Models, Languages and Logics for Concurrent Distributed Systems CEDISYS Workshop

Editors:
Uffe Engberg
Mogens Nielsen
Glynn Winskel

April 1992

CEDISYS Workshop in Aarhus, May 21-24, 1991

Introduction

The EEC Esprit Basic Research Action No 3011, Models, Languages and Logics for Concurrent Distributed Systems, CEDISYS, held its second workshop at Aarhus University in May, 1991, following the successful workshop in San Miniato in 1990.

The Aarhus Workshop was centered around CEDISYS research activities, and the selected themes of Applications and Automated Tools in the area of Distributed Systems. The 24 participants were CEDISYS partners, and invited guests with expertise on the selected themes.

The workshop was considered to be yet another successful CEDISYS event, and we would like to thank our invited guest speakers and the participants for their scientific contributions, the local organizers for their assistance, and the EEC Basic Research Actions initiative for its support.

This booklet contains the program of the workshop, short abstracts for the talks presented, and a list of participants.

Uffe Engberg Mogens Nielsen Glynn Winskel

CEDISYS Workshop in Aarhus, May 21-24, 1991

Program

Lecture room D3, Computer Science Department.

Tuesday, May 21

9.00– 9.45	*The Observation Algebra of Spatial Histories,* Ugo Montanari
9.45–10.30	*Transition systems and Petri nets,* Glynn Winskel
	Coffee Break
11.15–12.00	*A category of Petri nets as a model of linear logic,* Carolyn Brown
	Lunch
14.00–14.45	*Relating Location Equivalence and Causal Bisimulations,* Astrid Kiehn
14.45–15.30	*Testing Concurrent Processes,* Matthew Hennessy
	Coffee Break
16.15–17.00	*A new technique for proving translations correct,* Anders Gammelgaard

Wednesday, May 22

9.00– 9.45 *Experience with a process algebra tool,* Dirk Taubner

9.45–10.30 *Ecrins/Auto/Autograph: Verification Tools for Process Calculi,* Robert de Simone

Coffee Break

11.15–12.00 *An Action-Based Framework for Verifying Logical and Behavioural Properties of Concurrent Systems,* Rocco De Nicola

Lunch

14.00–14.45 *The New TAV System,* Kim Guldstrand Larsen

14.45–15.30 *Generating BDDs for Symbolic Model Checking in CCS,* Dirk Taubner

Coffee Break

16.15–17.00 Discussion

Thursday, May 23

9.00– 9.45 *Atomic Refinement,* Matthew Hennessy

9.45–10.30 *Atomic and Non-Atomic Action Refinement,* Pierpaolo Degano

Coffee Break

11.15–12.00 *Observing localities,* Ilaria Castellani

Lunch

14.00–14.45 *Priority in CCS,* Glynn Winskel

14.45–15.30 *Compositional Checking of Satisfaction,* Henrik Reif Andersen

Coffee Break

16.00–16.45 *Parametric Semantics for Process Description Languages,* Rocco De Nicola

16.45–17.30 *Three Equivalent Semantics for CCS,* Gerard Boudol

Dinner

Friday, May 24

9.00–12.00 Review meeting

1. A general overview of the achievements by Ugo Montanari
2. Models by Mogens Nielsen
3. Logics and Proof Systems by Glynn Winskel
4. Languages by Gerard Boudol
5. A general perspective by Ugo Montanari
6. Discussion

Lunch

CEDISYS Workshop in Aarhus, May 21-24, 1991

Abstracts

Tuesday, May 21

The Observation Algebra of Spatial Histories, Ugo Montanari

Abstract
Observations can be described in a uniform way by introducing certain algebras called observation algebras: they lift to an algebraic level the standard treatment of actions in the operational semantics of process algebras. We introduce an observation algebra for CCS whose elements are labelled partial orderings of events called Spatial Histories. As a result we automatically obtain a truly concurrent semantics for CCS. (work by G. Ferrari, R. Gorrieri and U. Montanari)

Transition systems and Petri nets, Glynn Winskel

Abstract
This work arose out of an attempt to explain in a uniform way the relationships between models of concurrency. The idea is that a (notion of) model of computation is presented as a category with operations of the kind used in semantics of concurrent processes arising as universal constructions; relations between models are expressed by functors, often forming an adjunction, In particular the fact that such categories can be viewed as fibrations, projecting to a category of label sets, provides universal characterisations of those constructions for which labelling is crucial. This talk will concentrate on some recent results on the relationship between a category of safe Petri nets and a category of “concurrent transition systems”. To some extent it generalises the work of Nielsen, Rosenberg and Thiagarajan, on an adjunction between elementary transition systems and elementary nets, to allow the occurrence of events with conditions which are both pre and post-conditions; an extension of their notion of “region” is required. This generalisation is needed in order to carry the structure necessary for a fibration and repair the “deficiency” that labelled safe nets do not form a fibration, essentially because they do not always have enough conditions. The new objects of “concurrent transi-

tion systems" arose from the observation that two adjunctions, one between nets and event structures and another between nets and transition systems, factored through this common class of structures.

A category of Petri nets as a model of linear logic, Carolyn Brown

Abstract
This paper defines a category **GNet** with object set the set of all Petri nets. A morphism in **GNet** from a net N to a net N gives a precise way of simulating every evolution of N by an evolution of N'. We exhibit a morphism from a simple message handler to one with error-correction, showing that the more refined message handler can simulate any behaviour of its simple counterpart. The existence of such a morphism proves the correctness of the refinement.

We have previously defined a modular theory of elementary Petri nets based on de Paiva's dialectica category models of linear logic. We here generalise her construction, defining categories $\mathbf{M}_{\mathrm{N}}\mathbf{C}$ which model linear logic without modalities or negation. **GNet** arises naturally from $\mathbf{M}_{\mathbb{N}}\mathbf{Set}$, inheriting the structure which models linear logic. We describe some net constructors which this yields.

This more general framework has several advantages over our previous one. The theory is simplified, we obtain precise results about morphisms as simulations, relating them to CCS, and we obtain a natural extension to marked nets.

Relating Location Equivalence and Causal Bisimulations, Astrid Kiehn

Abstract
Location Equivalence and Causal Bisimulations yield semantic equivalences for CCS which are incomparable even for finite restriction-free processes, To give a better understanding of the differences between locations and causes we introduce a new transition system based on local and global causes. Over this transition system we define a bisimulation parameterized by a function f which evaluates the information on local and global causes provided by the transitions. Choosing appropriate instantiations for f we obtain location equivalence and causal bisimulations.

Testing Concurrent Processes, Matthew Hennessy

Abstract
For a language very similar to CCS we give a natural characterisation of the largest equivalence cuntained in testing equivalence which is preserved by

action refinement, This is an extension of recent work by Aceto and Engberg.

A new technique for proving translations connect, Anders Gammelgaard

Abstract

An impurtant part of the ProCoS project is to prove a translation correct. The source language for the translation is a subset of occam-2 and the target language is an abstract assembly language for the transputer. The translation is specified recursively using the structure of source programs.

Correctness nutiuns as Milners observation equivalence and the refinement relation in failure semantics are inadequate for the chosen language (and hence for occam-2). Yet, we want to retain as much as possible from these notions and their associated proof techniques. To accommodate other parts of the ProCoS project we choose an external semantics much in the style of failure semantics but without the identification of divergence with chaos. Internally we define an ordinary sos-semantics for both the source and the target language. This is done because we believe that the close correspondence between executions of a source program and its translation is best reflected by operational models. Furthermore such models enable us to use a modified bisimulation technique in order to establish correct refinement in the external semantics.

The bisimulatiun technique has to be modified rather drastically, however. We end up with a technique where simulatiuns must be found, not for single transitions, but for whole chunks of transitions occurring in execution sequences. The new simulation relations are built up inductively through an inference system.

For parallel programs we get the problem that chunks from different processes may overlap very inconveniently. We solve the problem by giving operational semantics to parallel programs much as done in the chemical abstract machine by Berry and Boudul and in the grape semantics by Degano, De Nicola, and Montanari. We furthermore have to introduce the concept of a truly distributed execution in such semantics.

Wednesday, May 22

Experience with a process algebra tool, Dirk Taubner

Abstract

We describe the components of a typical tool for the verification of parallel processes based on process algebras. Process algebras such as CCS, TCSP, and ACP offer two means of verification, equivalence checking and model checking. For finite-state processes we describe algorithms for both problems and for the needed non-trivial preprocessing. We indicate how the approach may be exploited for verifying communication protocols.

Ecrins/Auto/Autograph: Verification Tools for Process Calculi, Robert de Simone

Abstract
Missing

An Action-Based Framework for Verifying Logical and Behavioural Properties of Concurrent Systems, Rocco De Nicola

Abstract
A system is described which supports proofs of both behavioral and logical properties of concurrent systems which are specified by means of a process algebra and its associated logics, The latter is an action based version interpreted over labelled transition systems, of the branching time logic CTL. The system is the result of integrating two existing tools, AUTO and EMC. The integration is realized by means of two transition functions from the action based branching time logic ACTL to CTL and from labelled transition systems to Kripkestructures. (work by R. De Nicola, A. Fantechi, S. Gnesi, G. Ristori)

The New TAV System, Kim Guldstrand Larsen

Abstract
TAV is a verification system for parallel and nondeterministic system expressed within the calculus of CCS. In particular TAV contains tools for deciding various notions of bisimilarity (equivalence checking) between processes, and contains tools for model–checking with respect to a rather powerful recursive extension of Hennessy–Milner Logic. A distinctive feature of the tools of TAV — important from a development point of view — is, that they all offer explanations for the answers they give.

In the New TAV system (which is almost finished) all tools has been extended to operate on modal transition systems. Modal transition systems extends ordinary labelled transitions systems in that transitions come in two flavors: transitions which are required of any implementation, and transitions which

are allowed by any implementation. As a result, modal transition systems enables loose specifications to be expressed in a "graphical" way. The relative strength of modal transition systems is captured by a notion of refinement extending the classical notion of bisimilarity.

An equation solving tool has also been added in the New TAV system (being heavily based on the introduction of modal transition systems).

In the talk I will present, motivate and demonstrate the tools of the New TAV system. Also, if time permits, I will comment on the (universal) algorithm which is used in the (New) TAV system both in the equivalence-checking and in the model-checking part. The algorithm is quite different from the usual partition-like algorithms; in fact checking is done in local fashion (exploring only the relevant state-space).

Generating BDDs for Symbolic Model Checking in CCS, Dirk Taubner

Abstract
Finite transition systems can easily be represented by binary derision diagrams (BDDs) through the characteristic function of the transition relation. Burch et al. have shown how model checking of a powerful version of the μ-calculus can be performed on such BDDs.

In this paper we show how a BDD can be generated from elementary finite transition systems given as BDDs by applying the CCS operations of parallel composition, restriction, and relabelling It appears that the resulting BDDs only grow linearly in the number of parallel components.

This way bisimilarity checking can be performed for processes out of the reach of conventional process algebra tools.

Thursday, May 23

Atomic Refinement, Matthew Hennessy

Abstract
We show that action refinement provides excellent motivation for semantic equivalences which distinguish nondeterminism from concurrency. We give a natural generalisation of two of the standard semantic equivalences, observational equivalence and testing equivalence and show that these generalisations are preserved by refinement. Indeed these generalisations are

characterised by action refinements in the sense that they are the largest equivalences preserved by refinement which are contained in the standard ones.

Atomic and Non-Atomic Action Refinement, Pierpaolo Degano

Abstract
The issue of action refinement in Process Description Languages is addressed, aiming at providing system designers with a programming feature that permits hierarchical system specification. Therefore, the semantic definitions of action refinement will be driven by methodological constraints. We will discuss atomic and non atomic action refinement, i.e., two notions according to which t he process replacing an action is executed atomically or otherwise. (work by P. Degano, R. Gorrieri, C. Laneve and U. Montanari)

Observing localities, Ilaria Castellani

Abstract
We introduce a new kind of transition system, where the actions are performed at explicit locations. Then we define notions of bisimulation preorder and equivalence for such transition systems. Using this model, we give an operational semantics for the CCS process description language, extended with a construct introducing locations. A complete axiomatization for both the preorder and location equivalence of finite terms is given.

Priority in CCS, Glynn Winskel

Abstract
This paper investigates an extension of Milner's CCS with a priority choice operator called prisum: this operator is very similar to the PRIALT construct of Occam. The new binary prisum operator only allows execution of its second component in the case where the environment is not ready to allow the first component to proceed. This dependency on the set of actions the environment is ready to perform goes beyond that encountered in traditional CCS. Its expression leads to a novel operational semantics in which transitions carry ready-sets (of the environment) as well as the normal action symbols from CCS. A notion of strong bisimulation is defined on agents with priority via this semantics. It is a congruence and satisfies new equational laws (including a new expansion law) which are shown to be complete for finite agents with prisum. The laws are conservative over agents of traditional CCS.

Compositional Checking of Satisfaction, Henrik Reif Andersen

Abstract

We present a compositional method for deciding whether a process satisfies an assertion. Assertions are formulae in a modal ν-calculus, and processes are drawn from a very general process algebra inspired by CCS and CSP. Well-known operators from CCS, CSP, and other process algebras appear as derived operators.

The method is *compositional in the structure of processes* and works purely on the syntax of processes. It consists of applying a sequence of *reductions*, each of which only take into account the top-level operator of the process. A reduction transforms a satisfaction problem for a composite process into equivalent satisfaction problems for the immediate subcomponents.

Using process variables, systems with undefined subcomponents can be defined, and given an overall requirement to the system, *necessary and suficient conditions* on these subcomponents can be found. Hence the process variables make it possible to specify and reason about what are often referred to as *contexts*, *environments*, and *partial implementations.*

As reductions are algorithms that work on syntax, they can be considered as forming a bridge between traditional non-compositional model checking and compositional proof systems.

Parametric Semantics for Process Description Languages, Rocco De Nicola

Abstract

PDL's can be given operational and axiomatic bisimulation semantics which are parameterized with respect to various notions of observation.
First, it is shown that observations can be described in a uniform way by introducing certain algebras called observation algebras. A parameterized form of the expansion theorem is defined, which is the heart of a finite axiomatization of a strong observational equivalence for finite CCS agents.
Second, certain node-labelled trees, called observation trees, are introduced as a general framework for capturing various observational equivalences for concurrent distributed systems. It is shown that several of the bisimulation-based equivalences known in the literature can be recast in terms of equivalences of observation trees. This recasting provides an axiomatization also for some equivalences which lacked one. (work by P. Degano, R. De Nicola, G. Ferrar i, R. Gorrieri and U. Montanari)

Three Equivalent Semantics for CCS, Gerard Boudol

Abstract

We define a semantics for CCS by means of flow event structures and, for terms without recursion, we define a flow net semantics. These two semantics provide us with domains of computations that are both isomorphic to the domain of "transitions up to permutations". This last interpretation is an adaptation of Berry and Levy's notion of equivalence by permutation of sequences of (prove d) transitions. In all these semantics, a CCS term may be regarded as performing posets of events instead of atomic actions.

CEDISYS Workshop in Aarhus, May 21-24, 1991

Participants

Sites

Henrik Reif Andersen	Tel:	+45 86127188
Aarhus University	Fax:	+45 86135725
Computer Science Department	E-mail:	henrikan@daimi.aau.dk
Ny Munkegade		
DK – 8000 Arhus C		
Gerard Boudol	Tel:	+33 9395 7474
INRIA - Sophia Antipolis	Fax:	
Route de Lucioles	E-mail:	gbo@cma.cma.fr
F – 6560 Valbonne		
Carolyn Brown	Tel:	+45 86127188
Aarhus University	Fax:	+45 86135725
Computer Science Department	E-mail:	cbrown@daimi.aau.dk
Ny Munkegade		
DK – 8000 Arhus C		
Ilaria Castellani	Tel:	+33 9395 7474
INRIA - Sophia Antipolis	Fax:	
Avenue Emile Hugues	E-mail:	ic@cma.cma.fr
F – 6560 Valbonne		
Søren Christensen	Tel:	
University of Edinburgh	Fax:	
Department of Computer Science	E-mail:	soc@lfcs.edinburgh.ac.uk
The King's Buildings		
UK – Edinburgh EH9 3JZ		

Pierpaolo Degano	Tel:	
University of Pisa	Fax:	
Dipartimento di Informatica	E-mail:	degano@di.unipi.it
Corso Italia, 40		
I – 56125 Pisa		

Rocco De Nicola	Tel:	+39 50 553159 (PISA)
IEI-CNR		+39 6 8841957
Via S. Maria 46		(Rome Tuesday to Thursday)
I – 56126 Pisa	E-mail:	denicola@icnucevm.cnuce.cnr.it

Uffe Engberg	Tel:	+45 86127188
Aarhus University	Fax:	+45 86135725
Computer Science Department	E-mail:	engberg@daimi.aau.dk
Ny Munkegade		
DK – 8000 Arhus C		

Anders Gammelgaard	Tel:	+45 86127188
Aarhus University	Fax:	+45 86135725
Computer Science Department	E-mail:	gammelg@daimi.aau.dk
Ny Munkegade		
DK – 8000 Arhus C		

Douglas Gurr	Tel:	+45 86127188
Aarhus University	Fax:	+45 86135725
Computer Science Department	E-mail:	dgurr@daimi.aau.dk
Ny Munkegade		
DK – 8000 Arhus C		

Matthew Hennessy	Tel:	(0273)678101
University of Sussex	Fax:	
CSAI	E-mail:	matthewh@cogs.sussex.ac.uk
Falmer		
UK – Brighton BNl 9QH		

Claus Torp Jensen
Aarhus University
Computer Science Department
Ny Munkegade
DK – 8000 Arhus C
Tel: +45 86127188
Fax: +45 86135725
E-mail: ctjmrr@daimi.aau.dk

Astrid Kiehn
Technische Universität München
Institut für Informatik
Postfach 20 24 20
W – 8000 Munchen 2
Tel: +49 89 2105 2389
Fax:
E-mail: kiehn@lan.informatik.tu-muenchen.dbp.de

Ugo Montanari
University of Pisa
Dipartimento di Informatica
Corso Italia, 40
I – 56100 Pisa
Tel: +39 5051 0221
Fax: +39 5051 0226
E-mail: ugo@dipisa.diunipi.it

Mogens Nielsen
Aarhus University
Computer Science Department
Ny Munkegade
DK – 8000 Arhus C
Tel: +45 86127188
Fax: +45 86135725
E-mail: mn@daimi.aau.dk

Glynn Winskel
Aarhus University
Computer Science Department
Ny Munkegade
DK – 8000 Arhus C
Tel: +45 86127188
Fax: +45 86135725
E-mail: gwisnkel@daimi.aau.dk

Speakers

Kim Guldstrand Larsen
Aalborg University Center
Dept. of Math. and Camp. Science
Frederik Bajersvej 7
DK – 9220 Aalborg
Tel: +45 98158522
Fax: +45 98158129
E-mail: kgl@iesd.auc.dk

Robert de Simone	Tel:	
INRIA - Sophia Antipolis	Fax:	
Route des Lucioles	E-mail:	rs@cma.cma.fr
Sophia-Antipolis		
F – 06561 Valbonne CEDEX		
Dirk Taubner	Tel:	+49 89 636 48480
Siemens AG	Fax:	+49 89 636 42284
ZFE IS INF2	E-mail:	taubner@ztivax.uucp
Otto-Hahn-Ring 6		
W – 8000 Munchen 83		

Reviewers

Jaco W de Bakker	Tel:	+31-20-592 4136/4058
CWI	Fax:	
Kruislaan 413	E-mail:	mieke@cwi.nl
NL – 1098 SJ Amsterdam		
Michel Bosco	Tel:	
EEC	Fax:	
	E-mail:	mbos@dg13.dg13.cec.be
Heinz Brix	Tel:	+49 89 636 44627
Siemens AG	Fax:	+49 89 636 45111/48322
ZFE IS SOF 11	E-mail:	brix@ztivax.uucp
Otto-Hahn Ring 6		
W – 8000 Muenchen 83		
Manfred Broy	Tel:	+49 89 2105 8161
Technische Universität München	Fax:	
Institut für Informatik	E-mail:	broy@informatik.tu-muenchen.dbp.de
Arcisstraße		
W – 8000 Munchen 2		

Local Participants

Claus Havelund	Tel:	+45 31396466
University of Copenhagen	Fax:	
Department of Comp. Science	E-mail:	havelund@diku.dk
Universitetsparken 1		
2100 København Ø		
Anna Ingolfsdottir	Tel:	+45 98158522
Aalborg University Center	Fax:	+45 98158129
Dept. of Math. and Camp. Science	E-mail:	annai@esd.auc.dk
Frederik Bajersvej 7		
DK – 9220 Aalborg		
Padmanabhan Krishnan	Tel:	+45 86127188
Aarhus University	Fax:	+45 86135725
Computer Science Department	E-mail:	paddy@daimi.aau.dk
Ny Munkegade		
DK – 8000 Arhus C		
Arne Skou	Tel:	+45 98158522
Aalborg University Center	Fax:	+45 98158129
Dept. of Math. and Comp. Science	E-mail:	ask@iesd.auc.dk
Frederik Bajersvej 7		
DK – 9220 Aalborg		
Joakim von Wright	Tel:	
Åbo Akademi	Fax:	
	E-mail:	jwright@