2$. Let $q \equiv q_1 + q_2$. $q_1, q_2 \in \mathsf{pf} \Rightarrow q_1, q_2 \in \mathsf{snf}$ wherefore $q \in \mathsf{snf}$. We just have to show $\mathsf{S}(q) = \mathsf{P}(q)$ in order to have $q \in \mathsf{pf}$.

$\mathsf{S}(q) = \mathsf{S}(q_1) \cup \mathsf{S}(q_2)$
$\quad = \mathsf{P}(q_1) \cup \mathsf{P}(q_2)$ because $q_1, q_2 \in \mathsf{pf}$
$\quad = \mathsf{P}(q_1 + q_2) = \mathsf{P}(q)$.

□

With the next lemma it is possible (by commutativity and associativity) to bring any summand of a sumnormal form, p, to the front of p.

Lemma 4.3.17 Let $p \in \mathsf{snf}$. Then $q \in \mathsf{S}(p)$ implies $p \equiv q$ or $\exists q' \in \mathsf{snf}.\ \vdash p =_s q + q'$.

Proof

$p \in \mathsf{cf}$: In this case we have $\mathsf{S}(p) = \{p\}$ so $q \in \{p\}$ implies $p \equiv q$.

$p \equiv p_1 + p_2$, $p_1, p_2 \in \mathsf{snf}$: Here we have $\mathsf{S}(p) = \mathsf{S}(p_1) \cup \mathsf{S}(p_2)$ so $q \in \mathsf{S}(p)$ gives us two cases to consider.

$q \in \mathsf{S}(p_1)$: Using the hypothesis of induction we also have two possibilities to consider here. $p_1 \equiv q_2$: Chose $q' \equiv p_2$. By reflexivity $\vdash p \equiv p_1 + p_2 \equiv q + q'$.
$\exists q'' \in \mathsf{snf}$: $\vdash p_1 =_s q + q''$: Chose $q' \equiv q'' + p_2$. We have:

$$\begin{aligned} \vdash p \equiv p_1 + p_2 &=_s (q + q'') + p_2 && \text{by } p_1 =_s q + q'' \text{ and congruence} \\ &= q + (q'' + p_2) && \text{by } +1 \\ &\equiv q + q' \end{aligned}$$

$q' \in \mathsf{snf}$ follows from $q'' \in \mathsf{snf}$ and $p_2 \in \mathsf{snf}$.

$q \in \mathsf{S}(p_2)$: Similar but with additional use of $+2$.

□

Any two sumnormal forms which are equal up to idempotents, commutativity and associativity have the same summands. Formally:

Lemma 4.3.18 Let $p, q \in \mathsf{snf}$. Then $\vdash p =_{si} q$ implies $\mathsf{S}(p) = \mathsf{S}(q)$.

Proof For the purpose of this proof it is convenient to extend S defined on snf to S' defined on PL. Let $\mathsf{S}' : PL \longrightarrow \mathcal{P}(\mathsf{cf})$ be defined by:

$$\begin{aligned} NIL &\mapsto \{\varepsilon\} \\ a.p &\mapsto a.\mathsf{S}'(p) \\ p_1 + p_2 &\mapsto \mathsf{S}'(p_1) \cup \mathsf{S}'(p_2) \\ p_1 \parallel p_2 &\mapsto \mathsf{S}'(p_1) \parallel \mathsf{S}'(p_2) \end{aligned}$$

A number of subproofs are necessary, but they are all inductive and quite trivial so we only give the principal line.
S' well-defined i.e., $\mathsf{S}'(p) \in \mathsf{cf}$ for all $p \in PL$ is proved by induction on the structure of p. That S' coincide with S on snf is shown by first proving

$$p \in \mathsf{cf} \Rightarrow \mathsf{S}'(p) = \mathsf{S}(p) \tag{4.8}$$

by induction on the structure of p considered as a member of cf and next

$$p \in \mathsf{snf} \Rightarrow \mathsf{S}'(p) = \mathsf{S}(p)$$

also by structural induction, but this time with p considered as a sumnormal form, using (4.8) in the basis.
Finally for $p, q \in PL$:

$$\vdash p =_{si} q \Rightarrow \mathsf{S}'(p) = \mathsf{S}'(p)$$

is shown by induction on the number of inferences used to prove $p =_{si} q$. Since S' coincide with S on snf the lemma then follows as a special case.

The reason that we need the extension is that if we tried to prove the lemma directly by induction on the inferences we would get the following problem by transitivity:
If $p, q \in \mathsf{snf}$ and $p =_{si} q$ was proved using $p =_{si} r$ and $r =_{si} q$ we cannot be sure that $r \in \mathsf{snf}$ and therefore could not use the hypothesis. □

By the definition of Θ we then have:

Corollary 4.3.19 If $p, q \in \mathsf{snf}$ then $\vdash p =_{si} q$ implies $\Theta(p) = \Theta(q)$.

Lemma 4.3.20 $p \in \mathsf{msnf}$ and $\vdash p =_s q$ implies $q \in \mathsf{msnf}$.

Proof Shown by induction on the number of inferences used to infer $p =_s q$. At first we consider the axioms.

Reflexivity: Evident since then $p = q$.

+1: $p \equiv p_1 + (p_2 + p_3) = (p_1 + p_2) + p_3 \equiv q$. $p \in \mathsf{msnf}$ implies $p_1, p_2, p_3 \in \mathsf{msnf}$, $\mathsf{S}(p_1) \cap \mathsf{S}(p_2 + p_3) = \mathsf{S}(p_1) \cap (\mathsf{S}(p_2) \cup \mathsf{S}(p_3)) = \emptyset$ and $\mathsf{S}(p_2) \cap \mathsf{S}(p_3) = \emptyset$. Clearly then also $\mathsf{S}(p_1 + p_2) \cap \mathsf{S}(p_3) = \emptyset$ and $\mathsf{S}(p_1) \cap \mathsf{S}(p_2) = \emptyset$ so $q \in \mathsf{msnf}$.

+2: Similar arguments using $\cap$ commutative.

Inferences:
Transitivity: Assume $\vdash p =_s q$ is obtained from $\vdash p =_s r$ and $\vdash r =_s q$. Using the hypothesis of induction on $\vdash p =_s r$ we have $r \in \mathsf{msnf}$ so we can use the induction once more to get $q \in \mathsf{msnf}$.
Congruences:

$a.$: Assume $p' =_s q'$ is used to show $p \equiv a.p' =_s a.q' \equiv q$. Now $p \equiv a.p' \in \mathsf{msnf}$ implies $p' \in \mathsf{cf}$. Since $p' =_s q'$ it then follows that $q' \in \mathsf{cf}$ and thereby $q \equiv a.q' \in \mathsf{cf}$. Hence $q \in \mathsf{msnf}$.

+: W.l.o.g. assume $p' =_s q'$ is used to infer $p \equiv p' + r =_s q' + r \equiv q$. $p \equiv p' + r \in \mathsf{msnf}$ implies $p', r \in \mathsf{msnf}$ and $\mathsf{S}(p') \cap \mathsf{S}(r) = \emptyset$. By hypothesis of induction we then from $p' =_s q'$ get $q' \in \mathsf{msnf}$. Then since $\mathsf{msnf} \subseteq \mathsf{snf}$: $p' =_s q'$ by lemma 4.3.18 implies $\mathsf{S}(p') = \mathsf{S}(q')$. Hence $\mathsf{S}(q') \cap \mathsf{S}(r) = \emptyset$ and we conclude $q \in \mathsf{msnf}$.

$\|$: Similar arguments as in the $a.$-case.

□

We now show that any sumnormal form can be minimalized.

Proposition 4.3.21 For $p \in \mathsf{snf}$ there exists a $q \in \mathsf{msnf}$ such that $\vdash p =_{si} q$.

Proof At first we prove:

$$q_1, q_2 \in \mathsf{msnf} \Rightarrow \exists q \in \mathsf{msnf}.\ \vdash q_1 + q_2 =_{si} q \tag{4.9}$$

by induction on the number $n = |\mathsf{S}(q_1) \cap \mathsf{S}(q_2)|$

$n = 0$: I.e., $\mathsf{S}(q_1) \cap \mathsf{S}(q_2) = \emptyset$ and $q_1, q_2 \in \mathsf{msnf}$. Let $q \equiv q_1 + q_2$. Clearly $q \in \mathsf{msnf}$ and by reflexivity $\vdash q_1 + q_1 = q$.

$n > 0$: Chose a $q' \in \mathsf{S}(q_1) \cap \mathsf{S}(q_2)$. By lemma 4.3.17 we obtain for $i \in \underline{2}$: $q_i \equiv q' \vee \exists q'_i \in \mathsf{snf}.\ \vdash q_i =_s q' + q'_i$, so there are four subcases to consider.

$q_1 \equiv q' \equiv q_2$: $\vdash q_1 + q_2 \equiv q' + q' =_i q'$. As $q' \equiv q_1 \in \mathsf{msnf}$ we can chose $q \equiv q'$.

$q_1 \equiv q'$, $(\exists q'_2 \in \mathsf{snf}.\ \vdash q_2 =_s q' + q'_2)$: We get $\vdash\ q_1 + q_2 \ \equiv\ q' + q_2 \ =_s\ q' + (q' + q'_2) \ =_s (q' + q') + q'_2 =_i q' + q'_2 =_s q_2$. Chose $q \equiv q_2$ and the result follows since $q_2 \in \mathsf{msnf}$.

$(\exists q'_1 \in \mathsf{snf}.\ \vdash q_1 =_s q' + q'_1)$, $q_2 \equiv q'$: Symmetric.

$\exists q'_i \in \mathsf{snf}.\ \vdash q_i =_s q' + q'_i$ for $i \in \underline{2}$: We get $\vdash q_1 + q_2 =_s (q' + q'_1) + (q' + q'_2) =_s (q' + q') + (q'_1 + q'_2) =_i q' + (q'_1 + q'_2) =_s (q' + q'_1) + q'_2 =_s q_1 + q'_2$. According to lemma 4.3.20 we have $q' + q'_2 \in \mathsf{msnf}$ because $q_2 \in \mathsf{msnf}$ and $\vdash q_2 =_s q' + q'_2$. $q' + q'_2 \in \mathsf{msnf}$ gives us $\mathsf{S}(q') \cap \mathsf{S}(q'_2) = \emptyset$. Hence $|\mathsf{S}(q_1) \cap \mathsf{S}(q'_2)| < n$. $q' + q'_2 \in \mathsf{msnf}$ also gives $q'_2 \in \mathsf{msnf}$ so we can use the hypothesis of induction on q_1, q'_2 to find a $q \in \mathsf{msnf}$ such that $\vdash q_1 + q'_2 =_{si} q$. All together we then have $\vdash q_1 + q'_2 =_{si} q$ thereby completing the inductive step in proving (4.9).

With (4.9) we can now prove the proposition by induction on the structure of p considered as a sumnormal form.

$p \in \mathsf{cf}$: We then also have $p \in \mathsf{msnf}$ and can chose $q \equiv p$.

$p \equiv p_1 + p_2$, $p_1, p_2 \in \mathsf{snf}$: Using the hypothesis of induction on p_1, p_2 we can find $q_1, q_2 \in \mathsf{msnf}$ such that $\vdash p_1 =_{si} q_1$ and $\vdash p_2 =_{si} q_1$. By congruence $\vdash p \equiv p_1 + p_2 =_{si} q_1 + q_2$ and from (4.9) we find a $q \in \mathsf{msnf}.\ \vdash q_1 + q_2 =_{si} q$. The result then follows by transitivity.

□

The next lemma establish the first connection to the denotations of terms.

Lemma 4.3.22 For $p \in PL$ we have $[\![p]\!]_\pi = \theta\mathsf{P}(p)$.

Proof Induction on the structure of p considered as a member of PL.

$p \equiv NIL$: We see $[\![p]\!]_\pi = NIL_\pi = \{\varepsilon\} = \{\theta(NIL)\} = \theta\{NIL\} = \theta\mathsf{P}(NIL)$.

$p \equiv a.p'$: $[\![p]\!]_\pi = a.[\![p']\!]_\pi \cup \{\varepsilon\} =$ (induction and the definition of θ) $a.\theta\mathsf{P}(p') \cup \{\theta(NIL)\} = \{a.\theta(q) \mid q \in \mathsf{P}(p')\} \cup \{\theta(NIL)\} =$ (definition of θ) $\{\theta(a.q) \mid q \in \mathsf{P}(p')\} \cup \{\theta(NIL)\} = \theta a.\mathsf{P}(p') \cup \theta\{NIL\} = \theta(a.\mathsf{P}(p') \cup \{NIL\}) = \theta\mathsf{P}(a.p') = \theta\mathsf{P}(p)$.

$p \equiv p_1 + p_2$: $[\![p]\!]_\pi = [\![p_1]\!]_\pi \cup [\![p_2]\!]_\pi =$ (hypothesis of induction) $\theta\mathsf{P}(p_1) \cup \theta\mathsf{P}(p_2) =$ (because θ is extended to sets in the natural way) $\theta(\mathsf{P}(p_1) \cup \mathsf{P}(p_2)) = \theta\mathsf{P}(p)$.

$p \equiv p_1 \parallel p_2$: $[\![p]\!]_\pi = [\![p_1]\!]_\pi \parallel [\![p_2]\!]_\pi =$ (by induction) $\theta\mathsf{P}(p_1) \parallel \theta\mathsf{P}(p_2) = \{\theta q_1 \parallel \theta q_2 \mid (q_1, q_2) \in \mathsf{P}(p_1) \times \mathsf{P}(p_2)\}$ = (by definition of θ) $\{\theta(q_1 \parallel q_2) \mid (q_1, q_2) \in \mathsf{P}(p_1) \times \mathsf{P}(p_2)\} = \theta(\mathsf{P}(p_1) \parallel \mathsf{P}(p_2)) = \theta\mathsf{P}(p)$.

□

From the last lemma, the definition of Θ and the fact that $p \in \mathsf{pf}$ implies $\mathsf{P}(p) = \mathsf{S}(p)$ we have:

Corollary 4.3.23 If $p \in \mathsf{pf}$ then $[\![p]\!]_\pi = \Theta(p)$.

Lemma 4.3.24 If $\vdash p = q'$ and $q' \in \mathsf{S}(q)$ for a $q \in \mathsf{snf}$ then $\vdash q = p + q$.

Proof By lemma 4.3.17 it is enough to consider the following two possibilities:

$q' \equiv q$: By +4 we have $\vdash q \equiv q' = q' + q' \equiv q' + q$. As $\vdash p = q'$ we get by congruence $\vdash q' + q = p + q$ from which the result follows.

$\exists q'' \in \mathsf{snf}.\ \vdash q =_s q' + q''$: Again by +4 we have $\vdash q' = q' + q'$ so by congruence $\vdash q = (q' + q') + q'' =_s q' + (q' + q'') =_s q' + q$. From $\vdash p = q'$ and congruence we now get $\vdash q' + q = p + q$ and thereby $\vdash q = p + q$.

□

Proposition 4.3.25 Suppose $p \in \mathsf{msnf}$ and $q \in \mathsf{snf}$. Then $\mathsf{S}(p) \subseteq_{nc} \mathsf{S}(q)$ implies $\vdash p + q = q$.

Proof Induction on the quantity $n = |\mathsf{S}(p)|$.

$n = 1$: From $p \in \mathsf{msnf}$ follows $\mathsf{S}(p) = \{p\}$. Now $\{p\} \subseteq_{nc} \mathsf{S}(q)$ means $\exists q' \in \mathsf{S}(q).\ \vdash p =_{nc} q'$. The result then follows from the last lemma.

$n > 1$: Chose a $p_1 \in \mathsf{S}(p)$. As $|\mathsf{S}(p)| > 1$ lemma 4.3.17 gives us that there exists a $p_2 \in \mathsf{snf}$ such that $\vdash p =_s p_1 + p_2$. By lemma 4.3.20 we get $p_1 + p_2 \in \mathsf{msnf}$ from $p \in \mathsf{msnf}$. According to the definition of msnf we then also have $p_1, p_2 \in \mathsf{msnf}$ and $\mathsf{S}(p_1) \cap \mathsf{S}(p_2) = \emptyset$. Since $\mathsf{S}(p_1) \neq \emptyset \neq \mathsf{S}(p_2)$ then $|\mathsf{S}(p_i)| < |\mathsf{S}(p_1) \cup \mathsf{S}(p_2)| = |\mathsf{S}(p_1 + p_2)| = |\mathsf{S}(p)| = n$ for $i \in \underline{2}$. Because $p_1, p_2 \in \mathsf{msnf}$ and $\mathsf{S}(p_1), \mathsf{S}(p_2) \subseteq \mathsf{S}(p_1) \cup \mathsf{S}(p_2) = \mathsf{S}(p) \subseteq_{nc} \mathsf{S}(q)$ the hypothesis of induction gives us

$$\vdash p_1 + q = q \textit{ and } \vdash p_2 + q = q. \tag{4.10}$$

Then
$$\begin{aligned} \vdash p + q &=_s (p_1 + p_2) + q \\ &= (p_1 + p_2) + (q + q) && \text{by } +4 \\ &=_s (p_1 + q) + (p_2 + q) \\ &= q + q && \text{by congruence and (4.10)} \\ &= q && \text{by } +4. \end{aligned}$$

□

Lemma 4.3.26 $\theta(p) = \varepsilon, p \in \mathsf{cf} \Rightarrow \vdash p =_n NIL$

Proof

$p \equiv NIL$: Trivial.

$p \equiv a.p'$: Then $\theta(p) = a.\theta(p') \neq \varepsilon$ so the premise is not fulfilled.

$p \equiv q \parallel r$: Then $\theta(p) = \theta(q) \parallel \theta(r)$. corollary$_T$ 1.2.14.c) gives us: $\varepsilon = \theta(q) \parallel \theta(r) \Rightarrow \theta(q) = \varepsilon = \theta(r)$. So we can use the hypothesis of induction to get $q =_n NIL$ and $r =_n NIL$. Then: $\vdash p \equiv q \parallel r =_n NIL \parallel NIL$ by congruence
$=_n NIL$ by $\parallel$3.

□

Lemma 4.3.27 If $p \in \mathsf{cf}$ and $\theta(p) = a.s$ then there exists a $p' \in \mathsf{cf}$ such that $\theta(p') = s$ and $\vdash p =_n a.p'$.

Proof

$p \equiv NIL$: $\theta(p) = \varepsilon \neq a.s$ so the premise is not fulfilled.

$p \equiv b.q$: $\theta(p) = b.\theta(q)$. From corollary$_T$ 1.2.3.b) we get $a = b$ and $s = \theta(q)$. So $p \equiv a.q$. Then just chose $p' \equiv q$. As $p \in \mathsf{cf}$ it follows that $p' \equiv q \in \mathsf{cf}$. By reflexivity $p =_n a.p'$.

$p \equiv q \parallel r$: $\theta(p) = \theta(q) \parallel \theta(r)$ so by corollary$_T$ 1.2.14.a) we then get w.l.o.g. $\theta(q) = a.s$ and $\theta(r) = \varepsilon$. Now $p \in \mathsf{cf} \Rightarrow q \in \mathsf{cf}$ so we can use the hypothesis of induction to find a $p' \in \mathsf{cf}$ such that $\vdash q =_n a.p'$ and $\theta(p') = s$. By lemma 4.3.26 $\theta(r) = \varepsilon \Rightarrow \vdash r =_n NIL$. Then by congruence and $\parallel$3: $\vdash p \equiv q \parallel r =_n q \parallel NIL =_n q =_n a.p'$.

□

Lemma 4.3.28 If $p \in \mathsf{cf}$ then $\theta(p) = s \parallel t$ implies that there exists $p_1, p_2 \in \mathsf{cf}$ such that $\theta(p_1) = s$, $\theta(p_2) = t$ and $\vdash p =_{nc} p_1 \parallel p_2$.

Proof

$p \equiv NIL$: $\theta(p) = \varepsilon$ and $s \parallel t = \varepsilon \Rightarrow s = \varepsilon = t$. Let $p_1 \equiv p_2 \equiv NIL \in \mathsf{cf}$. It is seen that $\theta(p_1) = s$ and $\theta(p_2) = t$. The result then follows by $\parallel$3.

$p \equiv a.p'$: $\theta(p) = a.\theta(p')$. By corollary$_T$ 1.2.14.a) we from $a.\theta(p') = s \parallel t$ get either

a) $s = a.\theta(p'), t = \varepsilon$ or
b) $s = \varepsilon, t = a.\theta(p')$.

We look at the two possibilities separately.

a) Let $p_1 \equiv a.p' \in \mathsf{cf}$ and $p_2 \equiv NIL \in \mathsf{cf}$. We have $\theta(p_1) = a.\theta(p') = s$, $\theta(p_2) = \theta(NIL) = \varepsilon = t$ and $\vdash p \equiv p_1 =_n p_1 \parallel NIL$ by $\parallel$3
$\equiv p_1 \parallel p_2$.
b) Symmetric with the addition that $\parallel$2 is used too.

$p \equiv q_1 \parallel q_2$: $\theta(p) = \theta(q_1) \parallel \theta(q_2)$. By corollary$_T$ 1.2.14.b) $\theta(q_1) \parallel \theta(q_2) = s \parallel t$ implies that there exists s', s'', t', t'' such that $s = s' \parallel s'', t = t' \parallel t'', \theta(q_1) = s' \parallel t', \theta(q_2) = s'' \parallel t''$. Using the induction on the last two equations we find $q_1', q_1'', q_2', q_2'' \in \mathsf{cf}$ such that

$$q_1 =_{nc} q_1' \parallel q_2', \; q_2 =_{nc} q_1'' \parallel q_2'' \tag{4.11}$$

and $\theta(q_1') = s', \theta(q_1'') = s'', \theta(q_2') = t', \theta(q_2'') = t''$. Now let $p_i \equiv q_i' \| q_i'', i \in \underline{2}$. Evidently $\theta(p_1) = \theta(q_1') \parallel \theta(q_1'') = s' \parallel s'' = s$ and similar $\theta(p_2) = t$. We also have:

$$\begin{aligned} \vdash p \equiv q_1 \parallel q_2 &=_{nc} (q_1' \parallel q_2') \parallel (q_1'' \parallel q_2'') && \text{by congruence and (4.11)} \\ &=_c \ (q_1' \parallel q_1'') \parallel (q_2' \parallel q_2'') && \text{by } \|1 \text{ and } \|2 \\ &\equiv \ p_1 \parallel p_2 \end{aligned}$$

As $q_1', q_1'', q_2', q_2'' \in \mathsf{cf}$ it follows that $p_1, p_2 \in \mathsf{cf}$. This concludes the inductive step.

□

Proposition 4.3.29 Let $p, q \in \mathsf{cf}$. Then $\theta(p) = \theta(q)$ implies $\vdash p =_{nc} q$.

Proof

$p \equiv NIL$: $\theta(p) = \theta(NIL) = \varepsilon$. By lemma 4.3.26 it is seen that $\theta(q) = \varepsilon$ implies $q =_n NIL$ and thereby $\vdash p =_{nc} q$.

$p \equiv a.p'$: By lemma 4.3.27 and $\theta(q) = a.\theta(p') = \theta(p), q \in \mathsf{cf}$ we can find a $q' \in \mathsf{cf}$ such that $q =_n a.q'$ and $\theta(q') = \theta(p')$. As $p \in \mathsf{cf} \Rightarrow p' \in \mathsf{cf}$ and $q' \in \mathsf{cf}$ we can use the induction to get $q' =_{nc} p'$. By congruence then $a.q' =_{nc} a.p' \equiv p$. As $q =_n a.q'$ we have $\vdash q =_{nc} p$.

$p \equiv p_1 \parallel p_2$: From lemma 4.3.28 and $\theta(q) = \theta(p_1) \parallel \theta(p_2), q \in \mathsf{cf}$ we see that there exists $q_1, q_2 \in \mathsf{cf}$ such that $q =_{nc} q_1 \parallel q_2$ and $\theta(q_1) = \theta(p_1), \theta(q_2) = \theta(p_2)$. As $p \in \mathsf{cf} \Rightarrow p_1, p_2 \in \mathsf{cf}$ we can use the hypothesis of induction to get $\vdash p_i =_{nc} q_i, i \in \underline{2}$. By congruence then $q =_{nc} q_1 \parallel q_2 =_{nc} p_1 \parallel p_2 \equiv p$.

□

It should be noticed that we up til now only have been using the π-inequalities. To emphasise this we will write $\vdash_\pi p = q$ whenever this is the case. So we could actual rewrite all the previous properties with this addition. If in addition to the π-inequalities also $\delta.\|$ is used in proving $p = q$ we write $\vdash_\delta p = q$.

Proposition 4.3.30 Let $q \in \mathsf{cf}$, $\theta(q) = t$ and $s \in TSW$. Then $s \prec\!\!\cdot\; t \Rightarrow \exists p \in \mathsf{cf}.\ \vdash_\delta p \leq q, \theta(p) = s$

Proof Recall proposition$_T$ 2.3.44:

$s \prec\!\!\cdot\; t$ implies $\exists u \in \gamma(s), D \subseteq \gamma(t).\ \gamma(s) \setminus \{u\} = \gamma(t) \setminus D$

and for some $a, b \in Act$, $s', s'', t' \in TSW$ either

a) $u = a.(s' \parallel b.s''), D = \{a.s', b.s''\}$

or

b) $u = a.s', D = \{a.t'\}, s' \prec\!\!\cdot\; t'$

This natural suggests to make the proof in the size of $A_{\theta(s)}$. Letting $t = \theta(q)$ we see from above that there is two cases to consider.

a) Then t can be written as $a.s' \parallel b.s'' \parallel (\parallel \gamma(t) \setminus D)$. Since $\theta(q) = t$ we can use lemma 4.3.28 to find $q_1, q_2, q_3 \in \mathsf{cf}$ such that

$$\vdash_\pi q = q_1 \parallel q_2 \parallel q_3 \tag{4.12}$$

and $\theta(q_1) = a.s', \theta(q_2) = b.s'', \theta(q_3) = (\parallel \gamma(t) \setminus D)$. We can then by lemma 4.3.27 find q_1', q_2' such that

$$\vdash_\pi q_1 = a.q_1', \vdash_\pi q_2 = b.q_2' \tag{4.13}$$

and $\theta(q_1') = s', \theta(q_2') = s''$. Let $p := a.(q_1' \parallel b.q_2') \parallel q_3$. We have:

$$\begin{aligned} \vdash_\delta p \equiv\ & a.(q_1' \parallel b.q_2') \parallel q_3 \\ \leq\ & a.q_1' \parallel b.q_2' \parallel q_3 && \text{by } \delta.\parallel \text{ and congruence} \\ =_\pi\ & q_1 \parallel q_2 \parallel q_3 && \text{by (4.13) and congruence} \\ =_\pi\ & q && \text{by (4.12)} \end{aligned}$$

We also have $\theta(p) = \theta(a.(q_1' \parallel b.q_2') \parallel q_3) = a.(\theta(q_1') \parallel b.\theta(q_2')) \parallel \theta(q_3) = a.(s' \parallel b.s'') \parallel (\parallel \gamma(t) \setminus D) = u \parallel (\parallel \gamma(t) \setminus D) = u \parallel (\parallel \gamma(s) \setminus \{u\}) = \parallel \gamma(s) = s$. In the proof of this case we actually did not use the inductive hypothesis.

b) In this case we can write t as $a.t' \parallel (\parallel \gamma(t) \setminus \{a.t'\})$. As in the a)-case we find $q_1, q_2, q_1' \in \mathsf{cf}$ such that

$$\vdash_\pi q = q_1 \parallel q_2, \vdash_\pi q_1 = a.q_1' \tag{4.14}$$

and $\theta(q_1) = a.t', \theta(q_2) = \parallel \gamma(t) \setminus \{a.t'\}, \theta(q_1') = t'$.
Clearly $|A_{s'}| < |A_s|$ so we can use the inductive assumption to find $p' \in \mathsf{cf}$ such that

$$\vdash_\delta p' \leq q_1' \tag{4.15}$$

and $\theta(p') = s'$. Let $p := a.p' \parallel q_2$. We have:

$$\begin{aligned} \vdash_\delta p \equiv\ & a.p' \parallel q_2 \\ \leq_\delta\ & a.q_1' \parallel q_2 && \text{by (4.15) and congruence} \\ =_\pi\ & q_1 \parallel q_2 && \text{by the second part of (4.14)} \\ =_\pi\ & q && \text{by the first part of (4.14)} \end{aligned}$$

Finally: $\theta(p) = a.\theta(p') \parallel \theta(q_2) = a.s' \parallel (\parallel \gamma(t) \setminus \{a.t'\}) = u \parallel (\parallel \gamma(s) \setminus \{u\}) = s$.
This also completes the inductive step. □

Proposition 4.3.31 Let $q \in \mathsf{snf}$. Then $\exists p \in \mathsf{snf}.\ \vdash_\delta p = q, \Theta(p) = \delta\Theta(q)$

Proof At first we prove an intermediate result. Notice that since $\prec\!\cdot^+ = \prec$ we see $s \prec t$ implies $s \prec\!\cdot^n t$ for some $n \geq 1$. Then we can use induction on n to prove:

$$q \in \mathsf{cf}, s \prec \theta(q) \Rightarrow \exists p \in \mathsf{cf}.\ \vdash_\delta p \leq q, \theta(p) = s \tag{4.16}$$

$n = 1$: I.e., $s \prec\!\cdot\ \theta(q)$. The result follows from proposition 4.3.30 above.

$n > 1$: Then there exists some u such that $s \prec\!\cdot^{n-1} u, u \prec\!\cdot\ \theta(q)$. By proposition 4.3.30 there exists some $r \in \mathsf{cf}.\ \vdash_\delta r \leq q, \theta(r) = u$. Hence $s \prec\!\cdot^{n-1} \theta(r)$, so by hypothesis $\exists p \in \mathsf{cf}.\ \vdash_\delta p \leq r, \theta(p) = s$. Then by transitivity $\vdash_\delta p \leq q$.

We now prove the proposition by induction on the structure of q considered as a sumnormal form.

$q \in \mathsf{cf}$: Then we by (4.16) have for every $s \prec \theta(q)$ a $p_s \in \mathsf{cf}$ such that $\vdash_\delta p_s \leq q, \theta(p_s) = s$. Since in general S-closure is finite and thereby $\{s \mid s \prec \theta(q)\}$ too, we have by congruence $\vdash_\delta \sum_{\{s \prec \theta(q)\}} p_s \leq \sum_{\{s \prec \theta(q)\}} q$ and $\vdash_\delta q + \sum_{\{s \prec \theta(q)\}} p_s \leq q + \sum_{\{s \prec \theta(q)\}} q$. +4 gives us $\vdash_\delta q + \sum_{\{s \prec \theta(q)\}} q = q$, so $\vdash_\delta q + \sum_{\{s \prec \theta(q)\}} p_s \leq q$ and from +5 we then deduce $\vdash_\delta q + \sum_{\{s \prec \theta(q)\}} p_s = q$. Now let $p := q + \sum_{\{s \prec \theta(q)\}} p_s$. Then $\vdash_\delta p = q$ and $\Theta(p) = \{\theta(q)\} \cup \{\theta(p_s) \mid s \prec \theta(q)\} = \{\theta(q)\} \cup \{s \mid s \prec \theta(q)\} = \{s \mid s \preceq \theta(q)\} = \delta\theta(q) = \delta\{\theta(q)\} = \delta\{\theta(q') \mid q' \in \{q\}\} =$ (since $q \in \mathsf{cf}$) $\delta(\{\theta(q') \mid q' \in \mathsf{S}(q)\} = \delta\Theta(q)$.

$q \equiv q_1 + q_2$, $q_1, q_2 \in \mathsf{snf}$: By hypothesis of induction we know that there exists $p_i \in \mathsf{snf}.\ \vdash_\delta p_i = q_i, \Theta(p_i) = \delta\Theta(q_i)$ for $i \in \underline{2}$. Define $p := p_1 + p_2 \in \mathsf{snf}$. Then $\vdash_\delta p \equiv p_1 + p_2 = q_1 + q_2 \equiv q$ by congruence. Furthermore $\Theta(p) = \Theta(p_1 + p_2) = \{\theta(p') \mid p' \in \mathsf{S}(p_1) \cup \mathsf{S}(p_2)\} = \Theta(p_1) \cup \Theta(p_2) = \delta\Theta(q_1) \cup \delta\Theta(q_2) = \delta(\Theta(q_1) \cup \Theta(q_2)) = \ldots = \delta\Theta(q_1 + q_2) = \delta\Theta(q)$.

□

Finally we are ready to prove the completeness in full detail.

Proof (of proposition 4.3.2) From the previous lemmas and propositions we at first show:

$$p \in PL \Rightarrow \exists q \in \mathsf{msnf}.\ \vdash_\delta p = q, [\![p]\!]_\delta = \Theta(q) \tag{4.17}$$

Given $p \in PL$. From proposition 4.3.9 we find a $p_1 \in \mathsf{snf}.\ \vdash_\pi p = p_1$. Then from proposition 4.3.16 also a $p_2 \in \mathsf{pf}$ is obtained such that $\vdash_\pi p_1 = p_2$. Since $\vdash_\pi p = p_2$ and the proof system is sound we have $[\![p]\!]_\delta = [\![p_2]\!]_\delta$. Now by proposition 4.3.31 we find a $p_3 \in \mathsf{snf}$ such that $\vdash_\delta p_2 = p_3$ and $\delta\Theta(p_2) = \Theta(p_3)$. Furthermore proposition 4.3.21 gives us a $q \in \mathsf{msnf}.\ \vdash_\pi p_3 =_{si} q$. By corollary 4.3.19 then $\Theta(p_3) = \Theta(q)$. Collecting the facts we have $\vdash_\delta p = q$ and

$[\![p]\!]_\delta = [\![p_2]\!]_\delta$
$\quad = \delta[\![p_2]\!]_\pi$ by lemma 3.3.16
$\quad = \delta\Theta(p_2)$ by corollary 4.3.23 and the fact that $p_2 \in \mathsf{pf}$
$\quad = \Theta(p_3) = \Theta(q)$

thereby establishing (4.17)

Now for the completeness. Assume $[\![p]\!]_\delta \trianglelefteq_\delta [\![q]\!]_\delta$.
By (4.17) we can find $p', q' \in \mathsf{msnf}.\ \vdash_\delta p = p', \vdash_\delta q' = q$ and $[\![p]\!]_\delta = \Theta(p'), [\![q]\!]_\delta = \Theta(q')$. Then by definition of $\trianglelefteq_\delta$ we have $[\![p]\!]_\delta \trianglelefteq [\![q]\!]_\delta$ implies $\Theta(p') \subseteq \Theta(q')$ which by the definition of Θ is the same as $\{\theta(p'') \mid p'' \in \mathsf{S}(p')\} \subseteq \{\theta(q'') \mid q'' \in \mathsf{S}(q')\}$ or equivalently

$$\forall p'' \in \mathsf{S}(p') \exists q'' \in \mathsf{S}(q').\ \theta(p'') = \theta(q'') \tag{4.18}$$

Since $\theta(p'') = \theta(q'')$, $p'', q'' \in \mathsf{cf}$ by proposition 4.3.29 implies $\vdash_\pi p'' =_{nc} q''$ we see that (4.18) can be written as

$$\mathsf{S}(p') \subseteq_{nc} \mathsf{S}(q')$$

Because $p', q' \in \mathsf{msnf} \subseteq \mathsf{snf}$ we can use proposition 4.3.25 to deduce $\vdash_\pi p' + q' = q'$. By +5 we have $\vdash_\pi p' \leq p' + q'$ so $\vdash_\pi p' \leq q'$. Now p' and q' where found such that $\vdash_\delta p = p'$ and $\vdash_\delta q' = q$ wherefore $\vdash_\delta p \leq q$ by transitivity and we are done. □

Chapter 5

Adding Recursion to PL

In this chapter we shall extend our process language, PL, to include recursively defined processes and investigate their semantics as in chapter 3.
All concepts introduced in the previous chapters should be adjusted to the new set-up and also new concepts are needed. We will make good use of the results obtained in these chapters so as results and notions of Hennessy [Hen85b] which the reader will be assumed to be acquainted with. Also some notions of [Hen87a]) are used.

5.1 Denotational Semantics

In order to define interpretations of the recursively defined processes the Σ-po algebras are extended to Σ-domains. The signature remains the same, but the function symbol NIL of rank 0 is nominated the rôle of Ω in [Hen87a], i.e., the syntactical representative of the least element of the interpretations. The carriers are in principle the old ones but extended to include infinite sets as well. Formally:

Definition 5.1.1 For $\star$ in $\{\delta, \upsilon, \chi\}$ the carrier $C_\star$ is defined by:

$\star$) $C_\star = \{S \neq \emptyset \mid \exists T \subseteq TSW.\, S = \star(\pi T)\}$

$\square$

Notice that we have left out the restriction of finiteness. The elements of the $\star$-carriers are of course $\star$-closed so corollary 3.1.4 remains true in this set-up. Also the definition and results definition 3.1.5—corollary 3.1.11 carry over since the proofs there makes no use of the finiteness of the sets. The only deviation is of course in definition 3.1.8 where elements of C_π now may be infinite sets as well.
The purpose of the next propositions is to show that the carriers are algebraic complete partial orders (algebraic cpos for short).

Proposition 5.1.2 For each $\star$ in $\{\delta, \upsilon, \chi\}$ the pair $(C_\star, \trianglelefteq_\star)$ is an algebraic cpo with least element $\perp_{C_\star} = \{\varepsilon\}$ and every nonempty subset D of $C_\star$ has a lub $\bigsqcup_\star D = \star(\bigcup_{d \in D} d)$

or for short $\bigsqcup_\star D = \star\bigcup_{d\in D} d$. In the case of $\star$ in $\{\delta, \upsilon\}$ $\bigsqcup_\star D$ actually equals $\bigcup_{d\in D} d$. Furthermore the compact elements are the finite sets of $C_\star$.

Proof At first we show $(C_\star, \trianglelefteq)$ to be a cpo. Recall that $\trianglelefteq_\star$ simply is $\subseteq$. Already in chapter 3 it was noticed (though for finite sets) that $\{\varepsilon\}$ is the least element $\bot_{C_\star}$ of $C_\star$ and that $(C_\star, \subseteq)$ is a partial order. Now let D be a nonempty subset of $C_\star$.

$\bigsqcup_\star D = \star\bigcup_{d\in D} d \in C_\star$ is seen as follows: $d \in C_\star$ implies $d = \star(\pi T_d)$ for some $\emptyset \neq T_d \subseteq TSW$. Clearly $\bigcup_{d\in D} T_d \subseteq TSW$ and is nonempty since D is. Then $\star(\pi(\bigcup_{d\in D} T_d)) = \star\bigcup_{d\in D} \pi T_d =$ (in case of χ: corollary$_T$ 1.3.23) $\star\bigcup_{d\in D} \star(\pi T_d) = \star\bigcup_{d\in D} d$. Notice that we actually have

$$\bigsqcup\nolimits_\star D = \bigcup_{d\in D} d \text{ if } \star \text{ in } \{\delta, \upsilon\} \tag{5.1}$$

since δ and υ distributes over $\cup$.
Also $\bigsqcup D$ is a lub of D: Obviously $\bigsqcup_\star D$ is a ub for D. Suppose there exists a ub $e \in C_\star$ of D i.e., $\forall d \in D.\, d \subseteq e$. We shall show $\bigsqcup_\star D \subseteq e$ and do this by proving $t \in \bigsqcup_\star D$ implies $t \in e$. By (5.1) this is clear in the case of $\star$ in $\{\delta, \upsilon\}$. So we are left with the case $\star = \chi$. Now $t \in \bigsqcup_\chi D = \chi\bigcup_{d\in D} d$ implies $\exists d, d' \in D.\, t \in \chi\{d, d'\}$. By assumption $d, d' \subseteq e$ and since $e \in C_\chi$ it must be χ-closed, so $\chi\{d, d'\} \subseteq e$ and thereby also $t \in e$.

Since every directed set is nonempty we then have that $(C_\star, \subseteq)$ is a cpo.

Let $a \in C_\star$ be a finite set. We shall show that a is compact. That is for a directed set D such that $a \subseteq \bigsqcup_\star D$ there exist an $e \in D$ such that $a \subseteq e$. At first we prove

$$\forall t \in a \exists d_t \in D.\, t \in d_t$$

Assume $t \in a$. Since $a \subseteq \bigsqcup_\star D$ we have $t \in \bigsqcup_\star D$ wherefore $\exists d, d' \in D.\, t \in \star\{d, d'\}$. D directed implies that there exists a $d_t \in D$ such that $d, d' \subseteq d_t$. Because d_t is $\star$-closed we must have $\star\{d, d'\} \subseteq d_t$ and hence also $t \in d_t$.
Now since a is finite $D_a = \{d_t \mid t \in a\}$ must be finite too. The directedness of D implies there is a ub e of D_a in D and clearly $a \subseteq e$. So a is indeed compact.

Conversely we show that all compact elements are finite sets.
Suppose $a \in C_\star$ is an infinite set. Then there exists a infinite subset T of TSW such that $a = \star\pi T$. Since T is infinite it contains a countable infinite subset $T' = \{t_n\}_{n\in\mathbb{N}}$ of different tree semiwords. For all $n \in \mathbb{N}$ define

$$T_n = \textstyle\bigcup_{i\leq n}\{t_i\}$$
$$S_n = \{s \in T \mid \forall j > n.\, A_{t_j} \not\subseteq A_s\}$$
$$d_n = \star\pi(T_n \cup S_n)$$

At first we show that $D = \{d_n\}_{n\in\mathbb{N}}$ forms an increasing chain.
Since $T_n \subseteq T_{n+1}$ and $S_n \subseteq S_{n+1}$ it follows that $d_n \subseteq d_{n+1}$, so it remains to show that the chain is increasing i.e.,

$$\forall n \exists m > n.\, d_n \subset d_m \tag{5.2}$$

Let n be given. In general for an arbitrary finite set V of (finite) semiwords there is only finite many semiwords s such that there is a $t \in V$ with $A_s \subseteq A_t$. So because T_n is finite and T' is infinite there then exists a $t_m \in T'$ such that $\forall t \in T_n.\, A_{t_m} \not\subseteq A_t$. It follows that

$m > n$ and $t_m \in d_m$. Also $t_m \notin d_n$.
Assume on the contrary $t_m \in d_n$. Then there exists $s, t \in T_n \cup S_n$ such that $t_m \in \star\pi\{s, t\}$. So either $A_{t_m} \subseteq A_s$ or $A_{t_m} \subseteq A_t$. Consider $A_{t_m} \subseteq A_s$. s must come from T_n or S_n. If $s \in T_n$ this means $A_{t_m} \subseteq A_{t_i}$ for some $t_i \in T_n$. But this is impossible by the way t_m is chosen. If $s \in S_n$ we have $\forall j > n.\, A_{t_j} \not\subseteq A_s$, especially $A_{t_m} \not\subseteq A_s$ as $m > n$. Again a contradiction. The case $A_{t_m} \subseteq A_t$ is ruled out in the same way. Hence the assumption was false.
Now where were we know that D is an increasing chain we convince ourselves that $a \subseteq \bigsqcup_\star D$
To begin with we show:

$$\forall t \in T \exists n.\, t \in T_n \cup S_n \tag{5.3}$$

If $t \in T'$ then $t = t_n$ for some n and $t \in T_n$. So suppose $t \in T \setminus T'$. There is only finite many $t_i (\in T')$ with $A_{t_i} \subseteq A_t$. So choose n to be the i of the last t_i with $A_{t_i} \subseteq A_t$. Then $\forall j > n.\, A_{t_j} \not\subseteq A_t$ and so $t \in S_n$.
Next to show $a \subseteq \bigsqcup_\star D$ let an $s \in a$ be given. Since $\forall n.\, d_n \subseteq \bigsqcup_\star D$ it is enough to find a d_m such that $s \in d_m$ in order to have $s \in \bigsqcup_\star D$. $s \in a$ implies $\exists t, t' \in T.\, s \in \star\pi\{t, t'\}$. Using (5.3) we obtain n and n' for t and t' respectively. W.l.o.g. assume $n \leq n'$. Then $T_n \cup S_n \subseteq T_{n'} \cup S_{n'}$ and $t, t' \in T_{n'} \cup S_{n'}$ wherefore $\star\pi\{t, t'\} \subseteq d_{n'}$. So $s \in \star\pi\{t, t'\} \subseteq d_{n'}$ and we can chose $m = n'$ to get the desired d_m.

We can now return to the question of the compactness of a. Since a is assumed to be compact and $a \subseteq \bigsqcup_\star D$ there should exists a $d_n \in D$ such that $a \subseteq d_n$. By (5.2) there exists a m such that $d_n \subset d_m$ or by the proof of (5.2) $\exists t_m.\, t_m \notin d_n$. But this is a contradiction to $t_m \in T' \subseteq a$, so our assumption of a being an infinite set was wrong.

Knowing how the compact elements of $C_\star$ looks like it easier to show $(C_\star, \subseteq)$ algebraic. We shall show $\forall a \in C_\star.\, a = \bigsqcup_\star D_a$, where $D_a = \{d \mid d \subseteq a, d \text{ compact}\}$.

Since $\{\varepsilon\} \subseteq a$ for all a it follows that D_a is nonempty, so $\bigsqcup_\star D_a$ is defined. It is clear that $\bigsqcup_\star D_a \subseteq a$ as a is a upper bound for D_a. To see the other inclusion let a $t \in a$ be given. We prove $t \in \bigsqcup_\star D_a$. $t \in a = \star\pi T$ implies there exists $s, s' \in T$ such that $t \in \star\pi\{s, s'\}$. $\star\pi\{s, s'\} \in D_a$ follows from $\star\pi\{s, s'\} \subseteq a$ and the finiteness of $\star\pi\{s, s'\}$. In general $d \in D$ implies $d \subseteq \bigsqcup_\star D$ so we have $t \in \star\pi\{s, s'\} \subseteq \bigsqcup_\star D_a$. □

In order to see that we actually have obtained Σ-domains corollary 3.1.11 must be strengthened to:

Proposition 5.1.3 All $op_\star \in \Sigma_\star$ are (relative) continuous on $C_\star$ w.r.t. $\trianglelefteq_\star$ for each $\star$ in $\{\delta, \upsilon, \chi\}$.

Proof Since the operators of Σ_π are natural extensions to sets they evidently are continuous w.r.t. $\trianglelefteq_\pi$ ($\subseteq$). The proofs of (3.11) and (3.12) on page 83 can be carried over to infinite sets wherefore we get:

$$\star op_\pi(\overline{\star S}) = \star\star op_\pi(\overline{S}) \tag{5.4}$$

where $op_\pi \in \Sigma_{\star n}$ and $\overline{\star S} \in {C_\star}^n$. It is then an easy matter to show the operators to be continuous. E.g., suppose D is a nonempty subset of $C_\star$. Then

$$\begin{aligned} a._\star \bigsqcup_\star D &= \star a._\pi(\star \bigcup D) && \text{definition of } \bigsqcup_\star \text{ and proposition 3.1.9} \\ &= \star\star a._\pi \bigcup D && \text{by (5.4)} \\ &= \star\star \bigcup a._\pi D && a._\pi \text{ continuous} \\ &= \star \bigcup \star a._\pi D && \text{by the nature of } \star \\ &= \bigsqcup_\star a._\star D \end{aligned}$$

That we in general for a nonempty set, D, of sets of tree-semiwords have:

$$\chi\chi \bigcup D = \chi \bigcup \chi D \ (= \chi \bigcup_{S \in D} (\chi S))$$

follows from c) of corollary$_T$ 1.3.23. Similarly it is shown that $+_\star$ and $\|_\star$ are left and right continuous ($\|_\star$ under the usual proviso). □

Corollary 5.1.4 For each $\star$ in $\{\delta, \upsilon, \chi\}$ $A_\star = (C_\star, \trianglelefteq_\star, \Sigma_\star)$ is a Σ-domain.

We now proceed by defining the language, $RPL(X)$, of the recursive process terms. $RPL(X)$ can be considered as the extension of $PL(X)$ obtained by adding constructors for recursion.

The recursive terms over Σ, $RBL(X)$, is the terms obtained from the following schema:

$$t ::= NIL \mid x, x \in X \mid a.t, a \in Act \mid t + t \mid t \parallel t \mid rec\, x.\, t, x \in X$$

$RPL(X)$ is then defined to be those terms of $RBL(X)$ where every subterm t meets the usual requirement:

(5.5) $$t = t_1 \parallel t_2 \Rightarrow L(t_1) \cap L(t_2) = \emptyset$$

and the additional requirement:

(5.6) $$t = rec\, x.\, t' \Rightarrow L(x) = L(t')$$

where L is extended to $RBL(X)$ by defining $L(rec\, x.\, t) = L(x) \cup L(t)$.

$FV(t)$ is defined in the normal way to be the free variables in t. The recursive processes is the subset of $RPL(X)$ with no free variables and is denoted RPL.

Notice

i) $PL(X), PL \subseteq RPL(X)$ and $PL = PL(X) \cap RPL$.

ii) Since $rec\, x.\, p$ intuitively stands for the "solution" of the equation $x = p$ the requirement in c) of equal sorts of x and p is natural.

□

We shall refer to $PL(X) \subseteq RPL(X)$ as the syntactic finite process terms.

The definitive consequences of the restriction on the arguments to $\parallel$ become clear by the introduction of $rec\, x.$ as can be seen by the following example.

Example: Consider the term $p = a.NIL \parallel b.x \in PL(X)$. No matter what sort x might have $rec\,x.\,p$ cannot be a (legal) term of $RPL(X)$ because the sort of x must contain a and b in which case we would not have $p \in PL(X)$. As a consequence $rec\,x.$ can only prefix terms containing $\parallel$ if at most one of the arguments has nonempty sort. This means that only terms like $rec\,x.\,(NIL \parallel p) \parallel NIL$ are possible. On the other hand $RPL(X)$ can contain terms like $(rec\,x.\,a.x + b.x) \parallel (rec\,y.\,c.y + d.y)$.

The denotational maps $[\![\_]\!]_\delta$, $[\![\_]\!]_\upsilon$ and $[\![\_]\!]_\chi$ are given in the standard way by means of environments as described by Hennessy. Of course the environments shall be modified as the $A_\star$-assignments in chapter 4. Notice that $[\![\_]\!]_\star$ is independant of the environement when used on closed terms.

5.2 Operational Semantics

With the definitions and results of section 3.2 extended in the natural way to the new setting we can take over most of them. We will in the following briefly state the main differences.

$\overline{RBL}(X)$ is defined to be the least set C such that:

$$\begin{aligned} RBL(X) &\subseteq C \\ \bar{a}.t &\in C \quad \text{if } t \in C \text{ and } a \in Act \\ t_1 \parallel t_2 &\in C \quad \text{if } t_1, t_2 \in C \end{aligned}$$

and $RCL(X)$—the recursive configuration terms—are defined to be the terms of $\overline{RBL}(X)$ that satisfies (5.5) and (5.6) above. The recursive process configurations RCL are simply the closed terms of $RCL(X)$.

Of course definition 3.2.2 has to have a inference rule for $rec\,x.\,-$:

$$6)\quad \frac{p[rec\,x.\,p/x] \xrightarrow{a} q, a \in Act}{rec\,x.\,p \xrightarrow{a} q}$$

and the test configurations, TC, shall be changed to RTC in order to include recursive process configurations (in (t, p)). The test language, TL, remain unchanged. The rest of the corresponding section of chapter 3 extends smoothly.

5.3 Full Abstractness

The map $\bar{\theta} : CL \longrightarrow TSW$ associating tree-semiwords with configurations is extended to $RCL \longrightarrow TSW$ simply by letting $\bar{\theta}(p) = \varepsilon$ if $p \in RPL$ and otherwise keeping it's compositional definition (page 75).

In this section we shall also use the notions of algebraic relations and syntactic preorders as explained in [Hen87a].

For $t, t' \in RPL(X)$ we write $t \preceq t'$ to mean that t is a *syntactic approximation* to t' where $\preceq$ is defined to be the least (relative) Σ-precongruence over $RPL(X)$ which satisfies:

$$\begin{aligned} NIL &\preceq t \\ t[rec\,x.\,t/x] &\preceq rec\,x.\,t \end{aligned} \tag{5.7}$$

For every $t \in RPL(X)$, $\mathrm{Fin}(t)$ denotes $\{t' \in PL(X) \mid t' \preceq t\}$; i.e., $\mathrm{Fin}(t)$ is the syntactical finite approximations to t. $\preceq$ is extended to $RCL(X)$ by taking it to be the least (relative) Σ-precongruence over $RCL(X)$ which satisfies (5.7) above.

A relation R over RPL is *algebraic* if for all $t, u \in RPL$:

$$t\ R\ u \textit{ iff } \forall t' \in \mathrm{Fin}(t) \exists u' \in \mathrm{Fin}(u).\ t'\ R\ u'$$

In the following it will prove useful to be able to limit the number of experiments necessary to distinguish processes. To this end we first investigate the possibilities to reduce the size of a test t in an experiment (A, t) on a process q without affecting the outcome of the experiment.
Looking at definition 3.2.9 we get some ideas. As an example consider inference rule 6) and the test $\bar{a}.t$. If $p \stackrel{\bar{a}}{\not\longrightarrow}$ the test $a.\top$ would have the same outcome. Whether $p \stackrel{\bar{a}}{\longrightarrow}$ or $p \stackrel{\bar{a}}{\not\longrightarrow}$ depends naturally on p, but if we can find some criterions under which we can deduce $p \stackrel{\bar{a}}{\not\longrightarrow}$ for all $a \in Act$ we can certainly reduce the test.
Now if we have signaled the multiset of actions A getting to p (i.e., $p \in D(A, q)$) it should be clear that $p \stackrel{\overline{w}}{\longrightarrow} p'$ implies $|\overline{w}| \leq |A|$. Hence we can make the reduction whenever we are sure that at least $|A|$ signaled actions have been tested. I.e., if $\bar{a}.t$ is a subterm of the test t' in the experiment (A, t') and the "path" leading to $\bar{a}.t$ is $|A|$ long we can replace $\bar{a}.t$ by $\bar{a}.\top$. So this limits the necessary depth of a test t' in an experiment (A, t').

Another idea to reduce the set of experiments has it's roots in the same inference rule. Consider the same example as above. Clearly it makes no difference to the outcome of the test if we replace a with b in $\bar{a}.t$ as long as we are sure $p \stackrel{\bar{b}}{\not\longrightarrow}$.

These considerations leads to the following definition and proposition.

Definition 5.3.1 Given $c \in Act$ and $B \subseteq Act$ we successively for each $n \in \mathbb{N}$ define $f^n_{c,B} : TL \longrightarrow TL$ structurally as follows:

$$\begin{aligned}
n = 0\text{:} \qquad \top &\mapsto \top \\
\bar{a}.t &\mapsto \bar{c}.\top \\
t \mathbin{\Box} t' &\mapsto f^0_{c,B}(t) \mathbin{\Box} f^0_{c,B}(t') \quad \text{for } \Box \in \{\&, \nabla\} \\
n > 0\text{:} \qquad \top &\mapsto \top \\
\bar{a}.t &\mapsto \bar{b}.f^{n-1}_{c,B}(t) \qquad \text{where } b = \begin{cases} a & \text{if } a \in B \\ c & \text{otherwise} \end{cases} \\
t \mathbin{\Box} t' &\mapsto f^n_{c,B}(t) \mathbin{\Box} f^n_{c,B}(t') \quad \text{for } \Box \in \{\&, \nabla\}
\end{aligned}$$

□

It should be clear that $f^n_{c,B}$ is well-defined.

Notice

i) A subsequent test to $\bar{a}$. is discarted and $\top$ inserted when $f^0_{c,B}$ is applied.

ii) If $\bar{a}$ occurs in $f^n_{c,B}(t)$ then $a \in B \cup \{c\}$.

$\square$

Proposition 5.3.2 Let $p \in RPL$ and $(A, t) \in E$. If $c \notin L(p)$ then:

$$p \text{ may x } (A, t) \textit{ iff } p \text{ may x } (A, f^{|A|}_{c,L(A)}(t))$$

where x either is accept or reject.

Before we prove this proposition we need the following definition and lemma. We will define a function, $\overline{\text{ad}}$, which given a $p \in RCL$ gives an upper bound of the length of $w \in \overline{Act}^*$ where $p \xrightarrow{\bar{w}}$. Notice there must not be any initiation of actions in the sequence ($\bar{w} \in \overline{Act}^*$). Latter we need the action depth, ad, of a closed syntactical finite term ($\in PL$) too so we introduce this notion here too. It will also be convenient with a function, $\bar{L}$, which yields the label set corresponding to the actions signaled to initiate.

Definition 5.3.3 The *action depth*, ad, of a process and the *barred depth*, $\overline{\text{ad}}$, of a recursive process configuration is defined as follows:

$$\begin{aligned} \text{ad} : PL &\longrightarrow \mathbb{N} \\ NIL &\mapsto 0 \\ a.p &\mapsto 1 + \text{ad}(p) \\ p + q &\mapsto \max\{\text{ad}(p), \text{ad}(q)\} \\ p \parallel q &\mapsto \text{ad}(p) + \text{ad}(q) \end{aligned}$$

$$\begin{aligned} \overline{\text{ad}} : RCL &\longrightarrow \mathbb{N} \\ p &\mapsto 0 && \text{if } p \in RPL \\ \bar{a}.p &\mapsto 1 + \overline{\text{ad}}(p) \\ p \parallel q &\mapsto \overline{\text{ad}}(p) + \overline{\text{ad}}(q) \end{aligned}$$

and the map $\bar{L} : RCL \longrightarrow Act$ is given by:

$$\begin{aligned} p &\mapsto \emptyset && \text{if } p \in RPL \\ \bar{a}.p &\mapsto \{a\} \cup \bar{L}(p) \\ p \parallel q &\mapsto \bar{L}(p) \cup \bar{L}(q) \end{aligned}$$

$\square$

So $\overline{\text{ad}}$ actually estimates the necessary maximal "length" of a test.

The following lemma tells that nothing is lost in reducing the test as informally argued previously.

Lemma 5.3.4 For $p \in RCL$, $B \subseteq Act$ and $c \in Act$ such that $\overline{\mathrm{ad}}(p) \leq n$, $\bar{L}(p) \subseteq B$, $c \notin B$ we have:

$$(t, p) \longrightarrow^* \top \textit{ iff } (f^n_{c,B}(t), p) \longrightarrow^* \top$$

Proof

only if: Assume $(t, p) \longrightarrow^* \top$. The proof will be by induction on n.

$n = 0$: Then clearly $p \stackrel{\bar{b}}{\not\longrightarrow}$ for all $b \in Act$. We proceed by induction on the structure of t.

$t = \top$: Follows directly from $f^0_{c,B}(\top) = \top$.

$t = \bar{a}.t'$: Since $p \stackrel{\bar{a}}{\not\longrightarrow}$ we can exclude this case.

$t = t' \,\&\, t''$: By lemma 3.2.11 $(t' \,\&\, t'') \longrightarrow^* \top$ implies $(t', p) \longrightarrow^* \top$ and $(t'', p) \longrightarrow^* \top$. Since $f^0_{c,B}(t' \,\&\, t'') = f^0_{c,B}(t') \,\&\, f^0_{c,B}(t'')$ the result now follows using the hypothesis of induction and lemma 3.2.11.

$t = t' \nabla t''$: Similar.

$n > 0$: Again we use structural induction.

$t = \top$: Immediate.

$t = \bar{a}.t'$: $(\bar{a}.t', p) \longrightarrow^* \top$ implies $p \stackrel{\bar{a}}{\longrightarrow} p'$, $(t', p') \longrightarrow^* \top$. Clearly $\overline{\mathrm{ad}}(p') \leq n - 1$, $a \in \bar{L}(p') \subseteq B$ and by corollary 3.2.3 $L(p') \subseteq L(p)$. Hence $\bar{L}(p') \subseteq B$ and we can use the hypothesis of (structural or natural) induction to get $(f^{n-1}_{c,B}(t'), p') \longrightarrow^* \top$. Since $a \in B$ we have $f^n_{c,B}(\bar{a}.p') = \bar{a}.f^{n-1}_{c,B}(t')$ and from $p \stackrel{\bar{a}}{\longrightarrow} p'$ it then follows that $(f^n_{c,B}(\bar{a}.t'), p') \longrightarrow^* \top$.

$t = t' \,\&\, t'', t' \nabla t''$: Similar as in the case $n = 0$ using the hypothesis of structural induction.

if: Suppose $(f^n_{c,B}(t), p) \longrightarrow^* \top$ for some t. Again we use natural induction on n.

$n = 0$: As for the other implication we use structural induction.

$t = \top$: Trivial.

$t = \bar{a}.t'$: Then $f^0_{c,B}(t) = \bar{c}.\top$. Since $p \stackrel{\bar{d}}{\not\longrightarrow}$ for all $d \in Act$ when $\overline{\mathrm{ad}}(p) \leq 0$ and because $(\bar{c}.\top, p) \longrightarrow^* \top$ implies $p \stackrel{\bar{c}}{\longrightarrow} p'$ for some p' we can exclude this case.

$t = t' \,\&\, t''$: We have $f^0_{c,B}(t' \& t'') = f^0_{c,B}(t') \& f^0_{c,B}(t'')$. From the assumption then $(f^0_{c,B}(t') \& f^0_{c,B}(t''), p) \longrightarrow^* \top$ and from lemma 3.2.11 $(f^0_{c,B}(t'), p) \longrightarrow^* \top$ and $(f^0_{c,B}(t''), p) \longrightarrow^* \top$. Using the hypothesis and the same lemma we get $(t' \,\&\, t'', p) \longrightarrow^* \top$.

$t = t' \nabla t''$: Similar.

$n > 0$: Structural induction on t.

$t = \top$: Immediate.

$t = \bar{a}.t'$: Then $f^n_{c,B}(t)$ equals $\bar{b}.f^{n-1}_{c,B}(t')$, where $b = a$ if $a \in B$ and $b = c$ otherwise. $(\bar{a}.f^{n-1}_{c,B}(t'), p) \longrightarrow^* \top$ implies $(f^{n-1}_{c,B}(t'), p') \longrightarrow^* \top$ where $p \stackrel{\bar{b}}{\longrightarrow} p'$. But $p \stackrel{\bar{b}}{\longrightarrow} p'$ clearly implies $\bar{L}(p') \subseteq \bar{L}(p)$, $b \in \bar{L}(p) \subseteq B$ and $\overline{\mathrm{ad}}(p') \leq \overline{\mathrm{ad}}(p)$ so by induction $(t', p') \longrightarrow^* \top$. Since $b \in B$ and $c \notin B$ we have $b = a$, i.e., $p \stackrel{\bar{a}}{\longrightarrow} p'$ and thereby $(\bar{a}.t', p) \longrightarrow^* \top$.

$t = t' \,\&\, t'', t' \nabla t''$: Similar to the case $n = 0$.

□

Whit this lemma we can give the promised proof.

Proof of proposition 5.3.2.
Assume $p \in RPL$, $(A, t) \in E$ and $c \notin L(p)$. We show:

$$p \text{ may x } (A, t) \textit{ iff } p \text{ may x } (A, F^{|A|}_{c,L(A)}(t))$$

only if: Suppose p may x (A, t). Then $\exists q \in D(A, p). (t, q) \longrightarrow^* \top$. Clearly $q \in D(A, p)$ implies $\overline{\text{ad}}(q) = |A|$ and $\bar{L}(q) = L(A)$. Because $L(q) \subseteq L(p)$ and $c \notin L(p)$ we can then use lemma 5.3.4 to get $(f^{|A|}_{c,L(A)}(t), q) \longrightarrow^* \top$ and we are done for this implication.
if: Similar. □

The next step in reducing the number of experiments is to observe that we just as well can use test normal forms in the experiments. Finally notice that all duplicates in a test normal form can be removed without affecting the outcome of the test. This leads to the following definition.

Definition 5.3.5 $t \in TL$ is a *reduced test normal form iff*

a) t is a test normal form. I.e., $t = \&_{j \in \underline{n}} (\nabla_{k \in \underline{n_j}} \overline{w_{jk}} \top)$.

b) $\forall j \in \underline{n} \forall k, l \in \underline{n_j} k \neq l \Rightarrow w_{jk} \neq w_{jl}$.

c) $\forall i, j \in \underline{n}.\ i \neq j \Rightarrow \{w_{i1}, \ldots, w_{in_i}\} \neq \{w_{j1}, \ldots, w_{jn_j}\}$.

□

From lemma 3.2.11 we get:

Corollary 5.3.6

a) $t \,\&\, t \cong t$

b) $t \nabla t \cong t$

With this, proposition 3.2.14 and proposition 3.2.16 we easely get:

Proposition 5.3.7 For every $t \in TL$ there is a reduced test normal form $t' \in TL$ such that $t \cong t'$.

Proof At first we use proposition 3.2.16 to find a test normal form t'' such that $t \cong t''$. Then if there exists $j \in \underline{n}$ and $k, l \in \underline{n_j}$ such that $k \neq l$ and $w_{jk} = w_{jl}$ we use proposition 3.2.14 and proposition 5.3.6.b) to remove e.g., $\overline{w_{jk}} \top$. Iterating this we eventually get a test normal form which fulfills b) of definition 5.3.5. Finally use proposition 3.2.14.a)-d) and corollary 5.3.6.a) to obtain a reduced test normal form t' with $t'' \cong t'$. By transitivity then $t \cong t'$. □

Using the definition of may x with x equal to either accept or to reject it is easy to see that:

(5.8) $t \cong t'$ implies $\forall p \in RPL \forall A.\, p$ may x $(A, t) \Leftrightarrow p$ may x (A, t')

Definition 5.3.8 Let F_c denote the set $\{(A,t) \in E \mid t$ is a reduced test normal form and $\exists t' \in TL.\, t = f^{|A|}_{c,L(A)}(t')\}$.

For $p, q \in PL$ we then write:

$$p \sqsubseteq^{F_c}_{\mathrm{x}} q \text{ iff } \forall e \in F_c.\, p \text{ may x } e \Leftrightarrow q \text{ may x } e$$

□

So if we consider p, q and $c \notin L(p) \cup L(q)$ then F_c denotes according to the previous ideas a reduced set of experiments sufficient to characterize a testing preorder between p and q. Formally we have:

Proposition 5.3.9 Given $p, q \in RPL$ and $c \in Act$ such that $c \notin L(p) \cup L(q)$. Then

$$p \sqsubseteq_{\mathrm{x}} q \text{ iff } p \sqsubseteq^{F_c}_{\mathrm{x}} q$$

Proof

only if: Immediate since $F_c \subseteq E$.

if: Assume $p \sqsubseteq^{F_c}_{\mathrm{x}} q$ and p may x (A, t) for some $(A, t) \in E$. Since $c \notin L(p)$ we have according to proposition 5.3.2 p may x $(A, f^{|A|}_{c,L(A)}(t))$. By proposition 5.3.7 there exists a reduced test normal form t' such that $t' \cong f^{|A|}_{c,L(A)}(t)$, so by (5.8) p may x (A, t'). Inspecting the proof of proposition 5.3.7 and definition 5.3.5 we see that in the process of converging $f^{|A|}_{c,L(A)}(t)$ to t' we get a member of F_c. Consequently by our assumption, $p \sqsubseteq^{F_c}_{\mathrm{x}} q$, we see q may x (A, t'). Now using (5.8) we get q may x $(A, f^{|A|}_{c,L(A)}(t))$. Since $c \notin L(q)$ we can also use proposition 5.3.2 to obtain q may x (A, t). □

So far when trying to limit the set of experiments we have concentrated on the test part of it. We now search for conditions which can limit the set of actions to signal when the process is known to be syntactical finite. The following lemma gives some limits for a class of processes.

Lemma 5.3.10 Suppose $p \in PL$ (i.e., finite or equally contains no occurrences of $rec\, x.$) then there exists an $n \in \mathbb{N}$ such that p may x $(A, t) \Rightarrow L(A) \subseteq L(p)$, $|A| \leq n$.

Proof The first consequent $L(A) \subseteq L(p)$ is immediate from corollary 3.2.3 and independent of n. The second is seen from p being finite as follows:

Let n be the maximal action depth of p $(n = \mathrm{ad}(p))$ and suppose p may x (A, t). This means $\exists q \in D(A, p).\, (t, q) \longrightarrow^* \top/\bot$. By definition $q \in D(A, p)$ implies $\exists w \in W.\, A_w \cong A$, $p \stackrel{w}{\longrightarrow} q$. We show

$$p \stackrel{w}{\longrightarrow} q \Rightarrow |w| \leq \mathrm{ad}(p)$$

by natural induction on the size, $\mathrm{ad}(p)$, and the result follows.

$\mathrm{ad}(p) = 0$: Inspecting the definition of ad we see that $\mathrm{ad}(p) = 0$ implies p is either NIL or combinations of NIL through $+$ or $\|$. So $p \xrightarrow{w} q$ must mean $q = p$ and $w = \varepsilon$. But $|\varepsilon| = 0$ so we are done.

$\mathrm{ad}(p) > 0$: We use structural induction on p.

$p = NIL$: Then $p \xrightarrow{w} q$ implies $w = \varepsilon$—ok.

$p = a.p'$: $a.p' \xrightarrow{w} q$ implies $w = aw'$ and $p' \xrightarrow{w'} q'$ for some q' such that $\bar{a}.q' = q$. Since $\mathrm{ad}(p') \leq 1 + \mathrm{ad}(p') = \mathrm{ad}(p)$ we by hypothesis get $|w'| \leq \mathrm{ad}(p')$. Clearly then $|w| \leq \mathrm{ad}(p)$.

$p = p_1 + p_2$: W.l.o.g. assume $p \xrightarrow{w} q$ is due to $p_1 \xrightarrow{w} q$. By the hypothesis of structural induction $|w| \leq \mathrm{ad}(p_1)$. Since $\mathrm{ad}(p_1 + p_2) \geq \mathrm{ad}(p)_1$ we are done for this case.

$p = p_1 \| p_2$: The case $w = \varepsilon$ is trivial, so suppose $w = aw'$. Clearly $p_1 \| p_2 \xrightarrow{aw'} q$ implies *either* $p_1 = a.p_1'$ and $p_1' \| p_2 \xrightarrow{w'} q_1' \| q_2$, $q = \bar{a}.q_1' \| q_2$ *or* similar for p_2. Suppose the former is true. Then since $\mathrm{ad}(p_1' \| p_2) = \mathrm{ad}(p_1') + \mathrm{ad}(p_2) < \mathrm{ad}(a.p_1' \| p_2)$ we by hypothesis of natural induction get $|w'| \leq \mathrm{ad}(p_1' \| p_2)$. But $|w| = |aw'| = 1 + |w'| \leq 1 + \mathrm{ad}(p_1' \| p_2) = \mathrm{ad}(p)$ and we have concluded the inductive step.

□

The following statements will elucidate some of the (mainly operational) implications when two terms are related via the syntactic preorder.

Lemma 5.3.11

a) $p \preceq q, q \in RPL \Rightarrow p \in RPL$

b) $\bar{\theta}(p) \neq \varepsilon \Rightarrow p \neq NIL$

c) $p \xrightarrow{a} q \Rightarrow \bar{\theta}(q) \neq \varepsilon$

Proof
a) is proved by induction on the length of the proof of $p \preceq q$.

b) follows immediately from $NIL \in RPL$ and the definition of $\bar{\theta}$.

c) follows by induction on the number of rules $p \xrightarrow{a} q$ is obtained with. □

Because recursion constructors only occurs in processes $p \in RPL$ we cannot have $q = rec\, x.\, q'$ for a $q' \in RCL \setminus RPL$. This enables us to deduce:

Corollary 5.3.12 If $p \in RCL \setminus RPL$ then:

a) $p = \bar{a}.p' \preceq q$ implies $q = \bar{a}.q'$ where $p' \preceq q'$

b) $p = p_1 \| p_2 \preceq q$ implies $q = q_1 \| q_2$ where $p_i \preceq q_i$ for $i \in \underline{2}$

It cause no problems to prove by structural induction:

Corollary 5.3.13 If $p \in RCL$ then there is a $p' \in CL$ such that:

$$p' \preceq p \text{ and } \bar{\theta}(p') = \bar{\theta}(p)$$

Lemma 5.3.14 Suppose $\bar{\theta}(p_1 \parallel p_2) = \bar{\theta}(q_1 \parallel q_2)$ and $p_1 \preceq q_1$, $p_2 \preceq q_2$ for $p_1, p_2, q_1, q_2 \in RCL$. Then $\bar{\theta}(p_i) = \bar{\theta}(q_i)$ for $i \in \underline{2}$.

Proof Let arbitrary $p, q \in RCL$ be given. An easy induction on the length of the proof of $p \preceq q$ shows that $\bar{\theta}(p)$ must be a prefix of $\bar{\theta}(q)$:

$$p \preceq q \Rightarrow \bar{\theta}(p) \sqsubseteq \bar{\theta}(q)$$

Hence $\bar{\theta}(p_i) \sqsubseteq \bar{\theta}(q_i)$ for $i \in \underline{2}$. We cannot have $\bar{\theta}(p_1) \sqsubset \bar{\theta}(q_1)$ since this clearly would imply $\bar{\theta}(p_1 \parallel p_2) = \bar{\theta}(p_1) \parallel \bar{\theta}(p_2) \sqsubset \bar{\theta}(q_1) \parallel \bar{\theta}(q_2) = \bar{\theta}(q_1 \parallel q_2)$ contradicting the assumption of the lemma. So we must have $\bar{\theta}(p_1) = \bar{\theta}(q_1)$. In the same way we infer $\bar{\theta}(p_2) = \bar{\theta}(q_2)$. □

Proposition 5.3.15 For $w \in Act^*$, $p \in RPL$ and $q \in RCL$ we have:

$$p \xrightarrow{w} q$$
$$\Downarrow$$
$$\exists p' \in PL, q' \in CL.\, p \succeq p' \xrightarrow{w} q', \bar{\theta}(q') = \bar{\theta}(q)$$

Proof With the basic case trivial and lemma 5.3.16 in the inductive step we prove:

$$w \in Act^*, p, q \in RCL, p \xrightarrow{w} q \succeq q' \in CL \text{ and } \bar{\theta}(q) = \bar{\theta}(q')$$
$$\Downarrow$$
$$\exists p' \in CL.\, \bar{\theta}(p) = \bar{\theta}(p') \text{ and } p \succeq p' \xrightarrow{w} q'$$

by induction on $|w|$. From this the proposition follows using corollary 5.3.13 and lemma 5.3.11. □

Lemma 5.3.16 For $a \in Act$ and $p, q \in RCL$ we have:

$$P \xrightarrow{a} q \succeq q' \in CL \text{ and } \bar{\theta}(q) = \bar{\theta}(q')$$
$$\Downarrow$$
$$\exists p' \in CL.\, \bar{\theta}(p) = \bar{\theta}(p') \text{ and } p \succeq p' \xrightarrow{a} q'$$

Proof By a) of lemma 5.3.11 we know $p \xrightarrow{a} q$ only if $\bar{\theta}(q) \neq \varepsilon$, so from $\bar{\theta}(q) = \bar{\theta}(q')$ and b) of the same lemma we get $q' \neq NIL$. We will use this fact when proving the lemma by induction on the number, n, of rules used to infer $p \xrightarrow{a} q$.

$n = 1$: since $a \in Act$ only one rule comes into consideration: $p = a.r \xrightarrow{a} \bar{a}.r = q$. From $NIL \neq q' \in CL$ and $q' \preceq \bar{a}.r$ follows $q' = \bar{a}.r'$ where $r' \preceq r$ and $r' \in CL$. $p = a.r$ implies $r \in RPL$ and by $r' \preceq r$ and $r' \in CL$ then $r' \in PL$. With $p' = a.r'$ therefore $p' \in PL \subseteq CL$ and $p' \xrightarrow{a} q'$. Because both p and p' belongs to RPL we have $\bar{\theta}(p) = \varepsilon = \bar{\theta}(p')$.

$n > 1$: We consider each inference rule in turn.

$p = \bar{b}.r$ and $q = \bar{b}.s$ where $r \xrightarrow{a} s$: As above we deduce $q' = \bar{b}.s'$ where $s' \preceq s$ and $s' \in CL$. $b.\bar{\theta}(s) = \bar{\theta}(q) = \bar{\theta}(q') = b.\bar{\theta}(s')$ clearly implies $\bar{\theta}(s) = \bar{\theta}(s')$ so by hypothesis of induction then $\exists r' \in CL.\, \bar{\theta}(r) = \bar{\theta}(r'), r \succeq r' \xrightarrow{a} s'$. Choose $p' = \bar{b}.r' \in CL$. Then $\bar{\theta}(p') = b.\bar{\theta}(r') = b.\bar{\theta}(r) = \bar{\theta}(p), p' \preceq p$ and $p' \xrightarrow{a} q'$.

$p = p_1 + p_2 \xrightarrow{a} q$: Suppose $p_1 \xrightarrow{a} q$. By hypothesis of induction $\exists p_1' \in CL$ such that $\bar{\theta}(p_1) = \bar{\theta}(p_1'), p_1 \succeq p_1' \xrightarrow{a} q'$. Corollary 5.3.13 gives us a $p_2' \in CL$ such that $p_2' \preceq p_2$ and $\bar{\theta}(p_2) = \bar{\theta}(p_2)$. Then just choose $p' = p_1' + p_2'$. Similar if $p_2 \xrightarrow{a} q$.

$p = p_1 \parallel p_2 \xrightarrow{a} q$: W.l.o.g. we assume $q = q_1 \parallel p_2$ and $p_1 \xrightarrow{a} q_1$. Since $q' \neq NIL$ we must have $q' = q_1' \parallel q_2'$ where $q_1 \succeq q_1' \in CL$ and $p_2 \succeq q_2' \in CL$. From lemma 5.3.14 we see $\bar{\theta}(q_1') = \bar{\theta}(q_1)$ and $\bar{\theta}(q_2') = \bar{\theta}(p_2)$. By hypothesis of induction then $\exists p_1' \in CL.\, \bar{\theta}(p_1) = \bar{\theta}(p_1'), p_1 \succeq p_1' \xrightarrow{a} q_1'$. Let $p' = p_1' \parallel q_1' \in CL$. Then $\bar{\theta}(p') = \bar{\theta}(p_1') \parallel \bar{\theta}(q_1') = \bar{\theta}(p_1) \parallel \bar{\theta}(p_2) = \bar{\theta}(p)$ and $p \succeq p' \xrightarrow{a} q'$ as we wanted.

□

Proposition 5.3.17 If $w \in Act^*$, $p, q \in RPL$ and $r \in RCL$ then:

$$p \preceq q, p \xrightarrow{w} r \Rightarrow \exists s \in RCL.\, q \xrightarrow{w} s, \bar{\theta}(r) = \bar{\theta}(s)$$

Proof Proved along the lines of proposition 5.3.15 but using lemma 5.3.18 below in place of lemma 5.3.16. □

Lemma 5.3.18 If $a \in Act$ and $p, p' \in RCL$ then:

$$\bar{\theta}(p) = \bar{\theta}(p') \text{ and } p \succeq p' \xrightarrow{a} q'$$
$$\Downarrow$$
$$\exists q.\, p \xrightarrow{a} q \succeq q' \text{ and } \bar{\theta}(q) = \bar{\theta}(q')$$

Proof By induction on the length of the proof of $p' \preceq p$. There are three cases in the basis:

$p = p'$: Let $q = q'$.

$p' = NIL$: Then $p' \stackrel{a}{\not\longrightarrow}$ and the implication holds vacuously.

$p' = p''[rec\, x.\, p''/x]$ and $p = rec\, x.\, p''$: By the recursion rule $rec\, x.\, p'' = p \xrightarrow{a} q'$ follows directly from $p''[rec\, x.\, p''/x] \xrightarrow{a} q'$. Let $q = q'$.

Now for the inductive step we consider the inference rules one by one.

$p' \preceq p'', p'' \preceq p$: By induction on the structure of $p' \in RCL$ (from the definition of RCL as extracted from $\overline{RBL}$) we prove:

$$\bar{\theta}(p') = \bar{\theta}(p'') = \bar{\theta}(p)$$

$p' \in RPL$: By definition of $\bar{\theta}$ then $\bar{\theta}(p') = \varepsilon$ and so $\bar{\theta}(p) = \varepsilon$. Hence also $p \in RPL$. From $p'' \preceq p$ and lemma 5.3.11 then $p'' \in RPL$. Therefore $\bar{\theta}(p'') = \varepsilon = \bar{\theta}(p') = \bar{\theta}(p)$.

$p' = \bar{a}.r'$: By corollary 5.3.12 $p' \preceq p''$ then implies $p'' = \bar{a}.r''$ and $r' \preceq r''$. Using the corollary once more we get $p = \bar{a}.r$ where $r'' \preceq r$. From $a.\bar{\theta}(r) = \bar{\theta}(p') = \bar{\theta}(p) = a.\bar{\theta}(r)$ clearly $\bar{\theta}(r') = \bar{\theta}(r)$ so by the hypothesis of structural induction $\bar{\theta}(r') = \bar{\theta}(r'') = \bar{\theta}(r)$. Therefore also $\bar{\theta}(p'') = a.\bar{\theta}(r'') = a.\bar{\theta}(r') = \bar{\theta}(p') = \bar{\theta}(p)$.

$p' = p'_1 \parallel p'_2$: We can assume $p' \in RCL \setminus RPL$ since we already have dealt with the case $p' \in RPL$. Similar as above we then from corollary 5.3.12 get $p'' = p''_1 \parallel p''_1$ and $p = p_1 \parallel p_2$ where $p'_i \preceq p''_i \preceq p_i$ for $i \in \underline{2}$. From lemma 5.3.14 we then conclude $\bar{\theta}(p'_i) = \bar{\theta}(p_i)$ and the rest follow by induction as in the last case.

Now where we know $\bar{\theta}(p') = \bar{\theta}(p'') = \bar{\theta}(p)$ we can use the main hypothesis of induction to find a q'' such that $p'' \xrightarrow{a} q'' \succeq q'$ and $\bar{\theta}(q'') = \bar{\theta}(q')$. Again by induction $\exists q.\, p \xrightarrow{a} q \succeq q'', \bar{\theta}(q) = \bar{\theta}(q'')$. Then also $q' \succeq q$ and $\bar{\theta}(q) = \bar{\theta}(q')$.

$p' = b.r'$, $p = b.r$ and $r' \preceq r$: $b.r' \xrightarrow{a} q'$ implies $a = b$ and $q' = \bar{a}.q'$. From $p' = a.r'$ and $p = a.r$ follows $r, r' \in RPL$ so $\bar{\theta}(r') = \varepsilon = \bar{\theta}(r)$. Then choose $q = \bar{a}.r$ and we clearly have $q' \preceq q$ and $\bar{\theta}(q') = \bar{\theta}(q)$ so as $p = a.r \xrightarrow{a} \bar{a}.r = q$.

$p' = \bar{b}.r'$, $p = \bar{b}.r$ and $r' \preceq r$: $\bar{b}.r \xrightarrow{a} q'$ only if $r \xrightarrow{a} s'$ where $q' = \bar{b}.s'$. Then $\bar{\theta}(p) = \bar{\theta}(p')$ implies $\bar{\theta}(r) = \bar{\theta}(r')$ so by induction $r \xrightarrow{a} s$ for some $s \succeq s'$ with $\bar{\theta}(s) = \bar{\theta}(s')$. With $q = \bar{b}.s$ then $p = \bar{b}.r \xrightarrow{a} q \succeq \bar{b}.s' = q'$ and $\bar{\theta}(q) = \bar{\theta}(q')$.

$p' = p'_1 + p'_2$, $p = p_1 + p_2$ and $p'_i \preceq p_i$: W.l.o.g. we assume $p' \xrightarrow{a} q'$ derives from $p'_1 \xrightarrow{a} q'$. From the form of p and p' we deduce $p_1, p'_1 \in RPL$ and therefore $\bar{\theta}(p_1) = \varepsilon = \bar{\theta}(p'_1)$. By hypothesis of induction we get a q such that $p_1 \xrightarrow{a} q \succeq q'$ and $\bar{\theta}(q) = \bar{\theta}(q')$. Because $p \xrightarrow{a} q$ this case is settled.

$p' = p'_1 \parallel p'_2$, $p = p_1 \parallel p_2$ and $p'_i \preceq p_i$: Suppose $p' \xrightarrow{a} q'$ is due to $p'_1 \xrightarrow{a} q'_1$ where $q' = q'_1 \parallel p'_2$. From $\bar{\theta}(p) = \bar{\theta}(p')$, $p'_i \preceq p_i$ and lemma 5.3.14 we get $\bar{\theta}(p_i) = \bar{\theta}(p'_i)$ so by induction $p_1 \xrightarrow{a} q_1$ for some $q_1 \succeq q'_1$ with $\bar{\theta}(q_1) = \bar{\theta}(q'_1)$. Letting $q = q_1 \parallel p_2$ it follows from $q'_1 \preceq q_1$ and $p'_1 \preceq p_1$ that $q' \preceq q$. Also $\bar{\theta}(q) = \bar{\theta}(q_1) \parallel \bar{\theta}(p_2) = \bar{\theta}(q'_1) \parallel \bar{\theta}(p'_2) = \bar{\theta}(q')$. Because $p_1 \xrightarrow{a} q_1$ we by the rules for $\xrightarrow{a}$ directly have $p \xrightarrow{a} q$. The other case where $p'_2 \xrightarrow{a} q'_2$ is symmetric.

□

We are now in a position to prove the fundamental proposition:

Proposition 5.3.19 For $p \in RPL$, $e \in E$ we have:

$$p \text{ may x } e \Rightarrow \exists d \in \text{Fin}(p).\, d \text{ may x } e$$

Proof Assume $e = (A, t)$. p may x (A, t) implies that there exists $q \in D(A, p).(t, q) \longrightarrow^* o_x$ where $o_x = \top$ if x = accept and $o_x = \bot$ if x = reject. Now $q \in D(A, p)$ implies $p \xrightarrow{w} q$ for some w such that $A_w \cong A$. Then from proposition 5.3.15 above there exists $d \in \text{Fin}(p)$ and $q' \in RCL$ such that $d \xrightarrow{w} q'$ and $\bar{\theta}(q) = \bar{\theta}(q')$. Clearly $q' \in D(A, d)$ and from lemma 3.3.9 we get $(t, q) \longrightarrow^* o_x$ implies $(t, q') \longrightarrow^* o_x$. Hence d may x $(A, t) = e$. □

We take full advantage of the previous results in the proof of the following.

Proposition 5.3.20 Let d be a syntactical finite process (i.e., $d \in PL$ and so contains no occurrences of rec.) and $p \in RPL$. If $d \sqsubseteq_{\mathrm{x}} p$ then $d \sqsubseteq_{\mathrm{x}} d'$ for some $d' \in \mathrm{Fin}(p)$ (x = a/ accept or x = r/ reject).

The proof is much like an equivalent proof of Hennessy with some minor adjustments to our set-up.

Proof Since $L(q)$ in general for $q \in RPL$ is finite there is a finite set $B \subseteq Act$ such that $L(d), L(p) \subseteq B$. Because Act is infinite we can chose a $c \notin B$. According to proposition 5.3.9 we have $q \sqsubseteq_{\mathrm{x}} q'$ *iff* $q \sqsubseteq_{\mathrm{x}}^{F_c} q'$ for arbitrary q, q' with $L(q), L(q') \subseteq B$. Now because d is finite lemma 5.3.10 ensures us a n such that d may x $(A,t) \Rightarrow L(A) \subseteq L(d)$, $|A| \leq n$. So let $F_c(d) = \{(A,t) \in F_c \mid L(A) \subseteq L(d), |A| \leq n\}$. Clearly then d may x e and $e \in F_c$ implies $e \in F_c(d)$. Therefore to show $d \sqsubseteq_{\mathrm{x}} q$ for a q with $L(q) \subseteq B$ it is sufficient to show q may x e for those e in $F_c(d)$ such that d may x e. Let F'_c denote this subset of $F_c(d)$.

A simple argument shows that $F_c(d)$ is finite and hence also F'_c. Since $L(d)$ is finite there is only finite many multisets A with $L(A) \subseteq L(d)$ and $|A| \leq n$. Also for a given finite A there can only be finite many t's with $(A,t) \in F_c$. $(A,t) \in F_c$ implies t is a test normal form i.e., of the form $t = \&_{j \in \underline{n}}(\nabla_{k \in \underline{n_j}} \bar{w}_{jk} \top)$. Furthermore since there exists a t' such that $t = f^{|A|}_{c,L(A)}(t')$ we must have $|w_{jk}| \leq |A|+1$ and $L(w_{jk}) \subseteq L(A) \cup \{c\}$. There is only finitely many strings with this property. Since there is no duplicates of the strings in $\nabla_{k \in \underline{n_j}} \bar{w}_{jk} \top$ there can only be finitely many on this form. Similar we see that there are finitely many tests of the form $\&_{j \in \underline{n}}(\nabla_{k \in \underline{n_j}} \bar{w}_{jk} \top)$. Hence a finite number of $e \in F_c$.

By the assumption of the lemma we know $d \sqsubseteq_{\mathrm{x}}^{F_c} p$ and so p may x e for every $e \in F'_c$. From the previous proposition (proposition 5.3.19) we find a $d(e) \in \mathrm{Fin}(p)$ for every $e \in F'_c$. $D = \{d(e) \mid e \in F'_c\}$ must be finite since F'_c has the same property. Then because $D \subseteq \mathrm{Fin}(p)$ and $\mathrm{Fin}(p)$ is directed we can take d' to be an upper bound of D and d' may x e for every $e \in F'_c$. In general $q \in \mathrm{Fin}(p)$ implies $L(q) \subseteq L(p)$ and it follows that $L(d') \subseteq B$ so we have $d \sqsubseteq_{\mathrm{x}} d'$. □

Proposition 5.3.21 The test preorders $\sqsubseteq_{\mathrm{a}}$ and $\sqsubseteq_{\mathrm{r}}$ extends $\preceq$ on RPL. I.e., $\preceq \subseteq \sqsubseteq_{\mathrm{a}}, \sqsubseteq_{\mathrm{r}}$.

Proof Suppose $p, q \in RPL$ and $p \preceq q$. Given $(A,t) \in E$ such that p may x we shall show q may x in order order to have $p \sqsubseteq_{\mathrm{x}} q$. Similar as in the proof of proposition 5.3.19 we see it is enough to show

$$\forall p' \in D(A,p) \exists q' \in D(A,q).\, \bar{\theta}(p') = \bar{\theta}(q')$$

But this follows immediately from proposition 5.3.17. □

Proposition 5.3.22 The test preorders $\sqsubseteq_{\mathrm{x}}$ are algebraic over RPL, where x either is a, r or left out.

With the results obtained so far the proof is very similar to a corresponding proof of Hennessy.

Proof Because $\lesssim_a$ and $\lesssim_r$ extends $\preceq$ it appears from the next proposition that $\lesssim$ is algebraic when $\lesssim_a$ and $\lesssim_r$ are. So assume $\mathrm{x} = \mathrm{a}$ or $\mathrm{x} = \mathrm{r}$ in the following.

Suppose $p \lesssim_x q$ and $d \in \mathrm{Fin}(p)$. We must find a $d' \in \mathrm{Fin}(q)$ such that $d \lesssim_x d'$. $d \in \mathrm{Fin}(p)$ implies $d \preceq p$. Then $d \lesssim_x p$ because $\lesssim_x$ extends $\preceq$ on RPL. Hence $d \lesssim_x q$ and by proposition 5.3.20 $d \lesssim_x d'$ for some $d' \in \mathrm{Fin}(q)$.

Conversely suppose for every $d \in \mathrm{Fin}(p)$ there exists a $d' \in \mathrm{Fin}(q)$ such that $d \lesssim_x d'$. We shall show $p \lesssim_x q$. At first we deduce $d \lesssim_x q$ for every $d \in \mathrm{Fin}(q)$. By assumption $d \lesssim_x d'$ for some $d' \in \mathrm{Fin}(q)$. $d' \in \mathrm{Fin}(q)$ only if $d' \preceq q$ and thus $d' \lesssim_x q$. From the transitivity of $\lesssim_x$ then $d \lesssim_x q$. Now to see $p \lesssim_x q$ suppose p may x e for some $e \in E$. By proposition 5.3.19 there is some $d \in \mathrm{Fin}(p)$ such that d may x e. From $d \lesssim_x q$ finally q may x e. □

The following proposition is actually more general than needed in the previous proof. In this proposition and a few others to follow we shall as in [Hen87a] use REC_Σ to denote the recursive terms that can be build from recursive combinators and a signature Σ. $FREC_\Sigma$ is just the syntactic finite terms.

Proposition 5.3.23 Suppose $\sqsubseteq'$ and $\sqsubseteq''$ are transitive relations over REC_Σ and $\sqsubseteq$ is defined as the intersection of $\sqsubseteq'$ and $\sqsubseteq''$. If $\sqsubseteq'$ and $\sqsubseteq''$ extends $\preceq$ and they both are algebraic then $\sqsubseteq$ is algebraic too.

Proof We shall show for $p, q \in REC_\Sigma$ that $p \sqsubseteq q$ *iff* $\forall d \in \mathrm{Fin}(p) \exists d' \in \mathrm{Fin}(q).\, d \sqsubseteq d'$.

if: $\forall d \in \mathrm{Fin}(p) \exists d' \in \mathrm{Fin}(q).\, d \sqsubseteq d'$ implies $\forall d \in \mathrm{Fin}(p) \exists d' \in \mathrm{Fin}(q).\, d \sqsubseteq' d', d \sqsubseteq'' d'$. Since $\sqsubseteq'$ and $\sqsubseteq''$ are algebraic this in turn implies $p \sqsubseteq' q$ and $p \sqsubseteq'' q$. By definition of $\sqsubseteq$ then $p \sqsubseteq q$.

only if: $p \sqsubseteq q \Rightarrow p \sqsubseteq' q, p \sqsubseteq'' q$. By the algebraicity of $\sqsubseteq'$ and $\sqsubseteq''$ the consequence of the implication gives: $\forall d \in \mathrm{Fin}(p) \exists d' \in \mathrm{Fin}(q).\, d \sqsubseteq' d'$ and $\forall d \in \mathrm{Fin}(p) \exists d'' \in \mathrm{Fin}(q).\, d \sqsubseteq'' d''$ from which we get: $\forall d \in \mathrm{Fin}(p) \exists d', d'' \in \mathrm{Fin}(q).\, d \sqsubseteq' d', d \sqsubseteq'' d''$. Because $\mathrm{Fin}(q)$ is directed under $\preceq$ there is an $e \in \mathrm{Fin}(q)$ such that $d' \preceq e$ and $d'' \preceq e$. Since $\sqsubseteq'$ and $\sqsubseteq''$ both extends $\preceq$ this means $d' \sqsubseteq' e$ and $d'' \sqsubseteq'' e$. From the transitivity of $\sqsubseteq'$ and $\sqsubseteq''$ then $d \sqsubseteq' e$ and $d \sqsubseteq'' e$ so by definition of $\sqsubseteq$ finally $d \sqsubseteq e$. □

Due to standard results as found in [Hen87a] it is now possible with a little elaboration on the denotational aspects to get the full abstractness results.

Hennessy shows the general corollary: $A[\![t[rec\,x.\,t/t]\!] = A[\![rec\,x.\,t]\!]$ from which it is quite easy to see:

Proposition 5.3.24 $\leq_A$ extends $\preceq$ on $REC_\Sigma(X)$. I.e.,

$$t \preceq t' \text{ implies } \forall \rho_A.\, A[\![t]\!]\rho_A \leq_A A[\![t]\!]\rho_A$$

Proposition 5.3.25 Given a Σ-domain, A, assume the functions preserve $\mathrm{Fin}(A)$. Then the denotational preorder, $\leq_A$, arrising from $A[\![]\!]$ is algebraic on REC_Σ. I.e.,

$$A[\![p]\!] \leq_A A[\![q]\!] \textit{ iff } \forall d \in \mathrm{Fin}(p) \exists e \in \mathrm{Fin}(q).\, A[\![d]\!] \leq_A A[\![e]\!]$$

Proof
if: A consequence of the proposition above is $\forall e \in \text{Fin}(q).\ A[\![e]\!] \leq_A A[\![q]\!]$ so from the assumption of the implication we get $\forall d \in \text{Fin}(p).\ A[\![d]\!] \leq_A A[\![q]\!]$. Hence also $\bigvee_A A[\![\text{Fin}(p)]\!] \leq_A A[\![q]\!]$. Since $\bigvee_A A[\![\text{Fin}(p)]\!]$ by Hennessy equals $A[\![p]\!]$ this actually reads $A[\![p]\!] \leq_A A[\![q]\!]$.

only if: $d \in \text{Fin}(p)$ and $p \in REC_\Sigma$ implies $d \in FREC_\Sigma$. So if $A[\![]\!]$ on elements of $FREC_\Sigma$ yields elements of $\text{Fin}(A)$ (i.e., finite elements) we get this implication as follows:
A is a Σ-domain and $\forall d \in \text{Fin}(p).\ A[\![d]\!] \leq_A A[\![p]\!] \leq_A A[\![q]\!]$ so $\forall d \in \text{Fin}(p).\ A[\![d]\!] \leq_A A[\![q]\!]$. Because $A[\![q]\!] = \bigvee_A A[\![\text{Fin}(q)]\!]$, $\text{Fin}(q)$ is directed and $A[\![d]\!]$ is assumed to denote a finite element there exists an $e \in \text{Fin}(q)$ such that $A[\![d]\!] \leq_A A[\![e]\!]$.

We owe to show $t \in FREC_\Sigma$ implies $A[\![t]\!] \in \text{Fin}(A)$. Using as hypothesis the assumption of the proposition that $\forall f \in \Sigma$ of arity k we have $f_A(A[\![\bar{t}\,]\!]) \in \text{Fin}(A)$ where $A[\![\bar{t}\,]\!] \in \text{Fin}(A)^k$ this easely follows by induction on the structure of t. □

From the results in chapter 3 and the characterization of the finite elements in proposition 5.1.2 it is seen that $op_\star$ preserve finite elements in $C_\star$ when $\star \in \{\delta, \upsilon, \chi\}$. We then have:

Corollary 5.3.26 The denotational preorders $\trianglelefteq_\delta$, $\trianglelefteq_\upsilon$ and $\trianglelefteq_\chi$ are algebraic on RPL.

Notice that from the proof of proposition 5.3.25 above it appears that $[\![\_]\!]_\star$ denotes finite elements when restricted to PL and by 3.3.19 on page 85 all finite elements are denotable by terms of PL so our different domains are actually finitary.

With the corollary it is now an easy matter to show the denotational models are fully abstract w.r.t. the corresponding preorders.

Theorem 5.3.27 If $p, q \in RPL$ then the different test preorders $\sqsubsim_{\mathrm{a}}$, $\sqsubsim_{\mathrm{r}}$ and $\sqsubsim$ are (relative) precongruences and:

δ) $[\![p]\!]_\delta \trianglelefteq_\delta [\![q]\!]_\delta$ *iff* $p \sqsubsim_{\mathrm{a}} q$

υ) $[\![p]\!]_\upsilon \trianglelefteq_\upsilon [\![q]\!]_\upsilon$ *iff* $p \sqsubsim_{\mathrm{r}} q$

χ) $[\![p]\!]_\chi \trianglelefteq_\chi [\![q]\!]_\chi$ *iff* $p \sqsubsim q$

Proof From the last corollary we know that $\trianglelefteq_\delta$, $\trianglelefteq_\upsilon$ and $\trianglelefteq_\chi$ are algebraic and from proposition 5.3.22 we also know that this is the case for $\sqsubsim_{\mathrm{a}}$, $\sqsubsim_{\mathrm{r}}$ and $\sqsubsim$ so by the corresponding result for syntactic finite processes, theorem 3.3.1, $\delta) - \chi)$ then follows.

The test preorders are seen to be precongruences because they now are know to agree with the corresponding denotational preorders which in turn are precongruence since the denotational maps are built from (relative) continuous and thereby monotone operators. □

Index

Constants

Functions

Operations

Relations

Alphabetical

Part II

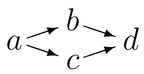
Tracing Partial Orders

Chapter 6

Pomsets

As mentioned in the presentation, the concept of labelled partial orders will be central for the models we are going to present. The basic idea is that labelled partial orders will represent individual behaviours of processes. In particular we will look at pomsets. We shall use the interpretation and graphical representation of pomsets from [Gra81]. That is

(6.1)



is used to represent a behaviour of a process with four action occurrences, where the d occurrence is causally dependent on the others, the b occurrence is causally dependent on a, but not on c, a.s.o.

6.0 Basic Definitions

Pomsets are usually defined as isomorphism classes of labelled partial orders ([Gis88, Pra86]). We will look at labelled partial orders, also known as labelled posets, over an action alphabet Δ—a countably infinite alphabet (fixed through out the rest of this part of the thesis). We assume Δ to be disjoint from $I\!N$—the nonnegative integers.

The labelled partial orders are defined on basis of a fixed ground set which is assumed to be closed under pairing and containing $I\!N$ and Δ (See e.g., [Hen87c] for a solution to the simple set equation $S = I\!N \cup \Delta \cup (S \times S)$).

Definition 6.0.1 *Labelled Poset*

A subset, X, of the ground set together with a partial order (reflexive, transitive and antisymmetric), $\leq$, and a labelling function $\ell : X \longrightarrow \Delta$ is called a *labelled poset* (lpo for short) and denoted $\langle X, \leq, \ell \rangle$.
Given a lpo p then $X_p = X$, $\leq_p = \leq$ and $\ell_p = \ell$ if $p = \langle X, \leq, \ell \rangle$.
The set of all such lpos is denoted LPO.

Given two lpos, a *morphism* $f : \langle X, \leq, \ell \rangle \longrightarrow \langle X', \leq', \ell' \rangle$ *of labelled posets* is a function $f : X \longrightarrow X'$ such that

$$x \leq y \Rightarrow f(x) \leq' f(y) \quad \text{for all } x, y \in X$$
$$\ell(x) = \ell'(f(x)) \quad \text{for all } x \in X$$

An *isomorphism* $f : p \longrightarrow q$ *of labelled posets* is a bijection $f : X_p \longrightarrow X_q$ such that f and f^{-1} are morphisms of labelled posets (then also $x \leq_p y$ *iff* $f(x) \leq_q f(y)$).
If such a isomorphism exists between p and q we write $p \cong q$.

The empty lpo, $\langle \emptyset, \emptyset, \emptyset \rangle$, is denoted ε. □

That *LPO* indeed is a set follows from the ground set being one. Observe that we use $x, y, \ldots$ to range over elements of X_p, where p is a lpo.

x and y are said to be concurrent/causally independent in a lpo p,

$$x \; co_p \; y \; \textit{iff} \; x \not\leq_p y \; \textit{and} \; y \not\leq_p x$$

Notice that co_p is *not* reflexive! We say that $Y \subseteq X_p$ is a co_p*-set* if all the elements of Y are concurrent in p, i.e., if $co_p|_{Y^2} = (Y \times Y) \setminus \{\langle y, y \rangle \mid y \in Y\}$ or alternatively if $\leq_p|_{Y^2} = \{\langle y, y \rangle \mid y \in Y\}$.

If Y is a set and $p = \langle X, \leq, \ell \rangle$ an lpo then the *restriction of* p *to* Y, $p|_Y$, is the lpo $\langle X|_Y, \leq|_{Y^2}, \ell|_Y \rangle$.
For $x \in X_p$ we sometimes (ambiguously) abbreviate $p|_{\{x\}}$ by x.

The definition of pomsets emerge almost immediately from that of lpos.

Definition 6.0.2 *Pomsets*

The equivalence class of a lpo p under $\cong$ is denoted $[p]$ and p is called a *representative* of the equivalence class. I.e., $[p] = \{q \in LPO \mid q \cong p\}$. Whenever an lpo is denoted by a single symbol, p, we define for convenience $\mathbf{p}$ to be $[p]$.

The set of all pomsets is then the quotient set of *LPO* by $\cong$, $LPO/\cong$ and is denoted $\mathbf{P}$.

A pomset $\mathbf{p}$ is contained in pomset $\mathbf{q}$ if a representative of $\mathbf{p}$ can be embedded in a representative of $\mathbf{q}$. Formally: $\mathbf{p}$ is a *subpomset* of $\mathbf{q}$, written $\mathbf{p} \hookrightarrow \mathbf{q}$, *iff* $\exists Y.\, \mathbf{p} = [q|_Y]$. □

We have defined the notion of subpomset by means of a single representative so one should check that the definition is independent of what representative used in the definition. E.g., if $q \cong q'$ it is easy to see that $q|_Y \cong q'|_{Y'}$ where Y' is the subset of $X_{q'}$ isomorphic to $Y \cap X_q$ under the lpo isomorphism holding between q and q'. It will often be left to the reader to check that definitions regarding pomsets are well-defined in this sense.

For a pomset $\mathbf{p}$ and a set of pomsets Q we denote by $Q(\mathbf{p})$ those pomsets of Q which are contained in $\mathbf{p}$, i.e., $Q(\mathbf{p}) = \{\mathbf{q} \in Q \mid \mathbf{q} \hookrightarrow \mathbf{p}\}$.

Example: If $\mathbf{p}$ is the pomset represented in (6.1) then e.g.,

$$\mathbf{p} \hookrightarrow \mathbf{p}, \quad a \rightarrow c \rightarrow d \hookrightarrow \mathbf{p}, \quad a \rightarrow d \hookrightarrow \mathbf{p}$$

and

$$\left\{ c, a \rightarrow d, \begin{smallmatrix} b \searrow \\ c \nearrow \end{smallmatrix} d \right\} \subseteq \mathbf{P}(\mathbf{p})$$

We overload notation and use ε and a to denote the empty pomset $[\langle\emptyset,\emptyset,\emptyset\rangle]$ and the singleton pomset $[\langle\{a\},\{\langle a,a\rangle\},a\mapsto a\rangle]$ respectively. Similarly a^n denote the multisingleton pomset $[\langle\{\langle a,k\rangle \mid 1\leq k\leq n\},\{\langle x,x\rangle \mid x=\langle a,k\rangle, 1\leq k\leq n\},\langle a,k\rangle\mapsto a\rangle]$.

For a set of pomsets P we adopt the notation P_ε for $P\cup\{\varepsilon\}$.

Below we list different types of pomsets we shall deal with together with the symbols we tend to use for them.

$A,B,\ldots\in\mathsf{M}$	—	the *multiset* pomsets: $\{\mathbf{p}\in\mathbf{P}\mid \mathbf{p}\neq\varepsilon$ *and* X_p is a co_p-set$\}$
$A,B,\ldots\in\mathsf{S}$	—	the *set* pomsets: $\{\mathbf{p}\in\mathsf{M}\mid \forall x,y\in X_p.\, x\neq y\Rightarrow \ell_p(x)\neq\ell_p(y)\}$
$a^n,b^m,\ldots\in\mathsf{N}$	—	the *multisingle multisingleton* $\{\mathbf{p}\in\mathsf{M}\mid \forall x,y\in X_p.\, \ell_p(x)=\ell_p(y)\}$
$a,b,\ldots\ \in\Delta$	—	the *singleton* pomsets: $\{\mathbf{p}\in\mathsf{M}\mid \forall x,y\in X_p.\, x=y\}$

Notice that we by this notation have $a=a^1=\{a\}$. The reader is obliged to sort out from the context the ambiguity arising from this notation in return for a more tractable presentation. The reader should also be aware that the sets $\Delta,\mathsf{N},\mathsf{S}$ and M are defined not to contain the empty pomset ε. As already stated e.g., M is augmented with ε by writing M_ε. In continuation with the notation above we then also have $\varepsilon=a^0=\emptyset$.

It will not be necessary to deal with infinite pomsets in the following so we will throughout the rest of this part assume *pomsets* to be *finite*. More precisely: we shall only consider pomsets $\mathbf{p}$ where X_p is finite.

Having restricted ourselves to finite pomsets we can now for a pomset associate a unique multiplicity function over Δ which for each action tells how many elements in the pomsets that are labelled whit this action.

Definition 6.0.3 *Multiplicity Function*

A *multiplicity function*, m, (over Δ) is a map $m:\Delta\longrightarrow I\!N$.
m is said to be *finite* if m is 0 everywhere except on a finite subset of Δ.
The set of multiplicity functions are partially ordered by

$$m\leq m' \text{ iff } \forall a\in\Delta.\, m(a)\leq m'(a)$$

Given a lpo p the associated multiplicity function, m_p, is defined by $\forall a\in\Delta.\, m_p(a)=|\{x\in X_p\mid \ell_p(x)=a\}|$.
The multiplicity function, $m_{\mathbf{p}}$, of a pomset $\mathbf{p}$ is simply m_p.
The preoder induced on pomsets by the partial order $\leq$ on multiplicity functions is (ambiguously) denoted $\leq$ and defined by $\mathbf{p}\leq\mathbf{q}$ *iff* $m_{\mathbf{p}}\leq m_{\mathbf{q}}$. □

It is easy to see that every finite set M of multiplicity functions has a lub (least upper bound) $\bigvee M=m'$ where m' is given by $\forall a\in\Delta.\, m'(a)=\max_{\leq}\{m(a)\mid m\in M\}$. If in addition every $m\in M$ is finite then so is $\bigvee M$. Also $m_{\mathbf{p}}$ is finite for every $\mathbf{p}\in\mathbf{P}$ because we only deal with finite pomsets.

Observe that multisets are nothing else than a pomset representation of multiplicity functions. It is mainly for convenience that we have chosen to work with both notions.

Definition 6.0.4 *Pomset Property*

A lpo property, P_*, is $\cong$-invariant if it is preserved under lpo isomorphism:

$$p \cong q, P_*(p) \text{ implies } P_*(q)$$

P_* is a *pomset property* if it is induced from a $\cong$-invariant lpo property, Q_*, in the following way:

$$P_*(\mathbf{p}) \textit{ iff } Q_*(p)$$

□

Observe that the $\cong$-invariance ensures the notion of pomset property to be well-defined. In the sequel we shall make no distinction between a pomset property and the lpo property it is induced from.

An example of a pomset property, P_*, is where $P_*(\mathbf{p})$ demands $\leq_p$ to satisfy the trichotomy law: $\forall x, y \in X_p.\ x \leq_p y \textit{ or } y \leq_p x$, i.e., that $\leq_p$ shall be total. The set of pomsets having this property is denoted W (words) and we write the property as P_{W}. Pomsets of W are by Gischer [Gis88] alternatively called tomsets. We shall often write w for $\mathbf{w} \in W$, because of the one to one correspondence between Δ^* and W (see [Sta81]).

We now give an example of a type of pomset property that can be defined in terms of a set of nonempty multisets.

Definition 6.0.5 *Multiset Induced Pomset Property*

Given $\mathsf{D} \subseteq \mathsf{M}$ we say that a pomset $\mathbf{p}$ has the $P_{\mathsf{M}\subseteq\mathsf{D}}$-*property* if the (nonempty) multisets contained in $\mathbf{p}$ are from D. Formally

$$P_{\mathsf{M}\subseteq\mathsf{D}}(\mathbf{p}) \textit{ iff } \mathsf{M}(\mathbf{p}) \subseteq \mathsf{D}$$

The $P_{\mathsf{M}\subseteq\mathsf{D}}$-*pomsets* are those with the $P_{\mathsf{M}\subseteq\mathsf{D}}$-property and they are denoted $\mathbf{P}_{\mathsf{M}\subseteq\mathsf{D}}$. □

It is easy to see that $P_{\mathsf{M}\subseteq\mathsf{D}}$ actually is a pomset property because it is induced from the lpo property:

$$P_{\mathsf{M}\subseteq\mathsf{D}}(\mathbf{p}) \textit{ iff } [p|_Y] \in \mathsf{D} \text{ for every nonempty } co_p\text{-set } Y \subseteq X_p$$

and *co*-sets are preserved by $\cong$.

Example: Suppose $\mathbf{p} = \begin{array}{ccc} a & \rightarrow & b \\ & \nearrow & \\ b & \rightarrow & c \end{array}$ and $\mathbf{q} = \begin{array}{ccc} a & \rightarrow & b \\ b & \rightarrow & c \end{array}$. Then $P_{\mathsf{M}\subseteq\mathsf{S}}(\mathbf{p})$ because

$$\mathsf{M}(\mathbf{p}) = \left\{ a, b, c, \genfrac{}{}{0pt}{}{a}{b}, \genfrac{}{}{0pt}{}{b}{c} \right\} \subseteq \mathsf{S}$$

but $P_{\mathsf{M}\subseteq\mathsf{S}}(\mathbf{q})$ does not hold because ($b^2 \notin \mathsf{S}$)

$$\mathsf{M}(\mathbf{q}) = \left\{ a, b, c, \genfrac{}{}{0pt}{}{a}{b}, \genfrac{}{}{0pt}{}{b}{b}, \genfrac{}{}{0pt}{}{a}{c}, \genfrac{}{}{0pt}{}{b}{c} \right\} \not\subseteq \mathsf{S}$$

6.1 Operations on Pomsets

Pomsets have been equipped with a variety of operations ([Gra81, Sta81, Gis88, Pra86]). In this part of the thesis we need only a few of these. Just as pomsets were defined on the basis of labelled posets we shall do so with the operations. The following two are both natural generalizations of concatenation of words: sequential and parallel composition.

Definition 6.1.1 *Sequential Composition of Pomsets*

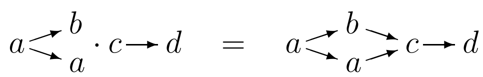
Given two pomsets, $\mathbf{p}$ and $\mathbf{q}$. Their sequential composition, $\mathbf{p} \cdot \mathbf{q}$, is obtained (informally) by taking their disjoint union (component wise), and making all elements of $\mathbf{q}$ causally dependent on all elements of $\mathbf{p}$. Formally:

For two lpos p_0 and p_1 we define their sequential composition $p_0 \cdot p_1 = \langle X, \leq, \ell \rangle$, where

$$\begin{array}{rl} X & \text{is the set } \{0\} \times X_{p_0} \cup \{1\} \times X_{p_1} \\ \leq & \text{is the partial order defined by} \\ & \langle i, x \rangle \leq \langle j, y \rangle \ \textit{iff} \quad i = j \ \textit{and} \ x \leq_{p_i} y \\ & \qquad\qquad\qquad\quad \textit{or} \ i = 0, j = 1 \\ \ell & \text{is the function } \langle i, x \rangle \mapsto \ell_{p_i}(x) \end{array}$$

For two pomsets $\mathbf{p}_0$ and $\mathbf{p}_1$ we then define $\mathbf{p}_0 \cdot \mathbf{p}_1$ to be $[p_0 \cdot p_1]$. □

Example:



Proposition 6.1.2 Suppose p, p_0, p_1 and p_2 are lpos. Then

- $p \cdot \varepsilon \cong p \cong \varepsilon \cdot p$
- $(p_0 \cdot p_1) \cdot p_2 \cong p_0 \cdot (p_1 \cdot p_2)$

Proof To see the last property use as isomorphism (from the left hand side to the right hand side) the function given by:

$$\begin{array}{rl} \langle 0, \langle 0, x \rangle \rangle & \mapsto \langle 0, x \rangle \\ \langle 0, \langle 1, x \rangle \rangle & \mapsto \langle 1, \langle 0, x \rangle \rangle \\ \langle 1, x \rangle & \mapsto \langle 1, \langle 1, x \rangle \rangle \end{array}$$

□

As a corollary we immediately get that for pomsets $\cdot$ is associative and has ε as left and right neutral element.

Definition 6.1.3 *Parallel Composition of Pomsets*

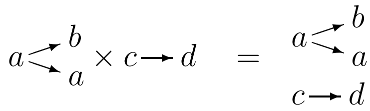
Given two pomsets, $\mathbf{p}$ and $\mathbf{q}$, their parallel composition, $\mathbf{p} \times \mathbf{q}$, is simply the union (component wise) of $\mathbf{p}$ and $\mathbf{q}$. Formally:

For lpos p_0 and p_1 we define $p_0 \times p_1 = \langle X, \leq, \ell \rangle$, where

X is the set $\{0\} \times X_{p_0} \cup \{1\} \times X_{p_1}$
$\leq$ is the partial order defined by
$\langle i, x \rangle \leq \langle j, y \rangle$ *iff* $i = j$ *and* $x \leq_{p_i} y$
ℓ is the function $\langle i, x \rangle \mapsto \ell_{p_i}(x)$

For pomsets $\mathbf{p}_0$ and $\mathbf{p}_1$ we define $\mathbf{p}_0 \times \mathbf{p}_1$ to be $[p_0 \times p_1]$. □

Example:



Proposition 6.1.4 Suppose p, p_0, p_1 and p_2 are lpos. Then

- $p \times \varepsilon \cong p \cong \varepsilon \times p$
- $p_0 \times p_1 \cong p_1 \times p_0$
- $p_0 \times (p_1 \times p_2) \cong (p_0 \times p_1) \times p_2$

Proof The second property is seen by using as isomorphism the function given by:

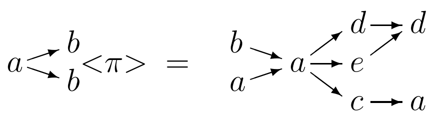
$$\langle 0, x \rangle \mapsto \langle 1, x \rangle$$
$$\langle 1, x \rangle \mapsto \langle 0, x \rangle$$

and the other properties are proved as in the last proposition. □

So for pomsets $\times$ is associative, commutative and has ε as left and right neutral element.

The next operator refines the different elements of a pomset into different pomsets (a formalization of the concept of "change of atomicity").

Example: Consider the pomset $a \!<\!{}^{b}_{b}$. Suppose we would like to refine the upper occurrence of b to ${}^{d}_{e}\!>\! d$, the lower to $c \to a$ and the a occurrence to ${}^{b}_{a}\!>\! a$. Call this refinement π and the associated operator $<\!\pi\!>$—then we would expect:



Actually it does not make sense talk about the upper, lower, etc. occurrence of b in a pomset, but for a particular representative each individual element can be replaced by "its own" pomset (representative) thus obtaining the representative of, a new pomset. We now give a definition of this construction and then in a moment utilize this for the definition of a function from pomsets.

The construction is not as simple as the others and we need to introduce some additional notions.

Definition 6.1.5 *Particular Refinement*
Let p be a lpo. A *particular refinement* for p is a mapping $\pi_p : X_p \longrightarrow LPO$.
Given a lpo p and a particular refinement (p.ref. for short), π_p, for p, we can construct a new lpo, $p{<}\pi_p{>}$,as follows: $p{<}\pi_p{>}$ is $\langle X, \leq, \ell \rangle$, where

$$\begin{array}{cll} X & \text{is the set } \{\langle x, x' \rangle \mid x \in X_p, x' \in X_{\pi_p(x)}\} & \\ \leq & \text{is the partial order defined by} & \\ & \langle x, x' \rangle \leq \langle y, y' \rangle \ \textit{iff} & x \leq_p y \ \textit{and} \\ & & x = y \Rightarrow x' \leq_{\pi_p(x)} y' \\ \ell & \text{is the function } \langle x, x' \rangle \mapsto \ell_{\pi_p(x)}(x') & \end{array}$$

Notice that $p{<}\pi_p{>}$ is a finite lpo. Following the idea of Gischer [Gis84] we introduce the following lpos

$$C = \langle \{0,1\}, \{\langle 0,0\rangle, \langle 0,1\rangle, \langle 1,1\rangle\}, i \mapsto a_i \rangle \text{ i.e., } [C] = a_0 \rightarrow a_1$$

$$S = \langle \{0,1\}, \{\langle 0,0\rangle, \langle 1,1\rangle\}, i \mapsto a_i \rangle \text{ i.e., } [S] = \begin{array}{c} a_0 \\ a_1 \end{array}$$

where a_0 and a_1 are two fixed elements of Δ. For lpos p_0 and p_1 let $\pi_{C(p_0,p_1)}$ denote the p.ref. for C given by $\pi_{C(p_0,p_1)}(i) = p_i$ for $i = 0,1$ and similar for $\pi_{S(p_0,p_1)}$.

From the definitions it immediately follows that sequential and parallel composition can be derived from particular refinements of C and S in the following sense:

$$p \cdot q = C{<}\pi_{C(p,q)}{>}$$
$$p \times q = S{<}\pi_{S(p,q)}{>}$$

Therefore also $\mathbf{p} \cdot \mathbf{q} = [C{<}\pi_{C(p,q)}{>}]$ and $\mathbf{p} \times \mathbf{q} = S{<}\pi_{S(p,q)}{>}$. That is to say with the words of Gischer [Gis88] $\cdot$ and $\times$ are pomset definable operations on pomsets. Gischer actually make refinement into a operation itself (called substitution) but it would not allow the type of refinements we shall need. We therefore prefer to postpone the definition of the pomset refinement operation to the section dealing with sets of pomsets.

6.2 Two Partial Orders on Pomsets

The first relation on pomsets we are going to present is used to compare the "concurrency" of two pomsets.

Definition 6.2.1 *$\preceq$-ordering on Pomsets*

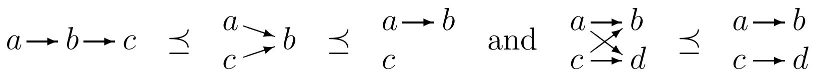
The preorder, $\preceq$, on lpos is defined: $p \preceq q$ *iff* there exists bijective function $X_q \longrightarrow X_p$ which also is a morphism of lpos.

This preorder induce a partial order, ambiguously denoted $\preceq$, on pomsets as follows:

$$\mathbf{p} \preceq \mathbf{q} \ \textit{iff} \ p \preceq q$$

$\mathbf{p} \preceq \mathbf{q}$ can be read: the pomset $\mathbf{p}$ is smoother than [Gra81]/ subsumed by [Gis88]/less nonsequential than the pomset $\mathbf{q}$. □

Notice that for lpos p and q, $p \preceq q$ does *not* imply $p \cong q$. It is also useful to observe that $\mathbf{p} \preceq \mathbf{q}$ implies $m_{\mathbf{p}} = m_{\mathbf{q}}$.

Example: 

The $\preceq$-downwards closure of a pomset $\mathbf{p}$, $\{\mathbf{p}' \in \mathbf{P} \mid \mathbf{p}' \preceq \mathbf{p}\}$, is denoted $\delta(\mathbf{p})$.
Suppose P_* is a property of pomsets then $\delta_*(\mathbf{p})$ will be a shorthand for the semi $\preceq$-downwards closure $\{\mathbf{p}' \in \mathbf{P} \mid \mathbf{p}' \preceq \mathbf{p}$ and $P_*(\mathbf{p}')\}$. E.g., $\delta_{\mathsf{W}}(\mathbf{p}) = \{\mathbf{p}' \in \mathbf{P} \mid \mathbf{p}' \preceq \mathbf{p}$ and $P_{\mathsf{W}}(\mathbf{p}')\} = \{\mathbf{p}' \in W \mid \mathbf{p}' \preceq \mathbf{p}\}$. Though we might have $\mathbf{p} \notin \delta_*(\mathbf{p})$ for some pomset property P_*, we call it the δ_*-closure.

From the definition of p.ref. we directly see:

Proposition 6.2.2 Let lpos p and q be given with $q = \langle X_p, \leq_q, \ell_p \rangle$ and $\leq_q \subseteq \leq_p$. Furthermore suppose π and π' are p.ref.'s for both p and q ($X_p = X_q$) such that $\forall x \in X_p.\ X_{\pi'(x)} = \langle X_{\pi(x)}, \leq, \ell_{\pi(x)} \rangle, \leq \subseteq \leq_{\pi(x)}$. Then

$$p{<}\pi{>} \preceq q{<}\pi{>}$$
$$p{<}\pi{>} \preceq p{<}\pi'{>}$$

The following alternative characterization of $\preceq$ will often be more convenient to work with.

Proposition 6.2.3 For pomsets $\mathbf{p}$ and $\mathbf{q}$ we have:

a) $\mathbf{p} \preceq \mathbf{q}$ *iff* $p = \langle X_{q'}, \leq_p, \ell_{q'} \rangle$ and $\leq_p \supseteq \leq_{q'}$ for some $q' \in \mathbf{q}$

b) $\mathbf{p} \preceq \mathbf{q}$ *iff* $\langle X_{p'}, \leq_q, \ell_{p'} \rangle = q$ and $\leq_{p'} \supseteq \leq_q$ for some $p' \in \mathbf{p}$

Proof Observe at first that in general $p \preceq q \cong r \Rightarrow p \preceq r$ and $p \cong q \preceq r \Rightarrow p \preceq r$.

a) *if*: $\mathrm{id}_{X_{q'}}$ is a label preserving bijective function from $X_{q'}$ to X_p because $X_p = X_{q'}$ and $\ell_p = \ell_{q'}$. By $\leq_p \supseteq \leq_{q'}$ it is also order preserving. Hence $p \preceq q'$ and since $q' \in \mathbf{q}$ means $q' \cong q$ we get $p \preceq q$ and so $\mathbf{p} \preceq \mathbf{q}$.
only if: By definition $\mathbf{p} \preceq \mathbf{q}$ implies the existence of a bijective function $f : X_q \longrightarrow X_p$ which also is a morphism of lpos. Then define q' to be $\langle X_p, \leq_{q'}, \ell_p \rangle$ where $\leq_{q'}$ is given by

$$x \leq_{q'} y \textit{ iff } f^{-1}(x) \leq_q f^{-1}(y)$$

Clearly $q' \cong q$ and $q' \in \mathbf{q}$. Also $p = \langle X_{q'}, \leq_p, \ell_{q'} \rangle$ so it remains to show $\leq_{q'} \subseteq \leq_p$. Assume $x \leq_{q'} y$. Then $f^{-1}(x) \leq_q f^{-1}(y)$ by definition of q' and because f is bijective and a morphism of lpos therefore $x = f \circ f^{-1}(x) \leq_p f \circ f^{-1}(y) = y$.

b) is proved similar. □

With the alternative characterization of $\preceq$, proposition 6.2.2 above and the observations made by the definition of particular refinement we get:

$$(6.2) \qquad \cdot \text{ and } \times \text{ are } \preceq\text{-monotone in their left and right arguments}$$

Similar we with appropriate p.ref.'s deduce from the above example that:

$$(6.3) \qquad (\mathbf{p} \times \mathbf{q}) \cdot (\mathbf{p}' \times \mathbf{q}') \preceq (\mathbf{p} \cdot \mathbf{p}') \times (\mathbf{q} \cdot \mathbf{q}')$$

We now turn to the second partial order on pomsets.

Definition 6.2.4 $\sqsubseteq$-*ordering on Pomsets*

Given two pomsets $\mathbf{p}$ and $\mathbf{q}$. Then $\mathbf{p}$ is a *prefix* of $\mathbf{q}$, $\mathbf{p} \sqsubseteq \mathbf{q}$, if $\mathbf{p}$ is a subpomset of $\mathbf{q}$ and the elements of $\mathbf{p}$ only dominates the elements of $\mathbf{p}$ in $\mathbf{q}$. Formally:

The lpo preorder, $\sqsubseteq$, is defined $p \sqsubseteq q$ *iff* there exists a $\leq_q$-downwards closed set Y such that p is isomorphic to the restriction of q to Y. I.e.,

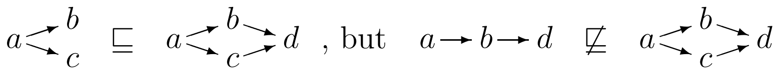
$$p \sqsubseteq q \textit{ iff } \exists Y.\, p \cong q|_Y \textit{ and } \{x \in X_q \mid \exists y \in Y.\, x \leq_q y\} \subseteq Y$$

The partial order, $\sqsubseteq \subseteq \mathbf{P} \times \mathbf{P}$, is induced from the lpo preorder by:

$$\mathbf{p} \sqsubseteq \mathbf{q} \textit{ iff } p \sqsubseteq q$$

π is defined to be the function which for a pomset $\mathbf{p}$ gives the $\sqsubseteq$-downwards closure of $\mathbf{p}$: $\pi(\mathbf{p}) = \{\mathbf{p}' \in \mathbf{P} \mid \mathbf{p}' \sqsubseteq \mathbf{p}\}$. $\square$

That $\mathbf{p} \sqsubseteq \mathbf{q}$ implies $\mathbf{p} \hookrightarrow \mathbf{q}$ follows from $p \cong q|_Y$. Notice that $\mathbf{p} \sqsubseteq \mathbf{p}$ and $\mathbf{p} \sqsubseteq \mathbf{q}$ implies $m_{\mathbf{p}} \leq m_{\mathbf{q}}$. Also observe that $\{x \in X_q \mid \exists y \in Y.\, x \leq_q y\} \subseteq Y$ just is a formalization of: Y is $\leq_q$-downwards closed.

Example: 

As for the partial order $\preceq$ there is an alternative characterization of $\sqsubseteq$:

Proposition 6.2.5 For pomsets $\mathbf{p}$ and $\mathbf{q}$ we have:

a) $\mathbf{p} \sqsubseteq \mathbf{q}$ *iff* $p' = q|_{X_{p'}}$ for some $p' \in \mathbf{p}$ with $\{x \in X_q \mid \exists y \in X_{p'}.\, x \leq_q y\} \subseteq X_{p'}$

b) $\mathbf{p} \sqsubseteq \mathbf{q}$ *iff* $p = q'|_{X_p}$ for some $q' \in \mathbf{q}$ with $\{x \in X_{q'} \mid \exists y \in X_p.\, x \leq_{q'} y\} \subseteq X_p$

Proof a), b) *if*: Immediate because $= \subseteq \cong$.

For the *only if* direction of a) and b) we by definition have

$$\exists Y.\, p \cong q|_Y \textit{ and } \{x \in X_q \mid \exists y \in Y.\, x \leq_q y\} \subseteq Y$$

W.l.o.g. we can assume $Y \subseteq X_q$ (because $q|_Y = q|_{(X_q \cap Y)}$).

a) *only if*: Define p' to be $q|_Y$. Obviously p' is a representative of $\mathbf{p}$ and because Y is a subset of X_q we have $Y = X_q|_Y = X_{p'}$. Hence $p' = q|_Y = q|_{X_{p'}}$ and $X_{p'}$ is $\leq_q$-downwards closed.

b) *only if*: Here we shall find a representative of $\mathbf{q}$ which p is a part of. The idea will be to find a representative q'' of $\mathbf{q}$ which has no elements in common with p and then just replace that part of q'' which is isomorphic to p with p to obtain q'. The elements of p are from the ground set which are composed of two-tuples. Clearly the "size", $|x|$, of an element x can be determined as follows

$$|x| = \begin{cases} |x_0| + |x_1| & \text{if } x = \langle x_0, x_1 \rangle \\ 1 & \text{otherwise } (x \in I\!N \text{ } or \text{ } x \in \Delta) \end{cases}$$

If p is empty we can just choose $q' = q$ so assume it is not. Then since we work with finite pomsets/ lpos it make sense choose a $z \in X_p$ with maximal size according to the measure above. Define $q'' = \langle X, \leq, \ell \rangle$ where

X is the set $\{\langle x, z \rangle \mid x \in X_q\}$
$\leq$ is the partial order defined by
$\langle x, z \rangle \leq \langle y, z \rangle$ *iff* $x \leq_q y$
ℓ is the function $\langle x, z \rangle \mapsto \ell_q(x)$

Evidently q'' is a representative of $\mathbf{q}$ and p is a lpo isomorphic to $q''|_{Y_z}$ where Y_z is the $\leq_{q''}$-downwards closed set $\{\langle x, z \rangle \mid x \in Y\}$. By construction all elements of $X_{q''}$ have size greater than those of X_p and so they cannot have any elements in common. The required q' is then obtained by replacing all elements from $X_{q''}$ which under the lpo isomorphism equals the elements of X_p with these corresponding elements of X_p. □

With this alternative characterization it is useful to observe for lpos p and q:

- $\{0\} \times X_p = X_{p \cdot \varepsilon} = X_{p \times \varepsilon}$ and $\{1\} \times X_p = X_{\varepsilon \cdot p} = X_{\varepsilon \times p}$
- $Y \subseteq X_{p \cdot \varepsilon}$ implies $(p \cdot q)|_Y = (p \cdot \varepsilon)|_Y$
- $X_{p \cdot \varepsilon} \subseteq Y$ implies $(p \cdot q)|_Y \cong p \cdot (\varepsilon \cdot q)|_Y$
- $Y \subseteq X_{p \times \varepsilon}$ implies $(p \times q)|_Y = (p \times \varepsilon)|_Y$ (symmetric for $\varepsilon \times p$)
- $X_{p \times \varepsilon} \subseteq Y$ implies $(p \times q)|_Y \cong p \times (\varepsilon \times q)|_Y$ (symmetric for $\varepsilon \times p$)

Then evidently a pomset is a prefix of two parallel composed pomsets *iff* it itself is the parallel composition of two prefixes of the two parallel composed pomsets. It is also easy to see $\mathbf{p} \sqsubseteq \mathbf{q}$ implies $\mathbf{p} \sqsubseteq \mathbf{q} \cdot \mathbf{r}$ and $\mathbf{r} \cdot \mathbf{p} \sqsubseteq \mathbf{r} \cdot \mathbf{q}$. It takes more effort to prove the "reverse":

Proposition 6.2.6 If $\mathbf{p} \sqsubseteq \mathbf{q} \cdot \mathbf{r}$ then either $\mathbf{p} \sqsubseteq \mathbf{q}$ or there exists a pomsets $\mathbf{r}'$ such that $\mathbf{p} = \mathbf{q} \cdot \mathbf{r}'$ and $\mathbf{r}' \sqsubseteq \mathbf{r}$.

Proof Let $\mathbf{p}', \mathbf{q}$ and $\mathbf{r}$ be given such that $\mathbf{p}' \sqsubseteq \mathbf{q} \cdot \mathbf{r}$. $\mathbf{q} \cdot \mathbf{r} = [q \cdot r]$ so by the alternative characterization of prefix we know there is a representative p of $\mathbf{p}'$ such that $p = (q \cdot r)|_{X_p}$ and X_p is $\leq_{q \cdot r}$-downwards closed.

If $X_p \subseteq \{0\} \times X_q$ then as observed $p = (q \cdot r)|_{X_p} = (q \cdot \varepsilon)|_{X_p}$. Of course X_p is $\leq_{q \cdot \varepsilon}$-downwards closed and by the alternative characterization of prefix then $\mathbf{p}' = \mathbf{p} \sqsubseteq [q \cdot \varepsilon] = \mathbf{q}$.

So assume $X_p \not\subseteq \{0\} \times X_q$. We show at first $\{0\} \times X_q \subseteq X_p$. Let an $x \in \{0\} \times X_q$ be given. From $p = (q \cdot r)|_{X_p}$ we have $X_p \subseteq \{0\} \times X_q \cup \{1\} \times X_r$, so because $X_p \not\subseteq \{0\} \times X_q$ there must be a y of X_p in $\{1\} \times X_r$. By definition of $q \cdot r$ then $x \leq_{q \cdot r} y$ wherefore the $\leq_{q \cdot r}$-downwards closure of X_p yields $x \in X_p$. Now since $X_{\varepsilon \cdot q} = \{0\} \times X_q \subseteq X_p$ we have $p \cong (q \cdot r)|_{X_p} \cong q \cdot (\varepsilon \cdot r)|_{X_p}$. That X_p is $\leq_{\varepsilon \cdot r}$-downwards closed is a simple consequence of $X_{\varepsilon \cdot r} \subseteq X_{q \cdot r}$ and the $\leq_{q \cdot r}$-downwards closure of X_p. Defining $r' = (\varepsilon \cdot r)|_{X_p}$ we get $\mathbf{r}' \sqsubseteq [\varepsilon \cdot r] = \varepsilon \cdot \mathbf{r} = \mathbf{r}$ by using the alternative characterization of prefix again. From $\mathbf{p}' \ni p = (q \cdot r)|_{X_p} \cong q \cdot (\varepsilon \cdot r)|_{X_p} = q \cdot r'$ then $\mathbf{p}' = [q \cdot r'] = \mathbf{q} \cdot \mathbf{r}'$ as desired. □

The next proposition will prove extremely useful in proving various connections between the two partial orders over pomsets.

Proposition 6.2.7 Given two pomsets $\mathbf{p}$ and $\mathbf{q}$. Then

$$\mathbf{p} \sqsubseteq \mathbf{q} \Rightarrow \exists \mathbf{r} \in \mathbf{P}.\, \mathbf{p} \cdot \mathbf{r} \preceq \mathbf{q}$$

Proof Assume $\mathbf{p} \sqsubseteq \mathbf{q}$. By the alternative characterization of prefix we can find a representative p' of $\mathbf{p}$ such that $p' = q|_{X_{p'}}$ and $X_{p'}$ is $\leq_q$-downwards closed. Define r to be $q|_{(X_q \setminus X_{p'})}$ and $q' = \langle X, \leq, \ell \rangle$, where

X	is the set $\{0\} \times X_{p'} \cup \{1\} \times (X_q \setminus X_{p'})$
$\leq$	is the partial order defined by
	$\langle i, x \rangle \leq \langle j, y \rangle$ *iff* $x \leq_q y$
ℓ	is the function $\langle i, x \rangle \mapsto \ell_q(x)$

Clearly $q' \cong q$—i.e., q' is a representative of $\mathbf{q}$, and from the definition of p', r and lpo sequential composition we see $X_{p' \cdot r} = X_{q'}$ and $\ell_{p' \cdot r} = \ell_{q'}$. To see $\leq_{q'} \subseteq \leq_{p' \cdot r}$ assume $\langle i, x \rangle \leq \langle j, y \rangle$. Then $x \leq_q y$ and if $i = 0 = j$ we have $x, y \in X_{p'}$ so $x \leq_{p'} y$ then follows from $\leq_{p'} = \leq_q|_{X_{p'}{}^2}$. Similar if $i = 1 = j$. If $i = 0$ and $j = 1$ then $\langle i, x \rangle \leq_{p' \cdot r'} \langle j, y \rangle$ by the definition of $p' \cdot r$. We are left with the case $i = 1$ and $j = 0$. This is means $x \leq_q y$, $x \in X_q \setminus X_{p'}$ and $y \in X_{p'}$—but this is impossible because $X_{p'}$ is $\leq_q$-downwards closed and we can exclude this case. So we actually have $q' = \langle X_{p' \cdot r}, \leq_{q'}, \ell_{p' \cdot r} \rangle$ and $\leq_{p' \cdot r} \supseteq \leq_{q'}$. The alternative characterization of $\preceq$ then gives us $\mathbf{p}' \cdot \mathbf{r} = [p' \cdot r] \preceq \mathbf{q}' = \mathbf{q}$ as we wanted. □

With this result it is easy to prove:

Proposition 6.2.8 For pomsets $\mathbf{p}$ and $\mathbf{q}$ we have

$$\exists \mathbf{r}.\, \mathbf{p} \preceq \mathbf{r} \sqsubseteq \mathbf{q} \text{ \textit{iff} } \exists \mathbf{s}.\, \mathbf{p} \sqsubseteq \mathbf{s} \preceq \mathbf{q}$$

In [Pra86, page 49] Pratt outlines an alternative proof. He defines prefix in another, but equivalent way: $\mathbf{p}$ is a prefix of $\mathbf{q}$ if $\exists Y.\, p \cong q|_Y$ and $X_q \setminus Y$ is $\leq_q$-upwards closed. So this proposition can be seen as just a reformulation of his theorem.

Proof *only if*: Assume $\mathbf{p} \preceq \mathbf{r} \sqsubseteq \mathbf{q}$. By the previous proposition we know there is a pomset $\mathbf{r}'$ such that $\mathbf{r} \cdot \mathbf{r}' \preceq \mathbf{q}$. From $\mathbf{p} \preceq \mathbf{r}$ and $\preceq$-monotonicity of $\cdot$ then $\mathbf{p} \cdot \mathbf{r}' \preceq \mathbf{q}$. But $\mathbf{p} \sqsubseteq \mathbf{p} \cdot \mathbf{r}'$ so we can just choose $\mathbf{s} = \mathbf{p} \cdot \mathbf{r}'$.

if: Suppose $\mathbf{p} \sqsubseteq \mathbf{s} \preceq \mathbf{q}$. Then there are representatives p' and q' of $\mathbf{p}$ and $\mathbf{q}$ respectively such that $p' = s|_{X_{p'}}$, $X_{p'}$ is $\leq_s$-downwards closed and $q' = \langle X_s, \leq_{q'}, \ell_s \rangle$ with $\leq_s \supseteq \leq_{q'}$. Define r to be $q'|_{X_{p'}}$. Then r is a lpo and to see that $X_{p'}$ is $\leq_r$-downwards closed assume $x \leq_r y \in X_{p'}$. Then $x \leq_{q'} y$ and from $\leq_{q'} \subseteq \leq_s$ also $x \leq_s y$. $x \in X_{p'}$ follows now from the $\leq_s$-downwards closure of $X_{p'}$. Hence $r \sqsubseteq q'$. We also have $r = \langle X_{p'}, \leq_{q'}|_{X_{p'}^2}, \ell_{p'} \rangle$, so from $\leq_{q'} \subseteq \leq_s$ then $\leq_r = \leq_q|_{X_{p'}^2} \subseteq \leq_s|_{X_{p'}^2} = \leq_{p'}$. Thus $p' \preceq r \sqsubseteq q'$ and $\mathbf{p} = \mathbf{p}' \preceq \mathbf{r} \sqsubseteq \mathbf{q}' = \mathbf{q}$. □

6.3 Sets of Pomsets

Sets of pomsets and operators on them are used extensively in the models we shall present, so we briefly treat them here. The two operations on pomsets $\cdot$ and $\times$ generalize to sets in the natural way e.g., $P \cdot Q = \{\mathbf{p} \cdot \mathbf{q} \mid \mathbf{p} \in P, \mathbf{q} \in Q\}$. We shall use $\cup$ to denote the normal set union and $\mathcal{P}(\_)$ the powerset operator. Also for a pomset property P_*, δ_* generalize to sets: $\delta_*(Q) = \bigcup_{\mathbf{q} \in Q} \delta_*(\mathbf{q})$.

The previously mentioned refinement operator for pomsets is defined using the particular refinement construction for lpos.

Definition 6.3.1 *Refinements*
A $\mathcal{P}(\mathbf{P})$-*refinement* is a mapping $\varrho : \Delta \longrightarrow \mathcal{P}(\mathbf{P})$.
We say that a $\mathcal{P}(\mathbf{P})$-refinement, ϱ, is *ε-free iff* $\forall a \in \Delta.\, \varepsilon \notin \varrho(a)$ and ϱ is *image finite* if $\varrho(a)$ is finite for every $a \in \Delta$.
A particular refinement π_p for a lpo p is *consistent* with a $\mathcal{P}(\mathbf{P})$-refinement ϱ *iff*

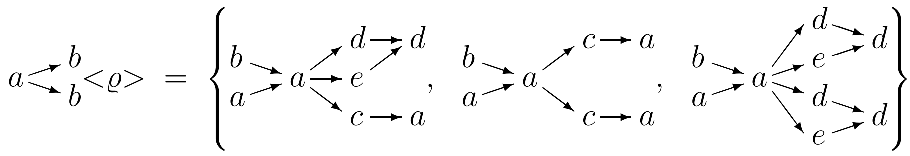
$$\forall x \in X_p.\, [\pi_p(x)] \in \varrho(\ell_p(x))$$

The mapping associated with ϱ is now defined as ${<}\varrho{>} : \mathbf{P} \longrightarrow \mathcal{P}(\mathbf{P})$ with $\mathbf{p}{<}\varrho{>} = \{[p{<}\pi_p{>}] \mid \pi_p \text{ is a } \varrho\text{-consistent p.ref. for } p\}$ and generalized to sets of pomsets by $P{<}\varrho{>} = \bigcup_{\mathbf{p} \in P} \mathbf{p}{<}\varrho{>}$.

For a finite lpo p and image finite refinement ϱ we notice that there is only finitely many different ϱ-consistent p.ref. for p (up to $\cong$ in each $x \in X_p$) and consequently in general $\mathbf{p}{<}\varrho{>}$ is a finite set of pomsets because we only work with finite pomsets. Also $P{<}\varrho{>}$ must be finite if P is a finite set of pomsets and ϱ is a image finite refinement.

Example: Consider the same pomset as in the example for particular refinement on page 135. Suppose ϱ is a $\mathcal{P}(\mathbf{P})$-refinement such that $a \mapsto \left\{ \begin{smallmatrix} b \\ a \end{smallmatrix} \rightrightarrows a \right\}$ and $b \mapsto \left\{ c \rightarrow a, \begin{smallmatrix} d \\ e \end{smallmatrix} \rightrightarrows d \right\}$.

Then



Whereas it was quite obvious that $\cdot$ and $\times$ defined operations on sets of pomsets this is not so easy to see for $<\varrho>$. But we now prove that $<\varrho>$ actually defines a operation on sets of pomsets.

Proposition 6.3.2 $<\varrho>$ is well-defined.

Proof From the definition of $<\varrho>$ we clearly see that is enough to show:

> If π_p is a ϱ-consistent p.ref. for a lpo p then $p \cong q$ implies the existence of a ϱ-consistent p.ref., π_q, for q such that $p<\pi_p> \cong q<\pi_q>$.

Let f be a isomorphism of lpos from q to p. If π_p is a p.ref. for p then $\pi_q := \pi_p \circ f$ is a p.ref. for q. Also π_q is consistent with ϱ because:

$$\begin{array}{ll} \forall x \in X_p.\ [\pi_p(x)] \in \varrho(\ell_p(x)) & \\ \Updownarrow & f \text{ is a bijection} \\ \forall x \in f(X_q).\ [\pi_p(x)] \in \varrho(\ell_p(x)) & \\ \Updownarrow & \\ \forall x \in X_q.\ [\pi_p(f(x))] \in \varrho(\ell_p(f(x))) & \\ \Updownarrow & f \text{ is label preserving, definition of } \pi_q \\ \forall x \in X_q.\ [\pi_q(x)] \in \varrho(\ell_q(x)) & \end{array}$$

To see $p<\pi_p> \cong t<\pi_q>$ we show $g : X_{q<\pi_q>} \longrightarrow X_{p<\pi_p>}$ given by $\langle x, x' \rangle \mapsto \langle f(x), x' \rangle$ is an isomorphism of lpos.

It is seen from: $g(X_{q<\pi_q>}) = \{g(\langle x, x' \rangle) \mid \langle x, x' \rangle \in X_{q<\pi_q>}\}$
$= \{\langle f(x), x' \rangle \mid x \in X_q, x' \in X_{\pi_p(f(x))}\}$
$= \{\langle x, x' \rangle \mid x \in f(X_q), x' \in X_{\pi_p(x)}\} = X_{p<\pi_p>}$

Clearly g is bijective and g^{-1} is $\langle x, x' \rangle \mapsto \langle f^{-1}(x), x' \rangle$.

We have
$\ell_{q<\pi_q>}(\langle x, x' \rangle) = \ell_{\pi_q(x)}(x') = \ell_{\pi_p(f(x))}(x') = \ell_{p<\pi_p>}(\langle f(x), x' \rangle) = \ell_{p<\pi_p>}(g(\langle x, x' \rangle))$,
so g is label preserving and since:

$$\begin{array}{ll}
\quad \langle x, x' \rangle \leq_{q<\pi_q>} \langle y, y' \rangle & \\
\Downarrow & \text{by construction of } q<\pi_q> \\
\quad x \leq_q y \textit{ and} & \\
\quad x = y \Rightarrow x' \leq_{\pi_q(x)} y' & \\
\Downarrow & f \text{ morphism from } q \text{ to } p \text{ and } \pi_q = \pi_p \circ f \\
\quad f(x) \leq_p f(y) \textit{ and} & \\
\quad f(x) = f(y) \Rightarrow x' \leq_{\pi_p(f(x))} y' & \\
\Downarrow & \text{by construction of } p<\pi_p> \\
\quad \langle f(x), x' \rangle \leq_{p<\pi_p>} \langle f(y), y' \rangle & \\
\Downarrow & \text{by definition of } g \\
\quad g(\langle x, x' \rangle) \leq_{p<\pi_p>} g(\langle y, y' \rangle) &
\end{array}$$

g is also order preserving and therefore a morphism of lpos. Similarly it is seen that g^{-1} is a morphism of lpos. $\square$

The difference between our refinement operation and Gischers substitution can be illustrated by the following example.

Example: Suppose $\mathbf{p} = a \rightarrow a$ and ϱ is a $\mathcal{P}(\mathbf{P})$-refinement with $\varrho(a) = \left\{ b, \begin{smallmatrix} c \\ d \end{smallmatrix} \right\}$. Then

$$\mathbf{p}<\varrho> = \left\{ b \rightarrow b,\; b \begin{smallmatrix} \nearrow c \\ \searrow d \end{smallmatrix},\; \begin{smallmatrix} c \searrow \\ d \nearrow \end{smallmatrix} b,\; \begin{smallmatrix} c \rightarrow c \\ \times \\ d \rightarrow d \end{smallmatrix} \right\}$$

whereas the result by Gischer substitution would be

$$\left\{ b \rightarrow b,\; \begin{smallmatrix} c \rightarrow c \\ \times \\ d \rightarrow d \end{smallmatrix} \right\}$$

The different operations enjoy a number of properties, many of them inherited from the corresponding properties of pomsets. Some of them are listed in:

Proposition 6.3.3

- $\cdot$, $\times$ and $\cup$ are associative
- $\times$ and $\cup$ are commutative
- $\{\varepsilon\}<\varrho> = \{\varepsilon\}$, $\{a\}<\varrho> = \varrho(a)$ and $<\varrho>$ distributes over $\cdot$, $\times$ and $\cup$

That $<\varrho>$ distributes over $\cdot$ may seem surprising. But if π is a ϱ-consistent p.ref. for $p_0 \cdot p_1$ then one can find ϱ-consistent p. refinements, π_{p_0} for p_0 and π_{p_1} for p_1 (just define $\pi_{p_i}(x) = \pi(\langle i, x \rangle)$ for $i = 0, 1$) such that

$$(p_0 \cdot p_1)<\pi> \cong p_0<\pi_{p_0}> \cdot p_1<\pi_{p_1}>$$

(the map $\langle \langle i, x \rangle, x' \rangle \mapsto \langle i, \langle x, x' \rangle \rangle$ is an isomorphism from $X_{(p_0 \cdot p_1)<\pi>}$ to $X_{p_0<\pi_{p_0}> \cdot p_1<\pi_{p_1}>}$). Then we have $[(p_0 \cdot p_1)<\pi>] = [p_0<\pi_{p_0}> \cdot p_1<\pi_{p_1}>]$. And of course then also $(\mathbf{p}_0 \cdot \mathbf{p}_1)<\varrho> = \mathbf{p}_0<\varrho> \cdot \mathbf{p}_1<\varrho>$ which generalize to sets as well.

The partial order $\subseteq$ on sets will be central to our models. $\cup$ and natural extensions to sets are $\subseteq$-monotone, so we get:

Proposition 6.3.4 The operators $\cdot$, $\cup$, $\times$, $<\varrho>$ and δ_* are $\subseteq$-monotone in all their arguments.

6.4 Two Types of Pomset Properties

The first type of pomset properties we shall consider is those where the property of a pomset is inherited to all subpomsets.

Definition 6.4.1 *Hereditary Pomset Properties*

A pomset property, P_*, is *hereditary*, *iff*

$$\forall \mathbf{p} \in \mathbf{P}.\ P_*(\mathbf{p}), \mathbf{q} \hookrightarrow \mathbf{p} \Rightarrow P_*(\mathbf{q})$$

□

A pomset being a singleton/ multisingleton/ set/ multiset pomset are examples of hereditary pomset properties because $\mathbf{p} \hookrightarrow \mathbf{q}$ implies $\mathsf{M}(\mathbf{p}) \subseteq \mathsf{M}(\mathbf{q})$. Also the P_{W}-property (page 133) is hereditary.

The following three propositions relates hereditary pomset properties with sequential and parallel composition of pomsets.

Proposition 6.4.2 Let P_* be a hereditary pomset property. Then

$$\begin{array}{l} \mathbf{q} \preceq \mathbf{p}_0 \cdot \mathbf{p}_1, P_*(\mathbf{q}) \\ \Downarrow \\ \exists \mathbf{q}_0, \mathbf{q}_1.\ \mathbf{q} = \mathbf{q}_0 \cdot \mathbf{q}_1 \text{ and } \mathbf{q}_i \preceq \mathbf{p}_i, P_*(\mathbf{q}_i) \text{ for } i = 0, 1 \end{array}$$

Proof We prove the proposition for lpos which then generalizes to pomsets. Let there be given lpos p_0, p_1 and q such that $q \preceq p_0 \cdot p_1$ and $P_*(q)$.

$q \preceq p_0 \cdot p_1$ implies the existence of a bijection $f : X_{p_0 \cdot p_1} \longrightarrow X_q$ which also is a morphism of lpos.

By definition of $\cdot$ we have $X_{p_0 \cdot p_1} = (\{0\} \times X_{p_0}) \cup (\{1\} \times X_{p_1})$ so we define q_i to be $\langle X_{p_i}, \leq_{q_i}, \ell_{p_i} \rangle$ where $\leq_{q_i} \subseteq X_{p_i} \times X_{p_i}$ is defined by:

$$x \leq_{q_i} y \textit{ iff } f(\langle i, x \rangle) \leq_q f(\langle i, y \rangle)$$

With this definition of q_i we only have to prove $\leq_{p_i} \subseteq \leq_{q_i}$ in order to have $q_i \preceq p_i$. This is seen as follows:

$$\begin{array}{lll} x \leq_{p_i} y & \Rightarrow \langle i, x \rangle \leq_{p_0 \cdot p_1} \langle i, y \rangle & \text{definition of } p_0 \cdot p_1 \\ & \Rightarrow f(\langle i, x \rangle) \leq_q f(\langle i, y \rangle) & f \text{ is order preserving} \\ & \Rightarrow x \leq_{q_i} y & \text{definition of } q_i \end{array}$$

Next we prove $q \cong q_0 \cdot q_1$. We show f is an isomorphism from $q_0 \cdot q_1$ to q. From $X_{q_0 \cdot q_1} = X_{p_0 \cdot p_1}$ we see that it makes sense. f is bijective and label preserving, so we just have to show that f and f^{-1} preserve order: At first notice

$$\begin{aligned}\langle i,x\rangle \leq_{q_0\cdot q_1} \langle i,y\rangle &\Leftrightarrow x \leq_{q_i} y && \text{definition of } q_0\cdot q_1\\ &\Leftrightarrow f(\langle i,x\rangle) \leq_q f(\langle i,y\rangle) && \text{definition of } q_i\end{aligned}$$

Now suppose $\langle i,x\rangle \leq_{q_0\cdot q_1} \langle j,y\rangle$, $i \neq j$. By definition of $q_0 \cdot q_1$ then $i = 0, j = 1$. But then also $\langle i,x\rangle \leq_{p_0\cdot p_1} \langle j,y\rangle$ and since f preserves the order of $p_0 \cdot p_1$ then $f(\langle i,x\rangle) \leq_q f(\langle j,y\rangle)$.

Suppose now $f(\langle i,x\rangle) \leq_q f(\langle j,y\rangle), i \neq j$. If $i = 0$ and $j = 1$ we by definition of $q_0 \cdot q_1$ also have $\langle i,x\rangle \leq_{q_0\cdot q_1} \langle j,y\rangle$. This settles the case because $i = 1$ and $j = 0$ would lead to a contradiction as follows:
If $i = 1$ and $j = 0$ we have $\langle j,y\rangle \leq_{p_0\cdot p_1} \langle i,x\rangle$ and $\langle j,y\rangle \neq \langle i,x\rangle$. Since f preserves the order of $p_0 \cdot p_1$ and is injective we get $f(\langle j,y\rangle) \leq_q f(\langle i,x\rangle)$ and $f(\langle j,y\rangle) \neq f(\langle i,x\rangle)$. But this together with $f(\langle i,x\rangle) \leq_q f(\langle j,y\rangle)$ contradicts the antisymmetry of $\leq_q$.

It remains to show $P_*(q_0)$ and $P_*(q_1)$. Clearly $q|_{f(\{i\}\times X_{q_i})} \cong q_i$ so because P_* is hereditary and invariant under $\cong$ the result follows. □

Proposition 6.4.3 Let P_* be a hereditary pomset property. Then

$$\begin{array}{l}\mathbf{q} \preceq \mathbf{p}_0 \times \mathbf{p}_1, P_*(\mathbf{q})\\ \Downarrow\\ \exists \mathbf{q}_0, \mathbf{q}_1.\ \mathbf{q} \preceq \mathbf{q}_0 \times \mathbf{q}_1 \textit{ and } \mathbf{q}_i \preceq \mathbf{p}_i, P_*(\mathbf{q}_i) \text{ for } i = 0,1\end{array}$$

Proof The definitions of q_0 and q_1 so as the arguments are exactly as in the proof of the previous proposition, except that $\cdot$ has to be exchanged to $\times$ and we *cannot* infer

$$f(\langle i,x\rangle) \leq_q f(\langle j,y\rangle), i \neq j \Rightarrow \langle i,x\rangle \leq_{q_0\times q_1} \langle j,y\rangle$$

because $i \neq j$ implies $\langle i,x\rangle\ co_{q_0\times q_1} \langle j,y\rangle$. For the same reason the proposition just states $\mathbf{q} \preceq \mathbf{q}_0 \times \mathbf{q}_1$. □

Proposition 6.4.4 Let P_* be hereditary pomset property. Then

a) $\delta_*(\mathbf{p}_0 \cdot \mathbf{p}_1) \subseteq \delta_*(\mathbf{p}_0) \cdot \delta_*(\mathbf{p}_1)$

b) $\delta_*(\mathbf{p}_0 \times \mathbf{p}_1) = \delta_*(\delta_*(\mathbf{p}_0) \times \delta_*(\mathbf{p}_1))$

Proof
a) Suppose $\mathbf{q} \in \delta_*(\mathbf{p}_0 \cdot \mathbf{p}_1)$—i.e., $\mathbf{q} \preceq \mathbf{p}_0 \cdot \mathbf{p}_1$ and $P_*(\mathbf{q})$. Then by the last but one proposition there exists pomsets $\mathbf{q}_0$ and $\mathbf{q}_1$ such that $\mathbf{q} = \mathbf{q}_0 \cdot \mathbf{q}_1$ and $\mathbf{q}_i \preceq \mathbf{p}_i, P_*(\mathbf{q}_i)$ for $i = 0,1$. This implies $\mathbf{q}_i \in \delta_*(\mathbf{p}_i)$ for $i = 0,1$ and $\mathbf{q} = \mathbf{q}_0 \cdot \mathbf{q}_1 \in \delta_*(\mathbf{p}_0) \cdot \delta_*(\mathbf{p}_1)$.

b) We prove each inclusion in turn.

- $\subseteq$: Suppose $\mathbf{q} \in \delta_*(\mathbf{p}_0 \times \mathbf{p}_1)$. Then $P_*(\mathbf{q})$ and $\mathbf{q} \preceq \mathbf{p}_0 \times \mathbf{p}_1$. Using the last proposition we find pomsets $\mathbf{q}_0$ and $\mathbf{q}_1$ such that $\mathbf{q} \preceq \mathbf{q}_0 \times \mathbf{q}_1$ and $\mathbf{q}_i \preceq \mathbf{p}_i, P_*(\mathbf{q}_i)$ for $i = 0,1$. This gives $\mathbf{q}_0 \times \mathbf{q}_1 \in \delta_*(\mathbf{p}_0) \times \delta_*(\mathbf{p}_1)$. From $P_*(\mathbf{q})$ and $\mathbf{q} \preceq \mathbf{q}_0 \times \mathbf{q}_1$ we then conclude $\mathbf{q} \in \delta_*(\delta_*(\mathbf{p}_0) \times \delta_*(\mathbf{p}_1))$ as desired.
- $\supseteq$: Given $\mathbf{q} \in \delta_*(\delta_*(\mathbf{p}_0) \times \delta_*(\mathbf{p}_1))$. Then $P_*(\mathbf{q})$ and $\mathbf{q} \preceq \mathbf{q}_0 \times \mathbf{q}_1$ for some $\mathbf{q}_i \in \delta_*(\mathbf{p}_i)$ and $i = 0,1$. This implies $\mathbf{q}_0 \preceq \mathbf{p}_0$ and $\mathbf{q}_1 \preceq \mathbf{p}_1$, so from the $\preceq$-monotonicity of $\times$ then
$$\mathbf{q} \preceq \mathbf{q}_0 \times \mathbf{q}_1 \preceq \mathbf{p}_0 \times \mathbf{q}_1 \preceq \mathbf{p}_0 \times \mathbf{p}_1$$
$\mathbf{q} \in \delta_*(\mathbf{p}_0 \times \mathbf{p}_1)$ then follows from $P_*(\mathbf{q})$.

$\square$

Proposition 6.4.5 If P_* is hereditary pomset property then $\pi\delta_*(\mathbf{p}) \subseteq \delta_*\pi(\mathbf{p})$.

Proof Let a $\mathbf{q} \in \pi\delta_*(\mathbf{p})$ be given. This means there is a $\mathbf{s}$ such that $P_*(\mathbf{s})$ and $\mathbf{q} \sqsubseteq \mathbf{s} \preceq \mathbf{p}$. $\mathbf{q} \sqsubseteq \mathbf{s}$ implies $\mathbf{q} \hookrightarrow \mathbf{s}$, so because P_* is hereditary we also have $P_*(\mathbf{q})$. By proposition 6.2.8 there is a pomset $\mathbf{r}$ with $\mathbf{q} \preceq \mathbf{r} \sqsubseteq \mathbf{p}$. Hence $\mathbf{q} \in \delta_*\pi(\mathbf{p})$. $\square$

Notice that P_* being hereditary was not used in $\supseteq$ of b) of proposition 6.4.4 and if we had closed the right hand side of a) similarly as in b) we would obtain equality.

But we shall deal with a certain type of pomset properties where it will not be necessarily to close in this way in order to obtain equality. For this type one can deduce/ synthesize the property for the sequential composition of two pomsets if they both have the property.

Definition 6.4.6 *Dot Synthesizable Pomset Properties*

A pomset property, P_*, is *dot synthesizable*, *iff*

$$\forall \mathbf{p}, \mathbf{q} \in \mathbf{P}.\ P_*(\mathbf{p}) \textit{ and } P_*(\mathbf{q}) \textit{ implies } P_*(\mathbf{p} \cdot \mathbf{q}) \tag{6.4}$$

$\square$

The following proposition states a condition that ensures a pomset property to be dot synthesizable.

Proposition 6.4.7 A pomset property P_* is dot synthesizable *if*

for every lpo p and $Y \subseteq X_p$ with $\forall x \in X_p \setminus Y \forall y \in Y.\ x\ c\not{o}_p\ y$ we have:

$$\begin{array}{c} P_*(p|_{X_p \setminus Y}) \textit{ and } P_*(p|_Y) \\ \Downarrow \\ P_*(p) \end{array} \tag{6.5}$$

Proof We show that $p \cdot q$ has the P_*-property if P_* fulfills the condition. By definition of $p \cdot q$ we have $\langle 0, z\rangle \leq_{p\cdot q} \langle 1, v\rangle$ for all $z \in X_p$ and $v \in X_q$ and as a consequence

$$\forall x \in \{0\} \times X_p \forall y \in \{1\} \times X_q.\ x\ c\not{o}_{p\cdot q}\ y$$

We also have $P_*(p)$ so from $p \cong \varepsilon \cdot p = (p \cdot q)|_{X_{p\cdot\varepsilon}}$ we see $P_*((p \cdot q)|_{X_{p\cdot\varepsilon}})$. Similar we get $P_*((p \cdot q)|_{X_{\varepsilon\cdot q}})$. Using (6.5) we then conclude $P_*(p \cdot q)$. $\square$

With this proposition it is easy to prove that the multiset induced pomset properties are examples of dot synthesizable pomset properties:

Proposition 6.4.8 The multiset induced pomset properties are dot synthesizable.

Proof Given a set of multisets, D, we show that $P_{\mathsf{M}\subseteq\mathsf{D}}$ satisfies the condition in proposition 6.4.7 above. Let p be any lpo and Y a subset of X_p with $\forall x \in X_p \setminus Y \forall y \in Y.\, x\ c\!\!\not{o}_p\ y$. The latter of course implies that any (nonempty) co_p-set, Z, must be contained in either $X_p \setminus Y$ or Y. So if $P_{\mathsf{M}\subseteq\mathsf{D}}(p|_{X_p\setminus Y})$ and $P_{\mathsf{M}\subseteq\mathsf{D}}(p|_Y)$ we conclude that $[p|_Z]$ must be contained in D, and so $P_{\mathsf{M}\subseteq\mathsf{D}}(p)$ as desired. □

That the condition in proposition 6.4.7 is not necessary for a pomset property to be dot synthesizable can be seen from the following example.

Example: Let P_{ex} be the dot synthesizable pomset property defined by: $P_{ex}(\mathbf{p})$ *iff* every element of $\mathbf{p}$ has an immediate neighbour with the same label. E.g., $P_{ex}(\varepsilon)$ and $P_{ex}(a \substack{\nearrow a \searrow \\ \searrow b \nearrow} b)$ but neither $P_{ex}(a)$ nor $P_{ex}(\mathbf{p})$ where $\mathbf{p} = a \to b \to a \to b$. If Y is the two elements of X_p labelled with b then $\forall x \in X_p \setminus Y \forall y \in Y.\, x\ c\!\!\not{o}_p\ y$. Also $[p|_{X_p\setminus Y}] = a \to a$ and $[p|_Y] = b \to b$ which both have the P_{ex}-property. As already stated $P_{ex}(\mathbf{p})$ does not hold wherefore the condition of the proposition is not satisfied.

Of course we cannot be sure that $\delta_*(\mathbf{p})$ is nonempty no matter whether we have to do with hereditary or dot synthesizable pomset properties. Take for instance the pomset property which is not fulfilled by any pomset. The next proposition states a condition which ensures $\delta_*(\mathbf{p})$ not to be empty.

Proposition 6.4.9 Let P_* be a dot synthesizable pomset property such that $P_*(\varepsilon)$ and for every singleton pomset a, $P_*(a)$. Then $\delta_*(\mathbf{p}) \neq \emptyset$ for every pomset $\mathbf{p}$.

Proof Let a pomset $\mathbf{p}$ be given. The proof is by induction on the number of elements in $\mathbf{p}$. The basis $\mathbf{p} = \varepsilon$ holds by the assumption of the proposition. So assume $\mathbf{p} \neq \varepsilon$. We can then choose an $x \in X_p$ minimal w.r.t. $\leq_p$. Then $\{x\}$ is $\leq_p$-downwards closed and by the alternative characterization of prefix then $a := [p|_{\{x\}}] \sqsubseteq \mathbf{p}$. By proposition 6.2.7 we find a $\mathbf{p}'$ such that $a \cdot \mathbf{p}' \preceq \mathbf{p}$. Clearly $\mathbf{p}'$ must have less elements than $\mathbf{p}$, so by hypothesis of induction $\exists \mathbf{q} \in \delta_*(\mathbf{p}')$—i.e., $\mathbf{q} \preceq \mathbf{p}'$ and $P_*(\mathbf{q})$. From the $\preceq$-monotonicity of $\cdot$ then $a \cdot \mathbf{q} \preceq a \cdot \mathbf{p}' \preceq \mathbf{p}$. By the assumption of the proposition $P_*(a)$ and we know $P_*(\mathbf{q})$ so $P_*(a \cdot \mathbf{q})$ follows from proposition 6.4.7. Hence $a \cdot \mathbf{q} \in \delta_*(\mathbf{p})$. □

As an example of the use of this proposition consider the pomset property P_{W}—a pomset being a word. Using proposition 6.4.7 one from the definition (trichotomy law) easely sees that P_{W} is dot synthesizable. Also the other assumptions of the lemma are fulfilled, so we conclude $\delta_{\mathsf{W}}(\mathbf{p}) \neq \emptyset$ for every pomset $\mathbf{p}$.

Proposition 6.4.10 Let P_* be a dot synthesizable pomset property. Then:

$$\delta_*(\mathbf{p}_0 \cdot \mathbf{p}_1) \supseteq \delta_*(\mathbf{p}_0) \cdot \delta_*(\mathbf{p}_1)$$

Proof Given $\mathbf{q} \in \delta_*(\mathbf{p}_0) \cdot \delta_*(\mathbf{p}_1)$. Then $\mathbf{q} = \mathbf{p}_0' \cdot \mathbf{p}_1'$ for some $\mathbf{p}_i' \in \delta_*(\mathbf{p}_i)$ and $i = 0, 1$. This implies $P_*(\mathbf{p}_i')$ and $\mathbf{p}_i' \preceq \mathbf{p}_i$ for $i = 0, 1$, so as a consequence of the $\preceq$-monotonicity of $\cdot$ then $\mathbf{p}_0' \cdot \mathbf{p}_1' \preceq \mathbf{p}_0 \cdot \mathbf{p}_1$, and $P_*(\mathbf{p}_0' \cdot \mathbf{p}_1')$ since P_* is dot synthesizable. Hence $\mathbf{q} \in \delta_*(\mathbf{p}_0 \cdot \mathbf{p}_1)$. □

So if a pomset property, P_*, is both hereditary and dot synthesizable we from this proposition and a) of proposition 6.4.4 see:

$$\delta_*(\mathbf{p}_0 \cdot \mathbf{p}_1) = \delta_*(\mathbf{p}_0) \cdot \delta_*(\mathbf{p}_1)$$

If in addition P_* holds for ε and the singleton pomsets we from proposition 6.4.5 and the following proposition get:

$$\delta_*\pi(\mathbf{p}) = \pi\delta_*(\mathbf{p})$$

In the following chapters we shall only meet such pomset properties.

Proposition 6.4.11 Suppose P_* is a dot synthesizable pomset property holding for ε and the singleton pomsets. Then for every pomset $\mathbf{p}$:

$$\delta_*\pi(\mathbf{p}) \subseteq \pi\delta_*(\mathbf{p})$$

Proof Suppose $\mathbf{q} \in \delta_*\pi(\mathbf{p})$. Then $P_*(\mathbf{q})$ and there is a pomset $\mathbf{r}$ with $\mathbf{q} \preceq \mathbf{r} \sqsubseteq \mathbf{p}$. As in the proof of proposition 6.2.8 we can find a $\mathbf{r}'$ such that $\mathbf{q} \cdot \mathbf{r}' \preceq \mathbf{p}$. We presume the same of P_* as in proposition 6.4.9, so there is a $\mathbf{p}' \in \delta_*(\mathbf{r}')$. Hence $P_*(\mathbf{p}')$ and by the $\preceq$-monotonicity of $\cdot$ also $\mathbf{q} \cdot \mathbf{p}' \preceq \mathbf{q} \cdot \mathbf{r}' \preceq \mathbf{p}$. $P_*(\mathbf{q} \cdot \mathbf{p}')$ follows from $P_*(\mathbf{q})$ and $P_*(\mathbf{p}')$. Because $\mathbf{q} \sqsubseteq \mathbf{q} \cdot \mathbf{p}'$ we actually have $\mathbf{q} \in \pi\delta_*(\mathbf{p})$. □

Chapter 7

BL—A Basic Process Language

As mentioned in the presentation we shall study degrees of nonsequentiality as "orthogonal" to existing study of branching, and as a consequence hereof the process expressions we shall use will be from a very basic language, BL, over the abstract set of action symbols, Δ, containing a combinator for internal nondeterminism beside combinators for sequencing and parallelism with auto-parallelism (but without communication).

BL consists of expressions of the form:

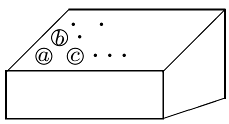
E	$::=$	a	individual process labelled $a \in \Delta$
		$E_0 \,;\, E_1$	sequential composition of E_0 and E_1
		$E_0 \oplus E_1$	internal nondeterministic composition of E_0 and E_1
		$E_0 \parallel E_1$	parallel composition of E_0 and E_1.

In all models to come these binary operators are associative, a fact we shall make use of in examples together with the combinator precedence:

$$\oplus \quad < \quad \parallel \quad < \quad ;$$

7.1 General Semantics

In the tradition as initiated in [HM80] our starting point will be the idea of an observer experimenting by doing tests on a black-box containing a process.



Tests consists in pushing buttons until some bulb is lightning up indicating the termination of the process. A direct test could be to try to push a button and a full test can then be considered as a maximal sequence of direct tests.

Within the branching tradition a widespread technique to increase an observers capability to distinguish nondeterministic processes is to provide the observer more sophisticated,

but natural means of making direct tests—e.g., in the readiness semantic where it is directly possible to test which buttons one successfully could push. How powerful these capabilities should be depends on the purpose and application [OH86].
In the line of this we shall look for natural direct tests which puts the observer in a position to discriminate degrees of nonsequentiality by processes, but remains faithful to the idea of an observer pushing buttons on a black-box.

Keeping the analogy of a human observer the weakest form of an direct test must be that of an observer pushing buttons using just óne finger. But also simultaneously observations are conceivable [Mil80]. Clearly some power of the direct test is gained if the observer uses two fingers at the same time thereby enabling the observer to direct test whether two different labelled individual processes could be started at the same time. Another approach would be to realize the force used to push the button—reflecting how many individual processes with equal label could be started at the same time. These two directions for increasing the power of the direct test seems to span the possibilities for an observer experimenting through pushing buttons by the fingers. Of course the combination of these directions opens up for a large variety with óne button direct tests at one extreme and finitely many button push with realized force for each, at the other extreme. It is difficult to argue which one to choose in this spectrum and in the end it must be a matter of application. As an example of one application consider the situation where more processes have access to a common store. Here it would be suitable if only direct tests with at most óne write in the common store is possible.

On the basis of sequences of direct test equivalences on a simple language, BL, will be defined. We can then investigate what consequences a choice of direct tests can have. However for an extension, RBL, of BL which allows change of atomicity, we shall later see that the actual choice is irrelevant if the equivalences are demanded to be congruences.

We now formalize the direct tests and add some "natural" requirements.

Definition 7.1.1 A set of *direct tests*, G, is a set of nonempty multisets satisfying:

$$\Delta \subseteq \mathsf{G}$$
$$A \hookrightarrow B, B \in \mathsf{G} \Rightarrow A \in \mathsf{G}_\varepsilon$$

The first demand says that an observer at least should be capable of doing the weakest direct test: push óne button. The second demand means that an observer capable of doing one direct test also should be able to do any weaker direct test.

Evidently our tests resembles the sequences of firing steps used to express nonsequential behaviours of processes in Petri nets [Rei85].

It is possible to carry through more quibbling observations as the partial order observations of [DM87] and in [BC87] transitions like $(a\,;b \parallel a)$ are possible. However one might argue that it is difficult to give "natural" intuition supporting such observations.

7.2 Operational Set-up

The sequence of direct test which can be performed will be build up from the direct test relation $\Rightarrow_{\mathsf{G}}$ holding through an $A \in \mathsf{G}$ between configurations, with each BL-expression being a possible start configuration. Configurations are expressions from CL, which is almost like BL with Δ extended with $\dagger$ (a symbol distinct from those of Δ). Intuitively $\dagger$ represents the extinct action. Formally CL is defined to be the least set C satisfying:

$$\begin{aligned} \dagger &\in C \\ BL &\subseteq C \\ E_0 \,;\, E_1 &\in C \quad \text{if } E_0 \in C \text{ and } E_1 \in BL \\ E_0 \parallel E_1 &\in C \quad \text{if } E_0, E_1 \in C \end{aligned}$$

The construction of CL reflects the idea that control cannot pass ; before all previous actions are extinct.

Example: $a \parallel (\dagger \,;\, b) \in CL$ but $\dagger \oplus a \notin CL$ and $a \,;\, (\dagger \,;\, b) \notin CL$.

We shall often prove properties by induction on the structure of an $E \in CL$. Strictly speaking we then first prove the property for expressions from BL and then look at $\dagger$ and sequential/ parallel composition afterwards. This implies that e.g., $E = E_0 \,;\, E_1$ shall be treated two times with the only difference that for the first time we can assume E_0 not to contain $\dagger$. We will therefore treat these cases together except at rare occasions where the distinction is crucial. The same applies for $E = E_0 \parallel E_1$.

So $\Rightarrow_{\mathsf{G}}$ is actually a subset of $CL \times \mathsf{G} \times CL$. If $\langle E, A, E' \rangle \in \Rightarrow_{\mathsf{G}}$ we write this as $E \overset{A}{\Rightarrow}_{\mathsf{G}} E'$. One can think of this as E can evolve to E' when the direct test A is performed.

We shall follow DeNicola [Nic87] and Hennessy [Hen88a] when defining $\Rightarrow_{\mathsf{G}}$. Hennessy does this in an extended labelled transition system by means of a relation $\succ\!\!\longrightarrow$, which reflects the step of an internal computation, and by a relation $\longrightarrow_{\mathsf{G}}$ for an external computation step corresponding to a direct test. The slight deviation from Hennessy in defining the relation, $\succ\!\!\longrightarrow$, for internal steps are manily due to differences in the languages considered.

Definition 7.2.1 $\rightarrowtail \subseteq CL \times CL$ and $\longrightarrow_{\mathsf{G}} \subseteq CL \times \mathsf{G} \times CL$ are defined as the least relations satisfying the following axioms and inference rules.

$$a \xrightarrow{a}_{\mathsf{G}} \dagger \qquad \frac{E_0 \xrightarrow{A}_{\mathsf{G}} E_0'}{E_0 \,;\, E_1 \xrightarrow{A}_{\mathsf{G}} E_0' \,;\, E_1}$$

$$\frac{E_0 \xrightarrow{A}_{\mathsf{G}} E_0'}{\begin{array}{c} E_0 \parallel E_1 \xrightarrow{A}_{\mathsf{G}} E_0' \parallel E_1 \\ E_1 \parallel E_0 \xrightarrow{A}_{\mathsf{G}} E_1 \parallel E_0' \end{array}} \qquad \frac{E_0 \xrightarrow{A_0}_{\mathsf{G}} E_0', E_1 \xrightarrow{A_1}_{\mathsf{G}} E_1', A_0 \times A_1 \in \mathsf{G}}{E_0 \parallel E_1 \xrightarrow{A_0 \times A_1}_{\mathsf{G}} E_0' \parallel E_1'}$$

$$\dagger \,;\, E \rightarrowtail E \qquad \frac{E_0 \rightarrowtail E_0'}{E_0 \,;\, E_1 \rightarrowtail E_0' \,;\, E_1}$$

$$\begin{array}{l} E_0 \oplus E_1 \rightarrowtail E_0 \\ E_0 \oplus E_1 \rightarrowtail E_1 \end{array}$$

$$\begin{array}{l} \dagger \parallel E \rightarrowtail E \\ E \parallel \dagger \rightarrowtail E \end{array} \qquad \frac{E_0 \rightarrowtail E_0'}{\begin{array}{c} E_0 \parallel E_1 \rightarrowtail E_0' \parallel E_1 \\ E_1 \parallel E_0 \rightarrowtail E_1 \parallel E_0' \end{array}}$$

In this way an internal step either resolves an internal nondeterministic choice or removes an extinct action. The idea of using $\rightarrowtail$ for other purposes than resolving internal nondeterministic choices is not inconsistent with Hennessy—he also uses $\rightarrowtail$ to unfold recursive definitions.

Notice that the definition of $\xrightarrow{A}_{\mathsf{G}}$ is well-defined because of the premise $A_0 \times A_1 \in \mathsf{G}$ in the rule for a composed action and because we assume $\Delta \subseteq \mathsf{G}$ (for $a \xrightarrow{a}_{\mathsf{G}} \dagger$).

Example: Let G be a set of direct test containing a^2. Then

$$a \,;\, b \parallel a \,;\, d \xrightarrow{a^2}_{\mathsf{G}} \dagger \,;\, b \parallel \dagger \,;\, d \rightarrowtail \dagger \,;\, b \parallel d \xrightarrow{d}_{\mathsf{G}} \dagger \,;\, b \parallel \dagger \rightarrowtail b \parallel \dagger \xrightarrow{b}_{\mathsf{G}} \dagger \parallel \dagger \rightarrowtail \dagger$$

The test relation, $\overset{A}{\Rightarrow}_{\mathsf{G}}$, is now defined as $\rightarrowtail^* \xrightarrow{A}_{\mathsf{G}} \rightarrowtail^*$ and for a sequence of direct tests $s \in \mathsf{G}^*$ and $E, E' \in CL$ we define:

$$E \overset{s}{\Rightarrow}_{\mathsf{G}} E', s = A_1 A_2 \ldots A_n$$

iff

$$\exists E_1, \ldots, E_n \in CL \exists A_1, \ldots, A_n \in \mathsf{G}, n \geq 0.$$
$$E \overset{A_1}{\Longrightarrow}_{\mathsf{G}} E_1 \overset{A_2}{\Longrightarrow}_{\mathsf{G}} \ldots \overset{A_n}{\Longrightarrow}_{\mathsf{G}} E_n = E'$$

where the case $n = 0$ means $E \rightarrowtail^* E'$.

With this notion of sequences of direct test it follows that any maximal sequence, s, of direct is of the form $E \overset{s}{\Rightarrow}_{\mathsf{G}} \dagger$, so we can define our basic operational preorder:

Definition 7.2.2 $\lesssim_{\mathsf{G}} \subseteq BL \times BL$

$$E_0 \lesssim_{\mathsf{G}} E_1$$

iff

$$E_0 \stackrel{s}{\Rightarrow}_{\mathsf{G}} \dagger \text{ implies } E_1 \stackrel{s}{\Rightarrow}_{\mathsf{G}} \dagger \text{ for all } s \in \mathsf{G}^* \qquad \Box$$

Notice that as expected the equivalence of $\lesssim_{\mathsf{G}}$, $\approx_{\mathsf{G}}$, identifies $a\,;(b \oplus c)$ and $a\,; b \oplus a\,; c$.

Throughout this section we will fix G and so will leave it out as a subscript of $\longrightarrow_{\mathsf{G}}$ and $\Rightarrow_{\mathsf{G}}$ except when dealing with certain G's. This will also be the case in the remaining sections whenever the direct test set G in question is clear from the context.

Given a concrete sequence of internal and external steps, written $\stackrel{s}{\Rightarrow}$, we define its length as the total number of steps in the sequence. If E under this sequence evolves to E' we also write this as $E \stackrel{s}{\Rightarrow} E'$. This allows us to make induction on the length of a concrete sequence.

As a first result notice that by an easy induction on the length of $\stackrel{s}{\Rightarrow}$ (where $\stackrel{s}{\Rightarrow}$ is a concrete sequence for $E_0 \stackrel{s}{\Rightarrow} E_0'$) one can prove:

Proposition 7.2.3 Suppose $E \in BL$, $E_0, E_1 \in CL$ and $E_0 \stackrel{s}{\Rightarrow} E_0'$. Then

- $E_0\,; E \stackrel{s}{\Rightarrow} E_0'\,; E$
- $E_0 \parallel E_1 \stackrel{s}{\Rightarrow} E_0' \parallel E_1$
- $E_1 \parallel E_0 \stackrel{s}{\Rightarrow} E_1 \parallel E_0'$

Since we only have a combinator for internal nondeterministic choice a natural question to raise is whether a processes reacts successful to a test *iff* one of the syntactic "controlled behaviours" of it does. Such a behaviour can be regarded as a deterministic process ($\in DBL$) or configuration ($\in DCL$)—deterministic in the sense that no internal nondeterminism is explicit present in the form of a $\oplus$-combinator, but of course their might be indirectly as in $a \parallel a$. A behaviour would in Petri net terms correspond to a possible process/ concurrent behaviour of a Petri net system—more accurately it would correspond to to an occurrence net of a place/ transition net [BF88]. Formally:

Definition 7.2.4 *Behaviours*

The set of configuration *behaviours*, DCL, is defined to be the $\oplus$-*free* expressions of CL. Similar $DBL = DCL \cap BL$ is the set of process behaviours.

The behaviours of a configuration expression is given by the map $\text{Beh} : CL \longrightarrow \mathcal{P}(DCL)$ defined as follows:

$$\begin{aligned}
\text{Beh}(\dagger) &= \{\dagger\} \\
\text{Beh}(a) &= \{a\} \\
\text{Beh}(E_0\,; E_1) &= \text{Beh}(E_0)\,; \text{Beh}(E_1) \\
\text{Beh}(E_0 \oplus E_1) &= \text{Beh}(E_0) \cup \text{Beh}(E_1) \\
\text{Beh}(E_0 \parallel E_1) &= \text{Beh}(E_0) \parallel \text{Beh}(E_1)
\end{aligned}$$

where $\mathrm{Beh}(E_0)\,;\mathrm{Beh}(E_1)$ denotes $\{E_0'\,;E_1' \mid E_0' \in \mathrm{Beh}(E_0), E_1' \in \mathrm{Beh}(E_1)\}$. Similar for $\|$. □

Notice that $E \in BL$ implies $\mathrm{Beh}(E) \subseteq DBL$.

Because $BL \subseteq CL$ we from the proposition below deduce a positive answer to the question whether a processes reacts successful to a test *iff* one of the syntactic "controlled behaviours" of it does.

Proposition 7.2.5 For a configuration $E \in CL$ and $s \in \mathsf{G}^*$ we have

$$E \stackrel{s}{\Rightarrow} \dagger \text{ iff } \exists F \in \mathrm{Beh}(E).\, F \stackrel{s}{\Rightarrow} \dagger$$

Because in general $\dagger \in \mathrm{Beh}(E)$ *iff* $E = \dagger$ this proposition is immediate from:

Proposition 7.2.6 Given $s \in \mathsf{G}^*$ and configurations E and F'. Then:

$$\begin{array}{l} \exists E' \stackrel{s}{\Rightarrow} E', F' \in \mathrm{Beh}(E') \\ \Updownarrow \\ \exists F \in \mathrm{Beh}(E).\, F \stackrel{s}{\Rightarrow} F' \end{array}$$

Proof Both implication are proven by induction on the length of $\stackrel{s}{\Rightarrow}$ using the following three propositions. □

Proposition 7.2.7 Given configurations E and F' we have:

$$\begin{array}{l} \exists E'.\, E \rightarrowtail E', F' \in \mathrm{Beh}(E') \\ \Downarrow \\ \exists F \in \mathrm{Beh}(E).\, F \rightarrowtail^* F' \end{array}$$

Proof By induction on the structure of E.

$E = \dagger$ *or* $E = a$: In both cases E cannot do any internal step so the implication holds vacuously.

$E = E_0\,; E_1$: According to the definition of $\rightarrowtail$ there are two subcases:

$E_0 = \dagger$: Then $E' = E_1$ and $F' \in \mathrm{Beh}(E_1)$. Now $\mathrm{Beh}(E) = \dagger\,; \mathrm{Beh}(E_1)$ so $F := \dagger\,; F' \in \mathrm{Beh}(E)$ and of course $D \rightarrowtail F'$.

$E_0 \rightarrowtail E_1'$: I.e., $E' = E_0'\,; E_1$, so $F \in \mathrm{Beh}(E')$ means $F' = F_1'\,; F_1$ for some $F_0' \in \mathrm{Beh}(E_0')$, $F_1 \in \mathrm{Beh}(E_1)$. By induction $\exists F_0 \in \mathrm{Beh}(E_0).\, F_0 \rightarrowtail^* F_0'$. By proposition 7.2.3 this implies $F := F_0\,; F_1 \rightarrowtail^* F_0'\,; F_1 = F'$. Since $F_i \in \mathrm{Beh}(E_i)$ for $i = 0, 1$ we also have $F \in \mathrm{Beh}(E)$.

$E = E_0 \oplus E_1$: By definition of $\rightarrowtail$ then either $E' = E_0$ or $E' = E_1$. W.l.o.g. assume $E' = E_0$. Then $F \in \mathrm{Beh}(E')$ means $F' \in \mathrm{Beh}(E_0) \subseteq \mathrm{Beh}(E_0) \cup \mathrm{Beh}(E_1) = \mathrm{Beh}(E_0 \oplus E_1)$ so we can just choose $F = F'$ since $F' \rightarrowtail^0 F'$.

$E = E_0 \parallel E_1$: Again according to the definition of $\rightarrowtail$ there are four possibilities. If the internal step $E_0 \parallel E_1 \rightarrowtail E'$ derives from one of the axioms for $\parallel$ the proof goes similar/ symmetric as in the first subcase of $E = E_0 ; E_1$ and if it derives from one of the inference rules for $\parallel$ it goes as in the second subcase.

□

Proposition 7.2.8 If E and F' are configurations then:

$$\exists F \in \mathrm{Beh}(E).\, F \rightarrowtail F'$$
$$\Downarrow$$
$$\exists E'.\, E \rightarrowtail^* E', F' \in \mathrm{Beh}(E')$$

Proof By induction on the structure of E.

$E = \dagger$ *or* $E = a$: Then $\mathrm{Beh}(E) = \{E\}$ and $F = E$. Since E of this form can do no internal step the implication holds trivially.

$E = E_0 ; E_1$: $F \in \mathrm{Beh}(E_0 ; E_1)$ means $F = F_0 ; F_1$ where $F_i \in \mathrm{Beh}(E_i)$ for $i = 0, 1$.
$E_0 = \dagger$: Since $F_0 \in \mathrm{Beh}(\dagger)$ implies $F_0 = \dagger$ we see that $F \rightarrowtail F'$ implies $F' = F_1$. Let $E' = E_1$. We have $E = \dagger ; E_1 \rightarrowtail E'$ with $F' \in \mathrm{Beh}(E')$ as desired.

$E_0 \neq \dagger$: Now $F_0 \in \mathrm{Beh}(E_0)$, $E_0 \neq \dagger$ implies $F_0 \neq \dagger$. Inspecting the definition of $\rightarrowtail$ we see that $F \rightarrowtail F'$ must be due to $F_0 \rightarrowtail F_0'$, $F' = F_0'; F_1$. By hypothesis of induction there exists E_0' such that $E_0 \rightarrowtail^* E_0'$ and $F_0' \in \mathrm{Beh}(E_0')$. Then also $E \rightarrowtail^* E_0' ; E_1$ and $F' \in \mathrm{Beh}(E_0') ; \mathrm{Beh}(E_1)$. Choosing $E' = E_0' ; E_1$ we are done.

$E = E_0 \oplus E_1$: $F \in \mathrm{Beh}(E)$ means $F \in \mathrm{Beh}(E_0)$ or $F \in \mathrm{Beh}(E_1)$. Suppose w.l.o.g. $F \in \mathrm{Beh}(E_0)$. Then by induction $\exists E'.\, E_0 \rightarrowtail^* E', F' \in \mathrm{Beh}(E')$. By definition of $\rightarrowtail$ then also $E_0 \oplus E_1 \rightarrowtail E_0 \rightarrowtail^* E'$ and thereby $E \rightarrowtail^* E'$.

$E = E_1 \parallel E_1$: Has similar/ symmetric subcases to those of $E = E_0 ; E_1$.

□

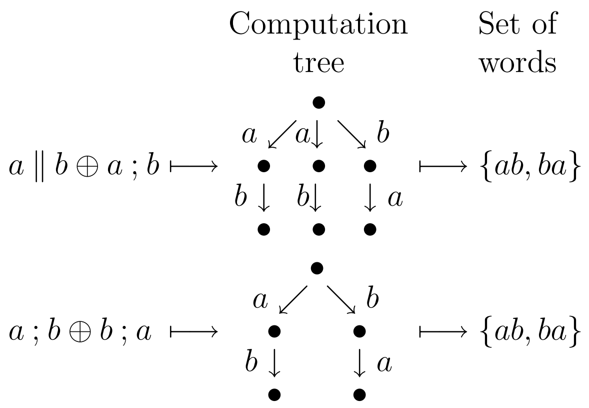
By an easy induction on the structure of E one can prove:

Proposition 7.2.9 For configurations E and F' we have:

$$\exists E'.\, E \xrightarrow{A} E', F' \in \mathrm{Beh}(E')$$
$$\Updownarrow$$
$$\exists F \in \mathrm{Beh}(E).\, F \xrightarrow{A} F'$$

7.3 Denotational Set-up

The well-known trace models (not necessarily maximal) e.g., [OH86, Hoa85] are based on sets of sequences of actions from Δ (words) and using the shuffle operator when dealing with $\parallel$. These and related models can be viewed as abstractions of computations trees canonicical associated with the process expressions. In the trace models for the equivalence corresponding to the smallest set of direct tests the abstraction would consist in taking the set of words which constitute the paths from the root to the leaves of the computation tree as illustrated in:



With these models in mind it offers it self for the generalized traces, to look for models based on sets of sequences of direct tests, i.e., subsets of G^*. However because immediate tests are directed towards discovering concurrency as mirrored in pomsets and in order to clear the way for the more complicated model in the next chapter we shall devise corresponding models in the pomset framework. So the idea is to obtain a similar picture as above using pomsets in stead.

Example: If we intuitively think of $\wp$ as associating pomsets to expressions and δ_{W} gives the linearizations of pomsets we expect:

	Set of pomsets		Set of pomsets
$a \parallel b \oplus a\,;b \overset{\wp}{\longmapsto}$	$\left\{\begin{matrix} a \\ b, \\ a \rightarrow b \end{matrix}\right\}$	$\overset{\delta_{\mathsf{W}}}{\longmapsto}$	$\left\{\begin{matrix} a \rightarrow b, \\ b \rightarrow a \end{matrix}\right\}$
$a\,;b \oplus b\,;a \overset{\wp}{\longmapsto}$	$\left\{\begin{matrix} a \rightarrow b, \\ b \rightarrow a \end{matrix}\right\}$	$\overset{\delta_{\mathsf{W}}}{\longmapsto}$	$\left\{\begin{matrix} a \rightarrow b, \\ b \rightarrow a \end{matrix}\right\}$

To make this picture precise and generalize to an arbitrary set of direct tests, G, we shall at first look for pomsets which only contains multisets from G. From $\mathsf{G} \subseteq \mathsf{M}$ and the definition of multiset induced pomset properties we know that $\mathbf{P}_{\mathsf{M} \subseteq \mathsf{G}}$ are the pomsets we are looking for.

For arbitrary pomset properties, P_* and $P_{*'}$, we denote $\mathbf{P}_* \cap \mathbf{P}_{*'}$ by $\mathbf{P}_{*,*'}$ and similar for the δ_*-closure we for a pomset $\mathbf{p}$ denote $\delta_*(\mathbf{p}) \cap \mathbf{P}_{*'}$ by $\delta_{*,*'}(\mathbf{p})$.

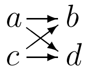
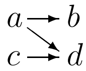
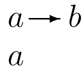
Notice that $\delta_{*,*'}(\mathbf{p})$ alternatively may be written as $\{\mathbf{q} \in \mathbf{P} \mid \mathbf{q} \preceq \mathbf{p}, P_*(\mathbf{p}) \textit{ and } P_{*'}(\mathbf{p})\}$.

Next it seems natural to seek a pomset property reflecting the general nature of when the multisets of a pomset are in sequence, i.e., pomsets of the form $A_1 \cdot A_2 \cdot \ldots \cdot A_n$. Pomsets of this form can be considered as "layered" in the sense that they may be viewed as "a linear order on top of a set of completely unordered pomsets (the individual multisets, A_i's)". One way to formalize this property is the following:

Definition 7.3.1 *P_{and}-Property for Pomsets*

A pomset $\mathbf{p}$ is said to have the P_{and}-property, $P_{and}(\mathbf{p})$, *iff* for all x, x', y, y' in X_p we have:

$$\text{if} \quad \begin{matrix} x \\ co_p \\ y \end{matrix} \quad \text{then} \quad \begin{matrix} \forall z.\, x <_p z \Rightarrow y <_p z \\ \textit{and} \\ \forall z.\, y <_p z \Rightarrow x <_p z \end{matrix}$$ □

Example:  has the P_{and}-property,  and  has not.

Proposition 7.3.2 The P_{and}-property is hereditary and dot synthesizable.

Proof Because of the universal quantification of x and y in the definition of $P_{and}(\mathbf{p})$ and because the partial order just is restricted in subpomsets it follows that the P_{and}-property is hereditary.

It is also dot synthesizable as can be seen by using proposition 6.4.7: Let a lpo p and a subset Y of X_p given such that $P_{and}(p|_{X_p \setminus Y})$,$P_{and}(p|_Y)$ and

$$\forall x \in X_p \setminus Y \forall y \in Y.\, x\ c\!\!\not{o}_p\ y \tag{7.1}$$

Suppose on the contrary $\neg P_{and}(p)$. By definition there must be $x, y, z \in X_p$ with $y\ co_p\ x <_p z$ and $y \not<_p z$. $y \not<_p z$ cannot mean $z \leq_p y$ because we then would get $x \leq_p y$ contradicting $y\ co_p\ x$, so actually:

$$y\ co_p\ x <_p z\ co_p\ y \tag{7.2}$$

If x, y and z all are in one of the two sets $X_p \setminus Y$ and Y, we get a contradiction to $P_{and}(p|_{X_p \setminus Y})$ and $P_{and}(p|_Y)$. Otherwise one element must be in one of the sets and the remaining two in the other set. From (7.2) we see that at least one of the two elements belonging to the same set must be concurrent to the element in the other set—a contradiction to (7.1). □

Proposition 7.3.3 The P_{and}-property has the following alternative characterization:

$$(\mathbf{p} \neq \varepsilon \textit{ and } P_{and}(\mathbf{p})) \quad \textit{iff} \quad \exists n \geq 1 \exists A_1, \ldots, A_n \in \mathsf{M}.\, A_1 \cdot \ldots \cdot A_n = \mathbf{p}$$

From this and the definition of $P_{\mathsf{M} \subseteq \mathsf{G}, and}$ we immediately get:

$$(\mathbf{p} \neq \varepsilon \textit{ and } P_{\mathsf{M} \subseteq \mathsf{G}, and}(\mathbf{p})) \quad \textit{iff} \quad \exists n \geq 1 \exists A_1, \ldots, A_n \in \mathsf{G}.\, A_1 \cdot \ldots \cdot A_n = \mathbf{p}$$

We abbreviate $P_{\mathsf{M} \subseteq \mathsf{G}, and}$ by P_{G}. $P_{\mathsf{M} \subseteq \mathsf{D}}$ is clearly hereditary so from the propositions 6.4.8 and 7.3.2 we get:

Corollary 7.3.4 The P_{G}-property is hereditary and dot synthesizable.

With the above biimplications it is not hard to see that G^* and $\mathbf{P}_{\mathsf{G}}$ coincide (are isomorphic), and as a consequence we shall often identify them in the sequel. So there is hope that we can base our models on subsets of $\mathbf{P}_{\mathsf{G}}$.

It only remains to establish a connection from BL-expressions to nonempty subsets of $\mathbf{P}_{\mathsf{G}}$. To this end we introduce a canonical map which give a natural association of sets of pomsets with BL-expressions.

Definition 7.3.5 *Canonical Pomset Association*
The *canonical associated pomsets* of a BL-expression is given by the map $\wp : BL \longrightarrow \mathcal{P}(\mathbf{P} \setminus \{\varepsilon\}) \setminus \emptyset$ defined compositionally as follows:

$$
\begin{aligned}
\wp(a) &= \{a\} \\
\wp(E_0 \,;\, E_1) &= \wp(E_0) \cdot \wp(E_1) \\
\wp(E_0 \oplus E_1) &= \wp(E_0) \cup \wp(E_1) \\
\wp(E_0 \parallel E_1) &= \wp(E_0) \times \wp(E_1)
\end{aligned}
$$

Example:
$$\wp((a \oplus b) \,;\, (a \parallel c)) = \left\{ a \genfrac{}{}{0pt}{}{\nearrow a}{\searrow c},\; b \genfrac{}{}{0pt}{}{\nearrow a}{\searrow c} \right\}$$

We can then let denotations in our models go via this map:

Definition 7.3.6 $[\![\_]\!]_{\mathsf{G}} : BL \longrightarrow \mathcal{P}(\mathbf{P}_{\mathsf{G}} \setminus \{\varepsilon\}) \setminus \emptyset$ with $[\![E]\!]_{\mathsf{G}} = \delta_{\mathsf{G}}(\wp(E))$. □

So, our G-model is finite sets of P_{G}-pomsets partially ordered by inclusion: $\subseteq$.

It is easy to check that the maps of the example on page 156 are correct and that they composed correspond to the denotational map just defined.

$[\![\_]\!]_{\mathsf{G}}$ together with the partial order induces a denotational preorder $\trianglelefteq_{\mathsf{G}}$ over BL by:

$$E_0 \trianglelefteq_{\mathsf{G}} E_1 \textit{ iff } [\![E_0]\!]_{\mathsf{G}} \subseteq [\![E_1]\!]_{\mathsf{G}}$$

Having models using sets of sequences from Δ^* in mind it is not hard to come up with:

Theorem 7.3.7 $[\![\_]\!]_{\mathsf{G}}$ can be defined compositionally by:

$$
\begin{aligned}
[\![a]\!]_{\mathsf{G}} &= \{a\} \\
[\![E_0 \,;\, E_1]\!]_{\mathsf{G}} &= [\![E_0]\!]_{\mathsf{G}} \cdot [\![E_1]\!]_{\mathsf{G}} \\
[\![E_0 \oplus E_1]\!]_{\mathsf{G}} &= [\![E_0]\!]_{\mathsf{G}} \cup [\![E_1]\!]_{\mathsf{G}} \\
[\![E_0 \parallel E_1]\!]_{\mathsf{G}} &= \delta_{\mathsf{G}}([\![E_0]\!]_{\mathsf{G}} \times [\![E_1]\!]_{\mathsf{G}})
\end{aligned}
$$

Notice that δ_{G} here acts as the natural generalization of the shuffle/ zip operator for Δ^*.

Proof At first notice that corollary 7.3.4 enables us to apply the propositions 6.4.4 and 6.4.10 in the proof. We look at the different cases:

a: Evident by inspection of the definitions.

$E_0 \,;\, E_1$: Follows directly from the fact that δ_{G} distributes over pomset sequential composition (proposition 6.4.4 and 6.4.10). See also the last case.

$E_0 \oplus E_1$: Similar because δ_{G} is a natural generalization to sets and therefore distributes over sets.

$E_0 \parallel E_1$:
$$\begin{aligned}
[\![E_0 \parallel E_1]\!]_{\mathsf{G}} &= \delta_{\mathsf{G}}(\wp(E_0 \parallel E_1)) && \text{definition of } [\![_]\!]_{\mathsf{G}} \\
&= \delta_{\mathsf{G}}(\wp(E_0) \times \wp(E_1)) && \text{definition of } \wp \\
&= \delta_{\mathsf{G}}(\delta_{\mathsf{G}}(\wp(E_0)) \times \delta_{\mathsf{G}}(\wp(E_1))) && \text{proposition 6.4.4} \\
&= \delta_{\mathsf{G}}([\![E_0]\!]_{\mathsf{G}} \times [\![E_1]\!]_{\mathsf{G}}) && \text{definition of } [\![_]\!]_{\mathsf{G}}
\end{aligned}$$

□

7.4 Full Abstractness

The first proposition says that $\trianglelefteq_{\mathsf{G}}$ is inherited in all BL-contexts.

Proposition 7.4.1 $\trianglelefteq_{\mathsf{G}}$ is a precongruence over BL.

Proof From theorem 7.3.7 we know a compositional definition of $[\![\_]\!]_{\mathsf{G}}$ using $\subseteq$-monotone operators (proposition 6.3.4), and hence $\trianglelefteq_{\mathsf{G}}$ is a precongruence. □

Theorem 7.4.2 $[\![\_]\!]_{\mathsf{G}}$ is fully abstract w.r.t. $\lesssim_{\mathsf{G}}$, because

a) $\lesssim_{\mathsf{G}}$ is a precongruence w.r.t. BL

b) $E_0 \lesssim_{\mathsf{G}} E_1$ *iff* $E_0 \trianglelefteq_{\mathsf{G}} E_1$

Proof a) is a consequence of proposition 7.4.1 and b) which in turn is a direct consequence of the proposition below. □

Proposition 7.4.3 For every $E_0, E_1 \in BL$ we have

$$[\![E_0]\!]_{\mathsf{G}} \subseteq [\![E_1]\!]_{\mathsf{G}} \textit{ iff } E_0 \lesssim_{\mathsf{G}} E_1$$

Proof In the last section we saw:

$$(\mathbf{p} \neq \varepsilon \textit{ and } P_{\mathsf{G}}(\mathbf{p})) \;\; \textit{iff} \;\; \exists n \geq 1 \exists A_1, \ldots, A_n \in \mathsf{G}.\, A_1 \cdot \ldots \cdot A_n = \mathbf{p}$$

from which we immediately get:

$$P_{\mathsf{G}}(\mathbf{p}) \;\; \textit{iff} \;\; \exists n \geq 1 \exists A_1, \ldots, A_n \in \mathsf{G}_{\varepsilon}.\, A_1 \cdot \ldots \cdot A_n = \mathbf{p}$$

Recalling our convention to identify G^* and $\mathbf{P}_{\mathsf{G}}$ we then from lemma 7.4.4 below get for $E \in BL$:

$$(7.3) \qquad [\![E]\!]_{\mathsf{G}} = \{A_1 \cdot \ldots \cdot A_n \in \mathsf{G}^*_{\varepsilon} \mid n \geq 1, E \overset{A_1}{\Longrightarrow} \ldots \overset{A_n}{\Longrightarrow} \dagger\} = \{s \in \mathsf{G}^* \mid E \overset{s}{\Rightarrow} \dagger\}$$

with $\overset{\emptyset}{\Rightarrow}$ interpreted according to the convention below.

The proof is now a simple matter: $[\![E_0]\!]_{\mathsf{G}} \subseteq [\![E_1]\!]_{\mathsf{G}}$ *iff* $\{s \in \mathsf{G}^* \mid E_0 \overset{s}{\Rightarrow} \dagger\} \subseteq \{s \in \mathsf{G}^* \mid E_1 \overset{s}{\Rightarrow} \dagger\}$ *iff* $E_0 \lesssim_{\mathsf{G}} E_1$. □

For simplicity of the following lemmas we shall temporarily adopt the notation $E \stackrel{\emptyset}{\longrightarrow} E'$ to mean $E = E'$ wherefore $E \stackrel{\emptyset}{\Rightarrow} E'$ also means $E \succ\!\!\longrightarrow^* E'$. For the same reason the lemmas are formulated slightly stronger than needed where they are used.

Lemma 7.4.4 Given $E \in BL$ and multisets $A_1, \ldots, A_n \in \mathsf{G}_\varepsilon$ $(n \geq 1)$. Then

$$E \stackrel{A_1}{\Longrightarrow} \ldots \stackrel{A_n}{\Longrightarrow} \dagger \;\; \textit{iff} \;\; \exists \mathbf{p} \in \wp(E).\, A_1 \cdot \ldots \cdot A_n \preceq \mathbf{p}$$

Proof Before proving each implication separately notice that $\mathbf{p} \in \wp(E)$ implies $\mathbf{p} \neq \varepsilon$ for $E \in BL$ and that every subexpression of $E \in BL$ belongs to BL too..

If: By induction on the structure of E.

$E = a$: $\wp(a) = \{a\}$ and we have $\mathbf{p} = a$. Clearly $A_1 \cdot \ldots \cdot A_n \preceq a$ implies that exactly one $A_i = \{a\}$, the rest of them equal to $\emptyset$. The result then follows from $a \stackrel{\emptyset}{\Rightarrow} \ldots \stackrel{\emptyset}{\Rightarrow} a \stackrel{a}{\longrightarrow} \dagger \stackrel{\emptyset}{\Rightarrow} \ldots \stackrel{\emptyset}{\Rightarrow} \dagger$.

$E = E_0 \,;\, E_1$: From $\wp(E) = \wp(E_0) \cdot \wp(E_1)$ we then see $\mathbf{p} = \mathbf{p}_0 \cdot \mathbf{p}_1$ where $\mathbf{p}_i \in \wp(E_i)$ for $i = 0, 1$. By lemma 7.4.7 $A_1 \cdot \ldots \cdot A_n \preceq \mathbf{p}_0 \cdot \mathbf{p}_1$ implies $n \geq 2$ and the existence of a $1 \leq j < n$ such that $A_1 \cdot \ldots \cdot A_j \preceq \mathbf{p}_0$ and $A_{j+1} \cdot \ldots \cdot A_n \preceq \mathbf{p}_1$. By hypothesis of induction then $E_0 \stackrel{A_1}{\Longrightarrow} \ldots \stackrel{A_j}{\Longrightarrow} \dagger$ and $E_1 \stackrel{A_{j+1}}{\Longrightarrow} \ldots \stackrel{A_n}{\Longrightarrow} \dagger$. By proposition 7.2.3 then $E_0 \,;\, E_1 \stackrel{A_1}{\Longrightarrow} \ldots \stackrel{A_j}{\Longrightarrow} \dagger \,;\, E_1$. Since $\dagger \,;\, E_1 \succ\!\!\longrightarrow E_1$ we get $E_0 \,;\, E_1 \stackrel{A_1}{\Longrightarrow} \ldots \stackrel{A_j}{\Longrightarrow} E_1 \stackrel{A_{j+1}}{\Longrightarrow} \ldots \stackrel{A_n}{\Longrightarrow} \dagger$ as desired.

$E = E_0 \oplus E_1$: $\mathbf{p} \in \wp(E) = \wp(E_0) \cup \wp(E_1)$ implies $\mathbf{p} \in \wp(E_0)$ or $\mathbf{p} \in \wp(E_1)$. Suppose w.l.o.g. $\mathbf{p} \in \wp(E_0)$. By hypothesis of induction $E_0 \stackrel{A_1}{\Longrightarrow} \ldots \stackrel{A_n}{\Longrightarrow} \dagger$ so from the rules of $\succ\!\!\longrightarrow$ then also $E_0 \oplus E_1 \succ\!\!\longrightarrow E_0 \stackrel{A_1}{\Longrightarrow} \ldots \stackrel{A_n}{\Longrightarrow} \dagger$.

$E = E_0 \parallel E_1$: $\mathbf{p} \in \wp(E) = \wp(E_0) \times \wp(E_1)$ implies $\mathbf{p} = \mathbf{p}_0 \times \mathbf{p}_1$ for some $\mathbf{p}_0 \in \wp(E_0)$ and $\mathbf{p}_1 \in \wp(E_1)$. According to proposition 7.4.10 $A_1 \cdot \ldots \cdot A_n \preceq \mathbf{p}_0 \times \mathbf{p}_1$ implies the existence of multisets $A^0_1, \ldots, A^0_n, A^1_1, \ldots, A^1_n$ such that $A^i_1 \cdot \ldots \cdot A^i_n \preceq \mathbf{p}_i$ for $i = 0, 1$ and $A_j = A^0_j \times A^1_j$ for $j = 1, \ldots, n$. This means $A^i_j \hookrightarrow A_j$, so because G has the closure property:

$$B \hookrightarrow C, C \in \mathsf{G} \Rightarrow B \in \mathsf{G}_\varepsilon$$

the A^i_j's actually belongs to G_ε. Hence we can use the hypothesis of induction to see $E_i \stackrel{A^i_1}{\Longrightarrow} \ldots \stackrel{A^i_n}{\Longrightarrow} \dagger$ for $i = 0, 1$. By proposition 7.2.3 then $E_0 \parallel E_1 \stackrel{A_1}{\Longrightarrow} \ldots \stackrel{A_n}{\Longrightarrow} \dagger \parallel \dagger$ and the result follows from $\dagger \parallel \dagger \succ\!\!\longrightarrow \dagger$.

Only if: We shall also prove this implication by induction on the structure of E.

$E = a$: Since $a \not\succ\!\!\longrightarrow$ and $a \stackrel{A}{\longrightarrow} F$ implies $A = a$ and $F = \dagger$ there is exactly one $A_i = a$ and the rest equal to $\emptyset$. Recalling $\emptyset (= \varepsilon)$ neutral to $\cdot$ we see from $a \preceq a$ and $\wp(a) = \{a\}$ that we are done.

$E = E_0 \,;\, E_1$: Because $E_0 \in BL$ we cannot have $E_0 \succ\!\!\longrightarrow^* \dagger$. For purely structural reasons we cannot for any E'_0 and E_1 have $E'_0 \,;\, E_1 = \dagger$ neither, so using lemma 7.4.5 on $E_0 \,;\, E_1 \stackrel{A_1}{\Longrightarrow} \ldots \stackrel{A_n}{\Longrightarrow} \dagger$ we deduce $n \geq 2$ and the existence of a $1 \leq j < n$ such that $E_0 \stackrel{A_1}{\Longrightarrow} \ldots \stackrel{A_j}{\Longrightarrow} \dagger$ and $E_1 \stackrel{A_{j+1}}{\Longrightarrow} \ldots \stackrel{A_n}{\Longrightarrow} \dagger$. By hypothesis then $A_1 \cdot \ldots \cdot A_j \preceq \mathbf{p}_0$

and $A_{j+1} \dots A_n \preceq \mathbf{p}_1$ where $\mathbf{p}_i \in \wp(E_i)$ for $i = 0, 1$. By $\preceq$-monotonicity of $\cdot$ then $A_1 \cdot \dots A_j \cdot A_{j+1} \cdot \dots \cdot A_n \preceq \mathbf{p}_0 \cdot A_{j+1} \cdot \dots \cdot A_n \preceq \mathbf{p}_0 \cdot \mathbf{p}_1 \in \wp(E_0) \cdot \wp(E_1)$.

$E = E_0 \oplus E_1$: Inspecting the definition of $\rightarrowtail$ and $\xrightarrow{A_1}$ one easely sees that $E_0 \oplus E_1 \stackrel{A_1}{\Longrightarrow} \dots \stackrel{A_n}{\Longrightarrow} \dagger$ implies $E_0 \oplus E_1 \rightarrowtail F \stackrel{A_1}{\Longrightarrow} \dots \stackrel{A_n}{\Longrightarrow} \dagger$ where $F = E_0$ or $F = E_1$. The result then follows from the hypothesis of induction and definition of $\wp$.

$E = E_0 \parallel E_1$: Then $E_0 \parallel E_1 \stackrel{A_1}{\Longrightarrow} \dots \stackrel{A_n}{\Longrightarrow} \dagger$. Chosing $E = \dagger$ in lemma 7.4.6 we see that there are $A_1^0, \dots, A_n^0, A_1^1, \dots, A_n^1 \in \mathsf{G}_\varepsilon$ such that $A_j = A_j^0 \times A_j^1$ for $j = 1, \dots, n$ and $E_i \stackrel{A_1^i}{\Longrightarrow} \dots \stackrel{A_n^i}{\Longrightarrow} \dagger$ for $i = 0, 1$. Using the hypothesis of induction together with (6.3):

$$(\mathbf{p} \times \mathbf{q}) \cdot (\mathbf{p}' \times \mathbf{q}') \preceq (\mathbf{p} \cdot \mathbf{p}') \times (\mathbf{q} \cdot \mathbf{q}')$$

the desired result is then obtained similarly as in the case $E = E_0 \,;\, E_1$.

□

Notice that we only used the closure property of G in the *if* part of the proof.

Lemma 7.4.5 Suppose $n \geq 1$ and $E_0 \,;\, E_1 \stackrel{A_1}{\Longrightarrow} \dots \stackrel{A_n}{\Longrightarrow} E$ for $A_1, \dots, A_n \in \mathsf{G}_\varepsilon$ and $E_0 \,;\, E_1 \in CL$. Then either

a) $E_0 \rightarrowtail^* \dagger$, $E_1 \stackrel{A_1}{\Longrightarrow} \dots \stackrel{A_n}{\Longrightarrow} E$ or

b) $E_0 \stackrel{A_1}{\Longrightarrow} \dots \stackrel{A_j}{\Longrightarrow} \dagger$, $E_1 \stackrel{A_{j+1}}{\Longrightarrow} \dots \stackrel{A_n}{\Longrightarrow} E$ for a $1 \leq j < n$ or

c) $E_0 \stackrel{A_1}{\Longrightarrow} \dots \stackrel{A_n}{\Longrightarrow} \dagger$, $E_1 \rightarrowtail^* E$ or

d) $E_0 \stackrel{A_1}{\Longrightarrow} \dots \stackrel{A_n}{\Longrightarrow} E_0'$, $E_0' \,;\, E_1 = E$ for some $E_0' \in CL$

Proof At first we prove by natural induction for arbitrary m and $F, E_0 \,;\, E_1 \in CL$:

$$\begin{array}{rl} & E_0 \,;\, E_1 \rightarrowtail^m F \\ \Downarrow & \\ i) & E_0 \rightarrowtail^* \dagger, E_1 \rightarrowtail^* F \text{ or} \\ ii) & E_0 \rightarrowtail^* F_0',\ F_0' \,;\, E_1 = F \end{array} \tag{7.4}$$

$m = 0$: Here $E_0 \,;\, E_1 = F$ and we can choose $F_0' = E_0$.

$m > 0$: Then for some $H \in CL$ we have $E_0 \,;\, E_1 \rightarrowtail H \rightarrowtail^{m-1} F$. According to the definition of $\rightarrowtail$ there are two cases:
$E_0 = \dagger$ and $H = E_1$: I.e., $E_1 \rightarrowtail^{m-1} F$ and $i)$ holds.
$E_0 \rightarrowtail H_0$ and $H_0;E_1 = H$: From the hypothesis of induction used on $H_0;E_1 \rightarrowtail^{m-1} F$ we get either $(H_0 \rightarrowtail^* \dagger, E_1 \rightarrowtail^* F)$ or $(H_0 \rightarrowtail^* F_0', F_0';E_1 = F)$. In the former case $i)$ is established because $E_0 \rightarrowtail H_0 \rightarrowtail^* \dagger$ and in the latter case we get $ii)$.

Using (7.4) we can now prove the lemma by induction on n.

$n = 1$: If $A_1 = \emptyset$ we have $E_0 \,;\, E_1 \rightarrowtail^* E$ and we can use (7.4) to see that c) or d) holds. So assume $A_1 \neq \emptyset$ and we have $E_0 \,;\, E_1 \rightarrowtail^* F \xrightarrow{A_1} F' \rightarrowtail^* E$ for some $F, F' \in CL$. We consider two case according to (7.4):

i) Here we have $E_0 \rightarrowtail^* \dagger, E_1 \rightarrowtail^* F \xrightarrow{A_1} F' \rightarrowtail^* E$ and a) holds.

ii) $E_0 \rightarrowtail^* F_0', F_0' \,;\, E_1 \xrightarrow{A_1} F' \rightarrowtail^* E$. Since $A_1 \neq \emptyset$ we must have $F' = F_0'' \,;\, E_1$ where $F_0' \xrightarrow{A_1} F_0''$. Using (7.4) on $F_0'' \,;\, E_1 \rightarrowtail^* E$ we see $F_0'' \rightarrowtail^* \dagger, E_1 \rightarrowtail^* E$ *or* $F_0'' \rightarrowtail^* E_0', E_0' \,;\, E_1 = E$. In the former case we have $E_0 \rightarrowtail^* F_0' \xrightarrow{A_1} F_0'' \rightarrowtail^* \dagger, E_1 \rightarrowtail^* E$ and c) holds. In the latter case $E_0 \rightarrowtail^* F_0' \xrightarrow{A_1} F_0'' \rightarrowtail^* E_0', E_0' \,;\, E_1 = E$ so here d) holds.

$n > 1$: Then $E_0 \,;\, E_1 \stackrel{A_1}{\Longrightarrow} F \stackrel{A_2}{\Longrightarrow} \ldots \stackrel{A_n}{\Longrightarrow} E$. From the case $n = 1$ we know that for $E_0 \,;\, E_1 \stackrel{A_1}{\Longrightarrow} F$ there are the following three main possibilities:

$E_0 \rightarrowtail^* \dagger, E_1 \stackrel{A_1}{\Longrightarrow} F$: Then also $E_1 \stackrel{A_1}{\Longrightarrow} \ldots \stackrel{A_n}{\Longrightarrow} E$ and a) holds.

$E_0 \stackrel{A_1}{\Longrightarrow} \dagger, E_1 \rightarrowtail^* F$: Here we get $E_1 \stackrel{A_2}{\Longrightarrow} \ldots \stackrel{A_n}{\Longrightarrow} E$ and b) is established with $j = 1$.

$E_0 \stackrel{A_1}{\Longrightarrow} F_0', F_0' \,;\, E_1 = F$: The hypothesis of induction used on $F_0' ; E_1 \stackrel{A_2}{\Longrightarrow} \ldots \stackrel{A_n}{\Longrightarrow} E$ yields:

a') $F_0' \rightarrowtail^* \dagger,\ E_1 \stackrel{A_2}{\Longrightarrow} \ldots \stackrel{A_n}{\Longrightarrow} E$ or
b') $F_0' \stackrel{A_2}{\Longrightarrow} \ldots \stackrel{A_j}{\Longrightarrow} \dagger,\ E_1 \stackrel{A_{j+1}}{\Longrightarrow} \ldots \stackrel{A_n}{\Longrightarrow} E$ for a $2 \leq j < n$ or
c') $F_0' \stackrel{A_2}{\Longrightarrow} \ldots \stackrel{A_n}{\Longrightarrow} \dagger,\ E_1 \rightarrowtail^* E$ or
d') $F_0' \stackrel{A_2}{\Longrightarrow} \ldots \stackrel{A_n}{\Longrightarrow} E_0',\ E_0' \,;\, E_1 = E$

Clearly we have $E_0 \stackrel{A_1}{\Longrightarrow} \dagger$ in the case a') thereby getting b) for $j = 1$. In the remaining cases b'), c') and d') we directly get b), c) and d) respectively.

□

Lemma 7.4.6 Suppose $n \geq 1$ and $E_0 \parallel E_1 \stackrel{A_1}{\Longrightarrow} \ldots \stackrel{A_n}{\Longrightarrow} E$ for $A_1, \ldots, A_n \in \mathsf{G}_\varepsilon$ and $E_0 \parallel E_1 \in CL$. Then there are $E_0', E_0' \in CL$ and $A_1^0, \ldots, A_n^0, A_1^1, \ldots, A_n' \in \mathsf{G}_\varepsilon$ such that

$$A_j = A_j^0 \times A_j^1 \text{ for } j = 1, \ldots, n \text{ and}$$
$$E_i \stackrel{A_i^i}{\Longrightarrow} \ldots \stackrel{A_n^i}{\Longrightarrow} E_i' \text{ for } i = 0, 1 \text{ and}$$
$$E_0' \parallel E_1' = E \text{ or } E_0', E_1' = \{\dagger, E\}$$

Proof At first we by natural induction prove for arbitrary m and $E_0, E_1 \in CL$ that $E_0 \parallel E_1 \rightarrowtail^m E$ implies the existence of some $E_0', E_1' \in CL$ such that:

$$\begin{aligned} &E_0 \rightarrowtail^* E_0', E_1 \rightarrowtail^* E_1' \text{ and} \\ &E_0' \parallel E_1' = E \text{ or } \{E_0', E_1'\} = \{\dagger, E\} \end{aligned} \tag{7.5}$$

$m = 0$: Then $E = E_0 \parallel E_1$ and we can choose $E_i' = E_i$.

$m > 0$: This means $E_0 \parallel E_1 \rightarrowtail F \rightarrowtail^{m-1} E$. According to the definition of $\rightarrowtail$ there are four cases:

$E_0 = \dagger$ and $E_1 = F$: Choose $E_0' = \dagger, E_1' = E$ and we are done.

$E_1 = \dagger$ and $E_0 = F$: Symmetric.

$E_0 \rightarrowtail F_0'$ and $F_0' \parallel E_1 = F$: Use the hypothesis of induction on $F_0' \parallel E_1 \rightarrowtail^{m-1} E$ to find E_0', E_1' such that $F_0' \rightarrowtail^* E_0', E_1 \rightarrowtail^* E_1'$ and $(E_0' \parallel E_1' = E$ *or* $\{E_0', E_1'\} = \{\dagger, E\})$. The result then follows because $E_0 \rightarrowtail^* F_0' \rightarrowtail^* E_0'$ or equally $E_0 \rightarrowtail^* E_0'$.

$E_1 \rightarrowtail F_1'$ and $E_0 \parallel F_1' = F$: Symmetric to the last case.

Next we prove the lemma by induction on n:

$n = 1$: We have $E_0 \| E_1 \rightarrowtail^* F \xrightarrow{A_1} H \rightarrowtail^* E$. We can now use (7.5) on $E_0 \| E_1 \rightarrowtail F$ to find $F_0', F_1' \in CL$ such that $E_i \rightarrowtail^* F_i'$ for $i = 0, 1$ and $(F_0' \parallel F_1' = F$ *or* $\{F_0', F_1'\} = \{\dagger, F\})$. According to this there are two subcases:

$\{F_0', F_1'\} = \{\dagger, F\}$: Suppose w.l.o.g. $F_0' = \dagger$ and $F_1' = F$. Choosing $E_0' = \dagger, E_1' = E$ and $A_1^0 = \emptyset, A_1^1 = A_1$ we are done because $E_0 \rightarrowtail^* E_0'$ implies $E_0 \overset{\emptyset}{\Rightarrow} E_0'$ or equally $E_0' \overset{A_1^0}{\Longrightarrow} E_0'$ and because $E_1 \rightarrowtail^* F \xrightarrow{A_1} H \rightarrowtail^* E$ implies $E_1 \overset{A_1^1}{\Longrightarrow} E_1'$.

$F_0' \parallel F_1' = F$: In this situation we have $F_0' \parallel F_1' \xrightarrow{A_1} H \rightarrowtail^* E$. Looking at the definition of $\xrightarrow{A_1}$ we see that $F_0' \| F_1' \xrightarrow{A_1} H$ implies $H = H_0' \| H_1'$ where $F_i' \xrightarrow{A_1^i} H_i, A_1^i \in \mathsf{G}_\varepsilon$ for $i = 0, 1$ and $A_1 = A_1^0 \times A_1^1$ (recall the convention $F_i' = H_i$ if $A_1^i = \emptyset$). Using (7.5) on $H_0 \parallel H_1 \rightarrowtail^* E$ we now find the desired E_0' and E_1' because $E_i \rightarrowtail^* F_i' \xrightarrow{A_1^i} H_i \rightarrowtail^* E_i'$.

$n > 1$: Then there must be a $F \in CL$ and $1 \leq j < n$ such that $E_0 \parallel E_1 \overset{A_1}{\Longrightarrow} \ldots \overset{A_j}{\Longrightarrow} F \overset{A_{j+1}}{\Longrightarrow} \ldots \overset{A_n}{\Longrightarrow} E$. By induction there are F_0', F_1' and $A_1^i, \ldots, A_j^i$ for $i = 0, 1$ such that $E_i \overset{A_1^i}{\Longrightarrow} \ldots \overset{A_j^i}{\Longrightarrow} F_i'$ and $(F_0' \parallel F_1' = F$ *or* $\{F_0', F_1'\} = \{\dagger, F\})$. Two cases:

$\{F_0', F_1'\} = \{\dagger, F\}$: Suppose w.l.o.g. $F_0' = \dagger$ and $F_1' = F$. Choosing $E_0' = \dagger, E_1' = E$ and $A_k^0 = \emptyset, A_k^1 = A_k$ for $k = j+1, \ldots, n$ we have $F_0' = \dagger \overset{\emptyset}{\Rightarrow} \ldots \overset{\emptyset}{\Rightarrow} \dagger = E_0'$ or equally $F_0' \overset{A_{j+1}^0}{\Longrightarrow} \ldots \overset{A_n^0}{\Longrightarrow} E_0'$ from which we get the result.

$F_0' \parallel F_1' = F$: Then we can apply the hypothesis of induction once more and find the desired E_0', E_1' and remaining A_k^i for $i = 0, 1$ and $k = j+1, \ldots, n$.

□

Lemma 7.4.7 Suppose $A_1, \ldots, A_n \in \mathsf{M}_\varepsilon$ and $\mathbf{p}_0, \mathbf{p}_1 \neq \varepsilon$. Then $A_1 \cdot \ldots \cdot A_n \preceq \mathbf{p}_0 \cdot \mathbf{p}_1$ implies $n \geq 2$ and there is a $1 \leq j < n$ such that

$$A_1 \cdot \ldots \cdot A_j \preceq \mathbf{p}_0 \text{ and } A_{j+1} \cdot \ldots \cdot A_n \preceq \mathbf{p}_1$$

Proof Let $\mathbf{q} = A_1 \cdot \ldots \cdot A_n$. By proposition 6.4.2 there are $\mathbf{q}_0$ and $\mathbf{q}_1$ such that $\mathbf{q} = \mathbf{q}_0 \cdot \mathbf{q}_1$ where $\mathbf{q}_i \preceq \mathbf{p}_i$ for $i = 0, 1$. Since $\mathbf{p}_i \neq \varepsilon$ we also have $\mathbf{q}_i \neq \varepsilon$. The lemma then follows from

$$\begin{array}{c} A_1 \cdot \ldots \cdot A_n = \mathbf{p}_0 \cdot \mathbf{p}_1, \mathbf{p}_0, \mathbf{p}_1 \neq \varepsilon \\ \Downarrow \\ n \geq 2, \exists 1 \leq j < n.\ A_1 \cdot \ldots \cdot A_j = \mathbf{p}_0, A_{j+1} \cdot \ldots \cdot A_n = \mathbf{p}_1 \end{array} \tag{7.6}$$

which we prove by induction on n:

$n = 1$: The situation is $A_1 = \mathbf{p}_0 \cdot \mathbf{p}_1$. The premise cannot hold since $\mathbf{p}_0, \mathbf{p}_1 \neq \varepsilon$ implies that there are at least two ordered elements of $\mathbf{p}_0 \cdot \mathbf{p}_1$, but this contradicts $A_1 = \mathbf{p}_0 \cdot \mathbf{p}_1$ because the elements of A_1 are unoreredered ($A_1 \in \mathsf{M}_\varepsilon$).

$n > 1$: Equally $n \geq 2$. Since $\mathbf{p}_0 \neq \varepsilon$ we can apply proposition 7.4.8 below to find a $\mathbf{p}_0'$ such that $A_1 \cdot \mathbf{p}_0' = \mathbf{p}_0$ and $A_2 \cdot \ldots \cdot A_n = \mathbf{p}_0' \cdot \mathbf{p}_1$.
If $\mathbf{p}_0' = \varepsilon$ we have $\mathbf{p}_0 = A_1$ and $A_2 \cdot \ldots \cdot A_n = \varepsilon \cdot \mathbf{p}_1 = \mathbf{p}_1$ wherefore we can choose $j = 1$. Otherwise if $\mathbf{p}_0' \neq \varepsilon$ we can use the induction to find $2 \leq j < n$ such that $A_2 \cdot \ldots \cdot A_j = \mathbf{p}_0'$ and $A_{j+1} \cdot \ldots \cdot A_n = \mathbf{p}_1$. Since $\mathbf{p}_0 = A_1 \cdot \mathbf{p}_0' = A_1 \cdot A_2 \cdot \ldots \cdot A_j$ we are done. □

Proposition 7.4.8 For $A \in \mathsf{M}_\varepsilon$ and a pomset $\mathbf{p}_0 \neq \varepsilon$ we have

$$\begin{array}{l} A \cdot \mathbf{q} = \mathbf{p}_0 \cdot \mathbf{p}_1 \\ \Downarrow \\ \exists \mathbf{p}_0'.\, A \cdot \mathbf{p}_0' = \mathbf{p}_0, \mathbf{q} = \mathbf{p}_0' \cdot \mathbf{p}_1 \end{array}$$

Proposition 7.4.9 Suppose $\mathbf{p} \cdot \mathbf{q} \preceq \mathbf{r}_0 \times \mathbf{r}_1$. Then there exists $\mathbf{p}_0, \mathbf{p}_1, \mathbf{q}_0, \mathbf{q}_1$ such that $\mathbf{p}_i \cdot \mathbf{q}_i \preceq \mathbf{r}_i$ for $i = 0, 1$ and $\mathbf{p} \preceq \mathbf{p}_0 \times \mathbf{p}_1$ and $\mathbf{q} \preceq \mathbf{q}_0 \times \mathbf{q}_0$.

Proof We prove it for lpos and the proposition follows immediately.

$p \cdot q \preceq r_0 \times r_1$ means that there exists a bijection $f : X_{r_0 \times r_1} \longrightarrow X_{p \cdot q}$ which also is a morphism of lpos.

By definition of $\cdot$ and $\times$ we have $X_{p \cdot q} = \{0\} \times X_p \cup \{1\} \times X_q$ and $X_{r_0 \times r_1} = \{0\} \times X_{r_0} \cup \{1\} \times X_{r_1}$. So define p_i as p restricted to $\{x \in X_p \mid \langle 0, x\rangle \in f(\{i\} \times X_{r_i})\}$ for $i = 0, 1$ and similar for q_0 and q_1.

$p_0 \cdot q_0 \preceq r_0$: Consider $g : X_{r_0} \longrightarrow X_{p_0 \cdot q_0}$ given by $g(x) = f(\langle 0, x\rangle)$. It is easy to see that g in- and surjective.

g order preserving: From f being order preserving and $x \leq_{r_0} y \Rightarrow \langle 0, x\rangle \leq_{r_0 \times r_1} \langle y, 0\rangle$ we see $f(\langle 0, x\rangle) \leq_{p \cdot q} f(\langle 0, y\rangle)$. $f(\langle 0, x\rangle)$ is of the form $\langle i, x'\rangle$ and $f(\langle 0, y\rangle)$ is of the form $\langle j, y'\rangle$, so $\langle i, x'\rangle \leq_{p \cdot q} \langle j, y'\rangle$. According to the definition of $\leq_{p \cdot q}$ then

$$(i = 0 = j, x' \leq_p y') \textit{ or } (i = 1 = j, x' \leq_q y') \textit{ or } (i = 0, j = 1)$$

In the case $x' \leq_p y'$ we have $x' \in X_p$, so we must have $x' \in X_{p_0}$. Also $y' \in X_{p_0}$. Similar considerations in the case $x' \leq_q y'$ leads us to:

$$(i = 0 = j, x' \leq_{p_0} y') \textit{ or } (i = 1 = j, x' \leq_{q_0} y') \textit{ or } (i = 0, j = 1)$$

But then $g(x) = f(\langle 0, x\rangle) = \langle i, x'\rangle \leq_{p_0 \cdot q_0} \langle j, y'\rangle = f(\langle 0, y\rangle) = g(y)$.
g is directly seen to be label preserving: $\ell_{r_0}(x) = \ell_{r_0 \times r_1}(\langle 0, x\rangle) = \ell_{p \cdot q}(f(\langle 0, x\rangle)) = \ell_{p_0 \cdot q_0}(g(x))$.

$p_1 \cdot q_1 \preceq r_1$: similar as above.

$p \preceq p_0 \times p_1$: Let $g : X_{p_0 \times p_1} \longrightarrow X_p$ be given by $g(\langle i, x\rangle) = x$. This time it is easy to see that g is a morphism of lpos.

g injective: Assume $\langle i, x\rangle \neq \langle j, y\rangle$. We are done if $x = y$ and $i \neq j$ is impossible. So suppose on the contrary $x = y, i \neq j$. Now $\langle i, x\rangle \in X_{p_0 \times p_1} \Rightarrow x \in X_{p_i} \Rightarrow \langle 0, x\rangle \in$

$f(\{i\}\times X_{r_i}) \Rightarrow \exists\langle i,x'\rangle \in X_{r_0\times r_1}. f(\langle i,x'\rangle) = \langle 0,x\rangle$ and similar $\langle j,x\rangle \in X_{p_0\times p_1} \Rightarrow \exists\langle j,x''\rangle \in X_{r_0\times r_1}.\, f(\langle j,x''\rangle) = \langle 0,x\rangle$. But since $i \neq j$ and thereby $\langle i,x'\rangle \neq \langle j,x''\rangle$ this is a contradiction to f being injective.

g surjective: $x \in X_p \Rightarrow \langle 0,x\rangle \in X_{p\cdot q} \Rightarrow$ (since f is surjective) $(\exists\langle i,y\rangle \in X_{r_0\times r_1}.\, f(\langle i,y\rangle) = \langle 0,x\rangle) \Rightarrow (\exists i \in \{0,1\}.\, x \in X_{p_i}) \Rightarrow \exists i \in \{0,1\}.\, \langle i,x\rangle \in X_{p_0\times p_1}$. From $g(\langle i,x\rangle) = x$ the result then follows.

$q \preceq q_0 \times q_1$: Similar as the last case. □

Proposition 7.4.10 If $n \geq 1$ and $A_1,\ldots,A_n \in \mathsf{M}_\varepsilon$ and $A_1 \cdot \ldots \cdot A_n \preceq \mathbf{p}_0 \times \mathbf{p}_1$ then there exists multisets $A^0_1,\ldots,A^0_n$ and $A^1_1,\ldots,A^1_n$ such that $A_j = A^0_j \times A^1_j$ for $j = 1,\ldots,n$, and $A^i_1 \cdot \ldots \cdot A^i_n \preceq \mathbf{p}_i$ for $i = 0,1$.

Proof By induction on n.

$n = 1$: $A_1 \preceq \mathbf{p}_0 \times \mathbf{p}_1$ clearly implies $\mathbf{p}_0$ and $\mathbf{p}_1$ are multisets and $A_1 = \mathbf{p}_0 \times \mathbf{p}_1$. Chose $A^0_1 = \mathbf{p}_0$ and $A^1_1 = \mathbf{p}_1$.

$n > 1$: $A_1 \cdot (A_2 \cdot \ldots \cdot A_n) \preceq \mathbf{p}_0 \times \mathbf{p}_1$ implies by the previous proposition the existence of pomsets $\mathbf{q}_0, \mathbf{q}_1, \mathbf{r}_0, \mathbf{r}_1$ such that $\mathbf{q}_0 \cdot \mathbf{r}_0 \preceq \mathbf{p}_0, \mathbf{q}_1 \cdot \mathbf{r}_1 \preceq \mathbf{p}_1, A_1 \preceq \mathbf{q}_0 \times \mathbf{q}_1$ and $A_2 \cdot \ldots \cdot A_n \preceq \mathbf{r}_0 \times \mathbf{r}_1$. The last implies by hypothesis of induction that there are multisets $A^0_2,\ldots,A^0_n$ and $A^1_2,\ldots,A^1_n$ with $A^0_2 \cdot \ldots \cdot A^0_n \preceq \mathbf{r}_0$, $A^1_2 \cdot \ldots \cdot A^1_n \preceq \mathbf{r}_1$ and $A^0_i \times A^1_i = A_i$ for $i = 2,\ldots,n$. From the case $n = 1$ we see that there exists A^0_1 and A^1_1 with $A^0_1 \preceq \mathbf{q}_0, A^1_1 \preceq \mathbf{q}_1$ and $A_1 = A^0_1 \times A^1_1$. By monotonicity and transitivity of $\preceq$ we now get

$$A^0_1 \cdot (A^0_2 \cdot \ldots \cdot A^0_n) \preceq \mathbf{q}_0 \cdot (A^0_2 \cdot \ldots \cdot A^0_n) \preceq \mathbf{q}_0 \cdot \mathbf{r}_0 \preceq \mathbf{p}_0$$

and similar for the A^1_i's.

□

7.5 Summary

In this section we show how the different G-semantics are related and some concrete examples of G-semantics are given.

At first notice that if G and G' are sets of direct observations then so are $\mathsf{G}\cup\mathsf{G}'$ and $\mathsf{G}\cap\mathsf{G}'$. Hence the direct observation sets forms a lattice under the inclusion relation; the meet/ glb being intersection and join/ lub being union.

This carry over to models as follows:

Proposition 7.5.1 $\mathsf{G} \subseteq \mathsf{G}'$ *iff* $=_{\mathsf{G}'} \subseteq =_{\mathsf{G}}$

Proof
only if: Suppose $\mathsf{G} \subseteq \mathsf{G}'$. Then $\mathsf{G} = \mathsf{G}' \cap \mathsf{G}$ and $\mathbf{P}_{\mathsf{G}} = \mathbf{P}_{\mathsf{G}'} \cap \mathbf{P}_{\mathsf{G}}$. By definition of $[\![\_]\!]_{\mathsf{G}}$ and $[\![\_]\!]_{\mathsf{G}'}$ therefore $[\![\_]\!]_{\mathsf{G}} = [\![\_]\!]_{\mathsf{G}'} \cap \mathbf{P}_{\mathsf{G}}$. As a consequence $\trianglelefteq_{\mathsf{G}'} \subseteq \trianglelefteq_{\mathsf{G}}$ and $=_{\mathsf{G}'} \subseteq =_{\mathsf{G}}$.

if: We start out by some general observations for an arbitrary set of direct test G. If A is any multiset let E_A be a BL-expression obtained by parallel composing the atomic actions, i.e., $\wp(E_A) = \{A\}$. From section 7.3 we know $\mathbf{p} \neq \varepsilon$, $\mathbf{p} \in \mathbf{P}_{\mathsf{G}}$ *iff* $\exists n \geq 1 \exists A_1, \ldots, A_n \in \mathsf{G}.\ A_1 \cdot \ldots \cdot A_n = \mathbf{p}$, so for all $\mathbf{p} \in \mathbf{P}_{\mathsf{G}} \setminus \{\varepsilon\}$ there are $E_{A_1}, \ldots, E_{A_n}$ such that $\{\mathbf{p}\} = \wp(E_{A_1} ; \ldots ; E_{A_n})$. Composing such expressions with $\oplus$ one can then for any finite and nonempty subset, P, of $\mathbf{P}_{\mathsf{G}} \setminus \{\varepsilon\}$ find an expression E_P with $\wp(E_P) = P$.

The proof of the implication is now by contradiction. Suppose $=_{\mathsf{G}'} \subseteq =_{\mathsf{G}}$ but $\mathsf{G} \not\subseteq \mathsf{G}'$. Then there is an $A \in \mathsf{G}$ with $A \notin \mathsf{G}'$. Of course then $A \in \mathbf{P}_{\mathsf{G}}$ and if $P = \delta_{\mathsf{G}'}(A)$ we have $[\![E_P]\!]_{\mathsf{G}'} = \delta_{\mathsf{G}'}(P) = \delta_{\mathsf{G}'}(A) = [\![E_A]\!]_{\mathsf{G}'}$—i.e., $E_P =_{\mathsf{G}'} E_A$. By assumption then also $E_P =_{\mathsf{G}} E_A$. But clearly $A \in [\![E_A]\!]_{\mathsf{G}}$, so we must have $A \in [\![E_P]\!]_{\mathsf{G}}$ too. $A \in [\![E_P]\!]_{\mathsf{G}}$ means $A \in \delta_{\mathsf{G}}(\delta_{\mathsf{G}'}(A))$ wherefore there must be a $\mathbf{p} \in \mathbf{P}_{\mathsf{G}'}$ such that $A \preceq \mathbf{p} \preceq A$. Hence $A = \mathbf{p} \in \mathbf{P}_{\mathsf{G}'}$—a contradiction to $A \notin \mathbf{P}_{\mathsf{G}'}$. □

From this proposition it immediately follows that $\mathsf{G} \subset \mathsf{G}'$ implies $[\![\_]\!]_{\mathsf{G}}$ is strictly more abstract than $[\![\_]\!]_{\mathsf{G}'}$ (i.e., all expressions identified by $[\![\_]\!]_{\mathsf{G}'}$ are identified by $[\![\_]\!]_{\mathsf{G}}$ and there is some expressions identified by $[\![\_]\!]_{\mathsf{G}}$ but not by $[\![\_]\!]_{\mathsf{G}'}$).

In fact the lattice of our G-models has a least and a greatest model (in the sense of their ability to distinguish expressions). The least model is of course the one generated from $\mathsf{G} = \Delta$ and the largest the one generated from $\mathsf{G} = \mathsf{M}$. It is not hard to see that $\mathbf{P}_{\Delta}$ agrees with W—the set of pomsets which are words (see page 133). As a consequence hereof we shall in the following (chapter) subscript with w rather than Δ when concerning the minimal model/ the operational semantics obtained by the least set of direct tests.

The variation of the operational semantics arising from the different sets of direct test manifest itself in the inference rule for a composed step: $\xrightarrow{A_0 \times A_1}_{\mathsf{G}}$. For W the inference rule totally vanish and for M it becomes

$$\frac{E_0 \xrightarrow{A_0}_{\mathsf{M}} E_0',\, E_1 \xrightarrow{A_1}_{\mathsf{M}} E_1'}{E_0 \parallel E_1 \xrightarrow{A_0 \| A_1}_{\mathsf{M}} E_0' \parallel E_1'}$$

I.e., no restrictions.

As examples of what models we might find in between the least and largest models we end this section by giving two almost contrasting models.

Starke [Sta81] has introduced one natural candidate for semiwords, i.e., to some extend half a word, half a (unordered) pomset. He defines a semiword to be a pomset, $\mathbf{p}$, where all *equally* labelled elements are ordered: $P_{\mathsf{sw}}(\mathbf{p})$ *iff* for all x, y in X_p we have:

$$\text{if} \quad \begin{matrix} x \\ co_p \\ y \end{matrix} \quad \text{then} \quad \begin{matrix} \ell_p(x) \\ \neq \\ \ell_p(y) \end{matrix}$$

One might just as well take the opposite standpoint and define a semiwords to be a pomset, $\mathbf{p}$, where all *unequally* labelled elements are ordered: $P_{\overline{\mathsf{sw}}}(\mathbf{p})$ *iff* for all x, y in X_p we have:

$$\text{if} \quad \begin{array}{c} x \\ co_p \\ y \end{array} \quad \text{then} \quad \begin{array}{c} \ell_p(x) \\ = \\ \ell_p(y) \end{array}$$

Notice that both properties are hereditary and dot synthesizable. However the candidate of Starke enjoys a number of nice properties: If $P_{\mathsf{sw}}(\mathbf{p})$ then there is a canonic representative, $\hat{p} \in LPO$, of $\mathbf{p}$ (in the sense that $\mathbf{p} = [\hat{p}]$ and if $\mathbf{p} = \mathbf{q}$ then $\hat{p} = \hat{q}$) and furthermore the partial order, $\preceq$, on such semiwords may be characterized by $\mathbf{p} \preceq \mathbf{q}$ *iff* $\hat{p} \preceq \hat{q}$ *iff* $\delta_{\mathsf{w}}(\mathbf{p}) \subseteq \delta_{\mathsf{w}}(\mathbf{q})$. We shall later in the next chapter see some consequences of a pomset having the P_{sw}-property.

On second thoughts one soon realizes that

$$\mathbf{P}_{\mathsf{sw},and} = \mathbf{P}_{\mathsf{S}} \text{ and } \mathbf{P}_{\overline{\mathsf{sw}},and} = \mathbf{P}_{\mathsf{N}}$$

So the most general linearizations of the semiwords of Starke corresponds to sequences of sets (of Δ) whereas the most general linearisations of the other type of semiwords corresponds to sequences of multisingletons (of Δ).

The inference rules for the composed step for these two sets of direct tests are particular simple:

$$\frac{E_0 \xrightarrow{A_0}_{\mathsf{S}} E_0', E_1 \xrightarrow{A_1}_{\mathsf{S}} E_1', A_0 \cup A_1 \subseteq \Delta}{E_0 \parallel E_1 \xrightarrow{A_0 \cup A_1}_{\mathsf{S}} E_0' \parallel E_1'}$$

and

$$\frac{E_0 \xrightarrow{a^n}_{\mathsf{N}} E_0', E_1 \xrightarrow{a^m}_{\mathsf{N}} E_1'}{E_0 \parallel E_1 \xrightarrow{a^{n+m}}_{\mathsf{N}} E_0' \parallel E_1'}$$

7.6 An Adequate Logic

In this section we shall for each G-semantics of BL give an adequate logic $\mathcal{L}_{\mathsf{G}}$. A logic for our process language will be a set of (logic) formulae together with a satisfaction relation which for each process and formula tells whether the process satisfies the formula. In the sense of Hennessy and Milner [HM80] such a logic is adequate for a G-semantics *iff* processes are identified by the G-semantics (by the equivalence, $\lessapprox\!\!\gtrapprox_{\mathsf{G}}$, of $\lessapprox_{\mathsf{G}}$) exactly when they satisfy the same set of formulae in the logic ($\mathcal{L}_{\mathsf{G}}$).

The branching aspect is on purpose left out of account and brought about partly by having only a combinator for internal nondeterminism and partly by constructing the operational preorders on the basis of sequences of direct tests. Pnueli [Pnu85] regards the latter as taking the linear view and shows how a *linear time logic* can be appropriate in this situation. In agreement with this we will define a process to satisfy a formula if all the “syntactic controlled” behaviours of the process satisfies the formula. We shall in a moment make these notions precise.

The different logics will share the same set of logic formulae, $\mathcal{L}_g$, but have individual satisfaction relation $\models_{\mathsf{G}}$—óne for each logic $\mathcal{L}_{\mathsf{G}}$. Mainly for proof technical reasons we shall define $\mathcal{L}_g$ as a subset of a larger formula language, $\mathcal{L}$, and base $\models_{\mathsf{G}}$ on a larger satisfaction relation for $\mathcal{L}$.

The set of formulae, $\mathcal{L}$, is defined in the BNF-like way:

$$f ::= \mathrm{tt} \mid \mathrm{ff} \mid \triangledown \mid \triangle \mid \langle\!\langle A \rangle\!\rangle f, A \in \mathsf{M} \mid \boxed{A} f, A \in \mathsf{M}$$

and $\mathcal{L}_g \subseteq \mathcal{L}$ is taken to be those formulae with no occurrence of the modality $\langle\!\langle A \rangle\!\rangle$.

Similarly as for Hennessy-Milner logic [HM85] we for each G-semantic define a satisfaction relation, $\models_{\mathsf{G}}$, between behaviours from DCL (see page 153)and formulae of $\mathcal{L}$ using the definitions of the operational G-semantics. The modalities $\langle\!\langle A \rangle\!\rangle$ and $\boxed{A}$ can be considered as generalizations of the corresponding Hennessy-Milner modalities (with $A = \{a\}$). tt (ff) has the standard interpretation that it always (never) is satisfied by a process. A process satisfies $\triangledown$ if it is terminated (i.e., no external computation step is possible) whereas $\triangle$ indicates that the process is alive. Formally:

Definition 7.6.1 $\models_{\mathsf{G}} \subseteq DCL \times \mathcal{L}$ is defined inductively:

$$\begin{array}{lll} E \models_{\mathsf{G}} \mathrm{tt} & \text{for all } E \in DCL \\ E \models_{\mathsf{G}} \triangledown & \textit{iff } \forall a \in \Delta.\ E \overset{a}{\not\Rightarrow}_{\mathsf{G}} \\ E \models_{\mathsf{G}} \triangle & \textit{iff } \exists a \in \Delta.\ E \overset{a}{\Rightarrow}_{\mathsf{G}} \\ E \models_{\mathsf{G}} \langle\!\langle A \rangle\!\rangle f & \textit{iff } \exists E'.\ E \overset{A}{\Rightarrow}_{\mathsf{G}} E' \textit{ and } E' \models_{\mathsf{G}} f \\ E \models_{\mathsf{G}} \boxed{A} f & \textit{iff } \forall E'.\ E \overset{A}{\Rightarrow}_{\mathsf{G}} E' \textit{ implies } E' \models_{\mathsf{G}} f \end{array}$$

where $E \overset{a}{\Rightarrow}_{\mathsf{G}}$ means $\exists E' \in CL.\ E \overset{a}{\Rightarrow}_{\mathsf{G}} E'$. □

Following the linear logic tradition we now for each logic, $\mathcal{L}_{\mathsf{G}}$, say that a process E ($\in BL$) satisfies a formula $f \in \mathcal{L}_g$,

$$E \models_{\mathsf{G}} f \textit{ iff } \forall E' \in \mathrm{Beh}(E).\ E' \models_{\mathsf{G}} f$$

The set of formulae from $\mathcal{L}_g$ which is satisfied in a logic $\mathcal{L}_{\mathsf{G}}$ by an $E \in BL$ will be denoted $\mathcal{L}_{\mathsf{G}}(E)$. I.e.,

$$\mathcal{L}_{\mathsf{G}}(E) = \{f \in \mathcal{L}_g \mid E \models_{\mathsf{G}} f\}$$

It will facilitate the proof of the the adequacy of the different $\mathcal{L}_{\mathsf{G}}$ logics to introduce some additional notions.

At first we give a syntactic map $\overline{\_} : \mathcal{L} \longrightarrow \mathcal{L}$ which yield the "dual" of a formula. For each $f \in \mathcal{L}$ define $\overline{f}$ by induction on the structure of f:

$$\begin{array}{ll} \overline{\mathrm{ff}} = \mathrm{tt} & \overline{\mathrm{tt}} = \mathrm{ff} \\ \overline{\triangle} = \triangledown & \overline{\triangledown} = \triangle \\ \overline{\boxed{A} f} = \langle\!\langle A \rangle\!\rangle \overline{f} & \overline{\langle\!\langle A \rangle\!\rangle f} = \boxed{A} \overline{f} \end{array}$$

Clearly $\overline{\overline{f}} = f$ and an easy induction on the structure of f shows that f and $\overline{f}$ are dual (for every satisfaction relation $\models_{\mathsf{G}}$) in the sense that

$$E \not\models_{\mathsf{G}} f \textit{ iff } E \models_{\mathsf{G}} \overline{f} \tag{7.7}$$

for all configuration behaviours $E \in DCL$. Now define

$$\overline{\mathcal{L}}_g = \{\overline{f} \in \mathcal{L} \mid f \in \mathcal{L}_g\}$$

and for every $E \in BL$

$$\overline{\mathcal{L}}_{\mathsf{G}}(E) = \{f \in \overline{\mathcal{L}}_g \mid \exists E' \in \mathrm{Beh}(E).\ E' \models_{\mathsf{G}} f\}$$

Notice that $\overline{\mathcal{L}}_g$ are the formulae of $\mathcal{L}$ whit no occurrence of the modality $\boxed{A}$.

The following lemma display the close relationship between $\overline{\mathcal{L}}_{\mathsf{G}}(\_)$ and $\mathcal{L}_{\mathsf{G}}(\_)$.

Lemma 7.6.2 For all $E_0, E_1 \in BL$ we have:

$$\overline{\mathcal{L}}_{\mathsf{G}}(E_0) \subseteq \overline{\mathcal{L}}_{\mathsf{G}}(E_1) \textit{ iff } \mathcal{L}_{\mathsf{G}}(E_0) \supseteq \mathcal{L}_{\mathsf{G}}(E_1)$$

Proof We start out by inferring for an arbitrary formula $f \in \mathcal{L}_g$ and process $E \in BL$:

$$\begin{array}{lll} f \notin \mathcal{L}_{\mathsf{G}}(E) & \textit{iff } E \not\models_{\mathsf{G}} f & \text{definition of } \mathcal{L}_{\mathsf{G}}(\_) \\ & \textit{iff } \exists E' \in \mathrm{Beh}(E).\ E' \not\models_{\mathsf{G}} f & \text{definition of } \models_{\mathsf{G}} \subseteq BL \times \mathcal{L}_g \\ & \textit{iff } \exists E' \in \mathrm{Beh}(E).\ E' \models_{\mathsf{G}} \overline{f} & \text{by (7.7)} \\ & \textit{iff } \overline{f} \in \overline{\mathcal{L}}_{\mathsf{G}}(E) & \text{definition of } \overline{\mathcal{L}}_{\mathsf{G}}(\_) \end{array}$$

Using the lemma below the proof is now merely logic rewriting:

$$\begin{array}{ll} & \overline{\mathcal{L}}_{\mathsf{G}}(E_0) \subseteq \overline{\mathcal{L}}_{\mathsf{G}}(E_1) \\ \Updownarrow & \\ & \forall f \in \overline{\mathcal{L}}_g.\ f \in \overline{\mathcal{L}}_{\mathsf{G}}(E_0) \Rightarrow f \in \overline{\mathcal{L}}_{\mathsf{G}}(E_1) \\ \Updownarrow & \qquad \text{definition of } \mathcal{L}_g \\ & \forall f \in \mathcal{L}_g.\ \overline{f} \in \overline{\mathcal{L}}_{\mathsf{G}}(E_0) \Rightarrow \overline{f} \in \overline{\mathcal{L}}_{\mathsf{G}}(E_1) \\ \Updownarrow & \qquad \text{from the above} \\ & \forall f \in \mathcal{L}_g.\ f \notin \overline{\mathcal{L}}_{\mathsf{G}}(E_0) \Rightarrow f \notin \overline{\mathcal{L}}_{\mathsf{G}}(E_1) \\ \Updownarrow & \\ & \mathcal{L}_{\mathsf{G}}(E_0) \supseteq \mathcal{L}_{\mathsf{G}}(E_1) \end{array}$$

□

The adequacy of the different $\mathcal{L}_{\mathsf{G}}$ logics can now be seen from:

Theorem 7.6.3 (Linear Logic Characterization)
For all $E_0, E_1 \in BL$:

$$E_0 \lesssim_{\mathsf{G}} E_1 \textit{ iff } \mathcal{L}_{\mathsf{G}}(E_0) \supseteq \mathcal{L}_{\mathsf{G}}(E_1)$$

Proof Immediate from the preceding lemma and the following. □

Lemma 7.6.4 Suppose $E_0, E_1 \in BL$. Then

$$E_0 \lesssim_{\mathsf{G}} E_1 \textit{ iff } \overline{\mathcal{L}}_{\mathsf{G}}(E_0) \subseteq \overline{\mathcal{L}}_{\mathsf{G}}(E_1)$$

Proof From the definition of $\models_{\mathsf{G}}$ it is almost trivial to prove for $F \in DCL$, $f \in \overline{\mathcal{L}}_g$ and $n \geq 1$ that

$$(7.8) \qquad F \models_{\mathsf{G}} \langle A_1 \rangle \cdots \langle A_n \rangle f \quad \textit{iff} \quad \exists F'.\, F \stackrel{A_1}{\Longrightarrow}_{\mathsf{G}} \ldots \stackrel{A_n}{\Longrightarrow}_{\mathsf{G}} F', F' \models_{\mathsf{G}} f$$

$$(7.9) \qquad F \models_{\mathsf{G}} \bigtriangledown \quad \textit{iff} \quad F \rightarrowtail^{*} \dagger$$

by induction on n in the case of (7.8) and induction on the structure of E in case of (7.9).

The *if* part of the lemma now follows the definition of $\lesssim_{\mathsf{G}}$ and by deducing for $E \in BL$ and $n \geq 1$

$$(7.10) \qquad E \stackrel{A_1}{\Longrightarrow}_{\mathsf{G}} \ldots \stackrel{A_n}{\Longrightarrow}_{\mathsf{G}} \dagger \textit{ iff } \langle A_1 \rangle \cdots \langle A_n \rangle \bigtriangledown \in \overline{\mathcal{L}}_{\mathsf{G}}(E)$$

as follows: $E \stackrel{A_1}{\Longrightarrow}_{\mathsf{G}} \ldots \stackrel{A_n}{\Longrightarrow}_{\mathsf{G}} \dagger$

iff $\exists F \in \mathrm{Beh}(E).\, F \stackrel{A_1}{\Longrightarrow}_{\mathsf{G}} \ldots \stackrel{A_n}{\Longrightarrow}_{\mathsf{G}} \dagger$	proposition 7.2.5
iff $\exists F \in \mathrm{Beh}(E) \exists F'.\, F \stackrel{A_1}{\Longrightarrow}_{\mathsf{G}} \ldots \stackrel{A_n}{\Longrightarrow}_{\mathsf{G}} F' \rightarrowtail^{*} \dagger$	definition of $\stackrel{A_n}{\Longrightarrow}$
iff $\exists F \in \mathrm{Beh}(E).\, F \models_{\mathsf{G}} \langle A_1 \rangle \cdots \langle A_n \rangle \bigtriangledown$	by (7.8) and (7.9)
iff $\langle A_1 \rangle \cdots \langle A_n \rangle \bigtriangledown \in \overline{\mathcal{L}}_{\mathsf{G}}(E)$	definition of $\overline{\mathcal{L}}_{\mathsf{G}}(\_)$

For the *only if* direction let an $f \in \overline{\mathcal{L}}_{\mathsf{G}}(E_0)$ be given. We consider each possible appearance of f in turn.

At first notice that $F \in \mathrm{Beh}(E)$ and $E \in BL$ implies $F \in DBL$, and that any $F \in DBL$ is capable of doing at least one action. So because $E_0 \in BL$ it follows that $f = \bigtriangledown$ is impossible and if $f = \bigtriangleup$ we also have $f \in \overline{\mathcal{L}}_{\mathsf{G}}(E_1)$ since $E_1 \in BL$.

ff is satisfied by no behaviour wherefore it should be clear that f cannot belong to $\overline{\mathcal{L}}_{\mathsf{G}}(E_1)$ because $E_1 \in BL$.

If $f = \mathrm{tt}$ then evidently $f \in \overline{\mathcal{L}}_{\mathsf{G}}(E_1)$ and if f is of the form $\langle A_1 \rangle \cdots \langle A_n \rangle \bigtriangledown$ the result follows from (7.10) and the definition of $\lesssim_{\mathsf{G}}$.

Now suppose f is of the form $\langle A_1 \rangle \cdots \langle A_i \rangle \mathrm{tt}$ for some $A_1, \ldots, A_i$ and $i \geq 1$. This means there is a $F \in \mathrm{Beh}(E_0)$ such that $F \models_{\mathsf{G}} \langle A_1 \rangle \cdots \langle A_i \rangle \mathrm{tt}$ and from (7.8) we conclude $F \stackrel{A_1}{\Longrightarrow}_{\mathsf{G}} \ldots \stackrel{A_i}{\Longrightarrow}_{\mathsf{G}} F'$ for some $F' \in DCL$. Using proposition 7.2.6 we get $E_0 \stackrel{A_1}{\Longrightarrow}_{\mathsf{G}} \ldots \stackrel{A_i}{\Longrightarrow}_{\mathsf{G}} E_0'$ for some E_0' (with $F' \in \mathrm{Beh}(E_0')$). Since we only have finite processes in BL and therefore also finite configurations there must be some $A_{i+1}, \ldots, A_n \in \mathsf{G}$ such that $E_0' \stackrel{A_{i+1}}{\Longrightarrow}_{\mathsf{G}} \ldots \stackrel{A_n}{\Longrightarrow}_{\mathsf{G}} \dagger$. The premise $E_0 \lesssim_{\mathsf{G}} E_1$ then gives $E_1 \stackrel{A_1}{\Longrightarrow}_{\mathsf{G}} \ldots \stackrel{A_i}{\Longrightarrow}_{\mathsf{G}} \ldots \stackrel{A_n}{\Longrightarrow}_{\mathsf{G}} \dagger$ and by (7.10) thus $\langle A_1 \rangle \cdots \langle A_i \rangle \cdots \langle A_n \rangle \bigtriangledown \in \overline{\mathcal{L}}_{\mathsf{G}}(E_1)$. A simple induction on i then shows that this implies $\langle A_1 \rangle \cdots \langle A_i \rangle \mathrm{tt} \in \overline{\mathcal{L}}_{\mathsf{G}}(E_1)$.

We are left with the case where f is of the form $\langle A_1 \rangle \cdots \langle A_i \rangle \bigtriangleup$. Using (7.8) it is easy to see by looking at the definition of $F' \models_{\mathsf{G}} \triangle$ that

$$\langle A_1 \rangle \cdots \langle A_i \rangle \triangle \in \overline{\mathcal{L}}_{\mathsf{G}}(E) \textit{ iff } \exists a \in \Delta.\ \langle A_1 \rangle \cdots \langle A_i \rangle \langle a \rangle \mathrm{tt} \in \overline{\mathcal{L}}_{\mathsf{G}}(E)$$

for $E \in BL$ and $i \geq 1$. Hence this case is reduced to the one we just have dealt with. $\square$

From the proof it is evident that the lemma still would hold if $\overline{\mathcal{L}}_g$ only had formulae of the form $\langle A_1 \rangle \cdots \langle A_n \rangle \triangledown$, and consequently a logic with formulae of $\mathcal{L}$ which only contained $\triangle$ and the modality $\boxed{A}$ would be sufficient to obtain the linear logic characterization. So why not be be content with this smaller formula language? Pnueli [Pnu85] argues that one advantage of logic is the ability to deal with partial specifications. Clearly there is more freedom to give partial specifications in the larger formula language. With little extra effort we could even include disjunction in $\overline{\mathcal{L}}_g$ without affecting the lemma. (7.7) would also hold if we added conjunction to $\mathcal{L}_g$ and would therefore obtain the characterization for this extended logic too.

Let us end this section by making the note that we easely could have obtained an alternative logic characterization of $\lesssim_{\mathsf{G}}$ by chosing as formula language $\overline{\mathcal{L}}_g$ and for each G take as satisfaction relation $\models'_{\mathsf{G}} \subseteq CL \times \overline{\mathcal{L}}_g$ with definition as 7.6.1 (on $\overline{\mathcal{L}}_g$) but with expressions from CL and not just DCL. We could then for $E \in BL$ let $\overline{\mathcal{L}}_{\mathsf{G}}(E)$ be $\{f \in \overline{\mathcal{L}}_g \mid E \models'_{\mathsf{G}} f\}$. From proposition 7.2.6 and the form of the formulae of $\overline{\mathcal{L}}_g$ it should be clear that lemma 7.6.4 still would hold for the changed set-up and could therefore serve as logic characterization. The reason why we have chosen to give the linear logic characterization is twofold. In the first place we want to show that a linear view (perhaps not surprisingly) is sufficient to capture the G-semantics. Secondly it prepares for a later logic characterization which we could not make so easely in the changed set-up.

Chapter 8

RBL—A Basic Process Language with Refinement

It is well-known ([BC87, vGV87, Hen87b]) that a distinction between concurrency and interleaving may be captured by adding a combinator to the process language, changing the atomicity of actions.

To give a simple concrete example assume the processes $E = \mathsf{Topneg} \,;\, \mathsf{Topneg}$ and $F = \mathsf{Topneg} \parallel \mathsf{Topneg}$ when run accesses a nonempty stack of logical values. With Topneg having the obvious effect on the stack E and F will run without problems leaving the stack as it was. If however Topneg is refined to $\mathsf{Pop} \,;\, \mathsf{Pushneg}$ (again with the effect as suggested by the name) in E and F getting E' and F' respectively the things change. There will be no difference between E and E', but when F' is run the value of the top element may have changed and in the case where the stack consists of one element stack underflow may occur.

We look at the different semantics for BL introduced in the previous chapter 7 and investigate the consequences of adding a combinator allowing an expansion of an individual action into a process.

Formally define a *BL-refinement* to be a mapping $\varrho : \Delta \longrightarrow BL$.

For each BL-refinement ϱ we introduce a combinator, $[\varrho]$, into our language, with the operational meaning that $E[\varrho]$ behaves operationally just like E with all a-occurrences substituted by $\varrho(a)$. We denote this extended language by RBL. The combinator precedence will be the same as for BL except that $[\varrho]$ binds stronger than the binary combinators.

In spite of we have not given a more explicit formulation of substitution yet, we shall look at an example which illustrates not only the idea of substitution but also a consequence for the preorders.

Example: Let $\varrho(a) = a\,;a$, $\varrho(b) = b$, and $E_0 = a \parallel b$, $E_1 = a\,;b \oplus b\,;a$. The "substituted" expressions F_0 and F_0 of $E_0[\varrho]\sigma$ and $E_1[\varrho]\sigma$ respectively will then be

$$F_0 = a\,;a \parallel b \text{ and } F_1 = a\,;a\,;b \oplus b\,;a\,;a$$

Clearly $E_0 \lesssim_{\mathsf{w}} E_1$ but $F_0 \stackrel{aba}{\Longrightarrow}_{\mathsf{w}} \dagger$ and $F_1 \stackrel{aba}{\not\Longrightarrow}_{\mathsf{w}}$. Hence $\lesssim_{\mathsf{w}}$ will not be a precongruence for RBL!

Though the example illustrates that $\lesssim_{\mathsf{W}}$ will not be a precongruence for RBL we cannot use it to conclude the same for $\lesssim_{\mathsf{G}}$ in general since many of the G-semantics would distinguish E_0 and E_1, e.g., $E_0 \not\lesssim_{\mathsf{S}} E_1$.

Our question is here: What is the precongruence associated with $\lesssim_{\mathsf{G}}$ for RBL, $\lesssim^c_{\mathsf{G}}$ (the largest precongruence contained in $\lesssim_{\mathsf{G}}$)? In the next section we give the different operational G-semantics and derive some results. We then pursue the question for $\lesssim_{\mathsf{W}}$ in the succeeding section through different considerations, gradually arriving at a model fully abstract with respect to $\lesssim^c_{\mathsf{W}}$. From the model considerations it then turns out that $\lesssim^c_{\mathsf{W}}$ equals $\lesssim^c_{\mathsf{G}}$ for every G-semantics.

8.1 Operational Set-up

We shall give different operational semantics for RBL similar as in the chapter with BL. In fact the extended labelled transition system will be the same except that the configuration language RCL now is the least set C satisfying:

$$\begin{array}{rl}
\dagger \in C & \\
RBL \subseteq C & \\
E_0 \,;\, E_1 \in C & \text{if } E_0 \in C \text{ and } E_1 \in RBL \\
E_0 \parallel E_1 \in C & \text{if } E_0, E_1 \in C
\end{array}$$

and the definition of $\rightarrowtail$ is augmented in order to cope with $[\varrho]$:

$$a[\varrho] \rightarrowtail \varrho(a)$$

$$(E_0 \,;\, E_1)[\varrho] \rightarrowtail E_0[\varrho] \,;\, E_1[\varrho] \qquad \frac{E \rightarrowtail E'}{E[\varrho] \rightarrowtail E'[\varrho]}$$

$$(E_0 \oplus E_1)[\varrho] \rightarrowtail E_0[\varrho] \oplus E_1[\varrho]$$

$$(E_0 \parallel E_1)[\varrho] \rightarrowtail E_0[\varrho] \parallel E_1[\varrho]$$

Notice that there is no rule to deal with a case like $E[\varrho'][\varrho]$. This is not necessary because the $[\varrho]$-inference rule allows "substitution" of $[\varrho']$ in E by internal steps before starting with $[\varrho]$.

Example: Suppose $\varrho'(b) = c \,;\, d$ and $\varrho(c) = e$. Then

$$\begin{aligned}
(a \parallel b)[\varrho'][\varrho] &\rightarrowtail (a[\varrho'] \parallel b[\varrho'])[\varrho] \\
&\rightarrowtail (a[\varrho'] \parallel c \,;\, d)[\varrho] \\
&\rightarrowtail a[\varrho'][\varrho] \parallel (c \,;\, d)[\varrho] \\
&\rightarrowtail a[\varrho'][\varrho] \parallel c[\varrho] \,;\, d[\varrho] \\
&\rightarrowtail a[\varrho'][\varrho] \parallel e \,;\, d[\varrho] \xrightarrow{e}_{\mathsf{G}} \cdots
\end{aligned}$$

Notice that the "substitution" not necessarily has to follow a unique route. E.g., above it is also possible with: $(a[\varrho'] \,\|\, b[\varrho'])[\varrho] \succ\!\!\longrightarrow a[\varrho'][\varrho] \,\|\, b[\varrho'][\varrho] \succ\!\!\longrightarrow a[\varrho'][\varrho] \,\|\, (c;d)[\varrho] \succ\!\!\longrightarrow \ldots$.

The definitions of $\stackrel{A}{\Rightarrow}_{\mathsf{G}}$, $\stackrel{s}{\Rightarrow}_{\mathsf{G}}$ and $\lesssim_{\mathsf{G}}$ are generalized to RBL in the obvious way. We keep the convention to leave out the subscript G in $\longrightarrow_{\mathsf{G}}$ and $\Rightarrow_{\mathsf{G}}$ except for certain G's.

Proposition 7.2.3 extends smoothly to RBL with one addition:

Proposition 8.1.1 Suppose $E \in RBL$, $E \succ\!\!\longrightarrow^* E'$ and $E_0, E_1 \in RCL$, $E_0 \stackrel{s}{\Rightarrow} E_0'$. Then

- $E_0 \,;\, E \stackrel{s}{\Rightarrow} E_0' \,;\, E$
- $E_0 \parallel E_1 \stackrel{s}{\Rightarrow} E_0' \parallel E_1$
- $E_1 \parallel E_0 \stackrel{s}{\Rightarrow} E_1 \parallel E_0'$
- $E[\varrho] \succ\!\!\longrightarrow^* E'[\varrho]$

We will now make it more precise what we mean by substitution. The substitution is "performed" by a compositionally defined mapping $\sigma : RCL \longrightarrow CL$, using $\{\varrho\} : BL \longrightarrow BL$ which (also compositionally) performs a single substitution in a refinement free expression. Because of their syntactic nature we write them postfix. The definitions of σ and $\{\varrho\}$ are in full:

$$
\begin{array}{rlrl}
\dagger\sigma & = \dagger & & \\
a\sigma & = a & a\{\varrho\} & = \varrho(a) \\
(E_0 \,;\, E_1)\sigma & = E_0\sigma \,;\, E_1\sigma & (E_0 \,;\, E_1)\{\varrho\} & = E_0\{\varrho\} \,;\, E_1\{\varrho\} \\
(E_0 \oplus E_1)\sigma & = E_0\sigma \oplus E_1\sigma & (E_0 \oplus E_1)\{\varrho\} & = E_0\{\varrho\} \oplus E_1\{\varrho\} \\
(E_0 \parallel E_1)\sigma & = E_0\sigma \parallel E_1\sigma & (E_0 \parallel E_1)\{\varrho\} & = E_0\{\varrho\} \parallel E_1\{\varrho\} \\
E[\varrho]\sigma & = (E\sigma)\{\varrho\} & &
\end{array}
$$

Notice that σ when restricted to RBL yield a map $\sigma : RBL \longrightarrow BL$. Because configurations only contains expressions like $E[\varrho]$ when $E \in RBL$ we then do not need a case for $\{\varrho\}$ similar to $\dagger\sigma = \dagger$ and we conclude that the definitions are well-defined.

Example: Suppose $\varrho'(a) = b \,;\, c$, $\varrho(b) = a \,;\, b$ and otherwise $\varrho'(e) = \varrho(e) = e$. Then

$$(a \parallel b)[\varrho'][\varrho]\sigma = (b \,;\, c \parallel b)\{\varrho\} = a \,;\, b \,;\, c \parallel a \,;\, b$$

The rest of this section is devoted the proof of the following proposition which essentially states: $E \in RBL$ behaves operationally as if the refinements were substituted in advance:

Proposition 8.1.2 Suppose $E \in RBL$. Then for $s \in \mathsf{G}^*$:

$$E \stackrel{s}{\Rightarrow} \dagger \textit{ iff } E\sigma \stackrel{s}{\Rightarrow} \dagger$$

Proof The proposition follow directly from:

$$E \in RCL, E \overset{s}{\Rightarrow} E'$$

(8.1) $\Downarrow$

$$E\sigma \overset{s}{\Rightarrow} E'\sigma$$

and

$$E \in RCL, E\sigma \overset{s}{\Rightarrow} E'$$

(8.2) $\Downarrow$

$$\exists E'' \in RCL, E \overset{s}{\Rightarrow} E'', E''\sigma = E'$$

which both are proven by induction on the length of $\overset{s}{\Rightarrow}$ using the following two lemmas in the inductive step of (8.1) and lemma 8.1.6 in the inductive step of (8.2). □

Lemma 8.1.3 For $E \in RCL$ we have: $E \rightarrowtail E'$ implies $E\sigma \rightarrowtail^* E'\sigma$

Proof Induction on the structure of E.

$E = \dagger$ or $E = a$: Then $E \not\rightarrowtail$ and the implication holds vacuously.

$E = E_0 \,;\, E_1$: From the definition of $\rightarrowtail$ we see that there are two cases:
$E_0 = \dagger$ and $E' = E_1$: We have $(\dagger \,;\, E_1)\sigma = \dagger \,;\, E_1\sigma \rightarrowtail E_1\sigma = E'$.
$E_0 \rightarrowtail E_0'$ and $E' = E_0' \,;\, E_1$: By induction $E_0\sigma \rightarrowtail^* E_0'\sigma$, so from proposition 8.1.1 then $E\sigma = E_0\sigma \,;\, E_1\sigma \rightarrowtail^* E_0'\sigma \,;\, E_1\sigma = E'\sigma$.

$E = E_0 \oplus E_1$: W.l.o.g. assume $E \rightarrowtail E_0 = E'$. Clearly $E\sigma = E_0\sigma \oplus E_1\sigma \rightarrowtail E_0\sigma = E\sigma$.

$E = E_0 \parallel E_1$: Similar and symmetric to the case $E = E_0 \,;\, E_1$.

$E = F[\varrho]$: In each case when the internal step derives from an axiom one easely from the definition of $\_\sigma$ and $\_\{\varrho\}$ show $F[\varrho]\sigma = E'\sigma$. Since $E'\sigma \rightarrowtail^0 E'\sigma$ the result then follows. It remains to look at the case where the internal step derives from the inference rule. Here we have $F \rightarrowtail F'$ and $E' = F'[\varrho]$. By hypothesis of induction $F\sigma \rightarrowtail^* F'\sigma$. Since $F\sigma \in BL$ we can use (8.3) below to get $(F[\varrho])\sigma = F\sigma\{\varrho\} \rightarrowtail^* F'\sigma\{\varrho\} = F'[\varrho]\sigma = E'\sigma$ as desired.

We used

(8.3) $$n \geq 0, E \in BL, E \rightarrowtail^n E' \text{ implies } E\{\varrho\} \rightarrowtail^* E'\{\varrho\}$$

which is proved by induction on n and in the inductive step one prove (8.3) for $n = 1$ by induction on the structure of E. The arguments are identical to the ones used above except that the things get easier because $E \in BL$ and we therefore do not have to deal with σ and the cases $E = \dagger$ and $E = F[\varrho]$. □

Lemma 8.1.4 For $E \in RCL$ we have: $E \xrightarrow{A} E'$ implies $E\sigma \xrightarrow{A} E'\sigma$

Proof By induction on the structure of E.

$E = \dagger$: Trivially true.

$E = a$: The only possibility is $A = a$ and $E' = \dagger$. We have $a\sigma = a \xrightarrow{a} \dagger = \dagger\sigma$.

$E = E_0 \,;\, E_1$: According to the definition of $\xrightarrow{A}$ we can only have $E_0 \,;\, E_1 \xrightarrow{A} E'$ if $E_0 \xrightarrow{A} E'_0$ and $E' = E'_0 \,;\, E_1$. By hypothesis of induction $E_0\sigma \xrightarrow{A} E'_0\sigma$. Using the inference rule for ; we get $E\sigma = E_0\sigma \,;\, E_1\sigma \xrightarrow{A} E'_0\sigma \,;\, E_1\sigma = E'\sigma$.

$E = E_0 \oplus E_1$: $\xrightarrow{A}$ is not defined for expressions of this form so the implication holds trivially.

$E = E_0 \parallel E_1$: There are three potential ways the step could have been produced. If only one part, E_0 or E_1, is involved the argument follow the case $E = E_0 \,;\, E_1$. Otherwise we have $E' = E'_0 \parallel E'_1$ where $A = A_0 \times A_1 \in \mathsf{G}$ and $E_i \xrightarrow{A_i} E'_i$ for $i = 0, 1$. By hypothesis of induction we then get $E\sigma = E_0\sigma \parallel E_1\sigma \xrightarrow{A} E'_0\sigma \parallel E'_1\sigma = E'\sigma$.

$E = F[\varrho]$: There are no axioms or inference rules for $\xrightarrow{A}$ when E is of this form.

□

In the statement and proofs of the lemma to follow we shall make extensive use of some special subsets $\overline{RBL}$ and $\overline{RCL}$ of the process expressions and the configuration expressions respectively. The idea is that $E \in \overline{RBL} \subseteq RBL$ if no internal step can bring any refinement combinator of E "inwards" in E. Similar if $E \in \overline{RCL}$. Looking at the rules for internal steps dealing with the refinement combinator one soon realize that the refinement then only can appear in the scope of a $\oplus$-combinator or the right hand side of a ;-combinator. This leads to the following inductive definition.

Definition 8.1.5 $\overline{RCL}$ is the least subset C of RCL which satisfies:

$$\begin{array}{rl} CL \subseteq C & \\ E_0 \,;\, E_1 \in C & \text{if } E_0 \in C \text{ and } E_1 \in RBL \\ E_0 \oplus E_1 \in C & \text{if } E_0, E_1 \in RBL \\ E_0 \parallel E_1 \in C & \text{if } E_0, E_1 \in C \end{array}$$

$\overline{RBL}$ is $RBL \cap \overline{RCL}$, i.e., $\overline{RBL}$ is the process expressions of $\overline{RCL}$ or equally those expressions of $\overline{RCL}$ that contains no †. □

Example: $a[\varrho] \oplus (b \,;\, c[\varrho]) \in \overline{RCL}$ but $a[\varrho] \parallel (b \,;\, c)[\varrho] \notin \overline{RCL}$ because $a[\varrho] \rightarrowtail \varrho(a)$ and $[\varrho]$ can be moved in over $b \,;\, c$.

Lemma 8.1.6 If $E \in RCL$ then

a) $E\sigma \rightarrowtail E'$ implies $\exists E'' \in RCL.\, E \rightarrowtail^* E'', E''\sigma = E'$

b) $E\sigma \xrightarrow{A} E'$ implies $\exists E'' \in RCL.\, E \overset{A}{\Rightarrow} E', E''\sigma = E'$.

Proof

a) Using lemma 8.1.7 we find a $F \in \overline{RCL}$ fulfilling $E \rightarrowtail^* F$ and $F\sigma = E\sigma$. So $F\sigma \rightarrowtail E'$. Since $F \in \overline{RCL}$ we can use lemma 8.1.9 to find a E'' with $F \rightarrowtail E''$ and $E''\sigma = E'$. Together we now have $E \rightarrowtail^* F \rightarrowtail E''$ and $E''\sigma = E'$.

b) Given $E\sigma \xrightarrow{A} E'$. As in a) we find a $F \in \overline{RCL}$ such that $E \rightarrowtail^* F$ and $F\sigma \xrightarrow{A} E'$. Because $F \in \overline{RCL}$ lemma 8.1.10 then yields $F \xrightarrow{A} E''$ for a E'' with $E''\sigma = E'$. Collecting the facts we have $E \rightarrowtail^* F \xrightarrow{A} E''$—i.e $E \stackrel{A}{\Rightarrow} E''$ and $E''\sigma = E'$.

□

The next lemma states that the refinement combinators can be brought entirely "inwards" by internal steps.

Lemma 8.1.7 Given $E \in RCL(RBL)$ there exists a $E' \in \overline{RCL}$ $(\overline{RBL})$ such that $E \rightarrowtail^* E'$ and $E'\sigma = E\sigma$.

Example: $E = a[\varrho] \parallel (b\,;c)[\varrho] \rightarrowtail \varrho(a) \parallel (b\,;c)[\varrho] \rightarrowtail^* \varrho(a) \parallel (\varrho(b)\,;c[\varrho]) = E' \in \overline{RCL}$ and $E\sigma = \varrho(a) \parallel (\varrho(b)\,;(c[\varrho]\sigma)) = E'\sigma$.

Proof By induction on the structure of E.

$E = \dagger$: Then $E \in RCL$. But also $E' := E = \dagger \in CL \subseteq \overline{RCL}$.

$E = a$: Just choose $E' = a \in BL \subseteq \overline{RBL} \subseteq \overline{RCL}$.

$E = E_0\,; E_1$: Here we must have $E_0 \in RCL(RBL)$ and $E_1 \in RBL$. By hypothesis of induction there is a $E_0' \in RCL(RBL)$ such that $E_0 \rightarrowtail^* E_0'$ and $E_0'\sigma = E_0\sigma$. Let $E' = E_0'\,; E_1$. From proposition 8.1.1 then $E_0\,; E_1 \rightarrowtail E'$ and of course $E' \in \overline{RCL}(\overline{RBL})$. Also $E'\sigma = E_0'\sigma\,; E_1\sigma = E_0\sigma\,; E_1\sigma = E\sigma$.

$E = E_0 \oplus E_1$: Clearly we can choose $E' = E$ here.

$E = E_0 \parallel E_1$: Similar arguments as in the case $E = E_0\,; E_1$.

$E = F[\varrho]$: E can only be of this form when $F \in RBL$. By hypothesis of induction $F \rightarrowtail^* F'$ for a $F' \in \overline{RBL}$ with $F'\sigma = F\sigma$. Since $F' \in \overline{RBL}$ we can use the following lemma to find a $E' \in \overline{RBL}$ such that $F'[\varrho] \rightarrowtail^* E'$ and $F'[\varrho]\sigma = E'\sigma$. From proposition 8.1.1 then $E = F[\varrho] \rightarrowtail^* F'[\varrho] \rightarrowtail^* E'$. We also have $E'\sigma = F'[\varrho]\sigma = (F'\sigma)\{\varrho\} = (F\sigma)\{\varrho\} = F[\varrho]\sigma = E\sigma$ as desired.

□

We need a measure, h, for the inductive proof of the next lemma. Intuitively h measure the number of internal steps necessary to move a refinement combinator $[\varrho]$ from outside "entirely inwards" in an expression $E \in \overline{RBL}$—i.e., if $\mathrm{h}(E) = n$ then there is a E' such that $E[\varrho] \rightarrowtail^n E' \in \overline{RBL}$. $\mathrm{h}(E) = 3$ in the example above. Formally $\mathrm{h} : \overline{RBL} \longrightarrow I\!N^+$ is given by:

$$
\begin{aligned}
\mathrm{h}(a) &= 1\\
\mathrm{h}(E_0\,; E_1) &= 1 + \mathrm{h}(E_0)\\
\mathrm{h}(E_0 \oplus E_1) &= 1\\
\mathrm{h}(E_0 \parallel E_1) &= \mathrm{h}(E_0) + \mathrm{h}(E_1)
\end{aligned}
$$

Notice that we do not have to define h for expressions of the form $E[\varrho]$ because they cannot belong to $\overline{RBL}$.

Lemma 8.1.8 If $E \in \overline{RBL}$ then there is an $E' \in \overline{RBL}$ such that

$$E[\varrho] \rightarrowtail^* E' \text{ and } E'\sigma = E[\varrho]\sigma$$

Proof By induction on $\mathrm{h}(E)$.

$\mathrm{h}(E) = 1$: There are two cases:

$E = a$: Then $E[\varrho] \rightarrowtail \varrho(a)$. Choose $E' = \varrho(a)$. Since $E' \in BL \subseteq \overline{RBL}$ by definition of BL-refinements we have $E'\sigma = E' = \varrho(a) = a\{\varrho\} = E\sigma$.

$E = E_0 \oplus E_1$: Then $E[\varrho] \rightarrowtail E_0[\varrho] \oplus E_1[\varrho] =: E' \in \overline{RBL}$ and the result follows from the compositional nature of σ and $\{\varrho\}$.

$\mathrm{h}(E) > 1$: Again there are two cases:

$E = E_0 \,;\, E_1$: Then $\mathrm{h}(E_0) < \mathrm{h}(E)$ and of course $E_0 \in \overline{RBL}$, so we can use the hypothesis of induction to find an E_0' such that $E_0[\varrho] \rightarrowtail^* E_0'$ and $E_0'\sigma = E_0[\varrho]\sigma$. Choosing $E' = E_0' ; E_1[\varrho]$ we get from proposition 8.1.1 $(E_0 ; E_1)[\varrho] \rightarrowtail E_0[\varrho] ; E_1[\varrho] \rightarrowtail^* E'$ and $E'\sigma = E_0'\sigma ; E_1[\varrho]\sigma = E_0[\varrho]\sigma ; E_1[\varrho]\sigma = (E_0\sigma ; E_1\sigma)\{\varrho\} = (E_0 ; E_1)\sigma\{\varrho\} = E[\varrho]\sigma$.

$E = E_0 \parallel E_1$: Here both $\mathrm{h}(E_0)$ and $\mathrm{h}(E_1)$ are less than $\mathrm{h}(E)$ so we can apply the hypothesis of induction on both and obtain the result with similar arguments as in the last case.

□

Lemma 8.1.9 Assume $E \in \overline{RCL}$. Then

$$\begin{array}{l} E\sigma \rightarrowtail E' \\ \Downarrow \\ \exists E''.\, E \rightarrowtail E'', E''\sigma = E' \end{array}$$

Proof By induction (from the definition of $\overline{RCL}$).

$E \in CL$: Then $E\sigma = E \rightarrowtail E' \in CL$. Hence also $E'\sigma = E'$ and we can chose $E'' = E'$.

$E = E_0 \,;\, E_1$, $E_0 \in \overline{RCL}$ and $E_1 \in RBL$: We have $E\sigma = E_0\sigma \,;\, E_1\sigma \rightarrowtail E'$. According to the definition of $\rightarrowtail$ there are two subcases to consider:

- $E_0\sigma = \dagger$ and $E' = E_1\sigma$: $E_0\sigma = \dagger$ implies $E_0 = \dagger$. Letting $E'' = E_1$ we get $E_0 \,;\, E_1 = \dagger \,;\, E_1 \rightarrowtail E''$ and $E''\sigma = E_1\sigma = E'$.
- $E_0\sigma \rightarrowtail E_0'$ and $E' = E_0' \,;\, E_1\sigma$: By hypothesis $\exists E_0''.E_0 \rightarrowtail E_0''$, $E_0''\sigma = E_0'$. With $E'' = E_0'' \,;\, E_1$ we get $E_0 \,;\, E_1 \rightarrowtail E''$ and $E''\sigma = E_0''\sigma \,;\, E_1\sigma = E_0' \,;\, E_1\sigma = E'$.

$E = E_0 \oplus E_1$ and $E_0, E_1 \in RBL$: Here the situation is $E\sigma = E_0\sigma \oplus E_1\sigma$. Inspecting the definition of $\rightarrowtail$ we see that there only is two possibilities. Assume w.l.o.g. that $E' = E_0\sigma$. Since $E_0 \oplus E_1 \rightarrowtail E_0$ the result then follows if we let $E'' = E_0$.

$E = E_0 \parallel E_1$ and $E_0, E_1 \in \overline{RCL}$: Similar/ symmetric argument as in the case $E = E_0 ; E_1$.

□

Lemma 8.1.10 Suppose $E \in \overline{RCL}$ and $A \in \mathsf{G}$. Then

$$E\sigma \stackrel{A}{\longrightarrow} E'$$
$$\Downarrow$$
$$\exists E''.\, E \stackrel{A}{\longrightarrow} E'', E''\sigma = E'$$

Proof By induction (from the definition of $\overline{RCL}$).

$E \in CL$: Then $E\sigma = E \stackrel{A}{\longrightarrow} E' \in CL$, so $E'\sigma = E'$ and we choose $E'' = E'$.

$E = E_0 ; E_1$, $E_0 \in \overline{RCL}$ and $E_1 \in RBL$: We have $E\sigma = E_0\sigma ; E_1\sigma \stackrel{A}{\longrightarrow} E'$. Since there only is one rule for $\stackrel{A}{\longrightarrow}$ and expressions of this form we deduce $E' = E'_0 ; E_1\sigma$ where $E_0\sigma \stackrel{A}{\longrightarrow} E'_0$. Because $E_0 \in \overline{RCL}$ we can use the hypothesis of induction to find an E''_0 such that $E_0 \stackrel{A}{\longrightarrow} E''_0$ and $E''_0\sigma = E'_0$. From the same rule we then get $E_0 ; E_1 \stackrel{A}{\longrightarrow} E''_0 ; E_1$. Choose $E'' = E''_0 ; E_1$ and we have $E''\sigma = E''_0\sigma ; E_1\sigma = E'_0 ; E_1\sigma = E'$ as we want.

$E = E_0 \oplus E_1$ and $E_0, E_1 \in RBL$: This means $E\sigma = E_0\sigma \oplus E_1\sigma$, but there is no rule for $\stackrel{A}{\longrightarrow}$ and expressions of this form wherefore the implication holds trivially.

$E = E_0 \parallel E_1$ and $E_0, E_1 \in \overline{RBL}$: Here we have $E\sigma = E_0\sigma \parallel E_1\sigma$. Three inference rules shall be taken into consideration. The ones with only one of $E_0\sigma$ and $E_1\sigma$ involved in $\stackrel{A}{\longrightarrow}$ goes similar/ symmetric as in the case $E = E_0 ; E_1$. If both are involved we have $E' = E_0 \parallel E_1$ where $A = A_0 \times A_1$ and for $i = 0, 1$, $E_i\sigma \stackrel{A_i}{\longrightarrow} E'_i$ and $A_i \in \mathsf{G}$. Since $E_0, E_1 \in \overline{RCL}$ the hypothesis of induction can be applied to get for each i an E''_i such that $E_i \stackrel{A_i}{\longrightarrow} E''_i$ and $E''_i\sigma = E'_i$. Since $A \in \mathsf{G}$ then also $E_0 \parallel E_1 \stackrel{A_0 \times A_1}{\longrightarrow} E''_0 \parallel E''_1$. Choosing $E'' = E''_0 \parallel E''_1$ this reads $E \stackrel{A}{\longrightarrow} E''$ and we see $E''\sigma = E''_0\sigma \parallel E''_1\sigma = E'_0 \parallel E'_1 = E'$ and we are done.

□

8.2 Denotational Set-up

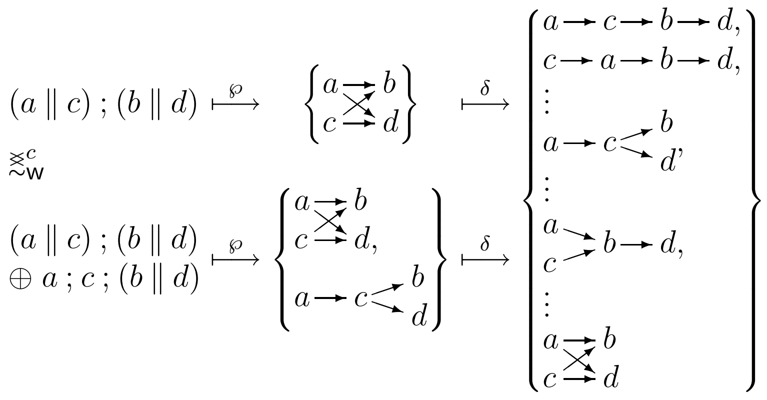
As mentioned in the introduction to this chapter we will start out by searching a model for $\lesssim^c_{\mathsf{W}}$. To this end $[\![\_]\!]_{\mathsf{W}}$ is extended to RBL by letting

$$[\![E]\!]_{\mathsf{W}} = [\![E\sigma]\!]_{\mathsf{W}} \text{ for } E \in RBL$$

The induced denotational preorder $\trianglelefteq_{\mathsf{W}}$ then also extends to RBL and it follows from proposition 8.1.2 and proposition 7.4.3 that $\lesssim_{\mathsf{W}} = \trianglelefteq_{\mathsf{W}}$ on RBL.

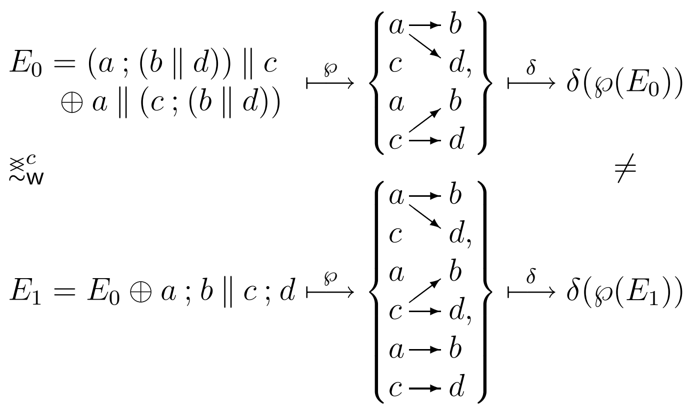
In the case of $\lesssim^c_{\mathsf{W}}$ it is much harder (than in the previous chapter) to see intuitively that $\lesssim^c_{\mathsf{W}}$ should have a pomset based model at all—and if so what it should look like. But following the pattern from the previous chapter we shall be looking for a model in which the denotation of E is expressible as $\delta_*(\wp(E))$ for a suitable pomset property P_*—but which?

Playing with examples, one soon realizes that a refinement combinator is quite a powerful tool in distinguishing expressions, because much of the information represented in $\wp(E)$ may be reflected by suitable refinement combinator $[\varrho]$, in the sense of "overlapping" occurrences of ϱ-images of concurrent elements in $\mathbf{p} \in \wp(E)$ (as indicated on page 172 in the example of $\lesssim_{\mathsf{w}}$ not being a precongruence for RBL). So clearly fewer identifications should be made. Through examples like:



one might be led to the conjecture that $\delta(\wp(E))$ ordered under inclusion could be a model for $\lesssim^c_{\mathsf{w}}$. However, this is *not* the case, as can be seen by looking at the example:

Example:



The inequallity follows from:

$$\mathbf{p} = \begin{array}{c} a \to b \\ c \to d \end{array} \in \delta(\wp(E_1)), \qquad \begin{array}{c} a \to b \\ c \to d \end{array} \notin \delta(\wp(E_0))$$

We do not intend to prove operational that $E_0 \lesssim^c_{\mathsf{w}} E_1$ and $E_1 \lesssim^c_{\mathsf{w}} E_0$ (it will follow easily from the denotational characterization to be developed), but invite the reader to find convincing arguments for this fact.

So, presumable $\mathbf{p}$ should not belong to the denotation of E in a model for $\lesssim^c_{\mathsf{w}}$. Intuitively, an argument could be that no single linearization of a refinement version from $\mathbf{p}{<}\varrho{>}$ can reflect the full structure of $\mathbf{p}$, in the sense that if the images of a and d overlap in such a linearization (reflecting a and d being concurrent) then the image of c must precede that of b, and vice versa. Following this intuition one may look for a property expressing when the full structure of a pomset may be reflected in a single linearization of a refined version of it (in the "overlapping" sense).

Based on our example, we suggest the following formalization of this property—expressed as a slight modification of the P_{and}-property.

Definition 8.2.1 *P_{or}-Property for Pomsets*

A pomset $\mathbf{p}$ is said to have the P_{or}-property, $P_{or}(\mathbf{p})$ *iff* for all x, x', y, y' in X_p we have:

$$\text{if} \quad \begin{array}{l} x <_p x' \\ co_p \\ y <_p y' \end{array} \quad \text{then} \quad \begin{array}{l} \forall z.\, y <_p z \Rightarrow x <_p z \\ or \\ \forall z.\, x <_p z \Rightarrow y <_p z \end{array} \qquad \square$$

Example: $\begin{matrix} a & \rightarrow & b \\ & \searrow & \\ c & \rightarrow & d \end{matrix}$ has the P_{or}-property, $\begin{matrix} a & \rightarrow & b \\ c & \rightarrow & d \end{matrix}$ has not.

P_{or} has an alternative characterization (used extensively in the following), the proof of which is trivial:

Proposition 8.2.2 A pomset $\mathbf{p} = [p]$ has the P_{or}-property *iff* for all x, x', y, y' in X_p:

$$\text{if} \quad \begin{array}{l} x <_p x' \\ co_p \\ y <_p y' \end{array} \quad \text{then} \quad \begin{array}{l} x <_p y' \\ or \\ y <_p x' \end{array}$$

Proposition 8.2.3 The P_{or}-property is hereditary and dot synthesizable.

Proof With the alternative characterization the proposition is proved with similar argumentation as the P_{and}-property was proved in proposition 7.3.2 to be hereditary and dot synthesizable. $\square$

After these manœuvres we now give the denotation of an expression $E \in RBL$, $[\![E]\!]_{or}$, in the model we informally arrived at, that is, finite sets of P_{or}-pomsets partial ordered under inclusion.

Definition 8.2.4 $[\![\_]\!]_{or} : RBL \longrightarrow \mathcal{P}(\mathbf{P}_{or} \setminus \{\varepsilon\}) \setminus \emptyset$ is defined by $[\![E]\!]_{or} = \delta_{or}(\wp(E\sigma))$. $\square$

As usual the induced denotational preorder is denoted $\trianglelefteq_{or}$.

Returning to the example on page 180 using δ_{or} in stead of δ we now get:

$$\begin{array}{lcl}
\begin{array}{l} E_0 = (a\,;(b \parallel d)) \parallel c \\ \quad \oplus\, a \parallel (c\,;(b \parallel d)) \end{array} \overset{\wp}{\longmapsto} \left\{ \begin{matrix} a & \rightarrow & b \\ & \searrow & \\ c & & d, \\ a & & b \\ & \nearrow & \\ c & \rightarrow & d \end{matrix} \right\} \overset{\delta_{or}}{\longmapsto} \delta_{or}(\wp(E_0)) \\
\lesssim^{c}_{\mathsf{w}} \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad = \\
E_1 = E_0 \oplus a\,; b \parallel c\,; d \overset{\wp}{\longmapsto} \left\{ \begin{matrix} a & \rightarrow & b \\ & \searrow & \\ c & & d, \\ a & & b \\ & \nearrow & \\ c & \rightarrow & d, \\ a & \rightarrow & b \\ c & \rightarrow & d \end{matrix} \right\} \overset{\delta_{or}}{\longmapsto} \delta_{or}(\wp(E_1))
\end{array}$$

where the equality follows from:

$$\delta_{or}(\wp(E_1)) = \delta_{or}(\wp(E_0)) \cup \delta_{or}(\begin{Bmatrix} a \rightarrow b \\ c \rightarrow d \end{Bmatrix}) = \delta_{or}(\wp(E_0)) \cup \delta_{or}(\begin{Bmatrix} a \rightarrow b & a \rightarrow b \\ \searrow & \nearrow \\ c \rightarrow d, & c \rightarrow d \end{Bmatrix}) = \delta_{or}(\wp(E_0))$$

For each BL-refinement ϱ we associate the corresponding $\mathcal{P}(\mathbf{P})$-refinement $\wp(\varrho)$, by letting $(\wp(\varrho))(a) = \wp(\varrho(a))$. Notice that $\wp(\varrho)$ is ε-free.

Theorem 8.2.5 $[\![\_]\!]_{or}$ has the following compositional definition:

$$\begin{array}{ll} [\![a]\!]_{or} & = \{a\} \\ [\![E_0 \,;\, E_1]\!]_{or} & = [\![E_0]\!]_{or} \cdot [\![E_1]\!]_{or} \\ [\![E_0 \oplus E_1]\!]_{or} & = [\![E_0]\!]_{or} \cup [\![E_1]\!]_{or} \\ [\![E_0 \parallel E_1]\!]_{or} & = \delta_{or}([\![E_0]\!]_{or} \times [\![E_1]\!]_{or}) \\ [\![E[\varrho]]\!]_{or} & = \delta_{or}([\![E]\!]_{or}{<}\wp(\varrho){>}) \end{array}$$

Proof Similar to that of theorem 7.3.7, but also using the compositional nature of σ. The case $E[\varrho]$ is more difficult, so we use some lemma's proved in the sequel.

$$\begin{array}{lll} [\![E[\varrho]]\!]_{or} & = \delta_{or}(\wp(E[\varrho]\sigma)) & \text{definition of } [\![\_]\!]_{or} \\ & = \delta_{or}(\wp((E\sigma)\{\varrho\})) & \text{definition of } \sigma \\ & = \delta_{or}((\wp(E\sigma)){<}\wp(\varrho){>}) & \text{lemma 8.2.7 and } E\sigma \in BL \\ & = \delta_{or}((\delta_{or}(\wp(E\sigma))){<}\wp(\varrho){>}) & \text{lemma 8.2.6} \\ & = \delta_{or}([\![E]\!]_{or}{<}\wp(\varrho){>}) & \text{definition of } [\![\_]\!]_{or} \end{array}$$

□

Lemma 8.2.6 Let P be a set of pomsets and ϱ an ε-free $\mathcal{P}(\mathbf{P})$-refinement. Then

$$\delta_{or}((\delta_{or}(P)){<}\varrho{>}) = \delta_{or}(P{<}\varrho{>})$$

Proof Clearly it is enough to prove $\delta_{or}((\delta_{or}(\mathbf{p})){<}\varrho{>}) = \delta_{or}(\mathbf{p}{<}\varrho{>})$ for a single pomset $\mathbf{p}$. Each inclusion is proven separately.

To see $\delta_{or}((\delta_{or}(\mathbf{p})){<}\varrho{>}) \subseteq \delta_{or}(\mathbf{p}{<}\varrho{>})$ let $\mathbf{q} \in \delta_{or}((\delta_{or}(\mathbf{p})){<}\varrho{>})$. Then $P_{or}(\mathbf{q})$ and there exists a $\mathbf{q}' \in (\delta_{or}(\mathbf{p})){<}\varrho{>}$ such that $\mathbf{q} \preceq \mathbf{q}'$. Therefore $\mathbf{q}' \in \mathbf{p}'{<}\varrho{>}$ for some $\mathbf{p}' \in \delta_{or}(\mathbf{p})$ and we have $\mathbf{p}' \preceq \mathbf{p}$. But by the nature of ${<}\varrho{>}$ this implies $\forall \mathbf{r}' \in \mathbf{p}'{<}\varrho{>} \exists \mathbf{r} \in \mathbf{p}{<}\varrho{>}. \mathbf{r}' \preceq \mathbf{r}$ (see proposition 6.2.2 and proposition 6.2.3). Hence there exists a $\mathbf{r} \in \mathbf{p}{<}\varrho{>}$ such that $\mathbf{q} \preceq \mathbf{q}' \preceq \mathbf{r}$. Since $P_{or}(\mathbf{q})$ we have $\mathbf{q} \in \delta_{or}(\mathbf{p}{<}\varrho{>})$.

$\delta_{or}((\delta_{or}(\mathbf{p})){<}\varrho{>}) \supseteq \delta_{or}(\mathbf{p}{<}\varrho{>})$: Suppose $\mathbf{q} \in \delta_{or}(\mathbf{p}{<}\varrho{>})$. This means $P_{or}(\mathbf{q})$ and $\mathbf{q} \preceq [p{<}\pi_p{>}]$, where ${<}\pi_p{>}$ is a ϱ-consistent particular refinement for a representative, p, of $\mathbf{p}$. So it is enough to find an $\mathbf{p}' \in \delta_{or}(\mathbf{p})$ such that $\mathbf{q} \preceq [p'{<}\pi_{p'}{>}]$, where $\pi_{p'}$ also is consistent with ϱ.

By proposition 6.2.3 $\mathbf{q} \preceq [p{<}\pi_p{>}]$ implies the existence of a representative, q, of $\mathbf{q}$ such that $q = \langle X_{p<\pi_p>}, \leq_q, \ell_{p<\pi_p>} \rangle$ and $\leq_q \supseteq \leq_{p<\pi_p>}$.

Define $p' := \langle X_p, \leq_{p'}, \ell_p \rangle$, where $\leq_{p'}$ is the reflexive closure of $<_{p'} \subseteq {X_p}^2$ defined by:

$$x <_{p'} y \quad \textit{iff} \quad \forall \langle x, x' \rangle, \langle y, y' \rangle \in X_q.\ \langle x, x' \rangle <_q \langle y, y' \rangle \tag{8.4}$$

That is, we order elements x, y in p' if and only if all elements from $\pi_p(y)$ are causally dependent on all elements $\pi_p(x)$ in q.

To see that p' in fact is a lpo notice that $\leq_{p'}$ by definition is reflexive, clearly also transitive and the antisymmetry is seen from (8.4), the ε-freeness of π_p (a consequence of ϱ being ε-free) and the antisymmetry of $\leq_q$.

$X_p = X_{p'}$ and $\ell_p = \ell_{p'}$ so $\mathbf{p'} \preceq \mathbf{p}$ follows by proving $\leq_{p'} \supseteq \leq_p$. By definition $x \leq_{p'} x$. If $x <_p y$ then $x \neq y$, so by the construction of $p{<}\pi_p{>}$ we have $\forall \langle x, x' \rangle, \langle y, y' \rangle \in X_{p<\pi_p>}.\ \langle x, x' \rangle <_{p<\pi_p>} \langle y, y' \rangle$ and from $\leq_q \supseteq \leq_{p<\pi_p>}$ this implies $\forall \langle x, x' \rangle, \langle y, y' \rangle \in X_q.\ \langle x, x' \rangle <_q \langle y, y' \rangle$. By definition of $<_{p'}$ then $x <_{p'} y$.

If p' have the P_{or}-property it then follows that $\mathbf{p'} \in \delta_{or}(\mathbf{p})$.

Assume that p' does not have the P_{or}-property. That is $X_{p'}$ contain elements x_1, x_2, y_1, y_2 such that:

$$(8.5)\quad \begin{array}{l} x_1 <_{p'} y_1 \\ co_{p'} \\ x_2 <_{p'} y_2 \end{array} \qquad\qquad (8.6)\quad \begin{array}{l} x_1 \not<_{p'} y_2 \\ \textit{and} \\ x_2 \not<_{p'} y_1 \end{array}$$

From the definition of p', the ε-freeness of ϱ and (8.6) it then follows that there exists x_1', x_2', y_1', y_2' such that:

$$(8.7)\quad \begin{array}{l} \langle x_1, x_1' \rangle \not<_q \langle y_2, y_2' \rangle \\ \textit{and} \\ \langle x_2, x_2' \rangle \not<_q \langle y_1, y_1' \rangle \end{array}$$

From (8.5) then:

$$(8.8)\quad \begin{array}{l} \langle x_1, x_1' \rangle <_q \langle y_1, y_1' \rangle \\ \\ \langle x_2, x_2' \rangle <_q \langle y_2, y_2' \rangle \end{array}$$

But from (8.7) and (8.8) it follows that:

$$\langle x_1, x_1' \rangle\ co_q\ \langle x_2, x_2' \rangle$$

and we have a contradiction to the fact that q has the P_{or}-property.

It remains to prove $\mathbf{q} \preceq [p'{<}\pi_{p'}{>}]$ for some ϱ-consistent p.ref., $\pi_{p'}$, for p'. Since $X_p = X_{p'}$, π_p is also a p.ref. for p' and we know that it is ϱ-consistent. For the same reason $X_{p'<\pi_p>} = X_{p<\pi_p>} = X_q$ and similarly $\ell_{p'<\pi_p>} = \ell_q$.

Next we show $\leq_q \supseteq \leq_{p'<\pi_p>}$. Assume $\langle x, x' \rangle \leq_{p'<\pi_p>} \langle y, y' \rangle$. By construction of $p'{<}\pi_p{>}$ this implies $x <_{p'} y$ or $(x = y, x' \leq_{\pi_p(x)} y')$. In the former case (8.4) directly gives $\langle x, x' \rangle <_q \langle y, y' \rangle$ and in the latter case we have $\langle x, x' \rangle <_{p<\pi_p>} \langle x, y' \rangle$ from the construction of $p{<}\pi_p{>}$. Since $\leq_q \supseteq \leq_{p<\pi_p>}$ this implies $\langle x, x' \rangle <_q \langle x, y' \rangle$. Hence $\leq_q \supseteq \leq_{p'<\pi_p>}$.
Collecting the facts we can use proposition 6.2.3 again to conclude $\mathbf{q} \preceq [p'{<}\pi_p{>}]$ as desired. □

Lemma 8.2.7 $\wp(E\{\varrho\}) = (\wp(E)){<}\wp(\varrho){>}$ for $E \in BL$.

Proof By induction on the structure of E.

$E = a$: $\wp(a\{\varrho\}) = \wp(\varrho(a)) = (\wp(\varrho))(a) = \{a\}\lt\wp(\varrho)\gt = \wp(a)\lt\wp(\varrho)\gt$.

$$\begin{array}{lll} E = E_0 \,;\, E_1\text{:} & \wp(E\{\varrho\}) = \wp(E_0\{\varrho\}) \cdot \wp(E_1\{\varrho\}) & \text{definition of } \wp \text{ and } \{\varrho\} \\ & = (\wp(E_0)\lt\wp(\varrho)\gt) \cdot (\wp(E_1)\lt\wp(\varrho)\gt) & \text{hypothesis} \\ & = (\wp(E_0) \cdot \wp(E_1))\lt\wp(\varrho)\gt & \text{proposition 6.3.3} \\ & = \wp(E)\lt\wp(\varrho)\gt & \text{definition of } \wp \end{array}$$

$E = E_0 \oplus E_1$ and $E = E_0 \parallel E_1$: Similar.

□

8.3 Full Abstractness

The connection between $[\![\_]\!]_{or}$ and $[\![\_]\!]_{\mathsf{W}}$ is indicated by:

Proposition 8.3.1 $[\![E]\!]_{\mathsf{W}} = \delta_{\mathsf{W}}([\![E]\!]_{or})$ for $E \in RBL$.

Proof
$$\begin{array}{lll} [\![E]\!]_{\mathsf{W}} & = \delta_{\mathsf{W}}(\wp(E\sigma)) & \text{definition} \\ & = \delta_{\mathsf{W}}(\delta_{or}(\wp(E\sigma))) & \text{since } \delta_{\mathsf{W}} \circ \delta_{or} = \delta_{\mathsf{W}} \\ & = \delta_{\mathsf{W}}([\![E]\!]_{or}) & \text{by definition} \end{array}$$
□

Furthermore:

Proposition 8.3.2 $\trianglelefteq_{or}$ is a precongruence.

Proof Similar to proposition 7.4.1, but with the additional case of $\lt\varrho\gt$, which also is $\subseteq$-monotone (proposition 6.3.4). □

And in fact:

Theorem 8.3.3 The denotation $[\![\_]\!]_{or}$ is fully abstract w.r.t. $\lesssim^c_{\mathsf{W}}$ on RBL.

Proof We show that $\trianglelefteq_{or}$ is the largest precongruence contained in $\lesssim_{\mathsf{W}}$ or equivalently the largest precongruence contained in $\trianglelefteq_{\mathsf{W}}$.

By proposition 8.3.2 $\trianglelefteq_{or}$ is a precongruence and the containment is seen as follows:
$$\begin{array}{lll} E_0 \trianglelefteq_{or} E_1 & \Rightarrow [\![E_0]\!]_{or} \subseteq [\![E_1]\!]_{or} & \text{by definition} \\ & \Rightarrow \delta_{\mathsf{W}}([\![E_0]\!]_{or}) \subseteq \delta_{\mathsf{W}}([\![E_1]\!]_{or}) & \delta_{\mathsf{W}} \text{ is } \subseteq\text{-monotone} \\ & \Rightarrow [\![E_0]\!]_{\mathsf{W}} \subseteq [\![E_1]\!]_{\mathsf{W}} & \text{proposition 8.3.1} \\ & \Rightarrow E_0 \trianglelefteq_{\mathsf{W}} E_1 & \text{by definition} \end{array}$$

To show that $\trianglelefteq_{or}$ is the largest precongruence contained in $\trianglelefteq_{\mathsf{W}}$ is harder, so we have deferred the crux of the matter to lemma 8.3.4 below, from which we see $E_0 \ntrianglelefteq_{or} E_1$ implies that there exists a BL-refinement ϱ such that $E_0[\varrho] \ntrianglelefteq_{\mathsf{W}} E_1[\varrho]$. This means that any preorder contained in $\trianglelefteq_{\mathsf{W}}$ larger than $\trianglelefteq_{or}$ would not be a precongruence w.r.t. this combinator: $[\varrho]$. □

In the following we will need a special type of refinements—*fission* refinements—which splits an atomic action into two. In this way the original action is no longer atomic. Our notation is inspired by Hennessy [Hen87b].

For each $k \in I\!N^+$ we shall in the sequel denote the "standard" finite set, $\{1, \ldots, k\}$, by $\underline{k}$. Now let a finite multiplicity function, m, be given. Because m is finite we can define $n(m) = \max\{k \mid k = 1 \text{ or } \exists a \in \Delta.\, m(a) = k\} \in I\!N^+$.
Since Δ is infinite, but countable, there exists an injective function $h : \Delta \times \{S, F\} \times \underline{n(m)} \longrightarrow \Delta$. I.e., $\forall a, a' \in \Delta \forall i, i' \in \{S, F\} \forall k, k' \in \underline{n(m)}$.

(8.9) $$\langle a, i, k\rangle \neq \langle a', i', k'\rangle \Rightarrow h(\langle a, i, k\rangle) \neq h(\langle a', i', k'\rangle)$$

For convenience we shall abbreviate $h(\langle a, i, k\rangle)$ by a_{i_k}.

With such a function we associate a BL-refinement, ϱ, by defining for all $a \in \Delta$:

$$\varrho(a) = a_{S_1} \,;\, a_{F_1} \oplus \ldots \oplus a_{S_{n(m)}} \,;\, a_{F_{n(m)}}$$

and call it a m-*fission* refinement.
The corresponding ε-free $\mathcal{P}(\mathbf{P})$-refinement, (ambiguously denoted) ϱ, has

$$\varrho(a) = \{a_{S_1} \cdot a_{F_1}, \ldots, a_{S_{n(m)}} \cdot a_{F_{n(m)}}\}$$

and is also called a m-*fission* refinement.

We shall refer to a_{S_k} and a_{F_k} as a fission *pair* of the m-fission refinement ϱ. I.e., the pair a_{S_k} and a_{F_k} is a fission of a.

If π_p is a ϱ-consistent p.ref. for a lpo p we can define two functions $\text{-}^{\pi_p}_{S}, \text{-}^{\pi_p}_{F} : X_p \longrightarrow X_{p<\pi_p>}$ as follows: $x^{\pi_p}_S$ (respectively $x^{\pi_p}_F$) is that element $\langle x, x'\rangle$ where $x' \in X_{\pi_p(x)}$ and $\ell_{\pi_p(x)}(x') = a_{S_k}$ (respectively a_{F_k}) for some $k \in \underline{n(m)}$, $a = \ell_p(x)$. We will drop the superscript, π_p, when it is clear from the context. Due to the construction of $p<\pi_p>$ and the definition of ϱ from h (fulfilling (8.9)) we have:

(8.10) $x_S = y_S \Leftrightarrow x = y \Leftrightarrow x_F = y_F$

(8.11) $\ell_{p<\pi_p>}(x_S) = a_{S_k} \Leftrightarrow \ell_{p<\pi_p>}(x_F) = a_{F_k}$

(8.12) $\ell_p(x) = a \Leftrightarrow \exists k \in \underline{n(m)}.\, \ell_{p<\pi_p>}(x_S) = a_{S_k}$

(8.13) $\ell_{p<\pi_p>}(x) = a_{S_k} \Rightarrow \exists y \in X_p.\, y_S = x$

(8.14) $\ell_{p<\pi_p>}(x_S) = b \Rightarrow \exists a \in \Delta, k \in \underline{n(m)}.\, b = a_{S_k}$(namely: $a = \ell_p(x)$)

(8.15) $x <_p y \Rightarrow x_F <_{p<\pi_p>} y_S$

(8.16) $x_S <_{p<\pi_p>} x_F$

Suppose p is a lpo with $m_p \leq m$ (i.e $\forall a \in \Delta.\, m_p(a) \leq m(a)$). Then there clearly are ϱ-consistent p. refinements, π_p, injective in the sense:

$$\forall x, y \in X_p.\, x \neq y \Rightarrow [\pi_p(x)] \neq [\pi_p(y)]$$

We call such a π_p for a ϱ-consistent particular *fission* refinement for p

Notice that as a consequence of (8.9) and π_p being injective we have for $p{<}\pi_p{>}$:

$$\forall x, y \in X_{p<\pi_p>}.\ x = y \Leftrightarrow \ell_{p<\pi_p>}(x) = \ell_{p<\pi_p>}(y) \tag{8.17}$$

We say that a lpo q is *p-reflecting* under the p. fission ref. π_p if and only if any pair of concurrent elements from p have overlapping Start/ Finish (fission pairs) occurrences in q, formally: *iff* $q = \langle X_{p<\pi_p>}, \leq_q, \ell_{p<\pi_p>}\rangle, \leq_q \supseteq \leq_{p<\pi_p>}$ (so $\mathbf{q} \preceq [p{<}\pi_p{>}] \in \mathbf{p}{<}\varrho{>}$) and for all $x, y \in X_p$:

$$\text{if} \quad \begin{matrix} x \\ co_p \\ y \end{matrix} \quad \text{then} \quad \begin{matrix} x_S <_q y_F \\ \textit{and} \\ y_S <_q x_F \end{matrix}$$

With this notation we can then say for pomsets $\mathbf{q}'$ and $\mathbf{p}'$ that $\mathbf{q}'$ is $\mathbf{p}'$-*reflecting* under the fission refinement ϱ *iff* there are representatives p and q of $\mathbf{p}'$ and $\mathbf{q}'$ respectively together with a ϱ-consistent p. fission ref., π_p, such that q is p-reflecting under π_p

Lemma 8.3.4 Given $E_0 \in RBL$. Then there exists a refinement combinator, $[\varrho]$, such that

$$\forall E_1 \in RBL.\ [\![E_0]\!]_{or} \not\subseteq [\![E_1]\!]_{or} \Rightarrow [\![E_0[\varrho]]\!]_{\mathsf{W}} \not\subseteq [\![E_1[\varrho]]\!]_{\mathsf{W}}$$

Proof Let m be the finite multiplicity function which is the lub for $\{m_{\mathbf{p}} \mid \mathbf{p} \in [\![E_0]\!]_{or}\}$ (finite set). Choose a m-fission refinement ϱ. The associated refinement combinator, $[\varrho]$, is the one we are after. To see this let an arbitrary $E_1 \in RBL$ be given such that $[\![E_0]\!]_{or} \not\subseteq [\![E_1]\!]_{or}$. The proof is by contradiction. Assume on the contrary $[\![E_0[\varrho]]\!]_{\mathsf{W}} \subseteq [\![E_1[\varrho]]\!]_{\mathsf{W}}$. $[\![E_0]\!]_{or} \not\subseteq [\![E_1]\!]_{or}$ only if there is a $\mathbf{p} \in [\![E_0]\!]_{or}$ such that $\mathbf{p} \notin [\![E_1]\!]_{or}$. $\mathbf{p} \in [\![E_0]\!]_{or}$ implies $P_{or}(\mathbf{p})$ and by definition also $m_{\mathbf{p}} \leq m$. By lemma 8.3.6 there is a $w \in \delta_{\mathsf{W}}(\mathbf{p}{<}\varrho{>})$ which is $\mathbf{p}$-reflecting.

Now $w \in \delta_{\mathsf{W}}(\mathbf{p}{<}\varrho{>})$ and $\mathbf{p} \in [\![E_0]\!]_{or}$ implies w in $\delta_{\mathsf{W}}([\![E_0]\!]_{or}{<}\varrho{>})$ which, because $\delta_{\mathsf{W}} \circ \delta_{or} = \delta_{\mathsf{W}}$, equals $\delta_{\mathsf{W}}(\delta_{or}([\![E_0]\!]_{or}{<}\varrho{>}))$. By theorem 8.2.5 and proposition 8.3.1 then also $w \in [\![E_0[\varrho]]\!]_{\mathsf{W}}$ and so $w \in [\![E_1[\varrho]]\!]_{\mathsf{W}}$ by the assumption. Reversing the arguments we find a $\mathbf{q} \in [\![E_1]\!]_{or}$ such that w is a linearization of a pomset, $\mathbf{r}$, of $\mathbf{q}{<}\varrho{>}$. Because w is $\mathbf{p}$-reflecting we then deduce from lemma 8.3.5 that $\mathbf{p} \preceq \mathbf{q}$. Since $P_{or}(\mathbf{p})$ and $[\![E_1]\!]_{or}$ is δ_{or}-closed then $\mathbf{p} \in [\![E_1]\!]_{or}$—a contradiction. □

Lemma 8.3.5 Let a fission refinement ϱ be given and suppose $\mathbf{w}'$ is $\mathbf{p}'$-reflecting. If $\mathbf{w}' \preceq \mathbf{r} \in \mathbf{q}{<}\varrho{>}$ then $\mathbf{p}' \preceq \mathbf{q}$.

Proof To see $\mathbf{p}' \preceq \mathbf{q}$ we at first elucidate the situation. $\mathbf{w}'$ being $\mathbf{p}'$-reflecting implies there are representatives w of $\mathbf{w}'$ and p of $\mathbf{p}'$ together with a ϱ-consistent p. fission ref., π_p, such that

$$w = \langle X_{p<\pi_p>}, \leq_w, \ell_{p<\pi_p>}\rangle, \leq_w \supseteq \leq_{p<\pi_p>}$$

We also have $\mathbf{w}' \preceq \mathbf{r} \in \mathbf{q}{<}\varrho{>}$ Therefore there is a ϱ-consistent p.ref., π_q, and a morphism of lpos $f : q{<}\pi_q{>} \longrightarrow w$.

We shall find a morphism $g : q \longrightarrow p$. Define

$$g(x) = y \ \textit{iff}\ \exists y \in X_p.\ y_S^{\pi_p} = f(x_S^{\pi_q}) \tag{8.18}$$

This gives sense since $X_q \overset{\pi_q}{\underset{S}{\longrightarrow}} X_{q<\pi_q>} \overset{f}{\longrightarrow} X_w = X_{p<\pi_p>} \overset{\pi_p}{\underset{S}{\longleftarrow}} X_p$.

To see that (8.18) actually defines a function $g : X_q \longrightarrow X_p$ we shall prove that there for a given $x \in X_q$ is one and only one $y \in X_p$ such that $y_S = f(x_S)$. At first we notice from f being label preserving and $\ell_w = \ell_{p<\pi_p>}$ that:

$$(8.19) \qquad \forall z \in X_{q<\pi_q>}.\ \ell_{q<\pi_q>}(z) = \ell_{p<\pi_p>}(f(z))$$

one: π_q is ϱ-consistent, so by (8.14) $\ell_{q<\pi_q>}(x_S) = a_{S_k}$ for some a and k. From (8.19) then $\ell_{p<\pi_p>}(f(x_S)) = a_{S_k}$ and by (8.13) there exists a $y \in X_p$ with $y_S = f(x_S)$.
only one: Follows directly from (8.10).

Before continuing we prove

$$(8.20) \qquad f(x_S) = g(x)_S, \qquad f(x_F) = g(x)_F$$

The first equation holds by definition of g and the second is seen from the first as follows: $f(x_S) = g(x)_S$ implies $\ell_{p<\pi_p>}(f(x_S)) = \ell_{p<\pi_p>}(g(x)_S)$ which by (8.19) is the same as $\ell_{q<\pi_q>}(x_S) = \ell_{p<\pi_p>}(g(x)_S)$. Because π_p and π_q both are ϱ-consistent we from (8.14) get $\ell_{q<\pi_q>}(x_S) = a_{S_k} = \ell_{p<\pi_p>}(g(x)_S)$ for some a and k, so by (8.11) and (8.19) then $\ell_{p<\pi_p>}(f(x_F)) = \ell_{p<\pi_p>}(g(x)_F)$. Now π_p is also a ϱ-consistent p. fission ref. for p, so we conclude $f(x_F) = g(x)_F$ from (8.17).

As the next step we show g to be bijective.

g injective:
$$\begin{aligned} x \neq y &\Rightarrow x_S \neq y_S && \text{by (8.10)} \\ &\Rightarrow f(x_S) \neq f(y_S) && f \text{ injective} \\ &\Rightarrow g(x)_S \neq g(y)_S && \text{by (8.20)} \\ &\Rightarrow g(x) \neq g(y) && \text{by (8.10)} \end{aligned}$$

g surjective: Given $y \in X_p$. By (8.14) then $\ell_{p<\pi_p>}(y_S) = a_{S_k}$ for some a and k. Since f is surjective and label preserving there is an $x' \in X_{q<\pi_q>}$ with $f(x') = y_S$ and $\ell_{q<\pi_q>}(x') = a_{S_k}$. From (8.13) we see that there must be an $x \in X_q$ with $x_S = x'$.

In proving g to be a morphism of lpos it remains to show that g is label and order preserving.

g label preserving: Suppose $x \in X_q$ and $\ell_q(x) = b$. Then from (8.12) $\ell_{q<\pi_q>}(x_S) = b_{S_k}$ for some k, and therefore $b_{S_k} = \ell_{p<\pi_p>}(f(x_S)) = \ell_{p<\pi_p>}(g(x)_S)$ by (8.19) and (8.20). Using (8.12) again we obtain $\ell_p(g(x)) = b = \ell_q(x)$.

g order preserving: Assume $x \leq_q y$. In the case $x = y$ the result follows from the reflexivity of $\leq_p$. In the case $x <_q y$ we have

$$(8.21) \qquad g(x)_F <_w g(y)_S$$

because
$$\begin{aligned} x <_q y &\Rightarrow x_F <_{q<\pi_q>} y_S && \text{by (8.15)} \\ &\Rightarrow f(x_F) <_w f(y_S) && f \text{ is order preserving} \\ &\Rightarrow g(x)_F <_w g(y)_S && \text{by (8.20)} \end{aligned}$$

We cannot have $g(y) <_p g(x)$ since it by (8.15) and (8.16) would imply $g(y)_S <_{p<\pi_p>} g(x)_F$ which in turn from $\leq_{p<\pi_p>} \subseteq \leq_w$ would imply $g(y)_S <_w g(x)_F$—contradicting (8.21). $g(x)\ co_p\ g(y)$ can also be excluded since we then from the fact that w is p-reflecting would get $g(y)_S <_w g(x)_F$—again contradicting (8.21). Hence we are left with $g(x) <_p g(y)$ as the only possibility and we are done. □

For a pomset $\mathbf{p}$, let in the sequel $M_p \subseteq X_p$ denote the set of minimal elements of p (w.r.t. $\leq_p$).

We state and prove the lemma referred to in the proof of lemma 8.3.4.

Lemma 8.3.6 Let $\mathbf{p}$ be a pomset with the P_{or}-property and $m_{\mathbf{p}} \leq m$, where m is some finite multiplicity function over Δ. Also let ϱ be a m-fission refinement. Then there exists a linearization w of $\mathbf{p}{<}\varrho{>}$ (i.e., $w \in \delta_{\mathsf{W}}(\mathbf{p}{<}\varrho{>})$) which is $\mathbf{p}$-reflecting under ϱ.

Proof If $\mathbf{p} = \varepsilon$ it is trivial that $w = \varepsilon$ will do, so we can assume $\mathbf{p} \neq \varepsilon$ in the following. Since $m_{\mathbf{p}} \leq m$ there is a ϱ-consistent p. fission ref., ${<}\pi_p{>}$, for p. The result is then a consequence of the corresponding statement for lpos:

> Let π_p be a p. fission ref. for $p \neq \varepsilon$. Assume the minimal elements M_p of p listed in some arbitrary order are: $x_1, \ldots, x_n$. Then there exists an p-reflecting linearization w of $p{<}\pi_p{>}$ isomorphic to a lpo of the form:
>
> $$x_{1S} \cdot \ldots \cdot x_{nS} \cdot v$$

The proof is by induction on the size of X_p.
The basis, X_p a singleton, is clear.
So assume $|X_p| > 1$. From proposition 8.3.7 we can find an element $x_i \in M_p$ such that x_i is dominated in X_p by all successors of M_p. Consider now the lpo, p', obtained by deleting x_i from p.

Notice that $M_p \setminus \{x_i\}$ is a subset of the minimal elements of p', hence we may list $M_{p'}$ as follows:

$$x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n, y_1, \ldots, y_k$$

Clearly $\pi_{p'} = \pi_p|_{X_{p'}}$ is a ϱ-consistent p. fission ref. for p', so because the P_{or} property is inherited to p' we can use the inductive hypothesis to find a p'-reflecting linearization w' of $p'{<}\pi_{p'}{>}$ isomorphic to a lpo of the form

$$x_{1S} \cdot \ldots \cdot x_{i-1S} \cdot x_{i+1S} \cdot \ldots \cdot x_{nS} \cdot y_{1S} \cdot \ldots \cdot y_{kS} \cdot v'$$

Since x_i is minimal in p there are no other elements before x_{iS} and x_{iF} in $p{<}\pi_p{>}$, and so $x_{iS} \cdot x_{iF} \cdot w'$ is isomorphic to a possible linearization of $p{<}\pi_p{>}$. By the way x_i was chosen, the elements concurrent to x_i are exactly $M_p \setminus \{x_i\}$. Then x_{iS} and x_{iF} are concurrent to $x_{1S}, \ldots, x_{i-1S}, x_{i+1S}, \ldots, x_{nS}$ in $p{<}\pi_p{>}$, from which it follows that

$$x_{1S} \cdot \ldots \cdot x_{iS} \cdot \ldots \cdot x_{nS} \cdot x_{iF} \cdot y_{1S} \cdot \ldots \cdot y_{kS} \cdot v'$$

must be isomorphic to a linearization, w, of $p{<}\pi_p{>}$, which quite easily is seen to be p-reflecting as desired. $\square$

Proposition 8.3.7 Let p be a nonempty lpo with the P_{or}-property and M a subset of the minimal elements M_p of p. Then there is an element z of M dominated by all the successors of M.

Proof By induction on the size of M. The basis where $M = \{z\}$ is evident, and for the inductive step choose an $x \in M$. By hypothesis of induction we can find a $y \in M \setminus \{x\}$, which is dominated by the successors of $M \setminus \{x\}$. If y is dominated by the successors of x too, we can choose $z = y$. Otherwise, since p has the P_{or}-property, and minimal elements are mutual concurrent, the successors of y must dominate x. But the successors of y are also the successors of $M \setminus \{x\}$ and we can choose $z = x$. □

8.4 Summary

Let us sum up the abstractness results we have proved in this chapter. If we let $\lessapprox\!\!\gtrsim$ denote the equivalence associated with an operational preorder $\lesssim$, and if we extend $[\![\_]\!]_{\mathsf{G}}$ to RBL in the same simple way as $[\![\_]\!]_{\mathsf{w}}$ were extended in section 8.2, we get the following immediate corollary:

Corollary 8.4.1 For all $E_0, E_1 \in RBL$:

$$\begin{array}{lcl} E_0 \lessapprox\!\!\gtrsim_{\mathsf{w}} E_1 & \textit{iff} & [\![E_0]\!]_{\mathsf{w}} = [\![E_1]\!]_{\mathsf{w}} \\ E_0 \lessapprox\!\!\gtrsim_{\mathsf{G}} E_1 & \textit{iff} & [\![E_0]\!]_{\mathsf{G}} = [\![E_1]\!]_{\mathsf{G}} \\ E_0 \lessapprox\!\!\gtrsim^c_{\mathsf{w}} E_1 & \textit{iff} & [\![E_0]\!]_{or} = [\![E_1]\!]_{or} \end{array}$$

It follows from $P_{\mathsf{M}} = P_{and}$, $P_{and}(\mathbf{p}) \Rightarrow P_{or}(\mathbf{p})$ and definitions that $[\![\_]\!]_{\mathsf{M}}$ is as abstract as $[\![\_]\!]_{or}$ on BL. The following two expressions:

$$\begin{array}{lcl} E_0 & = & a \,;\, b \parallel c \\ E_1 & = & (a \parallel c) \,;\, b \oplus a \,;\, (b \parallel c) \end{array}$$

show that $[\![_]\!]_{\mathsf{M}}$it is strictly more abstract than $[\![\_]\!]_{or}$ (identified by $[\![\_]\!]_{\mathsf{M}}$, but not by $[\![\_]\!]_{or}$).

Furthermore, from the full abstractness results, the fact that $P_{\mathsf{w}} \Rightarrow P_{\mathsf{G}} \Rightarrow P_{\mathsf{M}}(= P_{and}) \Rightarrow P_{or}$, and the examples in the summary from the last chapter we get:

Corollary 8.4.2 For all $E_0, E_1 \in RBL$:

$$(E_0 \lesssim^c_{\mathsf{w}} E_1) \Rightarrow (E_0 \lesssim_{\mathsf{G}} E_1) \Rightarrow (E_0 \lesssim_{\mathsf{w}} E_1)$$

and none of the implications hold in the other direction except of course in the last implication if G equals w.

At this stage, it seems very natural to ask what would have happened, if we had chosen to look for a denotational characterization of the RBL congruence associated with the different operational G-sequences rather than action-sequences, i.e., a characterization of $\lesssim^c_{\mathsf{G}}$ on RBL. Operationally it seems hard to say anything directly. However, from $\lesssim_{\mathsf{G}} \subseteq \lesssim_{\mathsf{w}}$ follows $\lesssim^c_{\mathsf{G}} \subseteq \lesssim^c_{\mathsf{w}}$ so $\lesssim^c_{\mathsf{w}}$ is at least as large as the largest precongruence contained in $\lesssim_{\mathsf{G}}$. $\lesssim^c_{\mathsf{w}}$ is a precongruence by definition and by $\lesssim^c_{\mathsf{w}} \subseteq \lesssim_{\mathsf{G}}$ it then follows that $\lesssim^c_{\mathsf{w}}$ is the largest precongruence contained in $\lesssim_{\mathsf{G}}$, i.e., $\lesssim^c_{\mathsf{G}} = \lesssim^c_{\mathsf{w}}$. From the last corollary but one we then obtain the full abstractness result:

Corollary 8.4.3 For all $E_0, E_1 \in RBL$:

$$\begin{array}{lcl} E_0 \lesssim^c_{\mathsf{G}} E_1 & \textit{iff} & E_0 \lesssim^c_{\mathsf{w}} E_1 \\ E_0 \lessapprox\!\!\gtrsim^c_{\mathsf{G}} E_1 & \textit{iff} & [\![E_0]\!]_{or} = [\![E_1]\!]_{or} \end{array}$$

8.5 An Adequate Logic for RBL without Auto-Parallelism

Along the lines of section 7.6 we would in this section for RBL like to find an adequate linear logic $\mathcal{L}^{\mathsf{r}}_{\mathsf{G}}$ for $\approx^{\mathsf{c}}_{\mathsf{G}}$. Unfortunately it seems insurmountable to devise such a logic for the full RBL language. We shall therefore confine ourselves to search for an adequate logic for the sublanguage, RBL', of RBL where processes have no auto-parallelism (see [vGV87]). That is the same action may not occur on both sides of a $\|$-combinator as in e.g., $a\,;b \parallel c\,;a$. Similar BL' will be the sublanguage of BL with no auto-parallelism.

Formally let $L(E)$ denote the sort/ label set/ set of actions of an $E \in BL$. Then BL' is those expressions of BL where all subexpressions of the form $E_0 \parallel E_1$ fulfills $L(E_0) \cap L(E_1) = \emptyset$. A BL'-refinement will be a mapping $\varrho : \Delta \longrightarrow BL'$ with the additional requirement:

$$a \neq b \Rightarrow L(\varrho(a)) \cap L(\varrho(b)) = \emptyset$$

RBL' is those $E \in RBL$ where $E\sigma \in BL'$ and if $[\varrho]$ is a refinement combinator of E then ϱ is a BL'-refinement. Due to the restrictions on the expressions of BL' and RBL' the congruence and full abstractness results to follow should be modified accordingly.

The decisive importance of BL' and RBL' is that the canonical pomset association map, $\wp$, when used on expressions of BL' yield pomsets with the P_{sw}-property (i.e., semiwords) we encountered in section 7.5. Recall that in a P_{sw}-pomset all equally labelled elements are ordered. Consequently one can speak of the i^{th} occurrence of a label $a \in \Delta$. The corresponding element of $\hat{p}$ will be $\langle i, a \rangle \in X_{\hat{p}}$, where $\hat{p}$ is the canonic representative of $\mathbf{p}$ we mentioned in section 7.5. For more details on this matter see Starke [Sta81]. From the results there a stronger version of the alternative characterization of $\preceq$ on P_{sw}-pomsets appears: If $\mathbf{p}, \mathbf{q} \in \mathbf{P}_{\mathsf{sw}}$ then

$$\mathbf{p} \preceq \mathbf{q} \;\; \textit{iff} \;\; X_{\hat{p}} = X_{\hat{q}} \;\; \textit{and} \; \leq_{\hat{p}} \supseteq \leq_{\hat{q}}$$

Another of the P_{sw}-pomsets characteristics is that:

if $\mathbf{p} \in \mathbf{P}$, $\mathbf{q} \in \mathbf{P}_{\mathsf{sw}}$ and $\mathbf{p} \preceq \mathbf{q}$ then $\mathbf{p} \in \mathbf{P}_{\mathsf{sw}}$.

Combining this with the fact that $\wp(E) \in \mathbf{P}_{\mathsf{sw}}$ when $E \in BL'$ we see that the denotational maps of the different models when restricted to BL' respectively RBL' only yield P_{sw}-pomsets. We can therefore choose to work with canonic representatives in stead.

To this end denote the set of lpos which are the canonic representatives of some P_{sw}-pomset by SW and call SW the set of semiwords (over Δ). For $p, q \in SW$ the partial order then becomes:

$$p \preceq q \;\; \textit{iff} \;\; X_p = X_q \;\; \textit{and} \; \leq_p \supseteq \leq_q$$

and the pomset operations inherits to SW via the canonic representatives. E.g., $p \times q := \widehat{p \times q}$ where the $\times$-operator under $\hat{}$ is the lpo parallel composition introduced in section 6.1. In order to insure this to be well-defined p and q must be disjoint. This will be assumed henceforth when writting $p \times q$. Because as we noticed in section 7.5 the P_{sw}-property is both hereditary and dot synthesizable it follows that all the purely denotational results carry over to BL' and RBL' (see also section 6.4).

On the operational side nothing changes except that CL and RCL are changed accordingly. But due to the nature of the process expressions we now focus on there is no point in regarding set of direct tests larger than $\mathsf{S} \subseteq \mathbf{P}_{\mathsf{sw}}$. We shall therefore assume

$$\Delta \subseteq \mathsf{G} \subseteq \mathsf{S}$$

The full abstractness results of course holds for BL'. It is however not so obvious that $\lesssim^c_{\mathsf{G}}$ will be fully abstract with the semiword version of $[\![\_]\!]_{or}$. That this is the case can be seen by passing through section 8.3 with semiwords in mind and observing that the BL-refinement used in lemma 8.3.4 actually is a BL'-refinement. The result for $\lesssim^c_{\mathsf{W}}$ and $\lesssim_{\mathsf{S}}$ is also reported in [NEL89]. There a simpler BL' refinement without $\oplus$ is used in the lemma corresponding to lemma 8.3.4 (this only works for semiwords—see the conclusion of that paper). Furthermore direct definitions of the different semiword operations is given—especially the definition of the refinement operator is not strait forward.

We will now introduce the linear logics. For BL' and $\lesssim_{\mathsf{G}}$ we can use the logic $\mathcal{L}_{\mathsf{G}}$ from section 7.6. Since $\mathsf{G} \subseteq \mathsf{S}$ it will do with modalities $\langle A\rangle$ and $\boxed{A}$ where $A \in \mathsf{S}$.

For RBL' a stronger modal language is needed. We shall also denote this language by $\mathcal{L}$ and define it to be the formulae obtained from:

$$f ::= \mathrm{tt} \mid \mathrm{ff} \mid \triangledown \mid \vartriangle \mid \overset{\frown}{A} f \mid \underset{\smile}{A} f \mid \langle A\rangle f \mid \boxed{A} f$$

where A can be any element of S.
$\mathcal{L}_{\mathsf{r}} \subseteq \mathcal{L}$ is defined to be those formulae with no occurrence of the modalities $\overset{\frown}{A}$ and $\langle A\rangle$. For each G the satisfaction relation is as in section 7.6 except for the modalities $\overset{\frown}{A}$ and $\underset{\smile}{A}$. The intuition behind $\underset{\smile}{A}$ is a kind of semi-deadlock. I.e., a process satisfies $\underset{\smile}{A} f$ if it *either* is what Stirling [Sti85] calls a-deadlocked for some $a \in A$ *or* it satisfies f. Dually a process satisfies $\overset{\frown}{A} f$ if it is able to perform every $a \in A$ and it also satisfies f. Formally:

Definition 8.5.1 $\models_{\mathsf{G}} \subseteq DCL' \times \mathcal{L}$ is defined:

$$
\begin{array}{lll}
E \models_{\mathsf{G}} \mathrm{tt} & \text{for all } E \in DCL' \\
E \models_{\mathsf{G}} \triangledown & \textit{iff } \forall a \in \Delta.\, E \stackrel{a}{\not\Rightarrow}_{\mathsf{G}} \\
E \models_{\mathsf{G}} \vartriangle & \textit{iff } \exists a \in \Delta.\, E \stackrel{a}{\Rightarrow}_{\mathsf{G}} \\
E \models_{\mathsf{G}} \overset{\frown}{A} f & \textit{iff } A \subseteq \{a \in \Delta \mid E \stackrel{a}{\Rightarrow}_{\mathsf{G}}\} \textit{ and } E \models_{\mathsf{G}} f \\
E \models_{\mathsf{G}} \underset{\smile}{A} f & \textit{iff } (\exists a \in A.\, E \stackrel{a}{\not\Rightarrow}_{\mathsf{G}}) \textit{ or } E \models_{\mathsf{G}} f \\
E \models_{\mathsf{G}} \langle A\rangle f & \textit{iff } \exists E'.\, E \stackrel{A}{\Rightarrow}_{\mathsf{G}} E' \textit{ and } E' \models_{\mathsf{G}} f \\
E \models_{\mathsf{G}} \boxed{A} f & \textit{iff } \forall E'.\, E \stackrel{A}{\Rightarrow}_{\mathsf{G}} E' \textit{ implies } E' \models_{\mathsf{G}} f
\end{array}
$$

□

As in section 7.6 we say that a process $E' \in BL'$ satisfies a formula $f \in \mathcal{L}_{\mathsf{r}}$,

$$E \models_{\mathsf{G}} f \ \textit{iff} \ \forall E' \in \mathrm{Beh}(E).\, E' \models_{\mathsf{G}} f$$

With the syntactic substitution $\sigma : RBL' \longrightarrow BL'$ it is then possible to extend $\models_{\mathsf{G}}$ further to RBL' as follows: $E \in RBL'$ satisfies a formula $f \in \mathcal{L}_{\mathsf{r}}$,

$$E \models_{\mathsf{G}} f \textit{ iff } E\sigma \models_{\mathsf{G}} f$$

For $E \in RBL'$ we define:

$$\mathcal{L}^{\mathsf{r}}_{\mathsf{G}}(E) = \{f \in \mathcal{L}_{\mathsf{r}} \mid E \models_{\mathsf{G}} f\}$$

and in order to prove the adequacy of $\mathcal{L}^{\mathsf{r}}_{\mathsf{G}}$ w.r.t. $\approxeq^{c}_{\mathsf{G}}$ on RBL' we shall also introduce for $E \in RBL'$:

$$\overline{\mathcal{L}}^{\mathsf{r}}_{\mathsf{G}}(E) = \{f \in \overline{\mathcal{L}}_{\mathsf{r}} \mid \exists E' \in \mathrm{Beh}(E\sigma).\, E' \models_{\mathsf{G}} f\}$$

where $\overline{\mathcal{L}}_{\mathsf{r}}$ is $\{\overline{f} \in \mathcal{L} \mid f \in \mathcal{L}_{\mathsf{r}}\}$ and $\overline{\_} : \mathcal{L} \longrightarrow \mathcal{L}$ is the syntactic map of section 7.6 extended to this larger formula language by:

$$\overline{\boxed{A} f} = \textcircled{A}\overline{f} \qquad\qquad \overline{\textcircled{A} f} = \boxed{A}\overline{f}$$

That is the formulae of $\overline{\mathcal{L}}_{\mathsf{r}}$ are:

$$f ::= tt \mid ff \mid \triangledown \mid \triangle \mid \boxed{A} f \mid \Diamond\!\!\!\!A\ f$$

Similarly as we proved lemma 7.6.2 we here for $E_0, E_1 \in RBL'$ get:

(8.22) $$\overline{\mathcal{L}}^{\mathsf{r}}_{\mathsf{G}}(E_0) \subseteq \overline{\mathcal{L}}^{\mathsf{r}}_{\mathsf{G}}(E_1) \textit{ iff } \mathcal{L}^{\mathsf{r}}_{\mathsf{G}}(E_0) \supseteq \mathcal{L}^{\mathsf{r}}_{\mathsf{G}}(E_1)$$

We are now ready to give the theorem from which that adequacy of $\mathcal{L}^{\mathsf{r}}_{\mathsf{G}}$ follows:

Theorem 8.5.2 (Linear Logic Characterization) For all $E_0, E_1 \in RBL'$:

$$E_0 \lesssim^{c}_{\mathsf{G}} E_1 \textit{ iff } \mathcal{L}^{\mathsf{r}}_{\mathsf{G}}(E_0) \supseteq \mathcal{L}^{\mathsf{r}}_{\mathsf{G}}(E_1)$$

Proof Immediate from (8.22) and lemma 8.5.17 at the end of the section which states:

(8.23) $$\overline{\mathcal{L}}^{\mathsf{r}}_{\mathsf{G}}(E_0) \subseteq \overline{\mathcal{L}}^{\mathsf{r}}_{\mathsf{G}}(E_1) \textit{ iff } E_0 \lesssim^{c}_{\mathsf{G}} E_1$$

□

Lemma 7.6.4 corresponding to (8.23) for BL was proved through operational argumentation. This is not so easely done here, but if we introduce a satisfaction relation based on semiwords we can utilize our knowledge of the models characterizing $\lesssim^{c}_{\mathsf{G}}$. For this purpose we introduce some additional concepts and conventions for semiwords.

As for the proof of lemma 8.3.6 M_p will for $p \in SW$ denote the minimal elements of X_p w.r.t. $\leq_p$. Suppose $a \in \Delta$ is the label of an element of X_p. Due to the nature of $p \in SW$ the first occurrence of an element of X_p labelled with a will then be $\langle 1, a\rangle \in M_p$. That is $\langle 1, a\rangle$ is the unique element of M_p labelled a (see page 190). For $A \subseteq \Delta$ or equally $A \in \mathsf{S}$ we can therefore make the convention to identify A with the uniquely determined set $\{\langle 1, a\rangle \mid a \in A\}$ such that it is sensible to write e.g., $A \subseteq M_p$ (or $a \in M_p$ for that matter). Obviously we then have:

Corollary 8.5.3

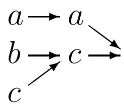
a) $M_p = \emptyset$ *iff* $p = \varepsilon$

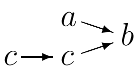
b) $M_a = \{a\}$ $(= a)$

c) $M_{p \cdot q} = \begin{cases} M_p & \text{if } p \neq \varepsilon \\ M_q & \text{if } p = \varepsilon \end{cases}$

d) $M_{p \times q} = M_p \cup M_q$

With the conventions we can if $A \subseteq M_p$ define the *complement semiword*, $\overline{A}^p$, of A in p to be the semiword $\hat{q}$ where q is $p|_{(X_p \setminus A)}$. The construction of $\overline{A}^p$ could of course be done directly from A and p.

Example: Suppose $A = \{a, b\}$ and $p =$  . Then $M_p = \{a, b, c\}$ and $\overline{A}^p =$



On second thoughts one realize the truth of

Corollary 8.5.4

a) $\overline{\varepsilon}^p = p$, $\overline{p}^p = \varepsilon$

b) $A \subseteq M_p$ implies $\overline{A}^{p \cdot q} = \overline{A}^p \cdot q$

c) $A \subseteq M_p$, $B \subseteq M_q$ and p disjoint to q implies
$A \cup B \subseteq M_{p \times q}$ and $\overline{A \times B}^{p \times q} = \overline{A}^p \times \overline{B}^q$

It is also easy to observe that $A \subseteq M_p$ and $A \subseteq M_q$ implies:

(8.24) $\quad \overline{A}^p \preceq \overline{A}^q$ *iff* $X_p \setminus A = X_q \setminus A$ *and* $\leq_p|_{(X_p \setminus A)^2} \supseteq \leq_q|_{(X_q \setminus A)^2}$

Proposition 8.5.5

a) $p \preceq q \Rightarrow M_p \subseteq M_q$

b) $p \preceq q$ and $A \subseteq M_p$ implies $\overline{A}^p \preceq \overline{A}^q$

Proof
a) Assume on the contrary that there exists a $x \in M_p$ such that $x \notin M_q$. Since $X_p = X_q$ and $x \in M_p$ we have $x \in X_q$. Hence $x \notin M_q$ implies there is a $y \in X_q$ with $y <_q x$. But then from $\leq_p \supseteq \leq_q$ also $y <_p x$—a contradiction to $x \in M_p$.

b) From a) and the hypothesis of the implication we see $A \subseteq M_q$. Hence $\overline{A}^q$ is well-defined. $p \preceq q$ gives us $X_p = X_q$ and $\leq_p \supseteq \leq_q$. b) is then immediate from (8.24). □

With the ability to regard elements of S as minimal elements of a semiword and with the notion of complement semiwords we can define the semiword satisfaction relation $\models^{\mathsf{sw}}_{\mathsf{G}}$:

Definition 8.5.6 $\models^{\mathsf{sw}}_{\mathsf{G}} \subseteq SW \times \overline{\mathcal{L}}_{\mathsf{r}}$ is defined inductively by:

$$\begin{array}{lll} p \models^{\mathsf{sw}}_{\mathsf{G}} \mathrm{tt} & \text{for all } p \in SW \\ p \models^{\mathsf{sw}}_{\mathsf{G}} \bigtriangledown & \textit{iff } p = \varepsilon \\ p \models^{\mathsf{sw}}_{\mathsf{G}} \bigtriangleup & \textit{iff } p \neq \varepsilon \\ p \models^{\mathsf{sw}}_{\mathsf{G}} \boxed{A} f & \textit{iff } A \subseteq M_p \textit{ and } p \models^{\mathsf{sw}}_{\mathsf{G}} f \\ p \models^{\mathsf{sw}}_{\mathsf{G}} \langle A \rangle f & \textit{iff } A \in \mathsf{G}, A \subseteq M_p \textit{ and } \overline{A}^p \models^{\mathsf{sw}}_{\mathsf{G}} f \end{array}$$

□

$\models^{\mathsf{sw}}_{\mathsf{G}}$ is extended to $\mathcal{P}(SW) \times \overline{\mathcal{L}}_{\mathsf{r}}$ by letting:

$$P \models^{\mathsf{sw}}_{\mathsf{G}} f \textit{ iff } \exists p \in P.\, p \models^{\mathsf{sw}}_{\mathsf{G}} f$$

At first we want to establish a connection between the operational based satisfaction relation $\models_{\mathsf{G}}$ (restricted to $RBL' \times \overline{\mathcal{L}}_{\mathsf{r}}$) and the semiword based $\models^{\mathsf{sw}}_{\mathsf{G}}$. $\wp$ is therefore extended to CL' by keeping it's compositional definition, but adding $\wp(\dagger) = \varepsilon$ for the extinct action. Naturally $\wp(E) = \{p\}$ when $E \in BL'$ so we shall often identify $\wp(E)$ with p in such situations. To get the connection some lemmas are needed.

Lemma 8.5.7 Suppose $E, E' \in DCL'$. Then:

a) $E \succ\!\!\longrightarrow^* E'$ implies $\wp(E) = \wp(E')$

b) $\wp(E) = \varepsilon$ *iff* $E \succ\!\!\longrightarrow^* \dagger$

c) $E \models_{\mathsf{G}} \bigtriangledown$ *iff* $E \succ\!\!\longrightarrow^* \dagger$

d) $\{a \in \Delta \mid E \overset{a}{\Rightarrow}_{\mathsf{G}}\} = M_{\wp(E)}$

Proof

a) Induction in the structure of E using the fact that ε is neutral to $\cdot$ and $\times$ on SW.

b) The *only if* part follows by a trivial induction on the structure of E and the *if* part is just a special case of a)

c) (7.9) on page 170.

d) By induction on the structure of E.

$E = \dagger$: $\{a \in \Delta \mid \dagger \overset{a}{\Rightarrow}_{\mathsf{G}}\} = \emptyset = M_\varepsilon = M_{\wp(E)}$.

$E = b$: $\{a \in \Delta \mid E \overset{a}{\Rightarrow}_{\mathsf{G}}\} = \{b\} = M_b = M_{\wp(E)}$.

$E = E_0\ ; E_1$: We consider two subcases:

$E_0 \not\rightarrowtail^* \dagger$: From lemma 7.4.5 and proposition 7.2.3 it then follows that $\{a \in \Delta \mid E \overset{a}{\Rightarrow}_{\mathsf{G}}\} = \{a \in \Delta \mid E_0 \overset{a}{\Rightarrow}_{\mathsf{G}}\}$ which by hypothesis of induction equals $M_{\wp(E_0)}$. Now from b) $E_0 \not\rightarrowtail^* \dagger$ implies $\wp(E_0) \neq \varepsilon$, so corollary 8.5.3 gives $M_{\wp(E_0)} = M_{\wp(E_0)\cdot\wp(E_1)}$. By definition of $\wp$ then also $M_{\wp(E_0)} = M_{\wp(E)}$.

$E_0 \rightarrowtail^* \dagger$: Similar we see $\{a \in \Delta \mid E \overset{a}{\Rightarrow}_{\mathsf{G}}\} = \{a \in \Delta \mid E_1 \overset{a}{\Rightarrow}_{\mathsf{G}}\}$ and $\wp(E_0) = \varepsilon$, so the result follows in the same way but using the hypothesis on E_1 instead.

$E = E_0 \parallel E_1$: This time we see $\{a \in \Delta \mid E \overset{a}{\Rightarrow}_{\mathsf{G}}\} = \{a \in \Delta \mid E_0 \overset{a}{\Rightarrow}_{\mathsf{G}}\} \cup \{a \in \Delta \mid E_1 \overset{a}{\Rightarrow}_{\mathsf{G}}\}$ from lemma 7.4.6 and proposition 7.2.3. The rest then follows along the lines above.

□

Lemma 8.5.8 For $E \in DCL'$:

a) $E \overset{A}{\Rightarrow}_{\mathsf{G}} E'$ implies $A \subseteq M_{\wp(E)}$ and $\overline{A}^{\wp(E)} = \wp(E')$

b) $A \subseteq M_{\wp(E)}$ and $A \in \mathsf{G}$ implies $\exists E'.\, E \overset{A}{\Rightarrow}_{\mathsf{G}} E'$

Proof
a) $E \overset{A}{\Rightarrow}_{\mathsf{G}} E'$ only if there are F and F' such that $E \rightarrowtail^* F \xrightarrow{A} F' \rightarrowtail^* E'$, so from a) of the preceding lemma we see it is enough to prove

$$E \xrightarrow{A}_{\mathsf{G}} E' \text{ implies } A \subseteq M_{\wp(E)} \text{ and } \overline{A}^{\wp(E)} = \wp(E')$$

By induction of the size of $E \xrightarrow{A} E'$ one easely shows $a \in A$ implies $E \xrightarrow{a}_{\mathsf{G}}$, so $A \subseteq \{a \in \Delta \mid E \overset{a}{\Rightarrow}_{\mathsf{G}}\}$. By d) of the preceding lemma therefore $A \subseteq M_{\wp(p)}$.
We now know that $\overline{A}^{\wp(E)}$ is well-defined and it make sense to prove

$$E \xrightarrow{A}_{\mathsf{G}} E' \text{ implies } \overline{A}^{\wp(E)} = \wp(E')$$

by induction on the size, m, of $E \xrightarrow{A}_{\mathsf{G}_m} E'$. Only the inductive step is interesting. We consider the different rules one by one.

$E = a \xrightarrow{A}_{\mathsf{G}_{m+1}} E'$: Clearly $A = a$ and $E' = \dagger$. Now $\overline{a}^a = \varepsilon$ and $\wp(a) = a$, so $\overline{a}^{\wp(E)} = \varepsilon = \wp(E')$.

$E = E_0 \,;\, E_1 \xrightarrow{A}_{\mathsf{G}_{m+1}} E_0' \,;\, E_1 = E'$ where $E_0 \xrightarrow{A}_{\mathsf{G}_m} E_0'$: By induction $\overline{A}^{\wp(E_0)} = \wp(E_0')$. So by corollary 8.5.4 c) $\overline{A}^{\wp(E)} = \overline{A}^{\wp(E_0)\cdot\wp(E_1)} = \overline{A}^{\wp(E_0)} \cdot \wp(E_1) = \wp(E_0') \cdot \wp(E_1) = \wp(E')$.

$E = E_0 \parallel E_1 \xrightarrow{A}_{\mathsf{G}_{m+1}} E'$: There are three ways $E_0 \parallel E_1 \xrightarrow{A}_{\mathsf{G}_{m+1}} E'$ could be obtained, and each case is proved essential as above but this time using corollary 8.5.4 d).

b) By induction on the structure of E.

$E = \dagger$: $\varepsilon \notin \mathsf{G}$ by definition of G so $A \neq \varepsilon$ $(= \emptyset)$ and we cannot have $A \subseteq M_{\wp(E)} = \emptyset$.

$E = a$: $M_{\wp(a)} = a$ and we must have $A = a$. From $a \xrightarrow{a}_{\mathsf{G}} \dagger$ the implication follows with $E' = \dagger$.

$E = E_0 \,;\, E_1$: We divide in two subcases.

$\wp(E_0) \neq \varepsilon$: By corollary 8.5.3 then $A \subseteq M_{\wp(E_0)\cdot\wp(E_1)}$ implies $A \subseteq M_{\wp(E_0)}$. The hypothesis of induction then gives us a E_0' such that $E_0 \stackrel{A}{\Rightarrow}_{\mathsf{G}} E_0'$. Choosing $E' = E_0' \,;\, E_1$ this subcase is settled with proposition 7.2.3.

$\wp(E_0) = \varepsilon$: Using corollary 8.5.3 we see that $A \subseteq M_{\wp(E_1)}$. By hypothesis of induction there exists a E' such that $E_1 \stackrel{A}{\Rightarrow}_{\mathsf{G}} E'$. From b) of the preceding lemma $E_0 \rightarrowtail^* \dagger$ when $\wp(E_0) = \varepsilon$. Applying proposition 7.2.3 then $E_0;E_1 \rightarrowtail^* \dagger;E_1 \rightarrowtail E_1 \stackrel{A}{\Rightarrow}_{\mathsf{G}} E'$ and so $E \stackrel{A}{\Rightarrow}_{\mathsf{G}} E'$.

$E = E_0 \parallel E_1$: $M_{\wp(E)} = M_{\wp(E_0)\times\wp(E_1)} = M_{\wp(E_0)} \cup M_{\wp(E_1)}$, so $A = A_0 \cup A_1$ where $A_i \subseteq M_{\wp(E_i)}$ for $i = 0, 1$. Because G has the property $A \hookrightarrow B, B \in \mathsf{G}$ implies $A \in \mathsf{G}_\varepsilon$ we see $A_0, A_1 \in \mathsf{G}_\varepsilon$. The subcases:

$A_0 = \varepsilon$ or $A_1 = \varepsilon$: Similar to the first subcase of $E = E_0 \,;\, E_1$.

$A_0 \neq \varepsilon \neq A_1$ Then $A_0, A_1 \in \mathsf{G}$ and by hypothesis of induction $\exists E_i'.\, E_i \stackrel{A_i}{\Longrightarrow}_{\mathsf{G}} E_i'$ for $i = 0, 1$. This also means there are F_0, F_0' such that $E_0 \rightarrowtail^* F_0 \stackrel{A_0}{\longrightarrow} F_0'$. Similar for E_1. By proposition 7.2.3 $E_0 \parallel E_1 \rightarrowtail^* F_0 \parallel F_1$ and because $A_0 \cup A_1 = A \in \mathsf{G}$, or equally by convention $A_0 \times A_1 = A \in \mathsf{G}$, we get $F_0 \parallel F_1 \stackrel{A}{\longrightarrow}_{\mathsf{G}} F_0' \parallel F_1'$. With $E' = F_0' \parallel F_1''$ this all together reads $E \stackrel{A}{\Rightarrow}_{\mathsf{G}} E'$.

□

With the previous two lemmas the connection between the two satisfaction relations can now be stated and proved:

Lemma 8.5.9 If $f \in \overline{\mathcal{L}}_{\mathsf{r}}$ and $E \in RBL'$ then

$$E \models_{\mathsf{G}} f \;\textit{iff}\; \wp(E\sigma) \models^{\mathsf{sw}}_{\mathsf{G}} f$$

Proof At first we prove a restricted/ modified version of the proposition:

If $E \in DCL'$ and $f \in \overline{\mathcal{L}}_{\mathsf{r}}$ then

$$f \in \overline{\mathcal{L}}^{\mathsf{r}}_{\mathsf{G}}(E) \;\textit{iff}\; \wp(E) \models^{\mathsf{sw}}_{\mathsf{G}} f \tag{8.25}$$

The proof of this will be by induction on the structure of f.

$f = \mathrm{tt}$ or $f = \mathrm{ff}$: Evident.

$f = \triangledown$:	$E \models_{\mathsf{G}} \triangledown$ *iff* $\forall a \in \Delta.\, E \stackrel{a}{\not\Rightarrow}_{\mathsf{G}}$	definition of $\models_{\mathsf{G}}$
	iff $M_{\wp(E)} = \emptyset$	lemma 8.5.7
	iff $\wp(E) = \varepsilon$	corollary 8.5.3
	iff $\wp(E) \models^{\mathsf{sw}}_{\mathsf{G}} \triangledown$	definition of $\models^{\mathsf{sw}}_{\mathsf{G}}$

$f = \triangle$: With the last case we see: $E \models_{\mathsf{G}} \triangle$ *iff* $E \not\models_{\mathsf{G}} \triangledown$ *iff* $\wp(E) \not\models^{\mathsf{sw}}_{\mathsf{G}} \triangledown$ *iff* $\wp(E) \models^{\mathsf{sw}}_{\mathsf{G}} \triangle$.

$f = \boxed{\mathrm{A}}g$: By hypothesis of induction $E \models_{\mathsf{G}} g$ *iff* $\wp(E) \models^{\mathsf{sw}}_{\mathsf{G}} g$ and from lemma 8.5.7 $\{a \in \Delta \mid E \stackrel{a}{\Rightarrow}_{\mathsf{G}}\} = M_{\wp(E)}$, so the result follows by the similarity of the definitions of $\models_{\mathsf{G}}$ and $\models^{\mathsf{sw}}_{\mathsf{G}}$.

$f = \langle\!\langle A \rangle\!\rangle$: Follows directly from lemma 8.5.8, hypothesis of induction and the definitions of $\models_{\mathsf{G}}$ and $\models_{\mathsf{G}}^{\mathsf{sw}}$.

With (8.25) the lemma now follows:

$$\begin{array}{lll} f \in \overline{\mathcal{L}}^{\mathsf{r}}_{\mathsf{G}}(E) & \textit{iff}\ \exists E' \in \mathrm{Beh}(E\sigma).\, E' \models_{\mathsf{G}} f & \text{definition of } \overline{\mathcal{L}}^{\mathsf{r}}_{\mathsf{G}} \\ & \textit{iff}\ \exists E' \in \mathrm{Beh}(E\sigma).\, \wp(E') \models_{\mathsf{G}} f & \text{(8.25) above} \\ & \textit{iff}\ \exists p \in \wp(E\sigma).\, p \models_{\mathsf{G}}^{\mathsf{sw}} f & \text{(8.26) below} \\ & \textit{iff}\ \wp(E\sigma) \models_{\mathsf{G}}^{\mathsf{sw}} f & \text{by extension of } \models_{\mathsf{G}}^{\mathsf{sw}} \text{ to sets} \end{array}$$

In the deduction we used

$$(8.26) \qquad \forall E \in BL'.\, \wp(E) = \{\wp(E') \mid E' \in \mathrm{Beh}(E)\}$$

which follows by induction on the structure of E using the compositional nature of $\wp$. □

It is appropriate here to recall the note at the end of section 7.6 where it was pointed out that an alternative logic characterization of $\lesssim_{\mathsf{G}}$ (on BL') could be obtained from $\overline{\mathcal{L}}_g$ by pretending definition 8.5.1 of $\models_{\mathsf{G}}$ was for $CL' \times \overline{\mathcal{L}}_g$ and not just $DCL' \times \overline{\mathcal{L}}_g$. The reason was that for $f \in \overline{\mathcal{L}}_g$ and $E \in CL'$ one would have:

$$E \models_{\mathsf{G}} f \ \textit{iff}\ \exists E' \in \mathrm{Beh}(E).\, E' \models_{\mathsf{G}} f$$

For the extended logic language here this would *not* be true. Just consider $E = a \oplus b$ and $f = \boxed{A}\mathrm{tt}$ where $A = \{a, b\}$. Then $E \models_{\mathsf{G}} f$, but for all $E' \in \mathrm{Beh}(E) = \{a, b\}$ $E' \not\models_{\mathsf{G}} f$.

With the last lemma we can now concentrate fully on properties of the semiword based satisfaction relation.

Lemma 8.5.10 Suppose $f \in \overline{\mathcal{L}}_{\mathsf{r}}$ and $p, q \in SW$. Then

$$p \models_{\mathsf{G}}^{\mathsf{sw}} f, p \preceq q \text{ implies } q \models_{\mathsf{G}}^{\mathsf{sw}} f$$

Proof Induction on the structure of f.

$f = \mathrm{tt}, \mathrm{ff}, \bigtriangledown, \bigtriangleup$: Either trivial or follows directly from $\varepsilon \preceq p$ *iff* $p = \varepsilon$.

$f = \boxed{A}g$: Then $A \subseteq M_p$ and $p \models_{\mathsf{G}}^{\mathsf{sw}} g$. By induction $q \models_{\mathsf{G}}^{\mathsf{sw}} g$ and from proposition 8.5.5 a) $M_p \subseteq M_q$, so $A \subseteq M_q$ and we get the result.

$f = \langle\!\langle A \rangle\!\rangle g$: This implies $A \subseteq M_p$ and $\overline{A}^p \models_{\mathsf{G}}^{\mathsf{sw}} g$. By proposition 8.5.5 b) it follows from $p \preceq q$ that $\overline{A}^q$ is well-defined and $\overline{A}^p \preceq \overline{A}^q$. By hypothesis of induction then $\overline{A}^q \models_{\mathsf{G}}^{\mathsf{sw}} g$. Using proposition 8.5.5 a) we have $A \subseteq M_q$, so q actually satisfies $\langle\!\langle A \rangle\!\rangle g = f$ as desired.

□

Lemma 8.5.11 If $p \in SW$ then $p \in \max_{\preceq}(\delta_{or}(q))$ implies $M_p = M_q$

Proof Assume $p \in \max_{\preceq}(\delta_{or}(q))$.

$\subseteq$: The assumption implies $p \preceq q$, so this inclusion follows from proposition 8.5.5.

$\supseteq$: Given $x \in M_q$. Suppose $x \notin M_p$. We show that this leads to a contradiction by finding a $r \in \delta_{or}(q)$ such that $p \prec r$. Define r to be $\langle X_p, \leq_r, \ell_p \rangle$, where $\leq_r = \leq_p \setminus \{\langle y, x \rangle \mid y <_p x\}$. By a moments reflection one sees that $\leq_r$ defines a partial order. $p \preceq r$ follows by definition and $p \prec r$ from $x \in M_r$ and $x \notin M_p$. Since we only have removed relations leading to x we see from $x \in M_q$ and $p \preceq q$ that $r \preceq q$ must hold. It remains to show that r has the P_{or}-property. Let $y, y', z, z' \in X_r$ be given such that

$$\begin{array}{c} y <_r y' \\ co_r \\ z <_r z' \end{array} \tag{8.27}$$

We shall then show $y \leq_r z'$ or $z \leq_r y'$. From $p \preceq r$ and (8.27) we see $y <_p y'$, $z <_p z'$ and since relations are removed from $\leq_p$ to obtain $\leq_r$ *iff* they lead to x, we conclude $x \neq y', z'$.

If $y \; co_p \; z$ then $P_{or}(p)$ implies $y \leq_p z'$ *or* $z \leq_p y'$. Since $x \neq y', z'$ we must then also have $y \leq_r z'$ *or* $z \leq_r y'$.

It remains to consider $y \not{co}_p z$—i.e., either $y \leq_p z$ or $z \leq_p y$. Suppose $y \leq_p z$. Since $z <_p z'$ the transitivity of $\leq_p$ yields $y \leq_p z'$ and from $x \neq z'$ we then conclude $y \leq_r z'$. Similar for $z \leq_p y$. □

Lemma 8.5.12 Suppose $A \in \mathsf{S}$ and $P \subseteq SW$ has the property $\forall p \in P.\ A \subseteq M_p$. Then $\max_{\preceq}\{\overline{A}^p \mid p \in P\} \subseteq \{\overline{A}^p \mid p \in \max_{\preceq} P\}$.

Proof Given $\overline{A}^q \in \max_{\preceq}\{\overline{A}^p \mid p \in P\}$. I.e., $q \in P$ and there is no $p \in P$ such that $\overline{A}^q \prec \overline{A}^p$ or by (8.24):

$$\nexists p \in P.\ X_q \setminus A = X_p \setminus A,\ \leq_q|_{(X_q \setminus A)^2} \supset \leq_p|_{(X_p \setminus A)^2}$$

So if $p \in P$ and $q \preceq p$ we must have $\leq_q|_{(X_q\setminus A)^2} = \leq_p|_{(X_p\setminus A)^2}$. Hence the partial order of any maximal element r of P ($r \in \max_{\preceq} P$) above q ($q \preceq r$) agrees on $X_q \setminus A$, wherefore $\overline{A}^q = \overline{A}^r$ and we are done. □

With $P = \left\{ \begin{matrix} a \searrow \\ c \nearrow \end{matrix} b,\ a \begin{matrix} \nearrow b \\ \searrow c \end{matrix} \right\}$ and $A = a$ it follows that the right hand side of the inclusion in the lemma may be different from the left hand side.

Lemma 8.5.13 If $A \subseteq M_q$ and $P = \{p \in \delta_{or}(q) \mid A \subseteq M_p\}$ then $\delta_{or}(\overline{A}^q) = \{\overline{A}^p \mid p \in P\}$.

Proof
$\subseteq$: Given $r \in \delta_{or}(\overline{A}^q)$, i.e., $P_{or}(r)$ and $r \preceq \overline{A}^q$. Let $p = A \cdot r$. Then $\overline{A}^p = r$ and we have $\overline{A}^p \preceq \overline{A}^q$ or equally by (8.24):

$$X_q \setminus A = X_p \setminus A,\ \leq_q|_{(X_q \setminus A)^2} \supseteq \leq_p|_{(X_p \setminus A)^2}$$

Then $X_p = X_q$ and because the elements of A in p are below all other elements of X_p we conclude $p \preceq q$. The P_{or}-property is dot synthesizable so from $P_{or}(A)$ and $P_{or}(r)$ follows

$P_{or}(A \cdot r)$, i.e., $p = A \cdot r \in \delta_{or}(q)$. Since $A \subseteq M_{A \cdot r} = M_p$ then $p \in P$ and this implication is settled.

$\supseteq$: $p \in \delta_{or}(q)$ implies $P_{or}(p)$ and $p \preceq q$. Since $A \subseteq M_p$ we get $\overline{A}^p \preceq \overline{A}^q$. The P_{or}-property inherits to $\overline{A}^p$, so $\overline{A}^p \in \delta_{or}(\overline{A}^q)$. $\square$

If $A \subseteq M_q$ we from the last two lemmas see that:

$$\max_{\preceq}(\delta_{or}(\overline{A}^q)) \subseteq \{\overline{A}^p \mid p \in \max_{\preceq}\{p \in \delta_{or}(q) \mid A \subseteq M_p\}\} \tag{8.28}$$

Since $A \subseteq M_q$ we by lemma 8.5.11 also have:

$$\max_{\preceq}\{p \in \delta_{or}(q) \mid A \subseteq M_p\} = \max_{\preceq}(\delta_{or}(q)) \tag{8.29}$$

Combining (8.28) and (8.29) then:

Corollary 8.5.14 If $A \subseteq M_q$ and $q \in SW$ then

$$\max_{\preceq}(\delta_{or}(\overline{A}^q)) \subseteq \{\overline{A}^p \mid p \in \max_{\preceq}(\delta_{or}(q))\}$$

That the converse inclusion does not hold can be seen as follows. Let $q = \begin{smallmatrix} a & \to & b \\ c & \to & d \end{smallmatrix}$ and $p = \begin{smallmatrix} a & \to & b \\ & \nearrow & \\ c & \to & d \end{smallmatrix}$. Then $p \in \max_{\preceq}(\delta_{or}(q))$ and $\overline{a}^p = \begin{smallmatrix} & & b \\ & \nearrow & \\ c & \to & d \end{smallmatrix}$, but $\overline{a}^p \notin \max_{\preceq}(\delta_{or}(\overline{a}^q)) = \left\{ \begin{smallmatrix} & & b \\ c & \to & d \end{smallmatrix} \right\}$.

Lemma 8.5.15 Given $f \in \overline{\mathcal{L}}_{\mathsf{r}}$ and $p \in SW$. Then

$$p \models^{\mathsf{sw}}_{\mathsf{G}} f \text{ implies } \exists q \in \delta_{or}(p).\ q \models^{\mathsf{sw}}_{\mathsf{G}} f$$

Proof The lemma follows by proving the stronger

$$p \models^{\mathsf{sw}}_{\mathsf{G}} f \text{ implies } \exists q \in \max_{\preceq}(\delta_{or}(p)).\ q \models^{\mathsf{sw}}_{\mathsf{G}} f$$

by induction in the structure of f.

$f = \mathrm{tt}, \mathrm{ff}, \triangledown, \triangle$: Either trivial or follows from $\max_{\preceq}(\delta_{or}(p)) = \{\varepsilon\}$ *iff* $p = \varepsilon$.

$f = \boxed{A}g$: Then $A \subseteq M_p$ and $p \models^{\mathsf{sw}}_{\mathsf{G}} g$. By hypothesis of induction there is a $q \in \max_{\preceq}(\delta_{or}(p))$ such that $q \models^{\mathsf{sw}}_{\mathsf{G}} g$. By lemma 8.5.11 we have $M_q = M_p$, so also $q \models^{\mathsf{sw}}_{\mathsf{G}} \boxed{A}g$.

$f = \langle A \rangle g$: Here we must have $A \in \mathsf{G}$, $A \subseteq M_p$ and $\overline{A}^p \models^{\mathsf{sw}}_{\mathsf{G}} g$. Using the hypothesis of induction we get a $q' \in \max_{\preceq}(\delta_{or}(\overline{A}^p))$ such that $q' \models^{\mathsf{sw}}_{\mathsf{G}} g$. By corollary 8.5.14 then $q' \in \{\overline{A}^q \mid q \in \max_{\preceq}(\delta_{or}(p))\}$. I.e., there is a $q \in \max_{\preceq}(\delta_{or}(p))$ such that $\overline{A}^q = q'$ and thereby $\overline{A}^q \models^{\mathsf{sw}}_{\mathsf{G}} g$. Because $q \in \max_{\preceq}(\delta_{or}(p))$ and $A \subseteq M_p$ we can use lemma 8.5.11 to see $A \subseteq M_q$. Hence $q \models^{\mathsf{sw}}_{\mathsf{G}} \langle A \rangle g$ and $q \in \max_{\preceq}(\delta_{or}(p))$ as desired.

$\square$

Lemma 8.5.16 Suppose $p \in SW$ and $P_{or}(p)$. Then there is a $f_p \in \overline{\mathcal{L}}_{\mathsf{r}}$ such that:

a) $p \models^{\mathsf{sw}}_{\mathsf{W}} f_p$

b) $q \models^{\mathsf{sw}}_{\mathsf{G}} f_p$ implies $p \preceq q$

Proof The proof is by induction on the size of p (i.e., of X_p).

In the basis $X_p = \emptyset$ we can only have $p = \varepsilon$. Choose $f_p = \triangledown$. By definition $\varepsilon \models^{\mathsf{sw}}_{\mathsf{W}} \triangledown$.

For the inductive step we can assume the proposition to hold for all semiwords of size less than the given p. Now consider M_p. Because $P_{or}(p)$ we by proposition 8.3.7 know there is a $a \in M_p$ such that

(8.30) $$\forall x \in M_p.\ x <_p y \Rightarrow a <_p y$$

Denote $\overline{a}^p$ by p'. Then the size of p' is less than the size of p and we can apply the hypothesis of induction to find a $f_{p'} \in \overline{\mathcal{L}}_{\mathsf{r}}$ such that $p' \models^{\mathsf{sw}}_{\mathsf{W}} f_{p'}$ and $q' \models^{\mathsf{sw}}_{\mathsf{G}} f_{p'}$ implies $p' \preceq q'$. Since $M_p \neq \emptyset$ (when $X_p \neq \emptyset$) we can define:

$$f_p = \boxed{\mathrm{A}} \diamondsuit\!\!\!\!\!\scriptstyle{\mathrm{a}}\ f_{p'}, \text{ where } A = M_p$$

Clearly $P \models^{\mathsf{sw}}_{\mathsf{W}} f_p$. So let a q be given such that $q \models^{\mathsf{sw}}_{\mathsf{G}} f_p$. This means $A \subseteq M_q$ and $q' = \overline{a}^q \models^{\mathsf{sw}}_{\mathsf{G}} f_{p'}$. From the hypothesis of induction we know $p' \preceq q'$ or equally $\overline{a}^p \preceq \overline{a}^q$. This means by (8.24):

(8.31) $X_q \setminus \{a\} = X_p \setminus \{a\}$ and (8.32) $\leq_q|_{(X_q\setminus\{a\})^2} \supseteq \leq_p|_{(X_p\setminus\{a\})^2}$

From $a \in M_p$, $a \in M_q$ and (8.31) follows $X_p = X_q$, so we just have to prove $\leq_p \supseteq \leq_q$ in order to obtain $p \preceq q$. By the reflexivity of $\leq_p$ is suffice to show $x <_q y$ implies $x <_p y$ for given $x, y \in X_q$ $(= X_p)$. We distinguish three cases:

$x, y \neq a$: Follows from (8.32).

$x \neq a, y = a$: I.e., $x <_q a$. But this contradicts $a \in M_q$, so we can exclude this case.

$x = a, y \neq z$: Then $x <_q y$ reads $a <_q y$ wherefore $y \notin M_q$. Because $M_p = A \subseteq M_q$ this also implies $y \notin M_p$. Hence there is a $z \in M_p$ such that $z <_p y$. By (8.30) then $a <_p y$ as we wish.

□

We can now prove the crucial lemma used in the proof of the linear logic characterization of $\lesssim^c_{\mathsf{G}}$.

Lemma 8.5.17 For $E_0, E_1 \in RBL'$ we have:

$$E_0 \lesssim^c_{\mathsf{G}} E_1 \textit{ iff } \overline{\mathcal{L}}^{\mathsf{r}}_{\mathsf{G}}(E_0) \subseteq \overline{\mathcal{L}}^{\mathsf{r}}_{\mathsf{G}}(E_1)$$

Proof

only if: $E_0 \lesssim^c_{\mathsf{G}} E_1$ implies $[\![E_0]\!]_{or} \subseteq [\![E_1]\!]_{or}$ by the full abstractness result of section 8.4. Now let $f \in \overline{\mathcal{L}}^{\mathsf{r}}_{\mathsf{G}}(E_0)$ be given. Then by lemma 8.5.9: $\wp(E_0\sigma) \models^{\mathsf{sw}}_{\mathsf{G}} f$ which means there is a $q \in \wp(E_0\sigma)$ such that $q \models^{\mathsf{sw}}_{\mathsf{G}} f$. Hence from lemma 8.5.15 $p \models^{\mathsf{sw}}_{\mathsf{G}} f$ for some $p \in \delta_{or}(q)$. This means there is a $p \in [\![E_0]\!]_{or}$ with $p \models^{\mathsf{sw}}_{\mathsf{G}} f$. Because $[\![E_0]\!]_{or} \subseteq [\![E_1]\!]_{or}$ then also $p \in [\![E_1]\!]_{or}$ and by definition of $[\![\_]\!]_{or}$ there must be a $r \in \wp(E_1\sigma)$ such that $p \preceq r$. So from lemma 8.5.10 $r \models^{\mathsf{sw}}_{\mathsf{G}} f$ and $\wp(E_1\sigma) \models^{\mathsf{sw}}_{\mathsf{G}} f$. Using lemma 8.5.9 again we get $f \in \overline{\mathcal{L}}^{\mathsf{r}}_{\mathsf{G}}(E_1)$.

if: $[\![E_0]\!]_{or} \subseteq [\![E_1]\!]_{or}$ implies $E_0 \lesssim^c_{\mathsf{G}} E_1$ so it is enough to prove $p \in [\![E_1]\!]_{or}$ for a given $p \in [\![E_0]\!]_{or}$. Then $P_{or}(p)$ and by the previous lemma there is $f_p \in \overline{\mathcal{L}}_{\mathsf{r}}$ such that $p \models^{\mathsf{sw}}_{\mathsf{W}} f_p$ and $q \models^{\mathsf{sw}}_{\mathsf{G}} f_p$ implies $p \preceq q$. Clearly $p \models^{\mathsf{sw}}_{\mathsf{W}} f_p$ implies $p \models^{\mathsf{sw}}_{\mathsf{G}} f_p$. $p \in [\![E_0]\!]_{or}$ means there is a $r \in \wp(E_0\sigma)$ such that $p \preceq r$. Lemma 8.5.10 gives $r \models^{\mathsf{sw}}_{\mathsf{G}} f_p$ because $p \models^{\mathsf{sw}}_{\mathsf{G}} f_p$, so from lemma 8.5.9 then $f_p \in \overline{\mathcal{L}}^{\mathsf{r}}_{\mathsf{G}}(E_0)$. Hence also $f_p \in \overline{\mathcal{L}}^{\mathsf{r}}_{\mathsf{G}}(E_1)$ by assumption. As above we see that there is a $q \in \wp(E_1\sigma)$ such that $q \models^{\mathsf{sw}}_{\mathsf{G}} f_p$. By the way f_p was chosen then $p \preceq q$. Because $P_{or}(p)$ we finally have $p \in [\![E_1]\!]_{or}$. $\square$

From this proof and lemma 8.5.16 ($p \models^{\mathsf{sw}}_{\mathsf{W}} f_p$) it appears that $\overline{\mathcal{L}}^{\mathsf{r}}_{\mathsf{W}}$ actually would suffice to characterize $\lesssim^c_{\mathsf{G}}$ and a closer look at f_p shows that formulae generated by:

$$f ::= \bigtriangledown \mid \boxed{A} f, A \in \mathsf{S} \mid \langle a \rangle f, a \in \Delta \tag{8.33}$$

would do. This can of course not come surprisingly because we from the full abstractnesss result already know $\lesssim^c_{\mathsf{W}} = \lesssim^c_{\mathsf{G}}$ for any set of direct tests G ($\Delta \subseteq \mathsf{G} \subseteq \mathsf{S}$). It can also be seen from the ability of $\boxed{A}$ and $\langle a \rangle$ to simulate the effect of the modality $\langle A \rangle$ under the satisfaction relation $\models_{\mathsf{G}}$ where $A \in \mathsf{G}$.

Example: Suppose $A = \{a, c\}$ and $A \in \mathsf{G}$. Then

$$E \models_{\mathsf{G}} \langle A \rangle \mathrm{tt} \;\textit{iff}\; E \models_{\mathsf{W}} \boxed{A}\langle a \rangle\langle c \rangle \mathrm{tt}$$

The reason why formulae from $\overline{\mathcal{L}}_{\mathsf{r}}$ are used in place of just those of (8.33) is as in section 7.6 because they provide more freedom in specifications. How forcible formulae can be of course depends on the available satisfaction relation.

Example: Suppose A and G are as in the example above. Furthermore let

$$E_0 = (a\,;a \parallel c)\,;e \text{ and } E_1 = (a \parallel c)\,;a\,;e \oplus a\,;(a \parallel c)\,;e$$

Then the formula

$$f = \boxed{A}\langle a \rangle\langle A \rangle \mathrm{tt}$$

would be sufficient to distinguish E_0 and E_1: $E_0 \models_{\mathsf{G}} f$ but $E_1 \not\models_{\mathsf{G}} f$. If only formulae of (8.33) could be used then a formula like

$$\boxed{A}\langle a \rangle\boxed{A}\langle c \rangle\langle a \rangle\langle e \rangle\bigtriangledown$$

should be used to differentiate E_0 and E_1.
Notice by the way that no formulae of $\overline{\mathcal{L}}_g$ can distinguish E_0 and E_1.

Striving towards more freedom in specifications it is tempting when looking at the definition for $E \models_{\mathsf{G}} \boxed{A} f$ to turn the modality $\boxed{A}$ into an atomic proposition with $E \models_{\mathsf{G}} \boxed{A}$ *iff* $A \subseteq \{a \in \Delta \mid E \stackrel{a}{\Rightarrow}_{\mathsf{G}}\}$ and add conjunction to $\overline{\mathcal{L}}_{\mathsf{r}}$ (disjunction to $\mathcal{L}_{\mathsf{r}}$ respectively). However this would make the modal logic to strong which can be seen as follows.
Let $E_0 = (a \parallel c) ; a \parallel e \oplus c \parallel a ; (a \parallel e)$ and $E_1 = a ; a \parallel c ; e$. Then $E_0 \lessapprox^c E_0 \oplus E_1$ (can be seen from the denotations) but with $f = \Diamond_{a} \Diamond_{a} \Diamond_{c} \mathrm{tt} \wedge \Diamond_{e} \Diamond_{a} \Diamond_{a} \mathrm{tt}$ we would have $E_0 \oplus E_1 \models_{\mathsf{w}} f$ and $E_0 \not\models_{\mathsf{w}} f$.
One way out would be to restrict the admissible formulae to be those where any subformula of the form $f \wedge g$ would have either $f \neq \Diamond_{A} f'$ or $g \neq \Diamond_{A} g'$.

We end the section with a comment regarding the logic characterization of $\lesssim^c_{\mathsf{G}}$ for the full *RBL* language.
Without problems the formula language could be extended to include formulae like $\boxed{A} f$ where $A \in \mathsf{M}$ and not just $A \in \mathsf{S}$ as it is now. Taking the same definition of $\models_{\mathsf{G}}$ but as if it was for *DCL* one could similarly extend $\models_{\mathsf{G}}$ to *RBL* and obtain a logic for *RBL*. Nevertheless the logic would not be strong enough to characterize $\lesssim^c_{\mathsf{G}}$ on *RBL*: If $E_0 = a ; a \parallel a ; a$ and $E_1 = a ; a ; a \parallel a \oplus (a \parallel a) ; (a ; a)$ then $E_0 \not\lesssim^c_{\mathsf{G}} E_1$ but no formula of $\overline{\mathcal{L}}_{\mathsf{r}}$ (or dually $\mathcal{L}_{\mathsf{r}}$) would be able to distinguish E_0 and E_1 as can be seen by an easy case analysis. How, if possible, to characterize $\lesssim^c_{\mathsf{G}}$ on *RBL* remains open.

Chapter 9

Adding Recursion to BL and RBL

In this chapter we shall equip the process languages BL and RBL of the two preceding chapters with constructors for recursion in order to deal with infinite behaviours. The crucial constructors will be of the form $rec\,x.\,\_$. If E is an expression with x at some places (where an action could have been) then one can roughly think of $rec\,x.\,E$ as the process which evolve like E until an x is meet where after it (repeatedly) can evolve like $rec\,x.\,E$. Example:

$$E = rec\,x.\,(a \oplus b\,;x) \overset{b}{\Rightarrow} E \overset{b}{\Rightarrow} E \overset{a}{\Rightarrow} \dagger$$

But of course E could just as well evolve infinitely performing b's

$$E \overset{b}{\Rightarrow} E \overset{b}{\Rightarrow} \ldots \overset{b}{\Rightarrow} E \overset{b}{\Rightarrow} \ldots$$

With our notion of (finite) maximal sequences of direct tests we would still be able to distinguish recursive BL processes like:

$$rec\,x.\,(a \oplus b\,;x) \quad \text{and} \quad rec\,x.\,(c \oplus b\,;x)$$

because they obviously can do different maximal sequences. On the other hand there will be no way to distinguish the processes:

$$rec\,x.\,(a\,;x) \quad \text{and} \quad rec\,x.\,(b\,;x) \tag{9.1}$$

This is satisfactory if nontermination is viewed as unimportant and only termination matters. Taking the opposite point of view, disregarding termination, they must be distinguished. One way to go would be to find some notion of infinite sequences. Against this one might argue that it breaks with the principle of finite observability: no (human) experimenter can carry out infinite sequences of direct tests. But there seems no reason to inhibit the experimenter from recording prefixes of a (possibly maximal) sequence. The preorder arrising when the experimenter is endowed with this capability will be denoted $\precapprox$ as opposed to $\lesssim$ from the previous chapters. The associated equivalence, $\approxeq$, of $\precapprox$ will be able to distinguish the expressions of (9.1) but in return identify

$$rec\,x.\,(b \oplus b\,;x) \quad \text{and} \quad rec\,x.\,(b\,;x)$$

which on the contrary would not be identified by $\gtrless$—the equivalence of $\lesssim$. The appropriate equivalence depends on what view is taken. However there is the serious drawback of $\precapprox$

that it is not a precongruence—not even on BL:

$$a \oplus a \,;\, b \precsim a \,;\, b \text{ but } (a \oplus a \,;\, b) \,;\, c \not\precsim (a \,;\, b) \,;\, c$$

We will therefore also be interested in $\precsim^c$—the largest precongruence contained in $\precsim$. Recalling the operational semantics of BL we know that a sequence of $E \,;\, F$ involving actions from F must contain a maximal sequence of E. The models from the previous chapters all to some extend mirrored maximal sequences so here we already get the clue that the previous models must be incorporated in the models which shall capture $\precsim^c$ (for the various operational G-semantics). For the recursive RBL processes we shall similarly look for models characterizing $\lesssim^c$ and $\precsim^c$.

There are standard ways of giving denotational semantics to recursive expressions and from the previous chapters we have an god idea of how the models for the preorders should look like. We will therefore in this chapter take the opposite angel and start out by constructing the infinite models and then use the finite parts of the models as link to the operational semantics.

9.1 General Set-up

In this section the definitions and results necessary for the remaining sections are introduced. We shall assign meaning to recursive expressions as done by Hennessy in [Hen88a]. Except for borrowing his notation and some results the section is intended to be self contained.

9.1.1 Denotations of Recursive Expressions

Given an infinite set, X, of variables and a signature, Σ, containing Ω which intuitively represent the completely undefined process. The language of recursive expressions over Σ, $REC_\Sigma(X)$, is the least set which satisfies

$$\begin{aligned} X &\subseteq REC_\Sigma(X) \\ f(t_1, \ldots, t_k) &\in REC_\Sigma(X) \quad \text{if } t_1, \ldots, t_n \in REC_\Sigma(X) \text{ and } f \in \Sigma \text{ is of arity } k \\ rec\, x.\, t &\in REC_\Sigma(X) \quad \text{if } t \in REC_\Sigma(X) \text{ and } x \in X \end{aligned}$$

The syntactically finite expressions are denoted $FREC_\Sigma(X)$—i.e., those expressions of $REC_\Sigma(X)$ with no occurrences of $rec\, x.$ for any $x \in X$. A variable x is free in t if x is not within the scope of a $rec\, y.$ combinator where $y = x$, and an expression t is called open (closed) if t contains (no) free variables. The set of free variables of an expression t is denoted $FV(t)$ and closed expressions of $REC_\Sigma(X)$ and $FREC_\Sigma(X)$ are denoted REC_Σ and $FREC_\Sigma$ respectively.

A syntactic substitution, ρ, is a $REC_\Sigma(X)$-assignment, i.e., a map from X to $REC_\Sigma(X)$, and is extended is extended to $REC_\Sigma(X)$ in the usual way, possible with renaming of bound variables to avoid clashes. $\rho[x \to t]$ denotes the substitution which maps x to t and otherwise is identical to ρ. $[t/x]$ is a shorthand for $I[x \to t]$ where I is the identity substitution.

An expression t is a approximation to u if it is in the relation $\preceq$, where the syntactic preorder, $\preceq$, is defined to be the least Σ-precongruence over $REC_\Sigma(X)$ which satisfies:

$$\Omega \preceq t$$
$$t[rec\,x.\,t/x] \preceq rec\,x.\,t$$

For every $t \in REC_\Sigma(X)$, $\mathrm{Fin}(t)$ denotes $\{t' \in FREC_\Sigma(X) \mid t' \preceq t\}$. Intuitively $\mathrm{Fin}(t)$ is the set of syntactic finite approximations to t and the meaning of t can thought of as the limit of these approximations.

Having syntactic finite approximations the notion of algebraic relations can be introduced:

A relation R over REC_Σ is *algebraic* if for all $t, u \in REC_\Sigma$:

$$t\ R\ u \ \mathit{iff}\ \forall t' \in \mathrm{Fin}(t) \exists u' \in \mathrm{Fin}(u).\, t'\ R\ u'$$

The preorder $\preceq$ enjoys the following properties ([Hen88a, page 218]):

- $\preceq$ is a partial order on $FREC_\Sigma$
- $t \preceq u$ implies $t\rho \preceq u\rho$
- $\mathrm{Fin}(t)$ is directed w.r.t. $\preceq$

A Σ-domain, A, is a triple $\langle A, \leq_A, \Sigma_A \rangle$ where

- $\langle A, \leq_A \rangle$ is an algebraic complete partial order (algebraic *cpo* for short)
- for each f in Σ of arity k there is an associated continuous function $f_A : A^k \longrightarrow A$ in Σ_A.
- Ω_A is the least element $\perp_A$ of $\langle A, \leq_A \rangle$

We shall use $\mathrm{Fin}(A)$ to denote the compact elements of A.

Given a Σ-domain, A, the expressions of $REC_\Sigma(X)$ are assigned a meaning using environments over A. An environment is an A-assignment, ρ_A, i.e., a map from X to A, and similar as for syntactic substitution $\rho_A[a/x]$ is the A-assignment which maps x to a and otherwise equals ρ_A. The set of all environments, $(X \longrightarrow A)$, is denoted ENV_A. Two A-environments ρ and ρ' from ENV_A are ordered by the induced pointwise ordering:

$$\rho \leq \rho' \ \mathit{iff}\ \forall x \in X.\, \rho(x) \leq_A \rho'(x)$$

Proposition 9.1.1 ENV_A is an algebraic complete partial order and the compact elements of ENV_A are those ρ_A where there exists a *finite* subset Y of X such that

a) $\forall x \in X \setminus Y.\, \rho_A(x) = \perp_A$

b) $\forall x \in Y.\, \rho_A(x) \in \mathrm{Fin}(A)$

Notice that $\bot_A$ is a compact elements so $\rho_A \in \text{Fin}(\text{ENV}_A)$ actually implies $\rho(x) \in \text{Fin}(A)$ for every $x \in X$.

Proof For convenience we will in this proof use $f, g, \ldots$ for elements of ENV_A. By Hennessy [Hen88a, page 123] $(X \longrightarrow A)$ is a complete partial order. We show that the compact elements of ENV_A are as described above. It is then a simple matter to check that every element is the lub of the compact elements below it.

For every compact element f there is a finite $Y \subseteq X$ fulfilling a) and b): Let $Y = \{x \in X \mid f(x) \neq \bot_A\}$. To see that Y is a finite set assume on the contrary it is infinite. Then Y contains a countable infinite subset $Z = \{z_i\}_{i \in I\!N}$. For each $i \in I\!N$ define $f_i \in \text{ENV}_A$ by

$$f_i(x) = \begin{cases} f(x) & \text{if } x \notin Z \\ f(z_j) & \text{if } x = z_j \text{ and } j < i \\ \bot_A & \text{otherwise} \end{cases}$$

Clearly $D = \{f_i\}_{i \in I\!N}$ is a chain $f_0 \leq_A f_1 \leq_A \ldots$ with lub f. Because $f \leq f$ and f is compact there is an f_i in D such that $f \leq f_i$, so $\forall x \in X.\ f(x) \leq_A f_i(x)$ and especially $f(z_i) \leq f_i(z_i) = \bot_A$. Hence $f(z_i) = \bot_A$—a contradiction to the definition of Z ($z \in Z \subseteq Y$ only if $f(z) \neq \bot_A$).
By definition Y fulfills a) and to see b) let an $y \in Y$ be given. We shall show $f(y) \in \text{Fin}(A)$. To this end let D_A be a directed set in A such that $f(y) \leq_A \bigvee_A D_A$. Then we shall find a $d \in D_A$ such that $f(y) \leq_A d$. For every $a \in D_A$ define f_a by

$$f_a(x) = \begin{cases} f(x) & \text{if } x \neq y \\ a & \text{if } x = y \end{cases}$$

and let D be $\{f_a \mid a \in D_A\}$. D is directed because D_A is and D has lub f_D in ENV_A where

$$f_D(x) = \begin{cases} f(x) & \text{if } x \neq y \\ \bigvee_A D_A & \text{if } x = y \end{cases}$$

So $f \leq f_D$ and because f is compact there is an $f_d \in D$ such that $f \leq f_d$. Then also $f(y) \leq f_d(y) = d \in D_A$.

That f is a compact element when there is a finite $Y \subseteq X$ fulfilling a) and b) is easier to see: Given a directed set D in ENV_A such that $f \leq f_D$ where f_D is the lub of D in ENV_A. We shall find a $g \in D$ such that $f \leq g$. $f \leq f_D$ implies $\forall x \in X. f(x) \leq_A \bigvee_A \{g(x) \mid g \in D\}$. Since $\forall y \in Y.\ f(y) \in \text{Fin}(A)$ we then have $\forall y \in Y \exists g \in D.\ f(y) \leq_A g(y)$. For each $y \in Y$ let g_y be such an g. Denote $\{g_y \mid y \in Y\}$ by G_Y. Since y is finite then so is G_Y and because D is directed we can then find a $g \in D$ such that $\forall g_y \in G_Y.\ g_y \leq g$. Hence $\forall y \in Y. f(y) \leq_A g(y)$. Because $\forall x \in X \backslash Y. f(x) = \bot_A$ we actually have $\forall z \in X \backslash Y. f(x) \leq_A g(x)$ and we conclude $f \leq g$. □

Since ENV_A is a cpo, meanings of expressions can now be given by means of the function: $A[\![\_]\!] : REC_\Sigma(X) \longrightarrow [\text{ENV}_A \longrightarrow A]$ defined as follows:

$$A[\![x]\!]\rho_A = \rho_A(x)$$
$$A[\![f(t_1, \ldots, t_k)]\!]\rho_A = f_A(A[\![t_1]\!]\rho_A, \ldots, A[\![t_k]\!]\rho_A)$$
$$A[\![rec\, x.\, t]\!]\rho_A = Y \lambda a.\ A[\![t]\!]\rho_A[a/x]$$

where Y is a function that yields the least fixpoint of $\lambda a. A[\![t]\!]\rho_A[a/x]$ in A and $[\text{ENV}_A \longrightarrow A]$ is the continuous functions of $(\text{ENV}_A \longrightarrow A)$.

We select some of the results Hennessy displays:

Proposition 9.1.2 $A[\![rec\,x.\,t]\!]\rho_A = A[\![t[rec\,x.\,t/x]]\!]\rho_A$ for all $\rho_A \in \text{ENV}_A$.

With this proposition it is easy to see for $t, u \in REC_\Sigma(X)$ that:

$$t \preceq u \text{ implies } \forall \rho_A \in \text{ENV}_A.\, A[\![t]\!]\rho_A \leq_A [\![u]\!]\rho_A$$

I.e., the preorder defined as on the right-hand side extends $\preceq$ on $REC_\Sigma(X)$.

Theorem 9.1.3 (finite approximations) For every t in $REC_\Sigma(X)$ and $\rho_A \in \text{ENV}_A$ the following holds: $A[\![t]\!]\rho_A = \bigvee_A A[\![\text{Fin}(t)]\!]\rho_A := \bigvee_A \{A[\![t']\!]\rho_A \mid t' \in \text{Fin}(t)\}$.

Lemma 9.1.4 If $\rho_A, \rho'_A \in \text{ENV}_A$ and $\rho_A \leq_{FV(t)} \rho'_A$ then $A[\![t]\!]\rho_A \leq_A A[\![t]\!]\rho'_A$

From this lemma it follows the value of $A[\![t]\!]\rho_A$ for a $t \in REC_\Sigma$ (the processes of $REC_\Sigma(X)$) is independent of ρ_A ($FV(t) = \emptyset$) and this value can be taken as the meaning of t. So $A[\![\_]\!]$ can be thought of as defining a map $REC_\Sigma \longrightarrow A$ and we will therefore just write $A[\![t]\!]$ when t is a closed expression.

Lemma 9.1.5 (Substitution Lemma) $A[\![t\rho]\!]\rho_A = A[\![t]\!](\rho_A \circ \rho)$
where the composition of the A-assignment ρ_A and the substitution ρ is the A-assignment: $(\rho_A \circ \rho)(x) = A[\![\rho(x)]\!]\rho_A$.

We are now ready to reflect on extending preorders from closed to open expressions.

A preorder, ambiguously denoted $\leq_A$, over REC_Σ can be induced from the partial order, $\leq_A$, of the Σ-domain by letting for $t, t' \in REC_\Sigma$:

$$t \leq_A t' \textit{ iff } \forall \rho_A \in \text{ENV}_A.\, A[\![t]\!]\rho_A \leq_A A[\![t']\!]\rho_A$$

Since Y and the functions of a Σ-domain are continuous and especially monotone it is self-evident from the definition of $A[\![\_]\!]$ that $\leq_A$ is a REC_Σ-precongruence in the sense that for closed expressions:

- for all f in Σ, $f(t_1, \ldots, t_k) \leq_A f(u_1, \ldots, u_k)$ whenever $t_i \leq_A u_i$ for $i = 1, \ldots, k$
- $t \leq_A u$ implies $rec\,x.\,t \leq_A rec\,x.\,u$

The latter actually does not tell us anything because $A[\![rec\,x.\,t]\!]\rho_A =$ (proposition 9.1.2 $A[\![t[rec\,x.\,t/x]]\!]\rho_A =$ (substitution lemma) $A[\![t]\!](\rho_A \circ I[rec\,x.\,t/x])$ which from lemma 9.1.4 and $FV(t) = \emptyset$ is seen to equal $A[\![t]\!]\rho_A$.

This is the motivation for extending the preorder to $REC_\Sigma(X)$. There is at least two ways to do this. For $t, u \in REC_\Sigma(X)$ define:

a) $t \leq_A u$ *iff* $\forall \rho_A \in \mathrm{ENV}_A.\ A[\![t]\!]\rho_A \leq_A A[\![u]\!]\rho_A$
b) $t \leq'_A u$ *iff* for every closed (syntactic) substitutions ρ, $A[\![t\rho]\!] \leq_A A[\![u\rho]\!]$

Notice that for closed substitutions $t\rho \in REC_\Sigma$.

With similar arguments as above it is now easy to see that $\leq_A$ is a $REC_\Sigma(X)$-precongruence and one might argue that it is the most natural extension in a denotational set-up whereas the other, $\leq'_A$, is more natural in an operational set-up. However we will now show that under certain circumstances $\leq_A$ and $\leq'_A$ coincide not only on REC_Σ but also on $REC_\Sigma(X)$.

We can then state:

Proposition 9.1.6 $\leq_A$ and $\leq'_A$ defined above coincide over $REC_\Sigma(X)$ provided there for every compact element of A is a $t \in FREC_\Sigma$ such that $A[\![t]\!] = a$.

Proof Let $t, t' \in REC_\Sigma(X)$ be given. Suppose that for all closed substitutions ρ, $A[\![t\rho]\!] \leq_A A[\![t'\rho]\!]$. We show this implies $A[\![t]\!] \leq_A A[\![t']\!]$—i.e., $\forall \sigma_A \in \mathrm{ENV}_A.\ A[\![t]\!]\sigma_A \leq_A A[\![t']\!]\sigma_A$. So let a $\sigma_A \in \mathrm{ENV}_A$ be given. From the proposition 9.1.1 ENV_A is algebraic because A is. This means $\sigma_A = \bigvee F$ where F is the directed set consisting of the compact elements in ENV_A below σ_A. From the proposition we know that for a compact element $\rho_A \in \mathrm{ENV}_A$ we have $\rho_A(x)$ is compact in A for all $x \in X$. By the proviso of the proposition there then is a $t_x \in FREC_\Sigma$ for each $x \in X$ such that $A[\![t_x]\!] = \rho_A(x)$. Letting ρ be the closed syntactic substitution with $\rho(x) = t_x$ for all $x \in X$ we then have $\rho_A \circ \rho = \rho_A$. I.e., for each $\rho_A \in F$ there is a closed substitution ρ with $\rho_A \circ \rho = \rho_A$. Our assumption was that for all closed substitutions ρ', $A[\![t\rho']\!] \leq_A A[\![t'\rho']\!]$ ($t\rho'$ and $t'\rho'$ are closed), so especially $A[\![t\rho]\!]\rho_A \leq_A A[\![t'\rho]\!]\rho_A$ for each $\rho_A \in F$. By the substitution lemma then $A[\![t]\!](\rho_A \circ \rho) \leq_A A[\![t']\!](\rho_A \circ \rho)$ for each $\rho_A \in F$, and since $\rho_A \circ \rho = \rho_A$ this actually reads:

$$\forall \rho_A \in F.\ A[\![t]\!]\rho_A \leq_A A[\![t']\!]\rho_A$$

From this the result then follows by the deduction

$$\forall \rho_A \in .\ A[\![t]\!]\rho_A \leq_A A[\![t']\!]\rho_A$$

$\Downarrow$

$$\{A[\![t]\!]\rho_A \mid \rho_A \in F\} \text{ dominated by } \{A[\![t']\!]\rho_A \mid \rho_A \in F\}$$

$\Downarrow$

$$\textstyle\bigvee_A\{A[\![t]\!]\rho_A \mid \rho_A \in F\} \leq_A \bigvee_A\{A[\![t']\!]\rho_A \mid \rho_A \in F\}$$

$\Downarrow$ $A[\![t]\!]$ and $A[\![t']\!]$ continuous ($\in [\mathrm{ENV}_A \longrightarrow A]$)

$$A[\![t]\!](\textstyle\bigvee F) \leq_A A[\![t']\!](\bigvee F)$$

$\Downarrow$

$$A[\![t]\!]\sigma_A \leq_A A[\![t']\!]\sigma_A$$

It remains to show the other direction $t \leq_A t' \Rightarrow t \leq'_A t'$.
Suppose $A[\![t]\!]\rho_A \leq_A [\![t']\!]\rho_A$ for all $\rho_A \in \mathrm{ENV}_A$. Given a closed substitution ρ and a $\rho_A \in \mathrm{ENV}_A$ we show $A[\![t\rho]\!]\rho_A \leq_A A[\![t'\rho]\!]\rho_A$. The substitution lemma directly gives us $A[\![t\rho]\!]\rho_A = A[\![t]\!](\rho_A \circ \rho)$ and similar for t' so because $\rho_A \circ \rho$ just is another A-assignment ($\in \mathrm{ENV}_A$) we are done. □

This result is closely related with the notion of substitutive relations as presented by Hennessy in [Hen83]:

A relation R over $REC_\Sigma(X)$ is *substitutive* if for all $t, u \in REC_\Sigma(X)$:

$$t\ R\ u \textit{ iff } \text{ for all closed syntactic substitutions } \rho,\ t\rho\ R\ u\rho$$

There Hennessy actually indicate the proposition above referring to [DNH84].

Proposition 9.1.7 When restricted to REC_Σ the preorder $\leq_A$ is algebraic provided for all $t \in FREC_\Sigma$, $A[\![t]\!]$ is a compact element of A.

Proof We show for given $t, u \in REC_\Sigma$:

$$A[\![t]\!] \leq_A A[\![u]\!] \textit{ iff } \forall t' \in \text{Fin}(t) \exists u' \in \text{Fin}(u).\ A[\![t']\!] \leq_A A[\![u']\!]$$

Each implication is proven separately.

if: If we show that $A[\![u]\!]$ is a ub for $A[\![\text{Fin}(t)]\!]$ the implication follows because $A[\![t]\!]$ by theorem 9.1.3 is a lub for $A[\![\text{Fin}(t)]\!]$. Let an arbitrary element $a \in A[\![\text{Fin}(t)]\!]$ be given. This means there is a $t' \in \text{Fin}(t)$ with $A[\![t']\!] = a$. By the antecedent of the implication there is a $u' \in \text{Fin}(u)$ such that $A[\![t']\!] \leq_A A[\![u']\!]$. $u' \in \text{Fin}(u)$ only if $u' \preceq u$ so $A[\![u']\!] \leq_A A[\![u]\!]$ and by the transitivity of $\leq_A$ then $a \leq_A A[\![u]\!]$.

only if: Assume $A[\![t]\!] \leq A[\![u]\!]$ and let a $t' \in \text{Fin}(t)$ be given. As above this implies $A[\![t']\!] \leq_A A[\![t]\!]$ and therefore also $A[\![t']\!] \leq_A A[\![u]\!]$. $A[\![u]\!]$ is also the lub for $A[\![\text{Fin}(u)]\!]$ so actually $A[\![t']\!] \leq_A \bigvee_A A[\![\text{Fin}(u)]\!]$. By the proviso of the proposition $A[\![t']\!] \in \text{Fin}(A)$. I.e., $A[\![t']\!]$ is compact wherefore $A[\![t']\!] \leq_A a$ for some $a \in A[\![\text{Fin}(u)]\!]$ or equally $A[\![t']\!] \leq_A A[\![u']\!]$ for some $u' \in \text{Fin}(u)$. □

A Σ-domain is *finitary* if the map $A[\![\_]\!] : REC_\Sigma \longrightarrow A$ when restricted to $FREC_\Sigma$ is surjective onto $\text{Fin}(A)$.

Corollary 9.1.8 If a Σ-domain, A, is finitary then the preorder over $REC_\Sigma(X)$ is substitutive and when restricted to REC_Σ it is algebraic.

9.1.2 Contexts

When considering a language a context, $\mathcal{C}[\,]$, is normally thought of as an expression with zero or more "holes", to be filled by some other expression of the language. Strictly speaking $\mathcal{C}[\,]$ is not an expression of the language, but if we think of a "hole" as a special constant symbol, a context will be an expression of the language extended with this constant. We illustrate the idea on the language of recursive expressions, $REC_\Sigma(X)$, built over the signature Σ.

We let the special constant symbol # (assumed not to be in Σ) take the rôle of a "hole". The set of $REC_\Sigma(X)$-contexts is then simply $REC_{\Sigma\cup\{\#\}}(X)$, written $REC_{\Sigma\#}(X)$ for short. Notice $REC_\Sigma \subseteq REC_{\Sigma\#}$. The context $\mathcal{C} \in REC_{\Sigma\#}(X)$ filled with (the context) $t \in REC_{\Sigma\#}(X)$, is denoted $\mathcal{C}[t]$ and defined by structural induction:

$$f[t] = \begin{cases} t & \text{if } f = \# \\ f & \text{otherwise (any other constant)} \end{cases}$$

$x[t] = x$

$f(\mathcal{C}_1, \ldots, \mathcal{C}_k)[t] = f(\mathcal{C}_1[t], \ldots, \mathcal{C}_k[t])$ for every $f \in \Sigma$ of arity $k \geq 1$.

$(rec\,x.\,\mathcal{C})[t] = rec\,x.\,(\mathcal{C}[t])$

Notice that, as opposed to syntactic substitution, free variables of t may become bound when filled in to $\mathcal{C}$. Also observe that if $t \in REC_\Sigma(X)$ then $\mathcal{C}[t] \in REC_\Sigma(X)$.

The advantage of considering contexts as ordinary expressions of an enlarged language is that it allows us to use the syntactic precongruence $\preceq$ on contexts just as we do on ordinary expressions. Recall that $\preceq$ is defined to be the least Σ-precongrueence over $REC_\Sigma(X)$ which satisfy

$$\Omega \preceq t$$
$$t[rec\,x.\,t/x] \preceq rec\,x.\,t$$

Clearly the least Σ-precongruence over $REC_{\Sigma\#}(X)$ which satisfies the two rules above will agree with $\preceq$ on REC_Σ so for convenience we shall make no distinction between them.

Lemma 9.1.9 If $\mathcal{C}$ and $\mathcal{C}'$ are $FREC_\Sigma(X)$-contexts and t, u are $REC_\Sigma(X)$-contexts then

a) $t \preceq u$ implies $\mathcal{C}[t] \preceq \mathcal{C}[u]$

b) $\mathcal{C} \preceq \mathcal{C}'$ implies $\mathcal{C}[t] \preceq \mathcal{C}'[t]$

Since $REC_\Sigma(X) \subseteq REC_{\Sigma\#}(X)$ the lemma of course applies for $t, u \in REC_\Sigma(X)$ (or $t, u \in FREC_\Sigma(X)$) too.

Proof a) By induction on the structure of $\mathcal{C}$.

$\mathcal{C} = f \neq \#$ or $\mathcal{C} = x$: Here we have $\mathcal{C}[t] = \mathcal{C} = \mathcal{C}[u]$.

$\mathcal{C} = \#$: Then $\mathcal{C}[t] = t \preceq u = \mathcal{C}[u]$.

$\mathcal{C} = f(\mathcal{C}_1, \ldots, \mathcal{C}_k)$:
$$\begin{aligned} \mathcal{C}[t] &= f(\mathcal{C}_1[t], \ldots, \mathcal{C}_k[t]) && \text{definition of } \_[t] \\ &\preceq f(\mathcal{C}_1[u], \ldots, \mathcal{C}_k[u]) && \text{hypothesis and definition of } \preceq \\ &= \mathcal{C}[u] \end{aligned}$$

b) Induction on the length of the proof of $\mathcal{C} \preceq \mathcal{C}'$. For the basis either $\mathcal{C} = \Omega$ or $\mathcal{C} = \mathcal{C}'$. The latter case is trivial and in the former we have $\mathcal{C}[t] = \Omega[t] = \Omega \preceq \mathcal{C}'[t]$. In the inductive step $\mathcal{C} \preceq \mathcal{C}'$ can because $\mathcal{C}' \in FREC_{\Sigma\#}$ only mean $\mathcal{C} = f(\mathcal{C}_1, \ldots, \mathcal{C}_k) \preceq f(\mathcal{C}'_1, \ldots, \mathcal{C}'_k) = \mathcal{C}$ where $\mathcal{C}_i \preceq \mathcal{C}'_i$ for $i = 1, \ldots, k$. The result then follows similar as in a). □

Lemma 9.1.10 Suppose $\mathcal{C}$ is a $FREC_\Sigma(X)$-context and $t \in REC_\Sigma(X)$. Then $u \in \text{Fin}(\mathcal{C}[t])$ implies there is a $FREC_\Sigma(X)$-context $\mathcal{C}' \preceq \mathcal{C}$ and a $t' \in \text{Fin}(t)$ such that $u \preceq \mathcal{C}'[t']$.

Proof By induction on the structure of $\mathcal{C}$. Recall $u \in \text{Fin}(\mathcal{C}[t])$ means $u \preceq \mathcal{C}[t]$ and $u \in FREC_\Sigma(X)$.

$\mathcal{C} = f \neq \#$ or $\mathcal{C} = x$: Then $\mathcal{C}[t] = \mathcal{C}$ and u equals Ω of $\mathcal{C}$. Letting $\mathcal{C}' = u$ and $t' = \Omega \in \mathrm{Fin}(t)$ we have $u = \mathcal{C}' = \mathcal{C}'[\Omega] = \mathcal{C}'[t']$.

$\mathcal{C} = \#$: I.e., $u \in \mathrm{Fin}(\mathcal{C}[t]) = \mathrm{Fin}(t)$. Choose $\mathcal{C}' = \#$ and $t' = u \in \mathrm{Fin}(t)$. Then $u = t' = \#[t'] = \mathcal{C}'[t']$.

$\mathcal{C} = f(\mathcal{C}_1, \ldots, \mathcal{C}_k)$ for an $f \in \Sigma$: Here we have $\mathcal{C}[t] = f(\mathcal{C}_1[t], \ldots, \mathcal{C}_k[t])$ so inspecting the definition of $\preceq$ we see $u \preceq \mathcal{C}[t]$ implies $u = f(u_1, \ldots, u_k)$ where $u_i \preceq_i \mathcal{C}_i[t]$ for $i = 1, \ldots, k$. By hypothesis of induction there for each $i = 1, \ldots, k$ is a $t'_i \in \mathrm{Fin}(t)$ and a $FREC_\Sigma(X)$-context $\mathcal{C}'_i \preceq \mathcal{C}_i$ such that $u_i \preceq \mathcal{C}'_i[t'_i]$. Since $\mathrm{Fin}(t)$ is directed there is ub $t' \in \mathrm{Fin}(t)$ for $t'_1, \ldots, t'_k$. By lemma 9.1.9 then $u_i \preceq \mathcal{C}'_i[t']$ for each i and because $\preceq$ is a Σ-precongruence we then have $f(u_1, \ldots, u_k) \preceq f(\mathcal{C}'_i[t'], \ldots \mathcal{C}'_k[t'])$. Letting $\mathcal{C}' = f(\mathcal{C}'_1, \ldots, \mathcal{C}'_k)$, $\mathcal{C}'$ is then a $FREC_\Sigma(X)$-context with $\mathcal{C}' \preceq \mathcal{C}$ and $u \preceq \mathcal{C}'[t']$.

□

9.1.3 Σ-precongruences

Suppose $\mathcal{L}_{\Sigma'}$ is a language constructed from a signature Σ'. Given a preorder, $\sqsubseteq$, over $\mathcal{L}_{\Sigma'}$ and a $\Sigma \subseteq \Sigma'$ we denote the largest Σ-precongruence over $\mathcal{L}_{\Sigma'}$ contained in $\sqsubseteq$ by $\sqsubseteq^\Sigma$. I.e.,

a) $\sqsubseteq^\Sigma \subseteq \sqsubseteq$

b) $\sqsubseteq^\Sigma$ is a Σ-precongruence

c) $\sqsubseteq' \subseteq \sqsubseteq^\Sigma$ for any other Σ-precongruence, $\sqsubseteq'$, contained in $\sqsubseteq$

Now define $\sqsubseteq^{\Sigma\#} \subseteq \mathcal{L}_{\Sigma'} \times \mathcal{L}_{\Sigma'}$ by

$$t \sqsubseteq^{\Sigma\#} u \textit{ iff } \forall \mathcal{L}_\Sigma\text{-contexts } \mathcal{C}.\, \mathcal{C}[t] \sqsubseteq \mathcal{C}[u]$$

Proposition 9.1.11 $\sqsubseteq^{\Sigma\#} = \sqsubseteq^\Sigma$, i.e., $\sqsubseteq^{\Sigma\#}$ is the largest Σ-precongruence contained in $\sqsubseteq$.

Proof We show that $\sqsubseteq^{\Sigma\#}$ fulfills a)—c).

a) Assume $t \sqsubseteq^{\Sigma\#} u$. With $\mathcal{C} = \#$ we especially have $t = \mathcal{C}[t] \sqsubseteq \mathcal{C}[u] = u$.

b) Suppose $f \in \Sigma$ of arity k and assume $t_i \sqsubseteq^{\Sigma\#} u_i$ for $i = 1, \ldots, k$. Let a $\mathcal{L}_\Sigma$-context $\mathcal{C}$ be given. We shall show $\mathcal{C}[f(t_1, \ldots, t_k)] \sqsubseteq \mathcal{C}[f(u_1, \ldots, u_k)]$. For $i = 1, \ldots, k$ define $\mathcal{C}_i$ to be the $\mathcal{L}_\Sigma$-context $\mathcal{C}[f(u_1, \ldots, u_{i-1}, \#, t_{i+1}, \ldots, t_k)]$. An easy induction on the structure of $\mathcal{C}$ shows $\mathcal{C}_1[t_1] = \mathcal{C}[f(t_1, \ldots, t_k)]$, $\mathcal{C}_k[u_k] = \mathcal{C}[f(u_1, \ldots, u_k)]$ and $\mathcal{C}_i[u_i] = \mathcal{C}_{i+1}[t_{i+1}]$ for $i = 1, \ldots, k-1$. By assumption $\mathcal{C}_i[t_i] \sqsubseteq \mathcal{C}_i[u_i]$ for $i = 1, \ldots, k$. The result then follows by the transitivity of $\sqsubseteq$.

c) Given another Σ-precongruence $\sqsubseteq' \subseteq \sqsubseteq$ suppose $t \sqsubseteq' u$. For every $\mathcal{L}_\Sigma$-context $\mathcal{C}$ it is easy to show $\mathcal{C}[t] \sqsubseteq' \mathcal{C}[u]$ by induction on the structure of $\mathcal{C}$ using the fact that $\sqsubseteq'$ is a Σ-precongruence. Since $\sqsubseteq' \subseteq \sqsubseteq$ we by definition of $\sqsubseteq^{\Sigma\#}$ then have $t \sqsubseteq^{\Sigma\#} u$. □

With this lemma we easely get

Proposition 9.1.12 Let a preorder, $\sqsubseteq$, over $\mathcal{L}_{\Sigma'}$ be given together with signatures $\Sigma^1 \subseteq \Sigma^2 \subseteq \Sigma'$. Assume $\sqsubseteq^{\Sigma^1}$ agrees on $\mathcal{L}_{\Sigma'}$ with another preorder, $\sqsubseteq'$, which is a Σ^2-precongruence. Then $\sqsubseteq^{\Sigma^1}$ equals $\sqsubseteq^{\Sigma^2}$.

Proof $\sqsubseteq^{\Sigma^2} \subseteq \sqsubseteq^{\Sigma^1}$ by definition. To see the opposite inclusion assume $t \sqsubseteq^{\Sigma^1} u$. By the previous proposition 9.1.11 is enough to show $\mathcal{C}[t] \sqsubseteq \mathcal{C}[u]$ for every $\mathcal{L}_{\Sigma^2}$ context $\mathcal{C}$. So let an arbitrary $\mathcal{L}_{\Sigma^2}$-context $\mathcal{C}$ be given. By the assumption of the lemma $t \sqsubseteq^{\Sigma^1} u$ implies $t \sqsubseteq' u$. Since $\sqsubseteq'$ is a precongruence w.r.t. to the combinators of Σ^2 we can then by induction on the structure of $\mathcal{C}$ show $\mathcal{C}[t] \sqsubseteq' \mathcal{C}[u]$. Then again by the assumption of the lemma $\mathcal{C}[t] \sqsubseteq^{\Sigma^1} \mathcal{C}[u]$. Because $\sqsubseteq^{\Sigma^1}$ by definition is contained in $\sqsubseteq$ we actually have $\mathcal{C}[t] \sqsubseteq \mathcal{C}[u]$ as desired. □

Before proceeding with the useful theorem below we need a definition:

Definition 9.1.13 Given a preorder, $\sqsubseteq$, over a language $\mathcal{L}$ and a subset $A \subseteq \mathcal{L}$. $\mathcal{L}$ is said to be A-*expressive* w.r.t. $\sqsubseteq$ *iff* for every $t \in \mathcal{L}$ there exists a *characteristic context* $\mathcal{C}_t[\,]$ such that

$$\forall u \in A.\, t \sqsubseteq^c u \textit{ iff } \mathcal{C}_t[t] \sqsubseteq \mathcal{C}_t[u]$$

where $\sqsubseteq^c$ is the largest precongruence w.r.t. to the combinators of $\mathcal{L}$ contained in $\sqsubseteq$. If $A = \mathcal{L}$ then $\mathcal{L}$ is simply said to be expressive w.r.t. $\sqsubseteq$.

Theorem 9.1.14 Let $\sqsubseteq$ be an algebraic preorder over REC_Σ containing the syntactic preorder $\preceq$. If $FREC_\Sigma$ is Fin(t)-expressive w.r.t $\sqsubseteq$ (restricted to $FREC_\Sigma$) for every $t \in REC_\Sigma$ then $\sqsubseteq^\Sigma$ is algebraic too.

Proof Given $t, u \in REC_\Sigma$ we show

$$t \sqsubseteq^\Sigma u$$
$$\Updownarrow$$
$$\forall t' \in \text{Fin}(t) \exists u' \in \text{Fin}(u).\, t' \sqsubseteq^\Sigma u'$$

$\Uparrow$: Assume $\forall t' \in \text{Fin}(t) \exists u' \in \text{Fin}(u).\, t' \sqsubseteq^\Sigma u'$. In order to have $t \sqsubseteq^\Sigma u$ it is by proposition 9.1.11 enough to show $\mathcal{C}[t] \sqsubseteq \mathcal{C}[u]$ for any given $FREC_\Sigma$-context $\mathcal{C}$. So suppose $\mathcal{C}$ is such a context. Let a $t'' \in \text{Fin}(\mathcal{C}[t])$ be given. By lemma 9.1.10 there is a $FREC_\Sigma$-context $\mathcal{C}' \preceq \mathcal{C}$ and a $t' \in \text{Fin}(t)$ such that $t'' \preceq \mathcal{C}'[t']$. By assumption there is a $u' \in \text{Fin}(u)$ with $t' \sqsubseteq^\Sigma u'$ and so also $\mathcal{C}'[t'] \sqsubseteq \mathcal{C}'[u']$ according to proposition 9.1.11. Clearly $\mathcal{C}'[u'] \in FREC_\Sigma$ and from $u' \preceq u$ it follows by lemma 9.1.9 that $\mathcal{C}'[u'] \preceq \mathcal{C}[u'] \preceq \mathcal{C}[u]$ so we actually have $\mathcal{C}'[u] \in \text{Fin}(\mathcal{C}[u])$. $\preceq \subseteq \sqsubseteq$ and the transitivity of $\sqsubseteq$ gives $t'' \sqsubseteq \mathcal{C}'[u']$. Hence for every $t'' \in \text{Fin}(\mathcal{C}[t])$ we have found a $u'' \in \text{Fin}(\mathcal{C}[u])$ such that $t'' \sqsubseteq u''$. Because $\sqsubseteq$ is algebraic this implies $\mathcal{C}[t] \sqsubseteq \mathcal{C}[u]$ as we wanted.

$\Downarrow$: Assume $t \sqsubseteq^\Sigma u$ and let a $t' \in \text{Fin}(t)$ be given. We shall find a $u' \in \text{Fin}(u)$ such that $t' \sqsubseteq^\Sigma u'$. Since $t' \in FREC_\Sigma$ and $FREC_\Sigma$ is Fin(t)-expressive there (for this t') is a $FREC_\Sigma$ context, $\mathcal{C}$, such that for all $u' \in \text{Fin}(u)$

$$\mathcal{C}[t'] \sqsubseteq \mathcal{C}[u'] \textit{ iff } t' \sqsubseteq^\Sigma u'$$

Let $\mathcal{C}$ be such a characteristic context for t'. We just have to find a $u' \in \text{Fin}(u)$ such that $\mathcal{C}[t'] \sqsubseteq \mathcal{C}[u']$. Since $t' \preceq t$ we by lemma 9.1.9 have $\mathcal{C}[t'] \preceq \mathcal{C}[t]$ and because $\mathcal{C}$ is a $FREC_\Sigma$-context this implies $\mathcal{C}[t'] \in \text{Fin}(\mathcal{C}[t])$. From $t \sqsubseteq^\Sigma u$ we by proposition 9.1.11 especially have $\mathcal{C}[t] \sqsubseteq \mathcal{C}[u]$ and by the algebraicity of $\sqsubseteq$ we deduce there must be a $u'' \in \text{Fin}(\mathcal{C}[u])$ such that $\mathcal{C}[t'] \sqsubseteq u''$. Using lemma 9.1.10 we find a $\mathcal{C}' \preceq \mathcal{C}$ and a $u' \in \text{Fin}(u)$ with $u'' \preceq \mathcal{C}'[u']$. By lemma 9.1.9 $u'' \preceq \mathcal{C}'[u'] \preceq \mathcal{C}[u']$ and from $\preceq \subseteq \sqsubseteq$ and transitivity of $\sqsubseteq$ we obtain $\mathcal{C}[t'] \sqsubseteq \mathcal{C}[u']$ as desired. $\square$

Proposition 9.1.15 Given a preorder $\sqsubseteq$ over $REC_{\Sigma'}$ extended to the open terms of $REC_{\Sigma'}(X)$ in the substitutive way. Suppose $\Sigma \subseteq \Sigma'$ and $\sqsubseteq$ identifies expressions equal up to rename of bound variables. Then $\sqsubseteq^\Sigma$ is substitutive.

Proof Given t and u we show

$$t \sqsubseteq^\Sigma u \quad \textit{iff} \quad \forall \rho \text{ (closed)}.\, t\rho \sqsubseteq^\Sigma u\rho$$

only if: For a particular ρ it will by proposition 9.1.11 do to show that for any closed ρ' and Σ-context, $\mathcal{C}$, we have $(\mathcal{C}[t\rho])\rho' \sqsubseteq (\mathcal{C}[u\rho])\rho'$ ($\mathcal{C}$ might contain free variables). Let such a context and syntactic substitutions be given. Because $FV(t\rho) = FV(u\rho) = \emptyset$ there is a closed context $\mathcal{C}'$ such that

$$\mathcal{C}'[t\rho] = (\mathcal{C}[t\rho])\rho' \quad \text{and} \quad \mathcal{C}'[u\rho] = (\mathcal{C}[u\rho])\rho'$$

Further more $\mathcal{C}'$ must be incapable of binding variables since it steems from the Σ-context $\mathcal{C}$.
Now $t \sqsubseteq^\Sigma u$ implies $\forall \rho$ (closed)$\forall\Sigma$-contexts $\mathcal{C}'.\,(\mathcal{C}'[t])\rho \sqsubseteq (\mathcal{C}'[u])\rho$. Since $\mathcal{C}'$ above is closed and $\mathcal{C}'$ cannot bind any variables, $(\mathcal{C}'[t])\rho$ must equal $\mathcal{C}'[t\rho]$ under $\sqsubseteq$; similar for u. The result then follows.

if: Assume $t\rho \sqsubseteq^\Sigma u\rho$ for all closed ρ. Given a context, $\mathcal{C}$, and some closed syntactic substitution, ρ', we show $\mathcal{C}[t]\rho' \sqsubseteq \mathcal{C}[u]\rho'$. Since $\mathcal{C}$ does not bind variables we must have $(\mathcal{C}[t\rho'])\rho' = (\mathcal{C}[t])\rho'$ and similar for u. As a particular case of the assumption we have $(\mathcal{C}[t\rho'])\rho' \sqsubseteq (\mathcal{C}[u\rho'])\rho'$ and are then done. $\square$

9.1.4 Obtaining Algebraic Complete Partial Orders

The algebraic cpos we are after can be obtained in a uniform way, so the construction of them will be presented here in a more general set-up.

Suppose $\langle P, \sqsubseteq \rangle$ is a preordered set and ϕ is a function $\phi : P \longrightarrow \mathcal{P}(P)\backslash\emptyset$ which is extended to $\mathcal{P}(P) \longrightarrow \mathcal{P}(P)$ in the natural way: $\phi(s) = \bigcup_{p\in s} \phi(p)$ for every $s \subseteq P$ ($\phi(\emptyset) = \emptyset$).

Also let there be given a collection, Π, of subsets of P—i.e., $\Pi \subseteq \mathcal{P}(P)$. About Π it is assumed that

- if $s \in \Pi$ and $p \in s$ then also $\{p\} \in \Pi$
- every nonempty $S \subseteq \Pi$ has lub w.r.t. $\subseteq$: $\bigcup S$ in Π—i.e $\bigcup S = \bigcup_{s\in S} s$ (Π closed under union)

- there is an element $s_\Pi \in \Pi$ with $\phi(s_\Pi) \subseteq \phi(s)$ for every $s \in \Pi$

Furthermore we shall assume that ϕ and $\sqsubseteq$ are interrelated such that for every $p \in s$ and $s \in \Pi$:

a) $\{q \in P \mid q \sqsubseteq p\}$ is finite

b) $q \in \phi(p)$ implies $q \sqsubseteq p$

c) $\exists q \in \phi(p).\, p \sqsubseteq q$

Finally define:

$\Phi \subseteq \mathcal{P}(P)$ to be the set $\{\phi(s) \in \mathcal{P}(P) \mid s \in \Pi\}$

In order to facilitate the overview it is intended to make use of symbols such that

$p, q, \ldots \in P$
$s, t, \ldots \in \Pi$ and $S, T, \ldots \subseteq \Pi$
$a, b, \ldots \in \Phi$ and $A, B, \ldots \subseteq \Phi$

The idea is now to make Φ into an algebraic cpo by ordering it under inclusion.

Lemma 9.1.16 $\langle \Phi, \subseteq \rangle$ is a cpo with least element $\phi(s_\Pi)$ and every nonempty subset A of Φ has a lub: $\bigcup A$. Furthermore for every nonempty $S \subseteq \Pi$ we have $\phi(\bigcup S) = (\bigcup_{s \in S} \phi(s) =)\bigcup \phi(S) \in \Phi$.

Proof Because $\phi(s_\Pi) \subseteq \phi(t)$ for every $t \in \Pi$ it is a $\subseteq$-least element of Φ. Now let a nonempty subset A of Φ be given. Of course $\bigcup A$ is a lub for A if it belongs to Φ. To see this notice at first that by definition of Φ there for each $a \in A$ exists a $s_a \in \Pi$ such that $a = \phi(s_a)$. Since ϕ is a natural extension we then get

$$\bigcup A = \bigcup_{a \in A} a = \bigcup_{a \in A} \phi(s_a) = \phi(\bigcup_{a \in A} s_a)$$

and because Π is closed under union and $A \neq \emptyset$ then $\bigcup_{a \in A} s_a \in \Pi$, so we conclude $\bigcup A \in \Phi$. From this it also appears that $\phi(\bigcup S) = \bigcup \phi(S) \in \Phi$ for every $\emptyset \neq S \subseteq \Pi$. $\square$

We are now concerned with the compact elements of Φ.

Recall that D is a directed set if it is nonempty and for all $d, d' \in D$ there exists an ub in D. An element $a \in \Phi$ is then compact if for every directed subset D and Φ such that $a \subseteq \bigcup D$ there is a $d \in D$ with $a \subseteq d$.

Lemma 9.1.17 The compact elements of $\langle \Phi, \subseteq \rangle$ is those $\phi(s)$, where $s \in \Pi$ and s is a finite set.

Notice that by a) and b) $\phi(s)$ must be finite too when s is.

Proof It is standard that a finite set $\phi(s) = a \in \Phi$ is compact when the partial order is $\subseteq$ and lub is $\bigcup$: Let $D \subseteq \Phi$ be a directed set such that $a \subseteq \bigcup D$. Then for each $p \in a$ we can select a $d_p \in D$ with $p \in d_p$. Denote the set of those d_p's by $D_a \subseteq D$. Since a is finite D_a must be finite too and has an ub $d \in D$ because D is directed, so $a \subseteq \bigcup D_a \subseteq d \in D$.

Now Φ is not a ordinary subset of $\mathcal{P}(P)$—it is induced from ϕ and a $\Pi \subseteq \mathcal{P}(P)$, so it is less standard to see that all compact elements $\phi(s) \in \Phi$ are such that s is a finite set—for instance it could be that ϕ "collapsed" an infinite set into a finite. But assume on the contrary there is a compact element $\phi(s) = a \in \Phi$ where s is infinite.

The idea will be to construct an increasing infinite chain D: $d_0, d_1, \ldots, d_i, \ldots$ with $a = \bigcup D$. Because a is compact and D is directed there will then be a $d \in D$ such that $a \subseteq d$. Since D is increasing then also $d \subset d'$ for a $d' \in D$. From $d' \subseteq \bigcup D = a$ then $a \subset a$—a contradiction.

We now construct the infinite increasing chain. Since s is infinite it contains a countable infinite subset $u = \{p_n\}_{n \in I\!N}$. For each $n \in I\!N$ define:

$$
\begin{aligned}
t_n &= \{p_i \in u \mid i \leq n\} \\
s_n &= \{p \in s \mid \forall j > n.\, p_j \not\sqsubseteq p\} \\
d_n &= \phi(s_n \cup t_n)
\end{aligned}
$$

By the assumption of Π each element of s is contained in Π as singleton sets, and Π is closed under union, so $s_n \cup t_n \in \Pi$ and $d_n \in \Phi$ for every $n \in I\!N$. Clearly $t_n \subseteq t_{n+1}$ and $s_n = \{p \in s \mid p_{n+1} \not\sqsubseteq p\} \cap s_{n+1} \subseteq s_{n+1}$ so because ϕ is $\subseteq$-monotone it follows that $D = \{d_n\}_{n \in I\!N}$ forms a nondecreasing chain in Φ.

To see that D in fact forms an increasing chain it is then enough for each $n \in I\!N$ to find an $m > n$ such that $d_n \neq d_m$.
Let n be given. By a) $\{q \in P \mid q \sqsubseteq p\}$ is finite for every $p \in t_n$. From the finiteness of t_n we see that $\{q \mid \exists p \in t_n.\, q \sqsubseteq p\}$ is finite too, so because $u \setminus t_n$ is infinite there must be a $p_m \in u \setminus t_n$ with $\forall p \in t_n.\, p_m \not\sqsubseteq p$. It follows that $m > n$ and by definition of s_n then $\forall p \in s_n.\, p_m \not\sqsubseteq p$, so we actually have

$$\forall p \in s_n \cup t_n.\, p_m \not\sqsubseteq p \tag{9.2}$$

Using c) we can now find a $q \in \phi(p_m)$ with $p_m \sqsubseteq q$. From (9.2) and the transitivity of $\sqsubseteq$ we conclude $\forall p \in s_n \cup t_n.\, q \not\sqsubseteq p$. By b) we then see that $q \notin \phi(p)$ for all $p \in s_n \cup t_n$ which, because $d_n = \phi(s_n \cup t_n) = \bigcup_{p \in s_n \cup t_n} \phi(p)$ implies $q \notin d_n$. Since $q \in \phi(p_m) \subseteq \phi(s_m \cup t_m) = d_m$ we then get $d_n \neq d_m$ as desired.

It remains to show $a = \bigcup D$. $\supseteq$ follows from a being a ub for D and to see $\subseteq$ let a $q \in a$ be given. Because $a = \phi(s)$ there must be a $p \in s$ with $q \in \phi(p)$. If $p \in u$ then $p = p_n$ for a $n \in I\!N$ and clearly then $q \in \phi(p_n) \subseteq d_n$. If on the other hand $p \in s \setminus u$ we know from a) that there only is finitely many $p_i \in u$ with $p_i \sqsubseteq p$. Suppose p_n is the last member of u with $p_n \sqsubseteq p$. Then for all $i > n$, $p_i \not\sqsubseteq p$ wherefore $p \in s_n$ and $q \in \phi(p) \subseteq \phi(s_n) \subseteq d_n$. In both cases $q \in d_n$ for some $d_n \in D$, so because $\bigcup D$ is a ub for D we arrive at $q \in \bigcup D$. □

Proposition 9.1.18 $\langle \Phi, \subseteq \rangle$ is an algebraic cpo.

Proof From the previous two lemmas we know that $\langle \Phi, \subseteq \rangle$ is a cpo and how the compact elements look like. So let an element $a \in \Phi$ be given. We shall show that a is the lub of the compact elements below a—i.e., that $a = \bigcup D_a$, where $D_a = \{\phi(s) \in \Phi \mid \phi(s) \subseteq a, s \text{ is finite}\}$. a being a ub for D_a gives $\bigcup D_a \subseteq a$ and to see $a \subseteq \bigcup D_a$ let a $q \in a$ be given. Then $q \in \phi(\{p\})$ for some $p \in s_a \in \Pi$, where $a = \phi(s_a)$. Hence $\phi(\{p\}) \subseteq a$ and because $\{p\}$ is finite, $\phi(\{p\})$ is compact. Therefore $\phi(\{p\}) \in D_a$ and $q \in \phi(\{p\}) \subseteq \bigcup D_a$. □

Proposition 9.1.19 Let Φ_1 and Φ_2 be two algebraic cpos constructed as above. Then $\Phi = \{\langle \phi_1(s), \phi_2(t) \rangle \mid s \in \Pi_1, t \in \Pi_2 \text{ and } t \subseteq s\}$ also is an algebraic cpo ordered under $\subseteq$ (component wise) with least element $\langle \phi_1(s_{\Pi_1}), \phi_2(s_{\Pi_2}) \rangle$ and every nonempty $D \subseteq \Phi$ has a lub $\bigcup D = \langle \bigcup D_1, \bigcup D_2 \rangle \in \Phi$, where $D_i = \{d_i \mid \langle d_1, d_2 \rangle \in D\}$. The compact elements of Φ are those $\langle \phi_1(s), \phi_2(t) \rangle \in \Phi$ where s and t are finite sets.

Proof $\Phi \subseteq \Phi_1 \times \Phi_1$ and the result can be derived from Hennessy [Hen88a, page 123]. At the first glance the result may seem obvious, but it has to be ensured that the lub actually belongs to Φ. $\phi_i(\bigcup D_i) = \bigcup \phi(D_i)$ is important here. Also the compact elements must be dealt with. □

9.2 Denotational Set-up

In this section we present two sets of models. One set will be the extension of the models of the previous two chapters: the models corresponding to the operational G-semantics ($\precsim_{\mathsf{G}}$) and the P_{or}-model for the precongruence ($\precsim^c_{\mathsf{G}}$) over RBL. For convenience we shall denote such a extended model by M_* where $*$ can be either *or* or G (a set of direct tests). M^p_* on the other hand will denote a model from the other set of models corresponding to the semantics ($\sqsubseteq_{\mathsf{G}}$) where the experimenter records prefix's of sequences. The domains of the M^p_* (M_*) models will be denoted A^p_* (A_*) and the operators corresponding to the combinators of the different languages in question will follow the same notational scheme.

9.2.1 The Recursive Languages

In the first section we have seen different pleasant consequences of having domains where the compact elements are denotable/ reachable. The goal will therefore be to extend the domains of the models from the preceding chapters to deal with "infinity" while at the same time enforcing constraints which ensures the reachability. The first subgoal is easely attained simply by considering infinite sets of pomsets instead of finite. Recalling that the different denotational maps were based on the canonical map, $\wp$, we get a clue for the second subgoal. At first we look at what pomsets we can get by $\wp$. Here we shall lean on a result of Grabowski [Gra81] which essentially states that the sets of pomsets generated from the singleton pomsets and ε by sequential and parallel composition exactly are the N-free pomsets.

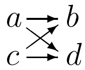
Definition 9.2.1 $P_{N\text{-free}}$*-Property for Pomsets*

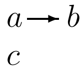
A pomset $\mathbf{p}$ is said to have the $P_{N\text{-free}}$-property , $P_{N\text{-free}}(\mathbf{p})$ *iff* for all x, x', y, y' in X_p we have:

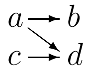
$$\text{if} \quad \begin{matrix} x <_p x' \\ co_p \quad co_p \\ y <_p y' \end{matrix} \quad \text{and } x <_p y' \text{ then} \quad y \leq_p x'$$

If a pomset $\mathbf{p}$ has the $P_{N\text{-free}}$-property we say that $\mathbf{p}$ is N-free.

We shall say that a $\mathcal{P}(\mathbf{P})$-refinement, ϱ, is N-free *iff* $\mathbf{p}$ is N-free for all $\mathbf{p} \in \varrho(a)$ and $a \in \Delta$. Similar a particular refinement for a lpo p, π_p, is N-free *iff* $\pi_p(x)$ is N-free for all $x \in X_p$. □

Example: 

and 

are N-free, but 

is not.

Gischer [Gis88] also calls these pomsets for the series-parallel pomsets and give an alternative and clear proof of the result, which (slightly modified for our set-up) can be formulated:

Theorem 9.2.2 For all pomsets $\mathbf{p}$:

$$P_{N\text{-free}}(\mathbf{p}) \textit{ and } \mathbf{p} \neq \varepsilon \textit{ iff } \exists E \in DBL.\ \wp(E) = \{\mathbf{p}\}$$

Because $\wp(E_0 \oplus E_1)$ equals the union of $\wp(E_0)$ and $\wp(E_1)$ we immediately get:

Corollary 9.2.3 If P is a finite and nonempty set of N-free pomsets such that $\varepsilon \notin P$ then $\exists E_P \in BL.\ \wp(E_P) = P$.

On top of the canonical map the relevant δ_*-closure were used top obtain the denotation. This suggests to let the elements of A_* be sets of pomsets which are obtained as the δ_*-closure of a set of N-free nonempty pomsets. As already argued in the introduction to this chapter, information of the M_*-models must be incorporated when it comes to the M_*^p-models for the semantics concerning prefix. Using the π-closure of pomsets to capture the idea of prefixes of sequences it appears that elements of A_*^p should be pairs where the second component is an element of A_* and the first component is the δ_*- and π-closure of a nonempty set of N-free pomsets with the additional constraint that this set of N-free pomsets shall be a superset of the other set which the second component is a δ_*-closure of. The additional constraint originates in the fact that if a maximal sequence can be recorded then so can any prefix of it. As we have seen in corollary 7.3.4 and proposition 8.2.3 the P_{G} and P_{or} properties are both hereditary and dot synthesizable. By proposition 6.4.5 and proposition 6.4.11 δ_* and π then commute so it make sense to talk about the δ_*-/ π-closure of a set. Formally

$$A_* = \{\delta_*(t) \mid t \subseteq \mathbf{P}_{N\text{-free}}, \varepsilon \notin t\}$$
$$A_*^p = \{\langle \delta_*\pi(s), \delta(t) \rangle \mid s, t \subseteq \mathbf{P}_{N\text{-free}}, \varepsilon \notin t \subseteq s \neq \emptyset\}$$

We shall often make use of the observation that $t \subseteq s \Rightarrow \delta_*(t) \subseteq \delta_*(s) \Rightarrow \delta_*(t) \subseteq \delta_*\pi(s)$ because δ_* is $\subseteq$-monotone and because in general $\mathbf{p} \in \pi(\mathbf{p})$.

We use the results of subsection 9.1.4 of this chapter to show that A_* and A_*^p are algebraic cpos. To this end notice that A_* and A_*^p can be written such that

$$
\begin{array}{ll}
A_* = \Phi_2 & \\
A_*^p = \{\langle \phi_1(s), \phi_2(t)\rangle \mid \phi_1(s) \in \Phi_1, \phi_2(t) \in \Phi_2, t \subseteq s\} & \\
\text{where} & \\
\Phi_1 = \{\phi_1(s) \mid s \in \Pi_1\} & \Phi_2 = \{\phi_2(t) \mid t \in \Pi_2\} \\
\phi_1 = \delta_* \circ \pi & \phi_2 = \delta_* \\
\Pi_1 = \{s \subseteq \mathbf{P}_{N\text{-free}} \mid s \neq \emptyset\} & \Pi_2 = \{t \subseteq \mathbf{P}_{N\text{-free}} \mid \varepsilon \notin t\}
\end{array}
$$

Clearly $s_{\Pi_1} = \{\varepsilon\} \in \Pi_1$ and $s_{\Pi_2} = \emptyset \in \Pi_2$ are elements such that $\phi_i(s_{\Pi_i}) \subseteq \phi_i(s)$ for every $s \in \Pi_i$ and $i = 0, 1$.
Recall that we for each pomset, $\mathbf{p}$, have a associated multiplicity function $m_{\mathbf{p}}$ which for each $a \in \Delta$ gives the number of elements in $\mathbf{p}$ that are labelled a. We saw that $\mathbf{p} \leq \mathbf{q}$ *iff* $\forall a \in \Delta.\, m_{\mathbf{p}}(a) \leq m_{\mathbf{q}}(a)$ defined a partial order on pomsets. Since we only are dealing with finite pomsets a moments reflection shows that $\{\mathbf{p}' \in \mathbf{P} \mid \exists \mathbf{q}.\, \mathbf{p}' \preceq \mathbf{q} \sqsubseteq \mathbf{p}\}$ is finite (given a multiplicity function, m', which only differs from 0 on finitely many $a \in \Delta$, there is only finitely many m below m', and for each such m there is only a finite number of pomsets $\mathbf{p}$ with $m_{\mathbf{p}} = m$). Obviously $\mathbf{q} \in \phi_i(\mathbf{p})$ implies $m_{\mathbf{q}} \leq m_{\mathbf{p}}$ for $i = 0, 1$. Since $\delta_*(\mathbf{p}) \neq \emptyset$ for the pomset properties we are dealing with and $\mathbf{q} \in \delta_*(\mathbf{p})$ only differs from $\mathbf{p}$ by the ordering of elements we have $m_{\mathbf{q}} = m_{\mathbf{p}}$ here. In the case of π we have $\mathbf{p} \in \pi(\mathbf{p})$ so we conclude that there for $i = 0, 1$ exists a $\mathbf{q} \in \phi_i(\mathbf{p})$ such that $m_{\mathbf{p}} \leq m_{\mathbf{q}}$. With the multiplicity preoder over pomsets and the other definitions above we from the results of the first section get:

Proposition 9.2.4 $\langle A_*, \subseteq\rangle$ and $\langle A_*^p, \subseteq\rangle$ (component wise) are algebraic cpos with least elements $\emptyset$ and $\langle\{\varepsilon\}, \emptyset\rangle$ respectively. The compact elements are those $\delta_*(s) \in A_*$ and $\langle \delta_*\pi(s), \delta_*(t)\rangle \in A_*^p$ where s and t are finite sets. Every nonempty $D \subseteq A_*$ has a lub: $\bigvee_* D = \bigcup D \in A_*$ and similar every nonempty $D \subseteq A_*^p$ has a lub $\bigvee_*^p D = \langle \bigcup D_1, \bigcup D_2\rangle \in A_*^p$ where $D_i = \{d_i \mid \langle d_1, d_2\rangle \in D\}$ for $i = 0, 1$.

The next step is to equip the algebraic cpos with operators corresponding to the different combinators of the languages in question.

In order to be in keeping with the notation we used in the previous chapters we will deviate from Hennessy by letting $RBL_\Omega^{rec}(X)$ denote the set of recursive expressions over RBL. I.e., in terminology of Hennessy $RBL_\Omega^{rec}(X)$ would be $REC_\Sigma(X)$ where the signature Σ is $a, ;, \oplus, \|$ and $[\varrho]$. The set of syntactic finite expressions $(FREC_\Sigma(X))$ will be denoted $RBL_\Omega(X)$. RBL_Ω^{rec} is the set of recursive processes over RBL, i.e., the closed expressions of $RBL_\Omega^{rec}(X)$. We use similar notation for the recursive expressions over BL, e.g., BL_Ω is the set of syntactic finite processes of $BL_\Omega^{rec}(X)$. The recursion combinators will be assumed to have lower precedence than the other combinators of the language in question.

Definition 9.2.5 Assume $d = \langle P, Q\rangle$ and $d_i = \langle P_i, Q_i\rangle$ for $i = 0, 1$ are elements of A^p_*. Then the operators of the M^p_* models are defined as follows:

$$\begin{aligned}
\Omega^p_* &= \langle\{\varepsilon\}, \emptyset\rangle \\
a^p_* &= \langle\{\varepsilon, a\}, \{a\}\rangle \\
d_0 ;^p_* d_1 &= \langle P_0 \cup Q_0 \cdot P_1, Q_0 \cdot Q_1\rangle \\
d_0 \oplus^p_* d_1 &= \langle P_0 \cup P_1, Q_0 \cup Q_1\rangle \\
d_0 \parallel^p_* d_1 &= \langle \delta_*(P_0 \times P_1), \delta_*(Q_0 \times Q_1)\rangle \\
d[\varrho]^p_{or} &= \langle \delta_{or}\pi(P{<}\wp(\varrho){>}), \delta_{or}(Q{<}\wp(\varrho){>})\rangle
\end{aligned}$$

The operators of the M_* models are derived from those of the M^p_* simply by projecting the second component. I.e., if $P_0, P_1 \in A_*$ then $P_0 \parallel_* P_1$ equals $\delta_*(P_0 \times P_1)$.

Proposition 9.2.6 The operators in the definition above are well-defined.

Proof We give a proof for the operators on A^p_*. That the A_*-operators are well-defined is then easely derived.

- Ω^p_*: This constant equals $\langle\{\varepsilon\}, \emptyset\rangle = \langle\delta_*\pi(\{\varepsilon\}), \delta_*(\emptyset)\rangle$ which is a member of A^p_* because $\{\varepsilon\}, \emptyset \subseteq \mathbf{P}_{N\text{-free}}$ and $\varepsilon \notin \emptyset \subseteq \{\varepsilon\} \neq \emptyset$.
- a^p_*: From $a \in \mathbf{P}_{N\text{-free}}$ and $\varepsilon \notin \{a\} \neq \emptyset$ we see $a^p_* = \langle\{\varepsilon, a\}, \{a\}\rangle = \langle\delta_*\pi(\{a\}), \delta_*(\{a\})\rangle \in A^p_*$.

For the binary operators on A^p_* assume $d_0, d_1 \in A^p_*$. Then $d_0 = \langle\delta_*\pi(s_0), \delta_*(t_0)\rangle$ for some $s_0, t_0 \in \mathbf{P}_{N\text{-free}}$ such that $\varepsilon \notin t_0 \subseteq s_0 \neq \emptyset$. Similar for d_1.

- $;^p_*$: From a) of proposition 9.2.7 below and the distributivity of δ_* over $\cdot$ we immediately get: $d_0 ;^p_* d_1 = \langle\delta_*\pi(s_0 \cup t_0 \cdot s_1), \delta_*(t_0 \cdot t_1)\rangle$. By Grabowski $\mathbf{p} \cdot \mathbf{q}$ is N-free when $\mathbf{p}$ and $\mathbf{q}$ are (can also be deduced from lemma 9.2.9 and the observations on page 136). So $d_0 ;^p_* d_1 \in A^p_*$ then follows from
$$\begin{aligned}
\varepsilon \notin t_0 \cdot t_1 & \quad \text{because } \varepsilon \notin t_0, t_1 \\
\subseteq s_0 \cup t_0 \cdot s_1 & \quad \text{since } t_1 \subseteq s_1 \\
\neq \emptyset & \quad \text{by } s_0 \neq \emptyset
\end{aligned}$$
- $\oplus^p_*$: Immediate from the distributivity of δ_* over $\cup$ and proposition 9.2.7.
- $\parallel^p_*$: From proposition 6.4.4 and proposition 9.2.7 we directly get $d_0 \parallel^p_* d_1 = \langle\delta_*\pi(s_0 \times s_1), \delta_*(t_0 \times t_1)\rangle$. Because the parallel composition of N-free pomsets are N-free $d_0 \parallel^p_* d_1 \in A^p_*$ is then easely deduced from the assumptions of s_0, s_1, t_0 and t_1.

It remains to show that the $[\varrho]^p_*$ operator on A^p_{or} is well-defined. Let a $d \in A^p_{or}$ be given and assume $d = \langle\delta_{or}\pi(s), \delta_{or}(t)\rangle$ where $s, t \subseteq \mathbf{P}_{N\text{-free}}$ and $\varepsilon \notin t \subseteq s \neq \emptyset$. Using lemma 8.2.6 for the second component and d) of proposition 9.2.7 below for the first we get $d[\varrho]^p_* = \langle\delta_{or}\pi(s{<}\wp(\varrho){>}), \delta_{or}(t{<}\wp(\varrho){>})\rangle$.
$\varrho(a) \in BL$ for every $a \in \Delta$, so from corollary 9.2.3 $(\wp(\varrho))(a)$ is a set of N-free nonempty pomsets when $a \in \Delta$. Hence from lemma 9.2.9 we know that $s{<}\wp(\varrho){>}$ and $t{<}\wp(\varrho){>}$ are sets of N-free pomsets because s and t are assumed to be N-free too. $\wp(\varrho)$ is ε-free so we conclude that $d[\varrho]^p_* \in A^p_*$. □

The following proposition is useful not only for the proof of the proposition above but also for other to come.

Proposition 9.2.7 Let ϱ be an ε-free $\mathcal{P}(\mathbf{P})$-assignment and suppose P, Q and R are sets of pomsets such that $P \supseteq R$. Then

a) $\delta_*\pi(P) \cup \delta_*(R) \cdot \delta_*\pi(Q) = \delta_*\pi(P \cup R \cdot Q)$

b) $\delta_*\pi(P) \cup \delta_*\pi(Q) = \delta_*\pi(P \cup Q)$

c) $\delta_*(\delta_*\pi(P) \times \delta_*\pi(Q)) = \delta_*\pi(P \times Q)$

d) $\delta_{or}\pi((\delta_{or}\pi(P)){<}\varrho{>}) = \delta_{or}\pi(P{<}\varrho{>})$

Proof a) At first we deduce:

$$\pi(\mathbf{p} \cdot \mathbf{q}) = \pi(\mathbf{p}) \cup \{\mathbf{p}\} \cdot \pi(\mathbf{q}) \tag{9.3}$$

from proposition 6.2.6 and the observations direct before that proposition. We then get:

$\delta_*\pi(P) \cup \delta_*(R) \cdot \delta_*\pi(Q)$	
$= \delta_*(\pi(P) \cup R \cdot \pi(Q))$	δ_* distributes over $\cdot$ and $\cup$
$= \delta_*(\pi(P) \cup \pi(R) \cup R \cdot \pi(Q))$	$R \subseteq P$ and π is $\subseteq$-monotone
$= \delta_*(\pi(P) \cup \pi(R \cdot Q))$	by (9.3)
$= \delta_*\pi(P \cup R \cdot Q)$	π distributes over $\cup$

b) Follows from the distributivity of δ_* and π over $\cup$.

c) $\delta_*(\delta_*\pi(P) \times \delta_*\pi(Q))$	
$= \delta_*(\pi(P) \times \pi(Q))$	proposition 6.4.4
$= \delta_*\pi(P \times Q)$	π distributes over $\times$

d) $\delta_{or}\pi((\delta_{or}\pi(P)){<}\varrho{>})$	
$= \pi\delta_{or}((\delta_{or}\pi(P)){<}\varrho{>})$	δ_{or} and π commutes
$= \pi\delta_{or}((\pi(P)){<}\varrho{>})$	lemma 8.2.6 (ϱ is ε-free)
$= \delta_{or}\pi((\pi(P)){<}\varrho{>})$	δ_{or} and π commutes
$= \delta_{or}\pi(P{<}\varrho{>})$	lemma 9.2.8 below

□

Lemma 9.2.8 Let P be a set of pomset and ϱ a $\mathcal{P}(\mathbf{P})$-refinement. Then

$$\pi((\pi(P)){<}\varrho{>}) = \pi(P{<}\varrho{>})$$

Proof π is a natural extension to sets of pomsets so it will do to show:

$$\pi((\pi(\mathbf{p})){<}\varrho{>}) = \pi(\mathbf{p}{<}\varrho{>})$$

$\supseteq$: Immediate from $\mathbf{p} \in \pi(\mathbf{p})$.

$\subseteq$: Let a $\mathbf{q} \in \pi((\pi(\mathbf{p})){<}\varrho{>})$ be given. Then $\mathbf{q} \sqsubseteq \mathbf{r}$ for some $\mathbf{r} \in \mathbf{s}{<}\varrho{>}$ where $\mathbf{s} \sqsubseteq \mathbf{p}$. By definition of $\_{<}\varrho{>}$, $\mathbf{r} \in \mathbf{s}{<}\varrho{>}$ implies there is a ϱ-consistent p.ref. π_s for s with $\mathbf{r} = [s{<}\pi_s{>}]$. Since $\mathbf{s} \sqsubseteq \mathbf{p}$ we can by the alternative characterization of $\sqsubseteq$ find a representative p' of $\mathbf{p}$ such that $s = p'|_{X_s}$ and X_s is $\leq_{p'}$-downwards closed. $X_s \subseteq X_{p'}$ so we can extend π_s to a ϱ-consistent p.ref. $\pi_{p'}$ for p'. Because $s = p'|_{X_s}$ and $\pi_{p'}$ equals π_s on X_s we see $s{<}\pi_s{>} = p'{<}\pi_{p'}{>}|_{X_{s<\pi_s>}}$.

We now show that $X_{s<\pi_s>}$ is $\leq_{p'<\pi_{p'}>}$-downwards closed. Suppose $\langle x, x'\rangle \leq_{p'<\pi_{p'}>} \langle y, y'\rangle$ and $\langle y, y'\rangle \in X_{s<\pi_s>}$. By construction of $p'<\pi_{p'}>$ the former implies $x \leq_{p'} y$. The latter similarly implies $y \in X_s$. Since X_s is $\leq_{p'}$-downwards closed then $x \in X_s$. Now $x' \in X_{\pi_{p'}(x)}$ so because $\pi_{p'}$ equals π_s on X_s we also have $x' \in X_{\pi_s(x)}$. Hence $\langle x, x'\rangle \in X_{s<\pi_s>}$.

Using the alternative characterization of $\sqsubseteq$ again we conclude $[s<\pi_s>] \sqsubseteq [p'<\pi_{p'}>]$. From the transitivity of $\sqsubseteq$, $\mathbf{q} \sqsubseteq \mathbf{r} = [s<\pi_s>]$ and $[p'<\pi_{p'}>] \in \mathbf{p}'<\varrho> = \mathbf{p}<\varrho>$ we then get $\mathbf{q} \in \pi(\mathbf{p}<\varrho>)$ as desired. □

Lemma 9.2.9 Suppose P is a set of N-free pomsets and ϱ is a N-free $\mathcal{P}(\mathbf{P})$-refinement. Then $P<\varrho>$ is a set of N-free pomsets too.

Proof The lemma follows from $\mathbf{p}<\varrho>$ being a set of N-free pomsets when $\mathbf{p}$ is N-free. To see this it is clearly enough to show that $p<\pi_p>$ is N-free for any ϱ-consistent p.ref. π_p for p (also N-free). Of course π_p is N-free when ϱ is. The proof that $p<\pi_p>$ is N-free is by contradiction. Assume $p<\pi_p> = \langle X, \leq, \ell\rangle$ is not N-free. By construction of $p<\pi_p>$ this implies the existence of $\langle x, x'\rangle, \langle y, y'\rangle, \langle z, z'\rangle, \langle v, v'\rangle \in X$ such that

$$\begin{array}{ccc} \langle x, x'\rangle & < & \langle z, z'\rangle \\ co & & co \\ \langle y, y'\rangle & < & \langle v, v'\rangle \end{array} \quad \text{and } \langle x, x'\rangle < \langle v, v'\rangle \tag{9.4}$$

holds, but $\langle y, y'\rangle \not< \langle z, z'\rangle$.

We consider the different cases:

$(x = v, y \neq v)$ or $(x \neq v, y = v)$: If $x \not{co}_p\, y$ then by the construction of $p<\pi_p>$ this implies $\langle x, x'\rangle \not{co} \langle y, y'\rangle$—a contradiction to the assumption. To see $x \not{co}_p\, y$ assume w.l.o.g. $(x = v, y \neq v)$ holds. From $y \neq v$ and the construction of $p<\pi_p>$ we deduce that $\langle y, y'\rangle < \langle v, v'\rangle$ only can be due to $y <_p v$. $v = x$ then gives $y <_p x$ and so $x \not{co}_p\, y$.

$(x = v, x \neq z)$ or $(x \neq v, x = z)$: Similar as above we find $z \not{co}\, v$ which leads to a contradiction in the same way.

$x \neq v, y \neq v, x \neq z$: From (9.4) and the construction of $p<\pi_p>$ we derive

$$\begin{array}{cc} x <_p z & \\ co_p \quad co_p & \text{and } x <_p v \\ y <_p v & \end{array}$$

Hence $y <_p z$ follows from the N-freeness of p and then $\langle y, y'\rangle < \langle z, z'\rangle$—a contradiction to $\langle y, y'\rangle \not< \langle z, z'\rangle$

$x = v, y = v, x = z$: I.e., $x = v = y = z$ so this time (9.4) and the construction of $p<\pi_p>$ gives:

$$\begin{array}{cc} x' <_{\pi_p(x)} z' & \\ co_{\pi_p(x)} \quad co_{\pi_p(x)} & \text{and } x' <_{\pi_p(x)} v' \\ y' <_{\pi_p(x)} v' & \end{array}$$

By the N-freeness of π_p then also $y' <_{\pi_p(x)} z'$. Since $x = y = z$ the construction of $p<\pi_p>$ yields $\langle y, y'\rangle < \langle z, z'\rangle$—again a contradiction.

A careful examination of the cases above shows that they actually exhaust all possible combinations of $x = / \neq v$, $y = / \neq v$ and $x = / \neq z$. Each time we reached a contradiction so the assumption, $p{<}\pi_p{>}$ is not N-free, was wrong. $\Box$

Proposition 9.2.10 The operators of A^p_* and A_* are continuous.

Proof The continuity of the A_*-operators is easely derived from the continuity of the A^p_*-operators which we now deal with. Constants are continuous. For the binary operators it is enough to show that they are left and right continuous. To this end let D' be a nonempty subset of A^p_* and suppose $\langle P, Q\rangle$ is a member of A^p_*.

$;^p_*$: Right continuous: Let $D = \langle P, Q\rangle \,;^p_*\, D' = \{\langle P \cup Q \cdot P', Q \cdot Q'\rangle \mid \langle P', Q'\rangle \in D\}$. Then $D_1 = \{R_1 \mid \langle R_1, R_2\rangle \in D\} = \{P \cup Q \cdot P' \mid \langle P', Q'\rangle \in D'\} = \{P \cup Q \cdot P'_1 \mid P'_1 \in D'_1\} = P \cup Q \cdot D'_1$ where the last equation follows from $D'_1 \neq \emptyset$ which in turn is a consequence of $D' \neq \emptyset$. Also $D_2 = \{Q \cdot Q' \mid \langle P', Q'\rangle \in D'\} = Q \cdot D'_2$. We then have: $\bigvee^p_*(\langle P, Q\rangle \,;^p_*\, D') = \langle \bigcup D_1, \bigcup D_2\rangle = \langle \bigcup(P \cup Q \cdot D'_1), \bigcup(Q \cdot D'_2)\rangle = \langle P \cup Q \cdot (\bigcup D'_1), Q \cdot (\bigcup D'_2)\rangle = \langle P, Q\rangle \,;^p_*\, \bigvee^p_* D'$.

Left continuous: Here we have: $\bigvee^p_*(D' ;^p_* \langle P, Q\rangle) = \langle \bigcup\{P' \cup Q' \cdot P \mid \langle P', Q'\rangle \in D'\}, \bigcup\{Q' \cdot Q \mid \langle P', Q'\rangle \in D'\}\rangle = \langle (\bigcup_{\langle P', Q'\rangle \in D'} P') \cup (\bigcup_{\langle P', Q'\rangle \in D'} Q') \cdot P, \bigcup(D'_2 \cdot Q)\rangle = \langle (\bigcup D'_1) \cup (\bigcup D'_2) \cdot P, (\bigcup D'_2) \cdot Q\rangle = \langle \bigcup D'_1, \bigcup D'_2\rangle \,;^p_*\, \langle P, Q\rangle = (\bigvee^p_* D') \,;^p_*\, \langle P, Q\rangle$.

$\oplus^p_*$: Obvious left and right continuous since it just is the union of the respective components.

$\|^p_*$: Right continuous: With $D = \langle P, Q\rangle \|^p_* D'$ we have $D_1 = \delta_*(P \times D'_1)$ and $D_2 = \delta_*(Q \times D'_2)$. Because δ_* is the natural extension to sets we have $\bigcup D_1 = \delta_*(\bigcup(P \times D'_1)) = \delta_*(P \times \bigcup D'_1)$ and similar for $\bigcup D_2$. It follows that $\bigvee^p_*(\langle P, Q\rangle \|^p_* D') = \langle \delta_*(P \times \bigcup D'_1), \delta_*(Q \times \bigcup D'_2)\rangle = \langle P, Q\rangle \|^p_* \bigvee^p_* D'$.

Left continuous: Symmetric.

Only the $[\varrho]^p_{or}$-operator is missing. $\bigvee^p_*(D'[\varrho]^p_{or}) = \bigvee^p_{or}\{\langle \delta_{or}\pi(P'{<}\wp(\varrho){>}), \delta_{or}(Q'{<}\wp(\varrho){>})\rangle \mid \langle P', Q'\rangle \in D'\} = \langle \bigcup \delta_{or}\pi(D'_1{<}\wp(\varrho){>}), \bigcup \delta_{or}(D'_2{<}\wp(\varrho){>})\rangle$. Since both δ_{or} and π as well as ${<}\wp(\varrho){>}$ are natural extensions to sets we immediately get: $\bigvee^p_*(D'[\varrho]^p_{or}) =$
$\langle \delta_{or}\pi((\bigcup D'_1){<}\wp(\varrho){>}), \delta_{or}((\bigcup D'_2){<}\wp(\varrho){>})\rangle = \langle \bigcup D'_1, \bigcup D'_2\rangle[\varrho]^p_{or} = (\bigvee^p_{or} D')[\varrho]^p_{or}$ $\Box$

Now where we have showed that A_* and A^p_* are algebraic cpos and that the different operators are continuous on the respective domains, we are in a position to apply the results presented in the previous section.

So for $BL^{rec}_\Omega(X)$ we get the denotational maps:

$$A_{\mathsf{G}}[\![\_]\!] : BL^{rec}_\Omega(X) \longrightarrow [\mathrm{ENV}_{A_{\mathsf{G}}} \longrightarrow A_{\mathsf{G}}]$$
$$A^p_{\mathsf{G}}[\![\_]\!] : BL^{rec}_\Omega(X) \longrightarrow [\mathrm{ENV}_{A^p_{\mathsf{G}}} \longrightarrow A^p_{\mathsf{G}}]$$

and for $RBL^{rec}_\Omega(X)$:

$$A_{or}[\![\_]\!] : RBL^{rec}_\Omega(X) \longrightarrow [\mathrm{ENV}_{A_{or}} \longrightarrow A_{or}]$$
$$A^p_{or}[\![\_]\!] : RBL^{rec}_\Omega(X) \longrightarrow [\mathrm{ENV}_{A^p_{or}} \longrightarrow A^p_{or}]$$

$A^p_*[\![\_]\!]_1$ and $A^p_*[\![\_]\!]_2$ will be used to refer to the first and second component of $A^p_*[\![\_]\!]$ respectively. Notice that if E is a closed expression then $A_*[\![E]\!] = A^p_*[\![E]\!]_2 \subseteq A^p_*[\![E]\!]_1$.

9.2.2 The Syntactic Finite Sublanguages

In this subsection we shall lift some of the results we obtained in the preceding chapters for BL and RBL to the corresponding syntactic finite expressions of BL^{rec}_Ω and RBL^{rec}_Ω respectively, namely BL_Ω and RBL_Ω.

When we dealt with BL and RBL the canonical map, $\wp$, associating sets of pomsets to BL-expressions, served as basis for the denotational maps in a natural way. It appears to be difficult to extend this idea (and $\wp$) to say BL^{rec}_Ω, but an extension to BL_Ω seems manageable. How such an extension should be depends on what we are aiming at. If we want to use $\wp$ as basis for maps concerned with different kinds of maximal pomsets it might be natural to let $\wp(\Omega)$ be the empty set, $\emptyset$, because Ω represents the process we know nothing about. If on the other hand we also want information about possible prefixes it might be just as natural to associate ε with Ω because ε is prefix of any pomset. Our M^p_*-models consists of pairs of pomsets with the "old" denotations from the M_*-models as second component. We therefore arrive at the following:

Definition 9.2.11 The map $\wp^p : BL_\Omega \longrightarrow \mathcal{P}(\mathbf{P}) \times \mathcal{P}(\mathbf{P})$ is defined inductively:

$$\begin{aligned}
\wp^p(\Omega) &= \langle \{\varepsilon\}, \emptyset \rangle \\
\wp^p(a) &= \langle \{\varepsilon, a\}, \{a\} \rangle \\
\wp^p(E_0\,;E_1) &= \langle \wp^p_1(E_0) \cup \wp^p_2(E_0) \cdot \wp^p_1(E_1), \wp^p_2(E_0) \cdot \wp^p_2(E_1) \rangle \\
\wp^p(E_0 \oplus E_1) &= \langle \wp^p_1(E_0) \cup \wp^p_1(E_1), \wp^p_2(E_0) \cup \wp^p_2(E_1) \rangle \\
\wp^p(E_0 \parallel E_1) &= \langle \wp^p_1(E_0) \times \wp^p_1(E_1), \wp^p_2(E_0) \times \wp^p_2(E_1) \rangle
\end{aligned}$$

where $\wp^p_1(E) = P$ and $\wp^p_2(E) = Q$ if $\wp^p(E) = \langle P, Q \rangle$, i.e., $\wp^p_1$ and $\wp^p_2$ are the projections of $\wp^p$ to the first and second component respectively.
The ordinary canonical map $\wp$ is extended to BL_Ω by $\wp = \wp^p_2$.

Observe that $\forall E \in BL_\Omega.\ \wp^p_2(E) \subseteq \wp^p_1(E)$.

Example: From $\wp^p(\Omega\,;d) = \langle \{\varepsilon\}, \emptyset \rangle$ follows

$$\wp^p((a\,;b)\,;(\Omega\,;d \oplus c)) = \langle \{\varepsilon, a, a \rightarrow b, a \rightarrow b \rightarrow c\}, \{a \rightarrow b \rightarrow c\} \rangle$$

and

$$\wp^p((a\,;(\Omega\,;d) \oplus b)\,;c) = \langle \{\varepsilon, a, b, b \rightarrow c\}, \{b \rightarrow c\} \rangle$$

The following proposition justifies to think of $\wp^p_1$ as the canonical association of pomset prefixes of an expression.

Proposition 9.2.12

a) If $E \in BL$ then $\wp^p_1(E) = \pi(\wp(E))$.

b) If $E \in BL_\Omega$ then $\wp^p_1(E) = \pi(\wp^p_1(E))$.

Proof By structural induction on E using (9.3) and $\wp_2^p = \wp$ in the case of $E = E_0\,;E_1$. □

Clearly the definition of $\wp^p$ is designed with the denotations of the M_*^p-models in mind. An easy structural induction in fact shows:

Proposition 9.2.13 Given an $E \in BL_\Omega$ then $\wp^p(E) = \langle P, Q\rangle$ implies $\varepsilon \notin Q \subseteq P \neq \emptyset$ and P, Q are finite subsets of $\mathbf{P}_{N\text{-free}}$.

Having generalized $\wp$ to BL_Ω we seek results like theorem 7.3.7 and theorem 8.2.5 for M_* and M_*^p. The situation is diametrically opposite here since compositional definitions of the denotational maps are given and we want an alternative definitions using $\wp$ and $\wp^p$.

Proposition 9.2.14 For any $E \in BL_\Omega$:

a) $A_*[\![E]\!] = \delta_*(\wp(E))$

b) $A_*^p[\![E]\!]_i = \delta_*(\wp_i^p(E))$ for $i = 1, 2$

Proof a) By induction on the structure of E. In the case $E = \Omega$ we have $A_*[\![E]\!] = \emptyset = \delta_*(\emptyset) = \delta_*(\wp(E))$. The remaining cases goes similar as in the proof of theorem 7.3.7.

b) Since $\wp = \wp_2^p$ and $A_*[\![E]\!]$ equals $A_*^p[\![E]\!]_2$, a) also reads

$$A_*^p[\![E]\!]_2 = \delta_*(\wp_2^p(E)) \tag{9.5}$$

Then b) follows from

$$A_*^p[\![E]\!]_1 = \delta_*(\wp_1^p(E))$$

which is proven by induction on the structure of E.

$E = \Omega$: $A_*^p[\![E]\!] = \{\varepsilon\} = \delta_*(\{\varepsilon\}) = \delta_*(\wp_1^p(E))$.

$E = a$: $A_*^p[\![E]\!] = \{\varepsilon, a\} = \delta_*(\{\varepsilon, a\}) = \delta_*(\wp_1^p(E))$.

$E = E_0\,;E_1$: Here we have:

$$\begin{aligned}
A_*^p[\![E]\!]_1 &= A_*^p[\![E_0]\!]_1 \cup A_*^p[\![E_0]\!]_2 \cdot A_*^p[\![E_1]\!]_1 && \text{definition of } A_*^p[\![_]\!] \\
&= \delta_*(\wp_1^p(E_0)) \cup \delta_*(\wp_2^p(E_0)) \cdot \delta_*(\wp_1^p(E_1)) && \text{induction and (9.5)} \\
&= \delta_*(\wp_1^p(E_0) \cup \wp_2^p(E_0) \cdot \wp_1^p(E_1)) && \delta_{or} \text{ distributes over } \cup \text{ and } \cdot \\
&= \delta_*(\wp_1^p(E_0\,;E_1)) = \delta_*(\wp_1^p(E)) && \text{definition of } \wp^p
\end{aligned}$$

$E = E_0 \oplus E_1$ and $E = E_0 \parallel E_1$: Similar, but without use of (9.5).

□

From a) of the proposition, definition 7.3.6 and definition 8.2.4 we as expected see $[\![E]\!]_* = A_*[\![E]\!]$ when $E \in BL$.

A simple consequence of this proposition and proposition 9.2.12 is:

Corollary 9.2.15 $A_*^p[\![E]\!] = \langle \delta_*\pi(\wp(E)), \delta_*(\wp(E))\rangle$ for every $E \in BL$

With the results obtained so far we are now able to show that the different models are surjective.

Proposition 9.2.16 Every compact element of A^p_* and A_* is the denotation of a syntactic finite expression.

Proof The result for A_* is easely derived from the corresponding proof for A^p_*. To see this we for a given compact element $a \in A^p_*$ just find an expression $E \in BL_\Omega \subseteq RBL_\Omega$ such that $A_*[\![E]\!] = a$. So it will actually not be necessary to involve the refinement combinators in order to denote the compact elements of A^p_{or}. Recall at first that a is an element of A^p_* in the M^p_* model when

$$a = \langle \delta_*\pi(s), \delta_*(t)\rangle \tag{9.6}$$

where s and t are two sets of N-free pomsets such that $\varepsilon \notin t \subseteq s \neq \emptyset$. Also the the compact elements were characterized to be those where s and t are finite sets.

If u is an arbitrary finite and nonempty set of N-free pomset such that $\varepsilon \notin u$ we from the last corollary and corollary 9.2.3 deduce there exists an $E_u \in BL$ with

$$A^p_*[\![E_u]\!] = \langle \delta_*\pi(u), \delta_*(u)\rangle \tag{9.7}$$

Now let a compact element a like (9.6) be given. We deal with different cases of s and t:

$\varepsilon \notin s$ and $t = \emptyset$: Then we can find an $E_s \in BL$ fulfilling (9.7). Hence $E = E_s\,;\Omega \in BL_\Omega$ and we get:

$$\begin{aligned} A^p_*[\![E]\!] &= \langle \delta_*\pi(s) \cup \delta_*(s)\cdot\{\varepsilon\}, \delta_*(s)\cdot\emptyset\rangle && \text{definition of } A^p_*[\![\_]\!] \\ &= \langle \delta_*(\pi(s)\cup s), \emptyset\rangle \\ &= \langle \delta_*\pi(s), \delta_*(\emptyset)\rangle = \langle \delta_*\pi(s), \delta_*(t)\rangle && s \subseteq \pi_*(s) \end{aligned}$$

$\varepsilon \notin s$ and $t \neq \emptyset$: Because $\varepsilon \notin t$ and $s \neq \emptyset$ we can then find $E_s, E_t \in BL$ fulfilling (9.7). Therefore $E = (E_s;\Omega) \oplus E_t \in BL_\Omega$ and $A_*[\![E]\!] = A^p_*[\![E_s;\Omega]\!] \oplus^p_* A^p_*[\![E_t]\!] = \langle \delta_*\pi(s), \emptyset\rangle \oplus^p_* \langle \delta_*\pi(t), \delta_*(t)\rangle = \langle \delta_*\pi(s) \cup \delta_*\pi(t), \emptyset \cup \delta_*(t)\rangle = \langle \delta_*\pi(s\cup t), \delta_*(t)\rangle = \langle \delta_*\pi(s), \delta_*(t)\rangle$ where the last equation follows from $t \subseteq s$.

$s = \{\varepsilon\}$: Because $t \subseteq s$ and $\varepsilon \notin t$ we must have $t = \emptyset$ in this situation and $E = \Omega$ will do.

$\varepsilon \in s$ and $s \setminus \{\varepsilon\} \neq \emptyset$: Then no matter whether $t = \emptyset$ or $t \neq \emptyset$ we can as above find a $E' \in BL_\Omega$ such that $A^p_*[\![E']\!] = \langle \delta_*\pi(s \setminus \{\varepsilon\}), \delta_*(t)\rangle$. Letting $E = \Omega \oplus E'$ we get $A^p_*[\![E]\!] = \langle \delta_*\pi(\{\varepsilon\} \cup (s \setminus \{\varepsilon\})), \delta_*(\emptyset \cup t)\rangle = \langle \delta_*\pi(s), \delta_*(t)\rangle$.

Inspecting how s and t can be for compact elements like (9.6) of A^p_* we see that all cases are covered. □

As for RBL and BL we are able to establish a connection between RBL_Ω and BL_Ω via the map σ which we together with $\{\varrho\}$ extend to RBL_Ω as follows:

$$\begin{aligned} \Omega\sigma &= \Omega & \Omega\{\varrho\} &= \Omega \\ a\sigma &= a & a\{\varrho\} &= \varrho(a) \\ (E_0\,;E_1)\sigma &= E_0\sigma\,;E_1\sigma & (E_0\,;E_1)\{\varrho\} &= E_0\{\varrho\}\,;E_1\{\varrho\} \\ (E_0 \oplus E_1)\sigma &= E_0\sigma \oplus E_1\sigma & (E_0 \oplus E_1)\{\varrho\} &= E_0\{\varrho\} \oplus E_1\{\varrho\} \\ (E_0 \parallel E_1)\sigma &= E_0\sigma \parallel E_1\sigma & (E_0 \parallel E_1)\{\varrho\} &= E_0\{\varrho\} \parallel E_1\{\varrho\} \\ E[\varrho]\sigma &= (E\sigma)\{\varrho\} \end{aligned}$$

Proposition 9.2.17 For every $E \in RBL_\Omega$ we have:

a) $A_{or}[\![E]\!] = A_{or}[\![E\sigma]\!]$

b) $A^p_{or}[\![E]\!] = A^p_{or}[\![E\sigma]\!]$

Proof a) Since $A^p_{or}[\![\_]\!]_2$ equals $A_{or}[\![\_]\!]$ this is just a simple consequence of b).

b) The proof is by induction on the structure of E. The basis is immediate because $\Omega\sigma = \Omega$ and $a\sigma = a$. In the inductive step there are four cases:

$E = E_0 \,;\, E_1$: Then:

$$\begin{array}{lll} A^p_{or}[\![E]\!] & = A^p_{or}[\![E_0]\!] \;;^p_{or}\; A^p_{or}[\![E_1]\!] & \text{definition of } A^p_{or}[\![\_]\!] \\ & = A^p_{or}[\![E_0\sigma]\!] \;;^p_{or}\; A^p_{or}[\![E_1\sigma]\!] & \text{induction} \\ & = A^p_{or}[\![E_0\sigma \,;\, E_1\sigma]\!] & \text{definition of } A^p_{or}[\![\_]\!] \\ & = A^p_{or}[\![(E_0 \,;\, E_1)\sigma]\!] = A^p_{or}[\![E\sigma]\!] & \text{definition of } \sigma \end{array}$$

$E = E_0 \oplus E_1$ and $E = E_0 \parallel E_1$: Similar

$E = F[\varrho]$: In this case we have:

$$\begin{array}{lll} A^p_{or}[\![E]\!] & = (A^p_{or}[\![F]\!])[\varrho]^p_{or} & \text{definition of } A^p_{or}[\![\_]\!] \\ & = (A^p_{or}[\![F\sigma]\!])[\varrho]^p_{or} & \text{induction} \\ & = (A^p_{or}[\![(F\sigma)\{\varrho\}]\!]) & \text{lemma 9.2.18 and } F\sigma \in BL_\Omega \\ & = (A^p_{or}[\![(F[\varrho]\sigma)]\!]) = A^p_{or}[\![E]\!] & \end{array}$$

□

Lemma 9.2.18 If $E \in BL_\Omega$ then

a) $A_{or}[\![E\{\varrho\}]\!] = (A_{or}[\![E]\!])[\varrho]_{or}$

b) $A^p_{or}[\![E\{\varrho\}]\!] = (A^p_{or}[\![E]\!])[\varrho]^p_{or}$

Proof a) The proof is like lemma 8.2.7 for RBL but with the additional case Ω (see also b)).

b) Since $A_{or}[\![E]\!]$ equals $A^p_{or}[\![E]\!]_2$ we from a) and the definition of $[\varrho]^p_{or}$ deduce

$$A^p_{or}[\![E\{\varrho\}]\!]_2 = \delta_{or}(A^p_{or}[\![E]\!]_2 {<} \wp(\varrho) {>}) \tag{9.8}$$

With this we then by induction on the structure of $E \in BL_\Omega$ prove

$$A^p_{or}[\![E\{\varrho\}]\!]_1 = \delta_{or}\pi(A^p_{or}[\![E]\!]_1 {<} \wp(\varrho) {>})$$

from which b) then follows using (9.8).

$E = \Omega$: $A^p_{or}[\![\Omega\{\varrho\}]\!]_1 = A^p_{or}[\![\Omega]\!]_1 = \{\varepsilon\} = \delta_{or}\pi(\{\varepsilon\} {<} \wp(\varrho) {>}) = \delta_{or}\pi(A^p_{or}[\![\Omega]\!]_1 {<} \wp(\varrho) {>})$.

$E = a$: Then:

$$\begin{array}{ll} A^p_{or}[\![a\{\varrho\}]\!]_1 = A^p_{or}[\![\varrho(a)]\!]_1 & \text{definition of } \{\varrho\} \\ = \delta_{or}(\wp^p_1(\varrho(a))) & \varrho(a) \in BL \text{ and proposition 9.2.14} \\ = \delta_{or}\pi(\wp(\varrho(a))) & \wp^p_1 = \pi \circ \wp \text{—proposition 9.2.12} \\ = \delta_{or}\pi((\wp(\varrho))(a)) & \text{definition of } \wp(\varrho) \\ = \delta_{or}\pi(\{\varepsilon, a\} {<} \wp(\varrho) {>}) & \text{proposition 6.3.3} \\ = \delta_{or}\pi(A^p_{or}[\![a]\!]_1 {<} \wp(\varrho) {>}) & \end{array}$$

$E = E_0 \,;\, E_1$: We get:

$$
\begin{array}{ll}
A^p_{or}[\![E\{\varrho\}]\!]_1 = A^p_{or}[\![E_0\{\varrho\} \,;\, E_1\{\varrho\}]\!]_1 & \text{definition of } \{\varrho\} \\
= A^p_{or}[\![E_0\{\varrho\}]\!]_1 \cup A^p_{or}[\![E_0\{\varrho\}]\!]_2 \cdot A^p_{or}[\![E_1\{\varrho\}]\!]_1 & \text{definition of } A^p_{or}[\![\_]\!] \\
= \delta_{or}\pi(A^p_{or}[\![E_0]\!]_1 < \wp(\varrho) >) & \\
\quad \cup \delta_{or}(A^p_{or}[\![E_0]\!]_2 < \wp(\varrho) >) \cdot \delta_{or}\pi(A^p_{or}[\![E_1]\!]_1 < \wp(\varrho) >) & \text{induction and (9.8)} \\
= \delta_{or}\pi(A^p_{or}[\![E_0]\!]_1 < \wp(\varrho) > & \text{proposition 9.2.7 and} \\
\quad \cup A^p_{or}[\![E_0]\!]_2 < \wp(\varrho) > \cdot A^p_{or}[\![E_1]\!]_1 < \wp(\varrho) >) & A^p_{or}[\![E_0]\!]_2 \subseteq A^p_{or}[\![E_0]\!]_1 \\
= \delta_{or}\pi((A^p_{or}[\![E_0]\!]_1 \cup A^p_{or}[\![E_0]\!]_2 \cdot A^p_{or}[\![E_1]\!]_1) < \wp(\varrho) >) & \text{proposition 6.3.3} \\
= \delta_{or}\pi(A^p_{or}[\![E_0 \,;\, E_1]\!]_1) & \text{definition of } A^p_{or}[\![\_]\!]
\end{array}
$$

$E = E_0 \oplus E_1$ and $E = E_0 \parallel E_1$: Similar.

□

Proposition 9.2.19 The denotation of a syntactic finite expression is a compact element.

Proof The proof for the M^p_* models is exemplary for the corresponding for the M_* models. Suppose $E \in BL_\Omega$. Then

$$
\begin{array}{ll}
A^p_*[\![E]\!] = \langle \delta_*(\wp^P_1(E)), \delta_*(\wp(^p_2(E))) \rangle & \text{proposition 9.2.14} \\
\qquad = \langle \delta_*\pi(\wp^P_1(E)), \delta_*(\wp(^p_2(E))) \rangle & \text{proposition 9.2.12}
\end{array}
$$

By proposition 9.2.13 it then follows that $A^p_*[\![E]\!] \in \mathrm{Fin}(A^p_*)$. Now if $E \in RBL_\Omega$, that is $* = or$, then by proposition 9.2.17 $A^p_*[\![E]\!] = A^p_*[\![E\sigma]\!]$ and because $E\sigma \in BL_\Omega$ it follows that $A^p_{or}[\![E]\!]$ denotes a compact element in A^p_{or}. □

From this proposition and proposition 9.2.16 we immediately have:

Corollary 9.2.20 The different M_* and M^p_* models are finitary.

We end this section with a proposition corresponding to proposition 8.3.1 which gives a connection between the denotations of the M^p_{or}-model and the M^p_{w}-model.

Proposition 9.2.21 Suppose $E \in RBL_\Omega$. Then

a) $A_{\mathsf{w}}[\![E\sigma]\!] = \delta_{\mathsf{w}}(A_{or}[\![E]\!])$

b) $A^p_{\mathsf{w}}[\![E\sigma]\!]_i = \delta_{\mathsf{w}}(A^p_{or}[\![E]\!]_i)$ for $i = 1, 2$

Proof As in the preceding proofs a) is just the special case of b) with $i = 2$.

b) The case $i = 2$ follows exactly as the case $i = 1$:

$$
\begin{array}{ll}
A^p_{\mathsf{w}}[\![E\sigma]\!]_1 = \delta_{\mathsf{w}}(\wp^p_1(E\sigma)) & \text{proposition 9.2.14} \\
\qquad = \delta_{\mathsf{w}}\delta_{or}(\wp^p_1(E\sigma)) & \delta_{\mathsf{w}} \circ \delta_{or} = \delta_{\mathsf{w}} \text{ (from } P_{\mathsf{w}} \Rightarrow P_{or}) \\
\qquad = \delta_{\mathsf{w}}(A^p_{or}[\![E\sigma]\!]_1) & \text{proposition 9.2.14} \\
\qquad = \delta_{\mathsf{w}}(A^p_{or}[\![E]\!]_1) & \text{proposition 9.2.17}
\end{array}
$$

□

9.3 Operational Set-up

9.3.1 The Recursive Languages

The configuration languages are as earlier obtained by adding †. With refinement the set of recursive configuration expressions, $RCL^{rec}_{\Omega}(X)$, is then in the usual way defined as the least set C which satisfies:

$$\begin{array}{rl} \dagger \in C & \\ RBL^{rec}_{\Omega}(X) \subseteq C & \\ E_0 \,;\, E_1 \in C & \text{if } E_0 \in C \text{ and } E_1 \in RBL^{rec}_{\Omega}(X) \\ E_0 \parallel E_1 \in C & \text{if } E_0, E_1 \in C \end{array}$$

RCL^{rec}_{Ω} is the set of recursive process configurations, i.e., the closed configuration expressions of $RCL^{rec}_{\Omega}(X)$. $RCL_{\Omega}(X)$, RCL_{Ω}, $CL^{rec}_{\Omega}(X)$ etc. can then be considered as $RCL^{rec}_{\Omega}(X)$ restricted to the appropriate sublanguage.

The different extended labelled transition systems are all changed in the same way to cope with the new situation and further the change only affects the definition of internal steps. The following rules are added:

$$\begin{array}{l} \Omega \rightarrowtail \Omega \\ \Omega[\varrho] \rightarrowtail \Omega \\ rec\,x.\,E \rightarrowtail E[rec\,x.\,E/x] \end{array}$$

The intuition behind the first and last rule is explained by Hennessy [Hen88a, pages 202–203] and the rule in the middle is there mainly for proof technical reasons. It can easely be shown to have no operational effect. It is worth to notice that (modulo this rule) no extra rules are needed for refinement to cope with recursion. From the next example one can see how it works.

Example: Suppose $E = a \oplus a\,;x$ and ϱ is a BL-refinement with $\varrho(a) = b$ and $\varrho(b) = a$. Then the following scenario shows a possible evolvement of $F = (rec\,x.\,E)[\varrho]$:

$$\begin{array}{llll} F & \rightarrowtail (a \oplus a\,; rec\,x.\,E)[\varrho] & \rightarrowtail a[\varrho] \\ & \rightarrowtail (a\,; rec\,x.\,E)[\varrho] & \rightarrowtail b \\ & \rightarrowtail a[\varrho]\,; F & \\ & \rightarrowtail b\,; F & \\ & \stackrel{b}{\Longrightarrow} F \cdots & \end{array}$$

With a slight change $F = rec\,x.\,(E[\varrho])$ we instead have:

$$\begin{array}{ll} F & \rightarrowtail^{*} (a\,; F)[\varrho] \\ & \rightarrowtail^{*} (b\,; F[\varrho]) \\ & \stackrel{b}{\Longrightarrow} F[\varrho] \\ & \rightarrowtail^{*} (b\,; F[\varrho])[\varrho] \\ & \rightarrowtail^{*} (a\,; F[\varrho][\varrho]) \\ & \stackrel{a}{\Longrightarrow} (F[\varrho])[\varrho] \cdots \end{array}$$

The definition of $\lesssim_{\mathsf{G}}$ remains the same, but of course it is now defined over RBL_{Ω}^{rec} and BL_{Ω}^{rec} respectively. The new operational preorder, $\precapprox_{\mathsf{G}}$, from the introduction to the chapter can be formulated:

Definition 9.3.1 $\precapprox_{\mathsf{G}} \subseteq RBL_{\Omega}^{rec} \times RBL_{\Omega}^{rec}$ is defined

$$E_0 \precapprox_{\mathsf{G}} E_1 \;\; \textit{iff} \;\; \forall s \in \mathsf{G}^*.\, E_0 \stackrel{s}{\Rightarrow} \text{ implies } E_1 \stackrel{s}{\Rightarrow}$$

Restricting $\precapprox_{\mathsf{G}}$ appropriately to the different sublanguages gives the remaining preorders. □

Up till now we have got along with mainly structural induction. When it comes to recursion it will be convenient with the notion of the size of a step. For an internal step $E \rightarrowtail E'$ the size, m, will be indicated by a relation $\rightarrowtail_m$, i.e., $E \rightarrowtail_m E'$. $\rightarrowtail_0$ is the empty relation. Similar for external steps. m in $E \rightarrowtail_m E'$ can be thought of as stating that there is a proof of $E \rightarrowtail E'$ from the rules of $\rightarrowtail$ with no more than m stages. E.g., if $E_0 \rightarrowtail_m E_0'$ then $E_0 \parallel E_1 \rightarrowtail_{m+1} E_0' \parallel E_1'$. See [Win85] for more details.

9.3.2 The Syntactic Finite Sublanguages

Not all notions can be carried over directly to the extended languages with recursion. For instance it is difficult to make sense in talking about the behaviours of a process of BL_{Ω}^{rec}/ RBL_{Ω}^{rec} since a process now may continue infinitely. At least it is hard to see how the map Beh should be extended to say BL_{Ω}^{rec} and we will not find any use for it. It will later turn out that the different operational preorders are determinated by their restriction to the sublanguages of syntactic finite expression, i.e., BL_{Ω} and RBL_{Ω}. We will therefore now look at how the previous obtained results for BL and RBL can be lifted to BL_{Ω} and RBL_{Ω}. If a proposition need no reformulation (except e.g., BL should be replaced with BL_{Ω}) will in the sequel simply be referred by writing it as: proposition$_{\Omega}$.

Both proposition 7.2.3 and proposition 8.1.1 extends directly to proposition$_{\Omega}$ 7.2.3 and proposition$_{\Omega}$ 8.1.1.

As we also saw for the denotational set-up, the syntactic map, σ, removing refinements will be useful to establish connections from RBL_{Ω} to BL_{Ω} or when it comes to configurations, from RCL_{Ω} to CL_{Ω}. We extend σ from RCL to RCL_{Ω} in the same way as σ was extended from BL to RBL_{Ω} namely by letting $\Omega\sigma = \Omega$ and similar for $\{\varrho\}$ we let $\Omega\{\varrho\} = \Omega$. It should be clear that the maps $\sigma : RCL_{\Omega} \longrightarrow CL_{\Omega}$ and $\{\varrho\} : CL_{\Omega} \longrightarrow CL_{\Omega}$ when restricted to RCL and BL respectively gives the old maps.

The first important result we shall extend to RBL_{Ω} is proposition 8.1.2:

$$\forall E \in RBL_{\Omega}.\, E \stackrel{s}{\Rightarrow} \dagger \;\textit{iff}\; E\sigma \stackrel{s}{\Rightarrow} \dagger$$

According to our convention we shall refer to it as proposition$_{\Omega}$ 8.1.2. Of course some care has to be taken because we now have to deal with one more extra case in the proofs: Ω, but because of the additional rule $\Omega[\varrho] \rightarrowtail \Omega$, this only give rise to minor changes. We briefly comment on selected parts of the results used to prove proposition$_{\Omega}$ 8.1.2. In lemma$_\Omega$ 8.1.4 the case $E = \Omega$ is trivially true and the same case is also easy in lemma$_\Omega$ 8.1.3. Here however there is also the case $E = F[\varrho]$ where $F = \Omega$. It is exactly here the rule $\Omega[\varrho] \succ\!\!\longrightarrow \Omega$ is considered: Since $F[\varrho]\sigma = \Omega\sigma\{\varrho\} = \Omega\{\varrho\} = \Omega \succ\!\!\longrightarrow \Omega$ this goes through smoothly and in $(8.3)_\Omega$ one just use $\Omega\{\varrho\} = \Omega$. In lemma$_\Omega$ 8.1.7 $E = \Omega$ is similar to $E = a$ and in the following lemma it is even simpler.

If we go back and look at the proof of proposition$_\Omega$ 8.1.2 we see that the proposition just is a special case of a more general result which also implies:

Proposition 9.3.2 For every $E \in RBL_\Omega$ and $s \in \mathsf{G}^*$:

$$E \stackrel{s}{\Rightarrow} \ \textit{iff} \ E\sigma \stackrel{s}{\Rightarrow}$$

The only remaining purely operational results from the preceding chapters are lemma 7.4.5 and lemma 7.4.6 which carry over totally unchanged, because Ω is not directly involved in the rules for expressions of the form $E_0 \,;\, E_1$ and $E_0 \parallel E_1$.

9.4 Full Abstractness

In this section we connect the denotational semantics with the operational through full abstractness results which are obtained by lifting via algebraicity of the involved preorders the corresponding results for the (syntactic) finite sublanguages.

9.4.1 The Recursive Languages

As mentioned in the beginning of the chapter we are after the largest precongruence contained in the relevant preorder. There we were just concerned with the ordinary combinators of the language in question, but of course we want the obtained preorder to be a precongruence w.r.t. to the recursive combinators too. If this shall make sense the operational preorders have to be extended to open expressions. This is usually done in what might be called the substitutive way:

$E_0 \lesssim_{\mathsf{G}} E_1$ *iff* for every closed syntactic substitution ρ, $E_0\rho \lesssim_{\mathsf{G}} E_1\rho$

$E_0 \sqsubseteq\!\!\!\!\sim_{\mathsf{G}} E_1$ *iff* for every closed syntactic substitution ρ, $E_0\rho \sqsubseteq\!\!\!\!\sim_{\mathsf{G}} E_1\rho$

The largest precongruence over $BL^{rec}_\Omega(X)$ contained in $\lesssim_{\mathsf{G}}$ will as usual be denoted $\lesssim^c_{\mathsf{G}}$. Similar for the other preorders. We can now formulate:

Theorem 9.4.1 The following denotations are fully abstract:

a) $A_{\mathsf{G}}[\![\_]\!]$ on $BL^{rec}_\Omega(X)$ w.r.t. $\lesssim_{\mathsf{G}}$

b) $A_{or}[\![\_]\!]$ on $RBL^{rec}_\Omega(X)$ w.r.t. $\lesssim^c_{\mathsf{W}}$

c) $A^p_{\mathsf{G}}[\![\_]\!]$ on $BL^{rec}_\Omega(X)$ w.r.t. $\sqsubseteq\!\!\!\!\sim^c_{\mathsf{G}}$

d) $A^p_{or}[\![\_]\!]$ on $RBL^{rec}_\Omega(X)$ w.r.t. $\sqsubseteq\!\!\!\!\sim^c_{\mathsf{W}}$

Proof oThe denotational preorders $\trianglelefteq_*$ and $\trianglelefteq_*^p$ are qua induced by the denotational maps, precongruences w.r.t. all the combinators—the recursion combinators inclusive. By proposition 9.1.12 it is then enough to show the theorem to hold where the operational precongruences now are understood to be the largest w.r.t. the ordinary combinators.

By corollary 9.1.8 the associated (denotational) induced preorders are then substitutive as well as algebraic. The different operational preorders are by definition substitutive and by proposition 9.1.15 then so are the associated precongruences. Hence if we can manage to show that the involved operational precongruences are algebraic and agrees with the denotational preorders on the syntactic finite sublanguages (closed expressions) the theorem then follows.

From theorem 9.4.18 we know that $\lesssim_{\mathsf{G}}$ and $\sqsubseteq\!\!\!\sim_{\mathsf{G}}$ are algebraic over RBL_Ω^{rec} and therefore also over BL_Ω^{rec}. Since theorem 9.4.19 gives the corresponding results full abstractness for the syntactic finite sublanguages it only remains to show the operational precongruences (w.r.t. the ordinary combinators) are algebraic:

a) $\lesssim_{\mathsf{G}}$ is algebraic on BL_Ω^{rec} and agrees on the finite expressions, BL_Ω, with $\trianglelefteq_{\mathsf{G}}$ so $\lesssim_{\mathsf{G}}$ is also a precongruence w.r.t. the ordinary combinators.

c) $\sqsubseteq\!\!\!\sim_{\mathsf{G}}$ is algebraic on BL_Ω^{rec} and by theorem 9.4.22 BL_Ω is $\{E' \in BL_\Omega \mid L(E') \subseteq A\}$-expressive w.r.t. $\sqsubseteq\!\!\!\sim_{\mathsf{G}}$ for every finite subset A of Δ. For every $E \in BL_\Omega^{rec}$, $L(E)$ is finite and if $E' \in \mathrm{Fin}(E)$ then $L(E') \subseteq L(E)$. Hence BL_Ω is $\mathrm{Fin}(E)$-expressive for all $E \in BL_\Omega^{rec}$ and from theorem 9.1.14 it then follows that $\sqsubseteq\!\!\!\sim_{\mathsf{G}}^c$ is algebraic over BL_Ω^{rec} and this case is done.

b) and d) Both $\lesssim_{\mathsf{W}}$ and $\sqsubseteq\!\!\!\sim_{\mathsf{W}}$ are algebraic and by theorem 9.4.22 RBL_Ω is expressive w.r.t. both preorders. Theorem 9.1.14 then gives us that $\lesssim_{\mathsf{W}}^c$ and $\sqsubseteq\!\!\!\sim_{\mathsf{W}}^c$ are algebraic over RBL_Ω^{rec}.

□

In the above proof we have just seen that $\lesssim_{\mathsf{G}}$ and $\sqsubseteq\!\!\!\sim_{\mathsf{G}}$ as well as $\lesssim_{\mathsf{W}}^c$ and $\sqsubseteq\!\!\!\sim_{\mathsf{W}}^c$ are algebraic. With arguments similar to those in the beginning of the proof we then from corollary 9.4.21 deduce:

Corollary 9.4.2 For $E_0, E_1 \in RBL_\Omega^{rec}(X)$ we have

$$(E_0 \lesssim_{\mathsf{W}}^c E_1) \Rightarrow (E_0 \lesssim_{\mathsf{G}} E_1) \Rightarrow (E_0 \lesssim_{\mathsf{W}} E_1)$$

and

$$(E_0 \sqsubseteq\!\!\!\sim_{\mathsf{W}}^c E_1) \Rightarrow (E_0 \sqsubseteq\!\!\!\sim_{\mathsf{G}} E_1) \Rightarrow (E_0 \sqsubseteq\!\!\!\sim_{\mathsf{W}} E_1)$$

Using the same considerations as in section 8.4 also:

Corollary 9.4.3 For all $E_0, E_1 \in RBL_\Omega^{rec}(X)$:

$E_0 \lesssim_{\mathsf{G}}^c E_1$ *iff* $E_0 \lesssim_{\mathsf{W}}^c E_1$ $\qquad$ $E_0 \sqsubseteq\!\!\!\sim_{\mathsf{G}}^c E_1$ *iff* $E_0 \sqsubseteq\!\!\!\sim_{\mathsf{W}}^c E_1$

$E_0 \approx_{\mathsf{G}}^c E_1$ *iff* $A_{or}[\![E_0]\!] = A_{or}[\![E_1]\!]$ $\qquad$ $E_0 \sqsupseteq\!\!\!\sqsubseteq\!\!\!\sim_{\mathsf{G}}^c E_1$ *iff* $A_{or}^p[\![E_0]\!] = A_{or}^p[\![E_1]\!]$

Since $A_{or}[\![E]\!] = A^p_{or}[\![E]\!]_2$ we from this corollary and the expressions of (9.1) in the introduction to the chapter get:

Corollary 9.4.4 For all $E_0, E_1 \in RBL^{rec}_\Omega(X)$:

$$E_0 \sqsubseteq^c_{\mathsf{G}} E_1 \Rightarrow E_0 \lesssim^c_{\mathsf{G}} E_1$$

and in general the implication does not hold in the other direction.

We shall now prove all the propositions we used to get the different full abstractness results.

In connection with the denotational set-up for recursion we have already meat the syntactic preoder $\preceq$. There it was used a relation telling what processes, E, that might be thought of as approximations to a process, F, possibly with recursion constructors, i.e., $E \preceq F$. We saw that the denotation of F was the limit of all its syntactic finite approximations. When it comes to the operational set-up here, $\preceq$ will play a similar rôle. Recall that $\preceq$ was defined to be the least relation over $RBL^{rec}_\Omega(X)$ satisfying:

$$E \preceq E \qquad \Omega \preceq E \qquad E[rec\,x.\,E/x] \preceq E$$

$$\frac{E \preceq F, F \preceq G}{E \preceq G} \qquad \frac{E_0 \preceq F_0, E_1 \preceq F_1}{\begin{array}{c} E_0\,;E_1 \preceq F_0\,;F_1 \\ E_0 \oplus E_1 \preceq F_0 \oplus F_1 \\ E_0 \parallel E_1 \preceq F_0 \parallel F_1 \end{array}} \qquad \frac{E \preceq F}{E[\varrho] \preceq F[\varrho]}$$

$\preceq$ is extended to $RCL^{rec}_\Omega(X)$ simply by letting $\preceq$ be the least relation over $RCL^{rec}_\Omega(X)$ which satisfies the rules above. Notice that in this way we may only have $E \preceq F[\varrho]$ if E and F comes from $RBL^{rec}_\Omega(X)$. It is also important to notice that $\dagger \preceq E$ implies $E = \dagger$ and that $\preceq$ contains the old precongruence over $RBL^{rec}_\Omega(X)$.

Having extended $\preceq$ to $RCL^{rec}_\Omega(X)$ we at first show that if E is an approximation of F then F can do all the sequences E can. A stronger formulation of this is

Lemma 9.4.5 Suppose $E, E' \in RCL^{rec}_\Omega$. Then

$$E \succeq E' \overset{s}{\Rightarrow} F' \text{ implies } \exists F.\, E \overset{s}{\Rightarrow} F \succeq F'$$

Proof As usual by induction on the size of $\overset{s}{\Rightarrow}$ using the analogous lemma 9.4.9 for single steps. □

Before proving lemma 9.4.9 consider the situation where $E \succeq E' \succ\!\!\longrightarrow F'$. We cannot expect that E immediately can do an internal step and evolve into F with $F \succeq F'$. This is because $E' \preceq E$ can imply that some of the recursive subexpressions of E have been "unwound" by $\preceq$ in order to obtain an expression equal to E' (up to Ω at some places in E'). By the recursion rule for $\succ\!\!\longrightarrow$ it is possible to do one unwinding, so given $E' \preceq E$ we would ideally like to unwind E by internal steps to a E'' which equals E' up to Ω. Then we could be sure that whatever internal (or external for that matter) step E' could do, E'' would be able to do a similar. There is however the snag about it that the definition of $\succ\!\!\longrightarrow$ *does not* open up for unwinding in the right hand argument of the ;-combinator and neither in the arguments of the $\oplus$-combinator. The situation is closely related with the one in chapter 8 where we wished to "perform" the substitutions of the refinement combinator of an expression. Thought by the experience there we define a subpreorder, $\preceq^u$, of $\preceq$ as the least relation over $RCL_\Omega^{rec}(X)$ which can be inferred from the rules:

$$E \preceq^u E \qquad \frac{E \preceq^u F, F \preceq^u G}{E \preceq^u G}$$

$$\Omega \preceq^u E$$

$$\frac{E_0 \preceq^u F_0, E_1 \preceq F_1}{E_0\,;E_1 \preceq^u F_0\,;F_1} \qquad \frac{E_0 \preceq F_0, E_1 \preceq F_1}{E_0 \oplus E_1 \preceq^u F_0 \oplus F_1}$$

$$\frac{E_0 \preceq^u F_0, E_1 \preceq^u F_1}{E_0 \parallel E_1 \preceq^u F_0 \parallel F_1} \qquad \frac{E \preceq^u F}{E[\varrho] \preceq^u F[\varrho]}$$

Example: $(rec\,y.\,E)\,;(a \parallel rec\,x.\,(a \parallel x)) \preceq^u (rec\,y.\,E)\,;rec\,x.\,a \parallel x$ but $(a \parallel rec\,x.\,(a \parallel x))\,;rec\,y.\,E \not\preceq^u (rec\,x.\,a \parallel x)\,;rec\,y.\,E$

This definition of $\preceq^u$ deserves several remarks:

- The requirement that $\preceq^u$ shall be over $RCL_\Omega^{rec}(X)$ has the implication that an inference rule only may be used when for the consequent, $E \preceq^u F$, it is ensured that $E, F \in RCL_\Omega^{rec}(X)$
- $\preceq^u \subseteq \preceq$
- The preorder $\preceq$ is used in the premisses of the ;- and $\oplus$-inference rule just in order to capture the unwindings which cannot be done by internal steps.
- There is no rule for $rec\,x.$. This reflects that the expressions are equal up to Ω (except of course in connection with ; and $\oplus$)

By the last remark the following useful lemma is reasonable:

Lemma 9.4.6

a) $\dagger \preceq^u F$ ($a \preceq^u F$) implies $F = \dagger$ ($F = a$)

b) $E_0\,;E_1 \preceq^u F$ implies $F = F_0\,;F_1$, $E_0 \preceq^u F_0$ and $E_1 \preceq F_1$

c) $E_0 \oplus E_1 \preceq^u F$ implies $F = F_0 \oplus F_1$, $E_0 \preceq F_0$ and $E_1 \preceq F_1$

d) $E_0 \parallel E_1 \preceq^u F$ implies $F = F_0 \parallel F_1$, $E_0 \preceq^u F_0$ and $E_1 \preceq^u F_1$

e) $rec\,x.\,E \preceq^u F$ implies $F = rec\,x.\,E$

Proof Using structural arguments each implication is proven by a simple induction on the number of rules used to prove the left hand side of the implication. $\square$

In the following we need to be able to see that an internal step solely originate in an unwinding of a recursive subexpression. We write this as $E \stackrel{u}{\rightarrowtail} F$.
Formally $\stackrel{u}{\rightarrowtail} \subseteq \rightarrowtail$ is defined to be the least relation over RCL_Ω^{rec} which can be deduced from $rec\,x.\,E \stackrel{u}{\rightarrowtail} E[rec\,x.\,E/x]$ and the $\stackrel{u}{\rightarrowtail}$ equivalent versions of the $\rightarrowtail$ inference rules.
The lemma now states:

Lemma 9.4.7 Given $E, E' \in RCL_\Omega^{rec}$ then

$$E \succeq E' \text{ implies } \exists F.\, E \stackrel{u}{\rightarrowtail}{}^* F \succeq^u E'$$

Before proving the lemma observe that there is an "unwind version" of proposition$_\Omega^{rec}$ 8.1.1.

Proof By induction on the number of rules used in the proof of $E' \preceq E$. There are three case in the basis:

$E = E'$: Let $F = E$ and $E \stackrel{u}{\rightarrowtail}{}^0 F \succeq^u F = E'$.

$E' = \Omega \preceq E$: Then also $\Omega \preceq^u E$ and we can choose $F = E$ as above.

$E' = G[rec\,x.\,G/x] \preceq rec\,x.\,G = E$: By the recursion rule for $\stackrel{u}{\rightarrowtail}$ it is seen that $E \stackrel{u}{\rightarrowtail} G[rec\,x.\,G/x] = E' \succeq^u E'$ so we can choose $F = E'$.

Now for the inductive step there are five ways $E' \preceq E$ could have been obtained.

$E' \preceq E'', E'' \preceq E$: By hypothesis of induction there are F' and F'' such that $E'' \stackrel{u}{\rightarrowtail}{}^* F' \succeq^u E'$ and $E \stackrel{u}{\rightarrowtail}{}^* F'' \succeq^u E''$. From lemma 9.4.8 below we know that $F'' \succeq^u E'' \stackrel{u}{\rightarrowtail}{}^* F'$ implies the existence of a F such that $F'' \stackrel{u}{\rightarrowtail}{}^* F \succeq^u F'$. Then we actually have $E \stackrel{u}{\rightarrowtail}{}^* F'' \stackrel{u}{\rightarrowtail}{}^* F \succeq^u F' \succeq^u E'$ as we want.

$E' = E'_0\,;E'_1, E = E_0\,;E_1$ and $E'_0 \preceq E_0, E'_1 \preceq E_1$: Using the inductive hypothesis on $E_0 \succeq E'_0$ we find a F_0 such that $E_0 \stackrel{u}{\rightarrowtail}{}^* F_0 \succeq^u E'_0$. The unwind version of proposition$_\Omega^{rec}$ 8.1.1 then gives $E = E_0\,;E_1 \stackrel{u}{\rightarrowtail}{}^* F_0\,;E_1$. Since $E'_1 \preceq E_1$ we by definition of $\preceq^u$ actually have $E' = E'_0\,;E'_1 \preceq^u F_0\,;E_1$ and we can let $F = F_0\,;E_1$.

$E' = E'_0 \oplus E'_1, E = E_0 \oplus E_1$ and $E'_0 \preceq E_0, E'_1 \preceq E_1$: Then also $E \preceq^u E'$ so we can choose $F = E$ because $E \stackrel{u}{\rightarrowtail}{}^0 F = E \succeq^u E'$.

$E' = E'_0 \parallel E'_1, E = E_0 \parallel E_1$ and $E'_0 \preceq E_0, E'_1 \preceq E_1$: By induction there for $i = 0, 1$ exists a F_i such that $E_i \stackrel{u}{\rightarrowtail}{}^* F_i \succeq^u E'_i$, so if we use the unwind version of proposition$_\Omega$ 8.1.1 we get $E = E_0 \parallel E_1 \stackrel{u}{\rightarrowtail}{}^* F_0 \parallel F_1$. Letting $F = F_0 \parallel F_1$ we have $F \succeq^u E'_0 \parallel E'_1 = E'$.

$E' = G'[\varrho], E = G[\varrho]$ and $G' \preceq G$ (and $G, G' \in RBL_\Omega^{rec}$): As above we find a H such that $G \stackrel{u}{\rightarrowtail}{}^* H \succeq^u G'$. By definition of $\preceq^u$ we then have $F := H[\varrho] \succeq^u G'[\varrho] = E'$ and of course $E \stackrel{u}{\rightarrowtail}{}^* F$.

□

Lemma 9.4.8 If $E, E' \in RCL_\Omega^{rec}$ then

$$E \succeq^u E' \overset{u}{\rightarrowtail}{}^* F' \text{ implies } \exists F.\, E \overset{u}{\rightarrowtail}{}^* F \succeq^u F'$$

Proof By induction on the number of unwinding steps using:

$$E \succeq^u E' \overset{u}{\rightarrowtail} F' \Rightarrow \exists F.\, E \overset{u}{\rightarrowtail} F \succeq^u F' \tag{9.9}$$

which in turn is proven by induction on the size, m, of $E' \overset{u}{\rightarrowtail}_m F'$.
Since $\overset{u}{\rightarrowtail}_0 = \emptyset$ the basic case is trivial and in the inductive step we can assume (9.9) holds for m when proving (9.9) for $m+1$. The different rules are handled one by one:

$E' = rec\,x.\,G \overset{u}{\rightarrowtail}_{m+1} G[rec\,x.\,G/x] = F'$: By lemma 9.4.6 $E \succeq^u rec\,x.\,G$ implies $E = rec\,x.\,G$. Let $F = F'$ and we use the same rule to get $E \overset{u}{\rightarrowtail} F \succeq^u F = F'$.

$E' = E'_0\,;E'_1 \overset{u}{\rightarrowtail}_{m+1} F'_0\,;E'_1 = F'$ where $E'_0 \overset{u}{\rightarrowtail}_m F'_0$: $E'_0;E'_1 \preceq^u E$ implies $E = E_0;E_1$ where $E'_0 \preceq^u E_0$ and $E'_1 \preceq E_1$. We can then use the hypothesis of induction to get an F_0 with $E_0 \overset{u}{\rightarrowtail} F_0 \succeq^u F'_0$. Then also $E = E_0\,;E_1 \overset{u}{\rightarrowtail} F_0\,;E_1 \succeq^u F'_0\,;E'_1 = F'$ and we can choose $F = F_0\,;E_1$.

$E' = E'_0 \parallel E'_1 \overset{u}{\rightarrowtail}_{m+1} F'$: There are two subcases which are handled similar/ symmetric as the rule for ;.

$E' = G'[\varrho] \overset{u}{\rightarrowtail}_{m+1} H'[\varrho] = F'$ where $G' \overset{u}{\rightarrowtail}_m H'$: By lemma 9.4.6 $E \succeq^u G'[\varrho]$ only if $E = G[\varrho]$ and $G' \preceq^u G$. Then by induction $G \overset{u}{\rightarrowtail} H \succeq^u H'$ for some H and we get $E \overset{u}{\rightarrowtail} H[\varrho] \succeq^u H'[\varrho] = F'$ as desired.

□

We are now ready to prove the equivalent lemma of 9.4.5 for single steps.

Lemma 9.4.9 Given $E, E' \in RCL_\Omega^{rec}$ and $A \in \mathsf{G}$. Then:

- $E \succeq E' \rightarrowtail F'$ implies $\exists F.\, E \rightarrowtail^* F \succeq F'$
- $E \succeq E' \xrightarrow{A} F'$ implies $\exists F.\, E \overset{A}{\Rightarrow} F \succeq F'$

Proof Immediate form the preceding lemma 9.4.7 and the two following lemmas. □

Lemma 9.4.10 If $E, E' \in RCL_\Omega^{rec}$ then

$$E \succeq^u E' \rightarrowtail F' \text{ implies } \exists F.\, E \rightarrowtail^* F \succeq F'$$

Proof By induction on the size, m, of the internal step $E' \rightarrowtail_m F'$.
The basic case is trivial and in the inductive case the lemma can be assumed to be true for all internal steps of size m. We now investigate all the rules.
Using the fact that $\preceq^u \subseteq \preceq$ and proposition$_\Omega^{rec}$ 8.1.1 the inference rules are handled exactly as in the proof of lemma 9.4.8. E.g., $E' = G'[\varrho] \rightarrowtail_{m+1} H'[\varrho] = F'$ where $G' \rightarrowtail_m H'$. By lemma 9.4.6 $E \succeq^u G'[\varrho]$ implies $E = G[\varrho]$ where $G' \preceq^u G$, so by hypothesis of induction then $G \rightarrowtail^* H$ for some $H \succeq H'$. By definition of $\preceq$ we have $F := H[\varrho] \succeq H'[\varrho] = F'$ and by proposition$_\Omega^{rec}$ 8.1.1 also $E = G[\varrho] \rightarrowtail^* H[\varrho] = F$. We will therefore just look at the ordinary rules for $\rightarrowtail$.

$E' = \Omega \rightarrowtail_{m+1} \Omega = F'$: Then $E' = F'$ and we can choose $E = F$. Then $E \rightarrowtail^0 F = E \succeq^u E' = F'$ and since $\preceq^u \subseteq \preceq$ we are done.

$E' = \dagger\, ; E'_1 \rightarrowtail_{m+1} E'_1 = F'$: By lemma 9.4.6 $\dagger\, ; E_1 \preceq^u E$ implies $E = \dagger\, ; E_1$ where $E'_1 \preceq E_1$. With $F = E_1$ we then get $E = \dagger\, ; E_1 \rightarrowtail F = E_1 \succeq E'_1 = F'$.

$E = E'_0 \oplus E'_1 \rightarrowtail_{m+1} F'$: Suppose w.l.o.g. $F' = E'_0$. $E'_0 \oplus E'_1 \preceq^u E$ only if $E = E_0 \oplus E_1$ where $E'_0 \preceq E_0$ and $E'_1 \preceq E_1$. But then also $E \rightarrowtail E_0 \succeq E'_0 = F'$.

$E' = E'_0 \parallel E'_1 \rightarrowtail_{m+1} F'$: Similar/ symmetric as the case with $E' = \dagger\, ; E'_1$ but with the addinnal use of $\preceq^u \subseteq \preceq$.

$E' = G'[\varrho] \rightarrowtail_{m+1} F'$: then $E' \preceq^u E$ means $E = G[\varrho]$ where $G' \preceq^u G$. There are five ordinary rules according to the structure of G':

$G' = \Omega$ and $F' = \Omega$: Let $F = E$. Since $E' \preceq^u E$ implies $E' \preceq E$ we then get $E \rightarrowtail^0 F = E \succeq E' = \Omega[\varrho] \succeq \Omega = F'$.

$G' = a$ and $F' = \varrho(a)$: $a \preceq^u G$ only if $G = a$, so we actually have $E = E'$ and one can choose $F = F'$.

$G' = G'_0\, ; G'_1$ and $F' = G'_0[\varrho]\, ; G'_1[\varrho]$: $G'_0 ; G'_1 \preceq^u G$ implies $G = G_0 ; G_1$ where $G'_0 \preceq^u G_1$ and $G'_1 \preceq G_1$. Again since $\preceq^u \subseteq \preceq$ we by letting $F = G_0[\varrho]\, ; G_1[\varrho]$ get $F' \preceq F$ and also $E = (G_0\, ; G_1)[\varrho] \rightarrowtail G_0[\varrho]\, ; G_1[\varrho] = F$.

$G' = G'_0 \oplus G'_1$ and $G' = G'_0 \parallel G'_1$: Similar as last case.

□

Lemma 9.4.11 Suppose $E, E' \in RCL_\Omega^{rec}$ and $A \in \mathsf{G}$. Then $E \succeq^u E' \xrightarrow{A} F'$ implies $\exists F.\, E \xrightarrow{A} F \succeq F'$

Proof By induction on the size of the step $E' \xrightarrow{A} F'$. The proof follows exactly the line of the previous lemma, except that we do not have to use proposition$_\Omega^{rec}$ 8.1.1. □

Up til now we have showed that if E is the approximation of F then F can do all the sequences E can. Now we take the opposite angel and show that if a (possible recursive) process is able to perform a sequence, then there is a syntactic finite approximation which also can do this sequence.

Lemma 9.4.12 Suppose $E \in RCL_\Omega^{rec}$. Then

$$E \stackrel{s}{\Rightarrow} F \succeq F'' \in RCL_\Omega \text{ implies } \exists E', F' \in RCL_\Omega.\, E \succeq E' \stackrel{s}{\Rightarrow} F' \succeq F''$$

Proof By induction on the size of $\stackrel{s}{\Rightarrow}$. In the basic case we have $E = F$ and can choose $E' = F' = F''$. In the inductive step there as usual are two maincases:

$E \rightarrowtail G \stackrel{s}{\Rightarrow}' F \succeq F''$: (where $\stackrel{s}{\Rightarrow} = \rightarrowtail\stackrel{s}{\Rightarrow}'$ and the length of $\stackrel{s}{\Rightarrow}'$ is less than that of $\stackrel{s}{\Rightarrow}$) By hypothesis of induction there are $G', H \in RCL_\Omega$ such that $G \succeq G' \stackrel{s}{\Rightarrow} H \succeq F''$. Now $E \rightarrowtail G \succeq G'$ implies by lemma 9.4.14 below the existence of $E', G'' \in RCL_\Omega$ with $E \succeq E' \rightarrowtail^* G'' \succeq G'$. We can then use lemma 9.4.5 on $G'' \succeq G' \stackrel{s}{\Rightarrow} H$ to find an F' which fulfills $G'' \stackrel{s}{\Rightarrow} F' \succeq H$. Collecting the facts so far we have $E \succeq E' \rightarrowtail^* G'' \stackrel{s}{\Rightarrow} F' \succeq H \succeq F''$ and so $E \succeq E' \stackrel{s}{\Rightarrow} F' \succeq F''$. For $E' \in RCL_\Omega$ we easely prove $E' \stackrel{s}{\Rightarrow} F'$ implies $F' \in RCL_\Omega$ so this case is settled.

$E \stackrel{A}{\longrightarrow} G \stackrel{s'}{\Rightarrow} F \succeq F''$: Similar but using lemma 9.4.15 in place of lemma 9.4.14.

□

Before proving the lemmas for single steps it will be useful to prove:

Lemma 9.4.13 For $E \in RCL_\Omega^{rec}(X)$ we have:

a) $\dagger \preceq E$ *iff* $E = \dagger$

b) If $E \neq \Omega$ then for $\diamond \in \{;, \oplus, \|\}$, $E \preceq F_0 \diamond F_1$ implies $E = E_0 \diamond E_1$ where $E_0 \preceq F_0$ and $E_1 \preceq F_1$

c) If $E \neq \Omega$ then $E \preceq F'[\varrho]$ implies $E = E'[\varrho]$ where $E' \preceq F'$

Proof Each implication is proven by a simple induction on the number of rules (from the definition of $\preceq$) used to prove the left hand side of the implication. □

We should mention that e.g., $E \preceq F'[\varrho]$ implies $F'[\varrho] \in RBL_\Omega^{rec}(X)$, because $\preceq$ only is defined on $RCL_\Omega^{rec}(X)$ and expressions only can be of this form when $F' \in RBL_\Omega^{rec}(X)$. Also notice that the opposite implications of b) – c) does not hold in general. E.g., from $E_0 \| E_1 \preceq F$ one cannot deduce that F is of the form $F = F_0 \| F_1$ where $E_0 \preceq F_0$ and $E_1 \preceq F_1$ because F might equal $rec\,x.\,G$ where $G[rec\,x.\,G/x] = E_0 \| E_1$. The only reason we can deduce something about E in a) when $\dagger \preceq E$ is because all recursive expressions are from $RBL_\Omega^{rec}(X)$ and because $\dagger \notin RBL_\Omega^{rec}(X)$.

Lemma 9.4.14 If $E \in RCL_\Omega^{rec}$ then

$$E \rightarrowtail F \succeq F'' \in RCL_\Omega \text{ implies } \exists E', F' \in RCL_\Omega.\ E \succeq E' \rightarrowtail^* F' \succeq F''$$

Proof If $F'' = \Omega$ the lemma follows by choosing $E' = F' = \Omega \in RCL_\Omega$. Hence we will assume $F'' \neq \Omega$ when proving the lemma by induction on the size, m, of $E \rightarrowtail_m F$. We assume the lemma holds for m when proving it for $m+1$ by considering the different rules.

$E = \Omega \rightarrowtail_{m+1} \Omega = F \succeq F''$: This can only mean $F'' = \Omega$ so we can choose $E' = F' = \Omega$.

$E = E_0 \,;\, E_1 \rightarrowtail_{m+1} F \succeq F''$: There are two subcases:

$E_0 = \dagger$ and $F = E_1$: Let $E' = \dagger \,;\, F'' \in RCL_\Omega$ and $F' = F''$.

$F = F_0 \,;\, E_1$ where $E_0 \rightarrowtail_m F_0$: We assume $F'' \neq \Omega$ so by lemma 9.4.13 we know that $F'' \preceq F_0 \,;\, E_1$ implies $F'' = F_0'' \,;\, E_1''$ for some $F_0'' \preceq F_0$ and $E_1'' \preceq E_1$. By hypothesis of induction there are $E_0', F_0' \in RCL_\Omega$ with $E_0 \succeq E_0 \rightarrowtail^* F_0' \succeq F_0''$. Because $F'' \in RCL_\Omega$ implies $E_1'' \in RBL_\Omega$ we then have $E' := E_0' \,;\, E_1'' \in RCL_\Omega$ and $F' := F_0' \,;\, E_1'' \in RCL_\Omega$. Also $E' \rightarrowtail^* F'$ and $E' = E_0' \,;\, E_1'' \preceq E_0 \,;\, E_1'' \preceq E_0 \,;\, E_1$ so as $F'' = F_0'' \,;\, E_1'' \preceq F_0' \,;\, E_1'' = F'$.

$E = E_0 \oplus E_1 \rightarrowtail_{m+1} F \succeq F''$: W.l.o.g. we just consider the case where $F = E_0$. Then let $E' = F'' \oplus \Omega \in RCL_\Omega$ and $F = F''$. Clearly $E' = F'' \oplus \Omega \preceq F \oplus E_1 = E$ and $E' \rightarrowtail F'' \succeq F''$. Let $F' = F''$.

$E = E_0 \parallel E_1 \succ\!\!\longrightarrow_{m+1} F \succeq F''$: The four subcases are handled similar/ symmetric to the rules for ; above.

$E = G[\varrho] \succ\!\!\longrightarrow_{m+1} F \succeq F''$: There are six subcases to be dealt with.

$G = \Omega$ and $F = \Omega$: Then $E, F \in BL_\Omega$ and we can choose $E' = E$ and $F' = F$.

$G = a$ and $F = \varrho(a)$: Again $E, F \in BL_\Omega$.

$G = G_0 \,;G_1$ and $F = G_0[\varrho]\,;G_1[\varrho]$: Since F'' is assumed to be different from Ω we by lemma 9.4.13 from $F \succeq F'' \in RBL_\Omega$ deduce $F'' = F_0''\,;F_1''$ where for $i = 0, 1$ either $F_i'' = \Omega$ or ($F_i'' = G_i''[\varrho]$ and $G_i \succeq G_i'' \in RBL_\Omega$). There are actually four subcases to consider, but we just treat $F_0'' = G_0''[\varrho]$ and $F_1'' = \Omega$ because the other follow in the same way. Choose $E' = (G_0''\,;\Omega)[\varrho] \in RBL_\Omega$ and $F' = G_0''[\varrho]\,;\Omega[\varrho] \in RBL_\Omega$. Then clearly $E' \succ\!\!\longrightarrow F'$ and $E' \preceq (G_0\,;\Omega)[\varrho] \preceq (G_0\,;G_1)[\varrho] = E$ and also $F'' = G_0'[\varrho]\,;\Omega \preceq G_0''[\varrho]\,;\Omega[\varrho] = F'$.

$G = G_0 \oplus G_1$ and $G = G_0 \parallel G_1$: Analogous to the last case.

$F = H[\varrho]$ where $G \succ\!\!\longrightarrow_m H$: By lemma 9.4.13 $\Omega \neq F'' \preceq H[\varrho]$ only if $F'' = H''[\varrho]$ for some $H'' \preceq H$. By hypothesis of induction there are $G', H' \in RBL_\Omega$ such that $G \succeq G' \succ\!\!\longrightarrow^* H' \succeq H''$. Now $G' \succeq G \in RBL_\Omega^{rec}$ and $G' \in RCL_\Omega$ implies $G' \in RBL_\Omega$ and similar for H' so we obtain $E' := G'[\varrho] \in RBL_\Omega$ and $F' := H'[\varrho] \in RBL_\Omega$ so as $E' \succ\!\!\longrightarrow^* F'$ from proposition$_\Omega$ 8.1.1. Clearly $E' \preceq E$ and $F'' = H''[\varrho] \preceq H'[\varrho] = F'$.

$E = rec\,x.\,G \succ\!\!\longrightarrow_{m+1} G[rec\,x.\,G/x] = F \succeq F''$: Choose $E' = F' = F'' \in RCL_\Omega$. Then of course $E' \succ\!\!\longrightarrow^0 F' \succeq F' = F''$ and because $E' \preceq F = G[rec\,x.\,G/x] \preceq rec\,x.\,G = E$ we also have $E' \preceq E$.

□

Lemma 9.4.15 If $E \in RCL_\Omega^{rec}$ and $A \in \mathsf{G}$ then

$$E \xrightarrow{A} F \succeq F'' \in RCL_\Omega \text{ implies } \exists E', F' \in RCL_\Omega.\ E \succeq E' \xrightarrow{A} F' \succeq F''$$

Proof At first the lemma is proven for the case $F'' \neq \Omega$. This will be done by induction on the size, m, of $E \xrightarrow{A}_m F$. As usual only the inductive step needs attention. We consider each rule in turn under the assumption $F'' \neq \Omega$ and that the lemma holds for m.

$E = a \xrightarrow{A}_{m+1} \dagger = F \succeq F''$: Clearly $A = a$ and $F'' = \dagger$. Choose $E' = a \in RCL_\Omega$ and $F' = \dagger \in RCL_\Omega$.

$E = E_0\,;E_1 \xrightarrow{A}_{m+1} F_0\,;E_1 = F$ where $E_0 \xrightarrow{A}_m F_0$: $\Omega \neq F'' \preceq F_0\,;E_1$ implies by lemma 9.4.13 $F'' = F_0''\,;E_1''$ where $F_0 \succeq F_0'' \in RCL_\Omega$ and $E_1 \succeq E_1'' \in RBL_\Omega$. By induction then $\exists E_0' \in RCL_\Omega.E_0 \succeq E_0' \xrightarrow{A} F_0' \succeq F_0''$. Letting $E' = E_0';E_1''$ we have $E' \in RCL_\Omega$ and $E' \preceq E_0'\,;E_1 \preceq E$ and using the same inference rule finally $E' = E_0'\,;E_1'' \xrightarrow{A} F_0'\,;E_1'' =: F'$ and also $F'' = F_0''\,;E_1'' \preceq F'$.

$E = E_0 \parallel E_1 \xrightarrow{A}_{m+1} F_0 \parallel F_1 = F \succeq F''$: The two rules where only one of the components of E is involved are handled similar/ symmetric as above. If $E \xrightarrow{A}_{m+1} F$ steams from the remaining inference rule we must have $A = A_0 \times A_1$ and $F'' \succeq F_0'' \parallel F_1''$

where for $i = 0, 1$, $E_i \xrightarrow{A_i}_m F_i \succeq F_i''$ and $A_i \in \mathsf{G}$. Like above we can apply the hypothesis of induction on each component and since $A_0 \times A_1 = A \in \mathsf{G}$ we can use the same rule again to obtain the result in a similar fashion.

Now from the rules of $\xrightarrow{A}$ obviously $E \xrightarrow{A} F$ only if $\dagger$ occurs in F. By structural induction on F an $F''' \in RCL_\Omega$ can then be found such that $F \succeq F''' \neq \Omega$. As above appropriate $E', F' \in RCL_\Omega$ are found for F'''. When $F'' = \Omega$ we have $F''' \succeq F''$ so this case is dealt with too. $\square$

The two key lemmas 9.4.12 and 9.4.5 enables us to establish the important properties which shall bring the different preorders in connection with our denotational models.

Proposition 9.4.16 $\lesssim_{\mathsf{G}}$ and $\sqsubseteq_{\mathsf{G}}$ extends $\preceq$ on RBL_Ω^{rec}.

Proof We shall show that when $\preceq$ is resticted to RBL_Ω^{rec} then $\preceq \subseteq \lesssim_{\mathsf{G}}$ and $\preceq \subseteq \sqsubseteq_{\mathsf{G}}$. So let $E, F \in RBL_\Omega^{rec}$ be given such that $E \preceq F$.

$\lesssim_{\mathsf{G}}$: Assume $E \overset{s}{\Rightarrow} \dagger$. By lemma 9.4.5 there is an F' such that $F \overset{s}{\Rightarrow} F' \succeq \dagger$. From lemma 9.4.13 $\dagger \preceq F'$ only if $F' = \dagger$.

$\sqsubseteq_{\mathsf{G}}$: Immediate from lemma 9.4.5. $\square$

Proposition 9.4.17 Given $E \in RBL_\Omega^{rec}$ then

a) $E \overset{s}{\Rightarrow} \dagger$ *iff* $\exists E' \in \mathrm{Fin}(E).\ E' \overset{s}{\Rightarrow} \dagger$

b) $E \overset{s}{\Rightarrow}$ *iff* $\exists E' \in \mathrm{Fin}(E).\ E' \overset{s}{\Rightarrow}$

Proof By definition $E' \in \mathrm{Fin}(E)$ means $E' \preceq E$ and $E' \in RBL_\Omega$, so the *if* part of a) and b) are just special cases of the previous proposition. *only if*:

a) Suppose $E \overset{s}{\Rightarrow} \dagger$. Because $\dagger \succeq \dagger$ we can use lemma 9.4.12 to find $E', F' \in RCL_\Omega$ such that $E \succeq E' \overset{s}{\Rightarrow} F' \succeq \dagger$. $\dagger \preceq F'$ only if $F' = \dagger$ so this means $E \succeq E' \overset{s}{\Rightarrow} \dagger$. Now $E' \preceq E \in RBL_\Omega^{rec}$ clearly implies $E' \in RBL_\Omega^{rec}$ wherefore we from $E' \in RCL_\Omega$ deduce $E' \in RBL_\Omega$ and thus $E' \in \mathrm{Fin}(E)$.

b) Suppose $E \overset{s}{\Rightarrow}$. This means $E \overset{s}{\Rightarrow} F$ for some $F \in RCL_\Omega^{rec}$. Using $F \succeq \Omega$ the rest goes as under a). $\square$

With the last to propositions it is easy to prove the main result of this section:

Theorem 9.4.18 The preorders $\lesssim_{\mathsf{G}}$ and $\sqsubseteq_{\mathsf{G}}$ over RBL_Ω^{rec} are algebraic.

Proof The preorder $\lesssim_{\mathsf{G}}$ is proved algebraic in exactly the same way as we now will prove $\sqsubseteq_{\mathsf{G}}$ algebraic. For $\sqsubseteq_{\mathsf{G}}$ we shall prove

$$E \sqsubseteq_{\mathsf{G}} F \textit{ iff } \forall E' \in \mathrm{Fin}(E) \exists F' \in \mathrm{Fin}(F).\ E' \sqsubseteq_{\mathsf{G}} F'$$

if: Assume the right hand side holds and let an $s \in \mathsf{G}^*$ be given such that $E \overset{s}{\Rightarrow}$. We prove $F \overset{s}{\Rightarrow}$. By proposition 9.4.17 above there is an $E' \in \mathrm{Fin}(E)$ with $E' \overset{s}{\Rightarrow}$. By assumption there is also an $F' \in \mathrm{Fin}(F)$ such that $E' \sqsubsim_{\mathsf{G}} F'$. Hence $F' \overset{s}{\Rightarrow}$ and using the same proposition again then $F \overset{s}{\Rightarrow}$.

only if: Assume $E \sqsubsim_{\mathsf{G}} F$ and let a $E' \in \mathrm{Fin}(E)$ be given.
At first we show that for each $s \in \mathsf{G}^*$ such that $E' \overset{s}{\Rightarrow}$ we can pick an $F_s \in \mathrm{Fin}(F)$ with $F_s \overset{s}{\Rightarrow}$. Suppose $E' \overset{s}{\Rightarrow}$. Applying the previous proposition we see that $E \overset{s}{\Rightarrow}$ and from the assumption then also $F \overset{s}{\Rightarrow}$. Using the proposition once more brings us an $F' \in \mathrm{Fin}(F)$ such that $F' \overset{s}{\Rightarrow}$. Let F_s be one of these F''s.
Now for any $H \in BL_\Omega$ it is an easy matter to prove by induction on the structure of H that $\{s \in \mathsf{G}^* \mid H \overset{s}{\Rightarrow}\}$ is finite. By proposition 9.3.2 we have $\{s \in \mathsf{G}^* \mid E' \overset{s}{\Rightarrow}\} = \{s \in \mathsf{G}^* \mid E'\sigma \overset{s}{\Rightarrow}\}$, so because $E'\sigma \in BL_\Omega$ we conclude $\{F_s \in \mathrm{Fin}(F) \mid E' \overset{s}{\Rightarrow}\}$ is finite too.
$\mathrm{Fin}(F)$ is directed so there is an ub $F' \in \mathrm{Fin}(F)$ for $\{F_s \mid E' \overset{s}{\Rightarrow}\}$. By proposition 9.4.16 $\preceq \subseteq \sqsubsim_{\mathsf{G}}$ this therefore means that for every F_s, F' can perform s. But there is exactly one F_s for each $E' \overset{s}{\Rightarrow}$ wherefore we conclude $E' \sqsubsim_{\mathsf{G}} F'$ as desired. □

9.4.2 The Syntactic Finite Sublanguages

In this subsection look at how the full abstractness results for BL and RBL can be carried over to BL_Ω and RBL_Ω.

Theorem 9.4.19 The following denotatons are fully abstract:

a) $A_{\mathsf{G}}[\![\_]\!]$ on BL_Ω w.r.t. $\lesssim_{\mathsf{G}}$

b) $A_{or}[\![\_]\!]$ on RBL_Ω w.r.t. $\lesssim^c_{\mathsf{W}}$

c) $A^p_{\mathsf{G}}[\![\_]\!]$ on BL_Ω w.r.t. $\sqsubsim^c_{\mathsf{G}}$

d) $A^p_{or}[\![\_]\!]$ on RBL_Ω w.r.t. $\sqsubsim^c_{\mathsf{W}}$

Proof

a) Since $A_{\mathsf{G}}[\![\_]\!]$ is defined compositionally and the operators are monotone, $\trianglelefteq_{\mathsf{G}}$ is a precongruence w.r.t. BL_Ω. a) then follows from proposition 9.4.23 below.

b) By definition $\lesssim^c_{\mathsf{W}} \subseteq RBL_\Omega \times RBL_\Omega$ is a precongruence w.r.t. the combinators of RBL_Ω. We then just have to show $\trianglelefteq_{or} = \lesssim^c_{\mathsf{W}}$. By proposition 9.1.11 this follows if we can prove for all $E_0, E_1 \in RBL_\Omega$

$$E_0 \trianglelefteq_{or} E_1 \text{ iff } \forall RBL_\Omega\text{-contexts } \mathcal{C}.\, \mathcal{C}[E_0] \lesssim_{\mathsf{W}} \mathcal{C}[E_1]$$

only if: Assume $E_0 \trianglelefteq_{or} E_1$ and let a RBL_Ω context, $\mathcal{C}$, be given. By the compositional nature of $A_{or}[\![\_]\!]$ and the monotonicity of the A_{or} operators it follows that $\trianglelefteq_{or}$ is a precongruence w.r.t. the combinators of RBL_Ω. Hence also $\mathcal{C}[E_0] \trianglelefteq_{or} \mathcal{C}[E_1]$ or equally $A_{or}[\![\mathcal{C}[E_0]]\!] \subseteq A_{or}[\![\mathcal{C}[E_1]]\!]$. From the $\subseteq$-monotonicity of δ_{W} then $\delta_{\mathsf{W}}(A_{or}[\![\mathcal{C}[E_0]]\!]) \subseteq \delta_{\mathsf{W}}(A_{or}[\![\mathcal{C}[E_1]]\!])$ which by proposition 9.4.23 implies $\mathcal{C}[E_0] \lesssim_{\mathsf{W}} \mathcal{C}[E_1]$.

if: Assume $E_0 \not\trianglelefteq_{or} E_1$ or equally $A_{or}[\![E_0]\!] \not\subseteq A_{or}[\![E_1]\!]$. From lemma 9.4.27 we see there is a RBL_Ω-context, $\mathcal{C}$, such that $\delta_{\mathsf{W}}(A_{or}[\![\mathcal{C}[E_0]]\!]) \not\subseteq \delta_{\mathsf{W}}(A_{or}[\![\mathcal{C}[E_1]]\!])$. Then also $\mathcal{C}[E_0] \not\lesssim_{\mathsf{W}} \mathcal{C}[E_1]$ by proposition 9.4.25.

c) Similar as b) with $\trianglelefteq^p_{\mathsf{G}}$ instead of $\trianglelefteq_{or}$ and only BL_Ω contexts are concerned. Proposition 9.4.23 is used in stead of proposition 9.4.25 (δ_{W} does not appear). With $A = L(E_1)$ the BL_Ω-context for the *if* part is found from lemma 9.4.26.

d) Similar to b) using lemma 9.4.28 to find the RBL_Ω-context in the *if* part.

□

From proposition 9.4.23, proposition$_\Omega$ 8.1.2 and proposition 9.3.2 so as proposition 9.2.17 and the theorem above we deduce the RBL_Ω equivalent of corollary 8.4.1 of section 8.4:

Corollary 9.4.20 For all $E_0, E_1 \in RBL_\Omega$:

$$\begin{array}{ll} E_0 \lesssim_{\mathsf{W}} E_1 \text{ iff } A_{\mathsf{W}}[\![E_0\sigma]\!] = A_{\mathsf{W}}[\![E_1\sigma]\!] & E_0 \sqsubseteq\!\!\!\sim_{\mathsf{W}} E_1 \text{ iff } A^p_{\mathsf{W}}[\![E_0\sigma]\!] = A^p_{\mathsf{W}}[\![E_1\sigma]\!] \\ E_0 \lesssim_{\mathsf{G}} E_1 \text{ iff } A_{\mathsf{G}}[\![E_0\sigma]\!] = A_{\mathsf{G}}[\![E_1\sigma]\!] & E_0 \sqsubseteq\!\!\!\sim_{\mathsf{G}} E_1 \text{ iff } A^p_{\mathsf{G}}[\![E_0\sigma]\!] = A^p_{\mathsf{G}}[\![E_1\sigma]\!] \\ E_0 \gtrless^c_{\mathsf{W}} E_1 \text{ iff } A_{or}[\![E_0\sigma]\!] = A_{or}[\![E_1\sigma]\!] & E_0 \sqsupseteq\!\!\!\sim^c_{\mathsf{G}} E_1 \text{ iff } A^p_{or}[\![E_0\sigma]\!] = A^p_{or}[\![E_1\sigma]\!] \end{array}$$

With the same argumentation as in section 8.4 then also:

Corollary 9.4.21 For $E_0, E_1 \in RBL_\Omega$ we have

$$(E_0 \lesssim^c_{\mathsf{W}} E_1) \Rightarrow (E_0 \lesssim_{\mathsf{G}} E_1) \Rightarrow (E_0 \lesssim_{\mathsf{W}} E_1)$$

and

$$(E_0 \sqsubseteq\!\!\!\sim^c_{\mathsf{W}} E_1) \Rightarrow (E_0 \sqsubseteq\!\!\!\sim_{\mathsf{G}} E_1) \Rightarrow (E_0 \sqsubseteq\!\!\!\sim_{\mathsf{W}} E_1)$$

Theorem 9.4.22

a) BL_Ω is $\{E \in BL_\Omega \mid L(E) \subseteq A\}$-expressive w.r.t. $\lesssim_{\mathsf{G}}$ for every finite subset A of Δ.

b) RBL_Ω is expressive w.r.t. both $\lesssim_{\mathsf{W}}$ and $\sqsubseteq\!\!\!\sim_{\mathsf{W}}$.

Proof

a) Suppose $A \subseteq \Delta$ is finite and $E_0 \in BL_\Omega$. Let $\mathcal{C}$ be the BL_Ω-context, $\#\,;e$, from lemma 9.4.26. Given an E_1 with $L(E_1) \subseteq A$ we show

$$E_0 \sqsubseteq\!\!\!\sim^c_{\mathsf{G}} E_1 \text{ iff } \mathcal{C}[E_0] \sqsubseteq\!\!\!\sim_{\mathsf{G}} \mathcal{C}[E_1]$$

only if: Since $\sqsubseteq\!\!\!\sim^c_{\mathsf{G}}$ by definition is a precongruence it follows that $\mathcal{C}[E_0] \sqsubseteq\!\!\!\sim^c_{\mathsf{G}} \mathcal{C}[E_1]$. Again by definition of $\sqsubseteq\!\!\!\sim^c_{\mathsf{G}}$ also $\sqsubseteq\!\!\!\sim^c_{\mathsf{G}} \subseteq \sqsubseteq\!\!\!\sim_{\mathsf{G}}$.

if: $\mathcal{C}[E_0] \sqsubseteq\!\!\!\sim_{\mathsf{G}} \mathcal{C}[E_1]$
$\Rightarrow A^p_{\mathsf{G}}[\![\mathcal{C}[E_0]]\!]_1 \subseteq A^p_{\mathsf{G}}[\![\mathcal{C}[E_1]]\!]_1$ proposition 9.4.23
$\Rightarrow A^p_{\mathsf{G}}[\![E_0]\!] \subseteq A^p_{\mathsf{G}}[\![E_1]\!]$ by choice of $\mathcal{C}$
$\Rightarrow E_0 \trianglelefteq^p_{\mathsf{G}} E_1$ definition of $\trianglelefteq^p_{\mathsf{G}}$
$\Rightarrow E_0 \sqsubseteq\!\!\!\sim^c_{\mathsf{G}} E_1$ by the theorem above

b) Similar as a) but using the RBL_Ω-context $\mathcal{C} = \#[\varrho]$ found by lemma 9.4.27 when dealing with $\lesssim_{\mathsf{W}}$ and $\mathcal{C} = \#[\varrho]\,;e$ from lemma 9.4.28 when concerned with $\sqsubseteq_{\mathsf{W}}$. In both cases proposition 9.4.25 is used in place of proposition 9.4.23.

□

Proposition 9.4.23 For every $E_0, E_1 \in BL_\Omega$:

a) $A_{\mathsf{G}}[\![E_0]\!] \subseteq A_{\mathsf{G}}[\![E_1]\!]$ *iff* $E_0 \lesssim_{\mathsf{G}} E_1$

b) $A^p_{\mathsf{G}}[\![E_0]\!]_1 \subseteq A^p_{\mathsf{G}}[\![E_1]\!]_1$ *iff* $E_0 \sqsubseteq_{\mathsf{G}} E_1$

Proof

a) This is nothing more than the extension of proposition 7.4.3 to BL_Ω. So of course a) holds if we can manage to extend lemma 7.4.4 to BL_Ω obtaining lemma$_\Omega$ 7.4.4. We will just comment on the main spots were the proof change:

only if: One additional case: $E = \Omega$: The implication holds trivially because $\Omega \overset{A_1}{\Rightarrow} \ldots \overset{A_n}{\Rightarrow} F$ implies $F = \Omega \neq \dagger$.

if: If $E = \Omega$ then $\wp(E) = \emptyset$ and we cannot have have $\mathbf{p} \in \wp(E)$. As above we can also here take over the corresponding proof for the other cases if we use lemma$_\Omega$ 7.4.5 and lemma$_\Omega$ 7.4.6.

b) Along the lines of the proof of proposition 7.4.3 in chapter 7 one from lemma 9.4.24 below for any $E \in BL_\Omega$ see:

$$\delta_{\mathsf{G}}(\wp^p_1(E)) = \{s \in \mathsf{G}^* \mid E \overset{s}{\Rightarrow}\} \tag{9.10}$$

From proposition 9.2.14 $A^p_{\mathsf{G}}[\![E]\!]_1 = \delta_{\mathsf{G}}(\wp^p_1(E))$ so the proposition then follows by the definition of $\sqsubseteq_{\mathsf{G}}$.

□

As the extended canonical map $\wp$ (by definition) agrees with $\wp^p_2$ we can use lemma$_\Omega$ 7.4.4 directly in the the proof of the next lemma. The same notation will also be used.

Lemma 9.4.24 Given $E \in BL_\Omega$ and $A_1, \ldots, A_n \in \mathsf{G}_\varepsilon$ $(n \geq 1)$. Then

$$E \overset{A_1}{\Longrightarrow} \ldots \overset{A_n}{\Longrightarrow} \;\textit{iff}\; \exists \mathbf{p} \in \wp^p_1(E).\, A_1 \cdot \ldots \cdot A_n \preceq \mathbf{p}$$

Proof Here we also start out by observing that any subexpression of a BL_Ω expression itself is from BL_Ω.

if: By induction on the structure of E.

$E = \Omega$: $\wp^p_1(\Omega) = \{\varepsilon\}$ and we must have $\mathbf{p} = \varepsilon$ and all $A_1, \ldots, A_n$ equal to $\emptyset$. Since for for every E, $E \rightarrowtail^0 E$ clearly $E \overset{\emptyset}{\Rightarrow} \ldots \overset{\emptyset}{\Rightarrow}$.

$E = a$: $\wp_1^p(a) = \{a, \varepsilon\}$. There are two possibilities for $\mathbf{p}$—either $\mathbf{p} = \varepsilon$ or $\mathbf{p} = a$. The former case goes as above and the latter as in the corresponding case of lemma$_\Omega$ 7.4.4.

$E = E_0 \,;\, E_1$: $\wp_1^p(E) = \wp_1^p(E_0) \cup \wp_2^p(E_0) \cdot \wp_1^p(E_1)$. If $\mathbf{p} \in \wp_1^p(E_0)$ the result follows from hypothesis of induction and proposition$_\Omega$ 7.2.3. Otherwise $\mathbf{p}$ must equal $\mathbf{p}_0 \cdot \mathbf{p}_1$ where $\mathbf{p}_0 \in \wp_2^p(E_0)$ and $\mathbf{p}_1 \in \wp_1^p(E_1)$. If $\mathbf{p}_1 = \varepsilon$ then $\mathbf{p} = \mathbf{p}_0 \cdot \varepsilon = \mathbf{p}_0 \in \wp_2^p(E_0)$ and the rest follows from lemma$_\Omega$ 7.4.4. We can then assume to be in the situation where $A_1 \cdot \ldots \cdot A_n \preceq \mathbf{p}_0 \cdot \mathbf{p}_1$ and $\mathbf{p}_1 \neq \varepsilon$. $\mathbf{p}_0 \neq \varepsilon$ because $\mathbf{p}_0 \in \wp_2^p(E_0)$. By lemma$_\Omega$ 7.4.7 then $n \geq 2$ and there exists a $1 \leq j < n$ such that $A_1 \cdot \ldots \cdot A_j \preceq \mathbf{p}_0$ and $A_{j+1} \cdot \ldots \cdot A_n \preceq \mathbf{p}_1$. Since $\mathbf{p}_0 \in \wp_2^p(E_0)$ we can use lemma$_\Omega$ 7.4.4 to get $E_0 \stackrel{A_1}{\Longrightarrow} \ldots \stackrel{A_j}{\Longrightarrow} \dagger$. From $A_{j+1} \cdot \ldots \cdot A_n \preceq \mathbf{p}_1 \in \wp_1^p(E_1)$ we by hypothesis of induction also have $E_1 \stackrel{A_{j+1}}{\Longrightarrow} \ldots \stackrel{A_n}{\Longrightarrow}$. Applying proposition$_\Omega$ 7.2.3 we finally get $E_0 \,;\, E_1 \stackrel{A_1}{\Longrightarrow} \ldots \stackrel{A_j}{\Longrightarrow} \dagger \,;\, E_1 \succ\!\!\longrightarrow E_1 \stackrel{A_{j+1}}{\Longrightarrow} \ldots \stackrel{A_n}{\Longrightarrow}$.

$E = E_0 \oplus E_1$ and $E = E_0 \parallel E_1$: On expressions of this form $\wp_1^p$ is defined like $\wp_2^p$ so the arguments are identical to those of lemma$_\Omega$ 7.4.4.

only if: Also by induction on the structure of E.

$E = \Omega$: Ω can only perform internal steps wherefore all A_1 through A_n must equal $\emptyset$ or by the alternative notation equal ε. But $\varepsilon \cdot \ldots \cdot \varepsilon \preceq \varepsilon \in \{\varepsilon\} = \wp_1^p(\Omega)$.

$E = a$: a cannot do any internal steps and if $a \stackrel{A}{\longrightarrow} F$ then $A = a$ and $F = \dagger$. $\dagger$ can do no steps at all so we conclude all $A_1, \ldots, A_n$ must equal $\varepsilon(= \emptyset)$ except for at most one which then only can be a. If all equals ε then $A_1 \cdot \ldots \cdot A_n = \varepsilon \preceq \varepsilon \in \{\varepsilon, a\} = \wp_1^p(a)$. Ohterwise $A_1 \cdot \ldots \cdot A_n = a \in \wp_1^p(a)$.

$E = E_0 \,;\, E_1$: Assume $E_0;E_1 \stackrel{A_1}{\Longrightarrow} \ldots \stackrel{A_n}{\Longrightarrow} F$. We want to use lemma$_\Omega$ 7.4.5 which mentions four ways such a sequence could be obtained. The first presuppose $E_0 \succ\!\!\longrightarrow^* \dagger$ so it can be excluded because $E_0 \in BL_\Omega$. The remaining three can for our purpose be summarized in two:

$$E_0 \stackrel{A_1}{\Longrightarrow} \ldots \stackrel{A_j}{\Longrightarrow} \dagger,\ E_1 \stackrel{A_{j+1}}{\Longrightarrow} \ldots \stackrel{A_n}{\Longrightarrow} F \text{ for a } 1 \leq j < n$$
$$E_0 \stackrel{A_1}{\Longrightarrow} \ldots \stackrel{A_n}{\Longrightarrow} F'$$

In the latter case we can apply the hypothesis of induction to find a $\mathbf{p} \in \wp_1^p(E_0)$ such that $A_1 \cdot \ldots \cdot A_n \preceq \mathbf{p}$. As $\wp_1^p(E_0) \subseteq \wp_1^p(E_0 \,;\, E_1)$ this case is settled. In the former case we can use lemma$_\Omega$ 7.4.4 to find a $\mathbf{p}_0 \in \wp_2^p(E_0)$ with $A_1 \cdot \ldots \cdot A_j \preceq \mathbf{p}_0$ and by induction there is a $\mathbf{p}_1 \in \wp_1^p(E_1)$ such that $A_{j+1} \cdot \ldots \cdot A_n \preceq \mathbf{p}_1$. From the $\preceq$-monotonicity of $\cdot$ we then deduce $A_1 \cdot \ldots \cdot A_n \preceq \mathbf{p}_0 \cdot \mathbf{p}_1 \in \wp_2^p(E_0) \cdot \wp_1^p(E_0) \subseteq \wp_1^p(E_0;E_1)$ as we want.

$E = E_0 \oplus E_1$ and $E = E_0 \parallel E_1$: Similar arguments as in lemma$_\Omega$ 7.4.4.

□

Proposition 9.4.25 For all $E_0, E_1 \in RBL_\Omega$:

a) $\delta_{\mathsf{w}}(A_{or}[\![E_0]\!]) \subseteq \delta_{\mathsf{w}}(A_{or}[\![E_1]\!])$ *iff* $E_0 \lesssim_{\mathsf{w}} E_1$

b) $\delta_{\mathsf{W}}(A^p_{or}[\![E_0]\!]_1) \subseteq \delta_{\mathsf{W}}(A^p_{or}[\![E_1]\!]_1)$ *iff* $E_0 \precsim_{\mathsf{W}} E_1$

Proof a) follows with exactly the same arguments as we now show b). b) follows from the definition of $\precsim_{\mathsf{W}}$ and the general deduction ($E \in RBL_\Omega$)

$$\begin{aligned}
\delta_{\mathsf{W}}(A^p_{or}[\![E]\!]_1) &= A^p_{\mathsf{W}}[\![E\sigma]\!]_1 && \text{proposition 9.2.21}\\
&= \delta_{\mathsf{W}}(\wp^p_1(E\sigma)) && \text{proposition 9.2.14 and } E\sigma \in BL_\Omega\\
&= \{w \in W \mid E\sigma \overset{w}{\Rightarrow}\} && \text{by (9.10)}\\
&= \{w \in W \mid E \overset{w}{\Rightarrow}\} && \text{proposition 9.3.2}
\end{aligned}$$

□

Lemma 9.4.26 Given an expression $E_0 \in BL_\Omega$ and a finite subset A of Δ. Then there is an $e \in \Delta$ such that for all $E_1 \in BL_\Omega$ with $L(E_1) \subseteq A$ we have

$$A^p_{\mathsf{G}}[\![E_0]\!] \not\subseteq A^p_{\mathsf{G}}[\![E_1]\!] \Rightarrow A^p_{\mathsf{G}}[\![E_0\,;e]\!]_1 \not\subseteq A^p_{\mathsf{G}}[\![E_1\,;e]\!]_1$$

Proof Let a $E_0 \in BL_\Omega$ be given. $L(E_0)$ is finite and containins $L(A^p_{\mathsf{G}}[\![E_0]\!]) = \bigcup\{L(\mathbf{p}) \mid \mathbf{p} \in A^p_{\mathsf{G}}[\![E_0]\!]_1 \cup A^p_{\mathsf{G}}[\![E_0]\!]_2\}$ where

$$a \in L(\mathbf{p}) \textit{ iff } m_{\mathbf{p}}(a) \neq 0$$

So since A is finite too, but Δ infinite, we can choose an $e \in \Delta$ that does occur in $L(A^p_{\mathsf{G}}[\![E_0]\!])$ or A. Before we start out proving that this e meats the requirement observe that for any $E \in BL_\Omega$ we have:

$$\begin{aligned}
A^p_{\mathsf{G}}[\![E\,;e]\!]_1 &= A^p_{\mathsf{G}}[\![E]\!]_1 \cup A^p_{\mathsf{G}}[\![E]\!]_2 \cdot \{\varepsilon, e\} && \text{definition of } ;^p_{\mathsf{G}} \text{ and } e^p_{\mathsf{G}}\\
&= A^p_{\mathsf{G}}[\![E]\!]_1 \cup A^p_{\mathsf{G}}[\![E]\!]_2 \cup A^p_{\mathsf{G}}[\![E]\!]_2 \cdot \{e\} && \varepsilon \text{ neutral to } \cdot\\
&= A^p_{\mathsf{G}}[\![E]\!]_1 \cup A^p_{\mathsf{G}}[\![E]\!]_2 \cdot \{e\} && A^p_{\mathsf{G}}[\![E]\!]_2 \subseteq A^p_{\mathsf{G}}[\![E]\!]_1 \text{ (in general)}
\end{aligned}$$

Now let an $E_1 \in BL_\Omega$ be given such that $L(E_1) \subseteq A$ and $A^p_{\mathsf{G}}[\![E_0]\!] \not\subseteq A^p_{\mathsf{G}}[\![E_1]\!]$. There are two ways how this can be:

$A^p_{\mathsf{G}}[\![E_0]\!]_2 \not\subseteq A^p_{\mathsf{G}}[\![E_1]\!]_2$: I.e., there is a $\mathbf{p} \in A^p_{\mathsf{G}}[\![E_0]\!]_2$ not in $A^p_{\mathsf{G}}[\![E_1]\!]_2$. By the observations above then $\mathbf{p} \cdot e \in A^p_{\mathsf{G}}[\![E_0\,;e]\!]_1$. $\mathbf{p} \cdot e \notin A^p_{\mathsf{G}}[\![E_1]\!]_1$ because $L(A^p_{\mathsf{G}}[\![E_1]\!]_1) \subseteq L(E_1) \subseteq A$ and $e \notin A$. $\mathbf{p} \notin A^p_{\mathsf{G}}[\![E_1]\!]_2$ implies $\mathbf{p} \cdot e \notin A^p_{\mathsf{G}}[\![E_1]\!]_2 \cdot \{e\}$ so from the observations above we conclude $\mathbf{p} \cdot e \notin A^p_{\mathsf{G}}[\![E_1\,;e]\!]_1$.

$A^p_{\mathsf{G}}[\![E_0]\!]_1 \not\subseteq A^p_{\mathsf{G}}[\![E_1]\!]_1$: Then let a $\mathbf{p} \in A^p_{\mathsf{G}}[\![E_0]\!]_1$ be given such that $\mathbf{p} \notin A^p_{\mathsf{G}}[\![E_1]\!]_1$. By the observations $\mathbf{p} \in A^p_{\mathsf{G}}[\![E_0\,;e]\!]_1$. We have $\mathbf{p} \notin A^p_{\mathsf{G}}[\![E_1]\!]_2 \cdot \{e\}$ because $\mathbf{p} \in A^p_{\mathsf{G}}[\![E_1]\!]_2 \cdot \{e\}$ would imply $e \in L(\mathbf{p}) \subseteq L(A^p_{\mathsf{G}}[\![E_0]\!])$ contradicting the way e is chosen. Hence $\mathbf{p} \notin A^p_{\mathsf{G}}[\![E_1]\!]_1 \cup A^p_{\mathsf{G}}[\![E_1]\!]_2 \cdot \{e\} = A^p_{\mathsf{G}}[\![E_0\,;e]\!]_1$. □

Lemma 9.4.27 Given an expression $E_0 \in RBL_\Omega$. Then there is a refinement combinator, $[\varrho]$, such that

$$\forall E_1 \in RBL_\Omega.\, A_{or}[\![E_0]\!] \not\subseteq A_{or}[\![E_1]\!] \Rightarrow \delta_{\mathsf{W}}(A_{or}[\![E_0[\varrho]]\!]) \not\subseteq \delta_{\mathsf{W}}(A_{or}[\![E_1[\varrho]]\!])$$

Proof From chapter 8 we already get the corresponding result for the M_{or}-model, but for the language RBL. All what can happen when Ω is added to the language is that $A_{or}[\![E_0]\!]$ might be empty in which case the implication holds vacuously. □

Lemma 9.4.28 Given an expression $E_0 \in RBL_\Omega$. Then there is a refinement combinator, ϱ, and an action $e \in \Delta$ such that

$$\forall E_1 \in RBL_\Omega.\, A^p_{or}[\![E_0]\!] \not\subseteq A^p_{or}[\![E_1]\!] \Rightarrow \delta_{\mathsf{w}}(A^p_{or}[\![E_0[\varrho]\,;e]\!]_1) \not\subseteq \delta_{\mathsf{w}}(A^p_{or}[\![E_1[\varrho]\,;e]\!]_1)$$

Proof Let $E_0 \in RBL_\Omega$ be given. As for $A_{or}[\![\_]\!]$ we are after a fission refinement, ϱ, such that any pomset, $\mathbf{p}$, associated with the denotation of E_0 can be reflected in a linearization of $\mathbf{q} \in \mathbf{p}{<}\varrho{>}$, but this time with the additional requirement that e does not occcur in any pomset which steems from a ${<}\varrho{>}$-refinement of a pomset associated with the denotation of an arbitrary $E_1 \in RBL_\Omega$. Since E_1 can be any syntactic finite expression there are practical no limitations on what singleton pomsets there may be in a pomset from its denotation. We can therefore just as well pick an arbitrary $e \in \Delta$ and seek a fission refinement ϱ for E_0 such that

(9.11) $$\forall a \in \Delta.\, e \notin L(\varrho(a))$$

Let m be the lub of the multiplicity functions associated with the pomsets of $A^p_{or}[\![E_0]\!]$, i.e., $m = \bigvee\{m_{\mathbf{p}} \mid \mathbf{p} \in A^p_{or}[\![E_0]\!]_1 \cup A^p_{or}[\![E_0]\!]_2\}$ (finite because $E_0 \in RBL_\Omega$). $\Delta\setminus\{e\}$ is (countable) infinite because Δ is, so from the arguments about the existence of fission refinements it should be clear we also can find a m-fission refinement ϱ with desired property (9.11). Remember when dealing with fission refinements we use the same symbol for the BL-fission refinement and the $\mathcal{P}(\mathbf{P})$-fission refinement.

Before we continue notice as in lemma 9.4.26 (for any $E \in RBL_\Omega$)

$$A^p_{or}[\![E[\varrho]\,;e]\!]_1 = A^p_{or}[\![E[\varrho]]\!]_1 \cup A^p_{or}[\![E[\varrho]]\!]_2 \cdot \{e\}$$

Now let any $E_1 \in RBL_\Omega$ be given and suppose $A^p_{or}[\![E_0]\!] \not\subseteq A^p_{or}[\![E_1]\!]$. Assume on the contrary $\delta_{\mathsf{w}}(A^p_{or}[\![E_0[\varrho]\,;e]\!]_1) \subseteq \delta_{\mathsf{w}}(A^p_{or}[\![E_1[\varrho]\,;e]\!]_1)$. There are two ways how $A^p_{or}[\![E_0]\!] \not\subseteq A^p_{or}[\![E_1]\!]$ can be:

$A^p_{or}[\![E_0]\!]_2 \not\subseteq A^p_{or}[\![E_1]\!]_2$: Then there is a $\mathbf{p} \in A^p_{or}[\![E_0]\!]_2$ with $\mathbf{p} \notin A^p_{or}[\![E_1]\!]_2$. Since $A^p_{or}[\![E_0]\!]_2$ is δ_{or}-closed $\mathbf{p}$ must have the P_{or}-property. Because ϱ is m-fission refinement and $m_{\mathbf{p}} \leq m$ we can use lemma 8.3.6 to find a $w \in \delta_{\mathsf{w}}(\mathbf{p}{<}\varrho{>})$ which is $\mathbf{p}$-reflecting. We then have:

$$\begin{aligned}
w \cdot e &\in \delta_{\mathsf{w}}(A^p_{or}[\![E_0]\!]_2{<}\varrho{>}) \cdot \{e\} & \\
&= \delta_{\mathsf{w}}(\delta_{or}(A^p_{or}[\![E_0]\!]_2{<}\varrho{>})) \cdot \{e\} & \delta_{\mathsf{w}} \circ \delta_{or} = \delta_{\mathsf{w}} \\
&= \delta_{\mathsf{w}}(A^p_{or}[\![E_0[\varrho]]\!]_2) \cdot \{e\} & \text{definition of } [\varrho]^p_* \\
&= \delta_{\mathsf{w}}(A^p_{or}[\![E_0[\varrho]]\!]_2 \cdot \{e\}) & \delta_{\mathsf{w}} \text{ distributes over } \cdot, \delta_{\mathsf{w}}(\{e\}) = \{e\} \\
&\subseteq \delta_{\mathsf{w}}(A^p_{or}[\![E_0[\varrho]\,;e]\!]_1) & \text{from notice and } \subseteq \text{ monotonicity of } \delta_{\mathsf{w}} \\
&\subseteq \delta_{\mathsf{w}}(A^p_{or}[\![E_1[\varrho]\,;e]\!]_1) & \text{assumption} \\
&= \delta_{\mathsf{w}}(A^p_{or}[\![E_1[\varrho]]\!]_1 \cup A^p_{or}[\![E_1[\varrho]]\!]_2 \cdot \{e\}) & \text{from notice} \\
&= \delta_{\mathsf{w}}(A^p_{or}[\![E_1[\varrho]]\!]_1) \cup \delta_{\mathsf{w}}(A^p_{or}[\![E_1[\varrho]]\!]_2) \cdot \{e\} &
\end{aligned}$$

Because $A^p_{or}[\![E_1[\varrho]]\!]_1 = \delta_{or}(A^p_{or}[\![E_1]\!]_1{<}\varrho{>})$ we see from (9.11) that $e \notin L(\delta_{\mathsf{w}}(A^p_{or}[\![E_1[\varrho]]\!]_1))$. Hence also $w \cdot e \notin \delta_{\mathsf{w}}(A^p_{or}[\![E_1[\varrho]]\!]_1)$ and we are left with $w \cdot e \in \delta_{\mathsf{w}}(A^p_{or}[\![E_1[\varrho]]\!]_2) \cdot \{e\}$. But then $w \in \delta_{\mathsf{w}}(A^p_{or}[\![E_1[\varrho]]\!]_2) = \delta_{\mathsf{w}}(\delta_{or}(A^p_{or}[\![E_1]\!]_2{<}\varrho{>}))$. This means there is a $\mathbf{p}_1 \in A^p_{or}[\![E_1]\!]_2$ and $\mathbf{q} \in \mathbf{p}_1{<}\varrho{>}$ such that $w \preceq \mathbf{q}$. Since w is $\mathbf{p}$-reflecting we by lemma 8.3.5 get $\mathbf{p} \preceq \mathbf{p}_1$. Because $P_{or}(\mathbf{p})$ and $A^p_{or}[\![E_1]\!]$ is δ_{or}-closed this implies $\mathbf{p} \in A^p_{or}[\![E_1]\!]_2$—a contradiction.

$A^p_{or}[\![E_0]\!]_1 \not\subseteq A^p_{or}[\![E_1]\!]_1$: We see there exists a $\mathbf{p} \in A^p_{or}[\![E_0]\!]_1$ such that $\mathbf{p} \notin A^p_{or}[\![E_1]\!]_1$ and $P_{or}(\mathbf{p})$ because $A^p_{or}[\![E_0]\!]_1$ is δ_{or}-closed (as well as π-closed). We can also here find a $\mathbf{p}$-reflecting linearization $w \in \delta_{\mathsf{w}}(\mathbf{p}{<}\varrho{>})$. Notice that because of (9.11) we have $e \notin L(w)$. We infer:

$$\begin{aligned}
w &\in \delta_{\mathsf{W}}(A^p_{or}[\![E_0]\!]_1 < \varrho >) \\
&\subseteq \delta_{\mathsf{W}}(\pi(A^p_{or}[\![E_0]\!]_1 < \varrho >)) && \delta_{\mathsf{W}} \text{ is } \subseteq\text{-montone and } P \subseteq \pi(P) \\
&= \delta_{\mathsf{W}}(\delta_{or}\pi(A^p_{or}[\![E_0]\!]_1 < \varrho >)) && \delta_{\mathsf{W}} \circ \delta_{or} = \delta_{\mathsf{W}} \\
&\subseteq \delta_{\mathsf{W}}(A^p_{or}[\![E_0[\varrho]\,;\,e]\!]_1) && \text{from notice and definition of } ;^p_* \\
&\subseteq \delta_{\mathsf{W}}(A^p_{or}[\![E_1[\varrho]\,;\,e]\!]_1) && \text{assumption} \\
&= \delta_{\mathsf{W}}(A^p_{or}[\![E_1[\varrho]]\!]_1) \cup \delta_{\mathsf{W}}(A^p_{or}[\![E_1[\varrho]]\!]_2) \cdot \{e\} && \text{as above}
\end{aligned}$$

$e \notin L(w)$ excludes $w \in \delta_{\mathsf{W}}(A^p_{or}[\![E_1[\varrho]]\!]_2) \cdot \{e\}$ and we are left with $w \in \delta_{\mathsf{W}}(A^p_{or}[\![E_1[\varrho]]\!]_1) = \delta_{\mathsf{W}}(\delta_{or}\pi((A^p_{or}[\![E_1]\!]_1) < \varrho >)) = \delta_{\mathsf{W}}\pi((A^p_{or}[\![E_1]\!]_1) < \varrho >)$. Then there must be pomsets such that

$$w \preceq \mathbf{q} \sqsubseteq \mathbf{q}' \in \mathbf{p}_1 < \varrho > \text{ where } \mathbf{p}_1 \in A^p_{or}[\![E_1]\!]_1$$

w is the linearization of some pomset refined by $<\varrho>$ and therefore must be balanced w.r.t. to the fission pairs of ϱ. Because $w \preceq \mathbf{q}$ they have the same labels and so $\mathbf{q}$ must be balanced w.r.t. to the fission pairs. With $\mathbf{q} \sqsubseteq \mathbf{q}' \in \mathbf{p}_1 < \varrho >$ we can then use the lemma below to conclude there is a pomset $\mathbf{p}'_1 \sqsubseteq \mathbf{p}_1$ such that $\mathbf{q} \in \mathbf{p}'_1 < \varrho >$. Because $w \preceq \mathbf{q} \in \mathbf{p}'_1 < \varrho >$ and w is $\mathbf{p}$-reflecting we can as in the case above conclude $\mathbf{p} \preceq \mathbf{p}'_1$. $A^p_{or}[\![E_1]\!]_1$ is both δ_{or}- and π-closed, so from $\mathbf{p}'_1 \sqsubseteq \mathbf{p}_1 \in A^p_{or}[\![E_1]\!]_1$ and $P_{or}(\mathbf{p})$ we then get $\mathbf{p} \in A^p_{or}[\![E_1]\!]_1$—again a contradiction. □

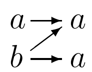
Lemma 9.4.29 Let a finite multiplicity function m over Δ be given together with a ε-free m-fission refinement ϱ. Suppose $\mathbf{p}, \mathbf{q}$ and $\mathbf{r}$ are pomsets such that $\mathbf{p} \sqsubseteq \mathbf{q} \in \mathbf{r} < \varrho >$. If $\mathbf{p}$ is *balanced* w.r.t. to the fission pairs of ϱ in the sense:

$$\forall a \in \Delta \forall k \in \underline{n(m)}.\ m_{\mathbf{p}}(a_{S_k}) = m_{\mathbf{p}}(a_{F_k})$$

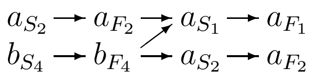
then there is a pomset $\mathbf{s} \sqsubseteq \mathbf{r}$ such that $\mathbf{p} \in \mathbf{s} < \varrho >$.

Proof By definition of the refinement operator, $\mathbf{q} \in \mathbf{r} < \varrho >$ means there is a ϱ-consistent p.ref., π_r, for r such that $\mathbf{q} = [r < \pi_r >]$. Then also $\mathbf{p} \sqsubseteq [r < \pi_r >]$.

We illustrate the situation by an example. Suppose r is the representative of the pomset



Then $[r < \pi_r >]$ typically may look like:



Evidently no matter how $\mathbf{p}$ is a ($\leq_{r<\pi_r>}$-downwards closed) prefix of $[r < \pi_r >]$ then for the fission pair a_{S_2}, a_{F_2} the number of times a_{S_2} occur in $\mathbf{p}$ must be greater than or equal the number of times a_{F_2} occur in $\mathbf{p}$. Similar for the other fission pairs. Clearly also if these numbers balance for every fission pair then there can be no element of $\mathbf{p}$ labelled say a_{S_1} without an immediate following element labelled a_{F_1}. By the nature of fission refinement these two elements must originate from the same element in $\mathbf{r}$ and then $\mathbf{p}$ must be the refinement of a prefix of $\mathbf{r}$.

Now formally $\mathbf{p} \sqsubseteq [r{<}\pi_r{>}]$ by the alternative characterization of $\sqsubseteq$ implies that we can find a representative p' of $\mathbf{p}$ such that

$$p' = r{<}\pi_r{>}|_{X_{p'}} \text{ and } X_{p'} \text{ is } \leq_{r<\pi_r>} \text{-downwards closed}$$

Notice this implies $X_{p'} \subseteq X_{r<\pi_r>}$. It then gives sense to define $Y = \{x \in X_r \mid \langle x, x' \rangle \in X_{p'}\}$ and $s = r|_Y$.

At first we show Y to be $\leq_r$-downwards closed thus gaining $\mathbf{s} \sqsubseteq \mathbf{r}$. Given an $x \in X_r$ and $y \in Y$ such that $x \leq_r y$. We shall show $x \in Y$. Now $y \in Y$ means there is some y' with $\langle y, y' \rangle \in X_{p'}$. Because ϱ is ε-free there must be an $x' \in X_{\pi_r(x)}$. By construction of $r{<}\pi_r{>}$ then $\langle x, x' \rangle \in X_{r<\pi_r>}$ and from $x \leq_r y$ also $\langle x, x' \rangle \leq_{r<\pi_r>} \langle y, y' \rangle$. Because $X_{p'}$ is $\leq_{r<\pi_r>}$-downwards closed this implies $\langle x, x' \rangle \in X_{p'}$ and by definition of Y then $x \in Y$ as we want.

Now $Y \subseteq X_r$ so we let π_s be $\pi_r|_Y$ which clearly is a ϱ-consistent particular fission refinement for s. We then wish to show

$$r{<}\pi_r{>}|_{X_{p'}} = s{<}\pi_s{>}$$

To do this we show at first that $X_{p'} = X_{s<\pi_s>}$.

- $\subseteq$: $\langle x, x' \rangle \in X_{p'} \subseteq X_{r<\pi_r>}$ implies $x \in Y$ and $x' \in X_{\pi_r(x)}$. Since $\pi_s = \pi_r|_Y$ and $x \in Y$ we have $x' \in X_{\pi_s(x)}$ and by construction of $s{<}\pi_s{>}$ also $\langle x, x' \rangle \in X_{s<\pi_s>}$.
- $\supseteq$: Calling in mind the observation we did on page 185 about ϱ-consistent particular fission refinements, we can especially for π_r make the following deductions: If $x \in X_r$ then there are exactly two elements in $X_{r<\pi_r>}$ with first component x, namely $x_S^{\pi_r}$ and $x_F^{\pi_r}$. Recall that $x_S^{\pi_r}$ is the $\langle x, x' \rangle \in X_{r<\pi_r>}$ where x' is that element of $X_{\pi_r(x)}$ with label $\ell_{\pi_r(x)}(x') = a_{S_k}$ whereto a equals $\ell_r(x)$ and $k \in \underline{n(m)}$. Similar for $x_F^{\pi_r}$ and we have seen $x_S^{\pi_r} <_{r<\pi_r>} x_F^{\pi_r}$. So $x_F^{\pi_r} \in X_{p'}$ implies $x_S^{\pi_r} \in \overline{X_{p'}}$ because $X_{p'}$ is $\leq_{r<\pi_r>}$-downwards closed. It follows that if there is one $x_S^{\pi_r}$ in $X_{p'}$ for some $x \in X_r$ without the corresponding $x_F^{\pi_r}$ in $X_{p'}$ then the number of a_{S_k}'s in $r{<}\pi_r{>}|_{X_{p'}}$ must be at least one less than the number of a_{F_k}'s where $\ell_{r<\pi_r>}(x_S^{\pi_r}) = a_{S_k}$. Hence $r{<}\pi_r{>}|_{X_{p'}}$ cannot be ballanced w.r.t. the fission pairs of ϱ if there is one $x_S^{\pi_r}$ in $X_{p'}$ without the corresponding $x_F^{\pi_r}$. By the proviso of the lemma $p' = r{<}\pi_r{>}|_{X_{p'}}$ is balanced so we conclude there can be no such $x_S^{\pi_r}$'s.
  We can now return to the question of $X_{s<\pi_s>} \subseteq X_{p'}$. Suppose $\langle x, x' \rangle \in X_{s<\pi_s>}$. This means $x \in Y$ and $x' \in X_{\pi_s(x)} = X_{\pi_r|_Y(x)} = X_{\pi_r(x)}$. $x \in Y$ implies by definition of Y the existence of an x'' such that $\langle x, x'' \rangle \in X_{p'}$. If $x' = x''$ we are done immediately. Now $\langle x, x'' \rangle \in X_{p'} \subseteq X_{r<\pi_r>}$ only if $x \in X_r$ and $x'' \in X_{\pi_r(x)}$, so when $x' \neq x''$ we get from $x \in X_r$ and $x' \in X_{\pi_r(x)}$, that either $(\langle x, x' \rangle = x_S^{\pi_r}, \langle x, x'' \rangle = x_F^{\pi_r})$ or $(\langle x, x' \rangle = x_F^{\pi_r}, \langle x, x'' \rangle = x_S^{\pi_r})$. As argued above we must have $\langle x, x' \rangle \in X_{p'}$ in the former case because $X_{p'}$ is $\leq_{r<\pi_r>}$-downwards closed and in the latter also $\langle x, x' \rangle \in X_{p'}$, but this time because $X_{p'}$ is balanced w.r.t. to the fission pairs of ϱ.

Having proved $X_{p'} = X_{s<\pi_s>}$ it is an easy matter to show $r{<}\pi_r{>}|_{X_{p'}} = s{<}\pi_s{>}$ from the definition of s, π_s and Y. For instance w.r.t. labels we have: $\langle x, x' \rangle \in X_{p'}$ implies $x \in Y$ so $\ell_{r<\pi_r>|_{X_{p'}}}(\langle x, x' \rangle) = \ell_{\pi_r(x)}(x') = \ell_{\pi_r(x)|_Y}(x') = \ell_{\pi_s(x)}(x') = \ell_{s<\pi_s>}(\langle x, x' \rangle)$—the last equality from $\langle x, x' \rangle \in X_{p'} = X_{s<\pi_s>}$.

Collecting the facts we have $p' = s{<}\pi_s{>}$ so of course $\mathbf{p} = \mathbf{p}' = [s{<}\pi_s{>}]$. Since π_s is ϱ-consistent then also $\mathbf{p} = [s{<}\pi_s{>}] \in \mathbf{s}{<}\varrho{>}$. We have already shown $\mathbf{s} \sqsubseteq \mathbf{r}$ wherefore the proof is completed. □

Index

Constants

Functions

Operations

Relations

Alphabetical