BRICS
Basic Research in Computer Science

BRICS RS-95-44 Courcelle & Walukiewicz: MSO Logic, Graphs and Unfoldings of Transition Systems

Monadic Second-Order Logic, Graphs and Unfoldings of Transition Systems

Bruno Courcelle
Igor Walukiewicz

BRICS Report Series **RS-95-44**

ISSN 0909-0878 **August 1995**

See back inner page for a list of recent publications in the BRICS Report Series. Copies may be obtained by contacting:

BRICS
Department of Computer Science
University of Aarhus
Ny Munkegade, building 540
DK - 8000 Aarhus C
Denmark

Telephone: +45 8942 3360
Telefax: +45 8942 3255
Internet: BRICS@brics.dk

BRICS publications are in general accessible through WWW and anonymous FTP:

`http://www.brics.dk/`
`ftp ftp.brics.dk (cd pub/BRICS)`

Monadic Second-Order Logic, Graph Coverings and Unfoldings of Transition Systems

Bruno Courcelle
LaBRI

Université Bordeaux I
351, Cours de la Libération,
F-33405 Talence Cedex, France
e-mail: courcell@labri.u-bordeaux.fr

Igor Walukiewicz
BRICS[1,2]
Department of Computer Science
University of Aarhus
Ny Munkegade
DK-8000 Aarhus C, Denmark
e-mail: igw@daimi.aau.dk

Abstract

We prove that every monadic second-order property of the unfolding of a transition system is a monadic second-order property of the system itself. We prove a similar result for certain graph coverings.

1 Introduction

A transition system can be seen as an abstract form of a program and the infinite tree obtained by unfolding (or unraveling) it, can be seen as its behavior. Since transition systems and their behaviors can be represented by logical structures, one can express their properties by logical formulas. We consider here monadic second-order logic as an appropriate logical language because it subsumes many other formalisms like μ-calculus or temporal logics (see Emerson and Jutla [6], Niwiński [8]) and it is decidable on many structures and in particular on infinite trees (by Rabin's Theorem, see Thomas [11]). It was conjectured in Courcelle [2] that for every monadic second-order property P of transition systems R defined by:

$$P(R) \Leftrightarrow Q(\mathsf{Un}(R))$$

[1] **B**asic **R**esearch **i**n **C**omputer **S**cience,
Centre of the Danish National Research Foundation.
[2] On leave from: Institute of Informatics, Warsaw University, Banacha 2,
02-097 Warsaw, POLAND

where $\mathsf{Un}(R)$ is the unfolding of R and Q is a monadic second-order property, is also monadic second-order (and is expressible by a formula constructible from that which defines Q, which is the same for all systems R).

This conjecture was proved in [2] for deterministic transition systems (possibly with infinitely many states) and we prove it here for the class of all systems.

This new proof is independent of that in [2] and uses a different technique, based on a notion of covering: a covering of a transition system (or more generally of a graph) G is a surjective homomorphism $h : G' \to G$ (where G' is another transition system or graph) the restriction of which to the "neighbourhood" of every state or vertex of G' is an isomorphism. We say that h is a k-covering if $h^{-1}(x)$ has cardinality $\leq k$ for each state or vertex x of G. For a transition system if we take as "neighbourhood" of a state the set of transitions outgoing from it, then there exists a universal covering which is precisely the unfolding. The main lemma says that every monadic second-order property of the universal covering of a transition system R is equivalent to a monadic second order property of a k-covering of R for some integer k depending only on the considered property (and not on R).

The notion of "neighbourhood" is a "parameter" of the notion of covering. In the case of graphs, we examine two possibilities for defining coverings. The first possibility consists of taking the set of edges incident to a vertex as its neighbourhood. Then the results concerning transition systems extend for this notion of covering but only for graphs of bounded degree: every monadic second-order property of the universal covering of a (finite or infinite) graph (relatively to this notion of neighbourhood) can be expressed as a monadic second order property of the graph.

A second possibility consists in taking as neighbourhood of a vertex the subgraph induced by the vertices at distance at most 1: there exists a corresponding notion of universal covering. However, we exhibit a finite graph G, the universal covering of which is the infinite grid. This shows that the result does not hold here because the monadic theory of the infinite grid is undecidable whereas that of G is decidable (because G is finite).

Finally we relate unfoldings of a transition systems with a construction by Shelah and Stupp, extended by Muchnik, about which we raise some questions that indicate possible developments of the present work.

This paper is organized as follows.

Section 1 deals with transition systems, their coverings and automata,

Section 2 deals with monadic second order logic,

Sections 3 and 4 present some technical lemmas,

Section 5 gives the main proof,

Section 6 discusses the Shelah-Stupp-Muchnik construction,

Section 7 concerns coverings of graphs,

Section 8 reviews some open questions.

2 Transition systems

Let $n, m \in \mathcal{N}$ and $m \geq 1$. A *transition system* of type (n, m) is a tuple $R = (G, x, P_1, \ldots, P_n, Q_1, \ldots, Q_m)$, where G is a directed graph, x is a vertex called the *root of* R from which all other vertices are accessible by a directed path, $P_1, \ldots, P_n$ are sets of vertices and $Q_1, \ldots, Q_m$ is a partition of the set of edges.

A vertex of G is called a *state* of R and an edge is called a *transition.* A transition in Q_i is said to be *of type* i.

In order to have uniform notations, we let:

S_R be the set of states of R,

T_R be its set of transitions,

$root_R$ be its root,

P_{iR} be the i-th set of states,

Q_{iR} be the i-th set of transitions,

$\mathsf{src}_R = \{(t, s) : t \in T_R, s \in S_R, s \text{ is the origin (or source) of } t\}$

$\mathsf{tgt}_R = \{(t, s) : t \in T_R, s \in S_R, s \text{ is the target of } t\}$

We shall also write $s = \mathsf{src}_R(t)$ (or $s = \mathsf{tgt}_R(t)$) if $(t, s) \in \mathsf{src}_R$ (or $(t, s) \in \mathsf{tgt}_R(t)$ respectively).

A *path* in R is a finite or infinite sequence of transitions $(t_1, t_2, \ldots)$ such that $root_R = \mathsf{src}_R(t_1)$ and for each i, $\mathsf{tgt}_R(t_i) = \mathsf{src}_R(t_{i+1})$. If it is finite, the target of the last transition is called the *end of the path.*

Let R and R' be two transition systems of type (n, m). We write $R \subseteq R'$ iff:

$$\begin{array}{rcl} S_R & \subseteq & S_{R'} \\ T_R & \subseteq & T_{R'} \\ root_R & = & root_{R'} \\ P_{iR} & = & P_{iR'} \cap S_R \\ Q_{iR} & = & Q_{iR'} \cap T_R \\ \mathsf{src}_R & = & \mathsf{src}_{R'} \cap (T_R \times S_R) \\ \mathsf{tgt}_R & = & \mathsf{tgt}_{R'} \cap (T_R \times S_R) \end{array}$$

A *homomorphism* $h : R \to R'$ is a mapping $S_R \cup T_R \to S_{R'} \cup T_{R'}$ such that:

$$\begin{array}{rcl} h(S_R) & \subseteq & S_{R'} \\ h(T_R) & \subseteq & T_{R'} \\ h(\mathsf{src}_R(t)) & = & \mathsf{src}_{R'}(h(t)) \;\; \text{for all } t \in T_R \\ h(\mathsf{tgt}_R(t)) & = & \mathsf{tgt}_{R'}(h(t)) \;\; \text{for all } t \in T_R \\ h(root_R) & = & root_{R'} \\ s \in P_{iR} & \textit{iff} & h(s) \in P_{iR'}, \;\; \text{for all } s \in S_R \text{ and } i = 1, \ldots, n \\ t \in Q_{iR} & \textit{iff} & h(t) \in Q_{iR'}, \;\; \text{for all } t \in T_R \text{ and } i = 1, \ldots, m \end{array}$$

A homomorphism $h : R \to R'$ is a *covering* (we shall also say that R *is a covering of* R'), if it is surjective and for every state $s \in S_R$, h is a bijection of $\mathsf{out}_R(s)$ onto $\mathsf{out}_{R'}(h(s))$. (We denote by $\mathsf{out}_R(s)$ the set of transitions t of R such that $\mathsf{src}_R(t) = s$.) It is a k-covering if each set $h^{-1}(s)$, where $s \in S_{R'}$, has at most k elements.

Fact 1 *If h is a homomorphism $R \to R'$, the image of every path of R is a path of R'. If furthermore, h is a covering, then every path in R' is an image by h of the unique path in R.*

We now define the *unfolding* $\mathsf{Un}(R)$ of a transition system R; this is a tree, and we shall consider it as the *behavior* of R.

We let N_R be the set of finite paths in R. We have in particular the empty path linking the root to itself. N_R is the set of nodes of $\mathsf{Un}(R)$.

If p and $p' \in N_R$, we define an edge $p \to p'$ (equivalently a transition) of type i iff p' extends p by exactly one transition of R of type i. We let Q_i^* denote the set of such transitions.

We let $h_R : N_R \to S_R$ associate with every finite path its end. We obtain a transition system $\mathsf{Un}(R)$ of type (n, m) by defining:

$$
\begin{array}{rcl}
S_{\mathsf{Un}(R)} & = & N_R \\
T_{\mathsf{Un}(R)} & = & Q_1^* \cup \ldots \cup Q_m^* \\
root_{\mathsf{Un}(R)} & = & \varepsilon \\
P_{i\mathsf{Un}(R)} & = & P_i^* = h_R^{-1}(P_{iR}) \\
Q_{i\mathsf{Un}(R)} & = & Q_i^*
\end{array}
$$

Fact 2 *$h_R : \mathsf{Un}(R) \to R$ is a covering*

Fact 3 *If $m : R \to R'$ is a covering then there exists a unique isomorphism $\bar{m} : \mathsf{Un}(R) \to \mathsf{Un}(R')$ such that $h_{R'} \circ \bar{m} = m \circ h_R$.*

Because of these properties, $\mathsf{Un}(R)$ will be called the *universal covering* of R.

A transition system of type (n, m) is *deterministic* if no two transitions with the same source belong to the same set Q_i. It is *complete deterministic* if in addition each state has exactly m outgoing transitions.

Fact 4 *Let R and R' be complete deterministic transition systems of the same type. There is at most one homomorphism $R \to R'$ and such a homomorphism is a covering. It exists iff there exists a mapping $h : S_R \to S_{R'}$ such that: (a) $h(root_R) = root_{R'}$, (b) for every transition $x \to x'$ of R there is in R' a transition $h(x) \to h(x')$ of the same type, (c) for every $x \in S_R$ and every i, $x \in P_{iR}$ iff $h(x) \in P_{iR'}$.*

2.1 Parity automata and transition systems

We denote by $\mathcal{T}$ the infinite complete binary tree. Its nodes are (as usual) defined as words from $\{1, 2\}^*$. It is a complete deterministic transition system of type $(0, 2)$. We denote by $\mathcal{T}_n$ the set of tuples of the form $(\mathcal{T}, P_1, \ldots, P_n)$, where $P_1, \ldots, P_n$ are sets of nodes of $\mathcal{T}$. These tuples can be considered as infinite complete binary trees the nodes of which are labeled by subsets of $\{1, \ldots, n\}$; they are complete deterministic transition systems of type $(n, 2)$.

A *parity-automaton* is a tuple $\mathcal{PA} = \langle S, \Sigma, s_0, \delta, \Omega \rangle$ where:

- S is a finite nonempty set of *states*,

- Σ is a finite set called *alphabet*, we will assume that it is the set of subsets of $\{1,\ldots,n\}$ for some $n \in \mathcal{N}$,

- $s_0 \in S$ is the *initial state*,

- $\delta \subseteq S \times \Sigma \times S \times S$ is a transition relation.

- $\Omega : S \to \mathcal{N}$ is a function defining acceptance condition.

A *run* of $\mathcal{PA}$ on a tree $\mathcal{B} \in \mathcal{T}_n$ is a function $r : \mathcal{T} \to S$ such that $r(root_{\mathcal{B}}) = s_0$ and for any node x of $\mathcal{T}$ (i.e. $x \in \{1,2\}^*$):

$$(r(x), \{i : P_{i\mathcal{B}}(x)\}, r(x1), r(x2)) \in \delta$$

here $x1$ and $x2$ denote nodes obtained from x by appending 1 and 2 respectively at the end of x.

For a given run r as above and a path P of $\mathcal{T}$ let us define by $\mathsf{Inf}(r(P))$ the set of states which appear infinitely often in the sequence $r(P)$. We say that run r is *accepting* if for every path P of $\mathcal{T}$, the number $\min\{\Omega(\mathsf{Inf}(r(P)))\}$ is even. We say that $\mathcal{PA}$ *accepts* $\mathcal{B}$ if there is an accepting run of $\mathcal{PA}$ on $\mathcal{B}$. The language *recognized* by $\mathcal{PA}$ is the set of trees accepted by $\mathcal{PA}$.

We will say that a run r is *regular* if for every two nodes x, y of $\mathcal{B}$:

> if $r(x) = r(y)$ and $\mathcal{B}/x$ is isomorphic to $\mathcal{B}/y$ (where $\mathcal{B}/x$ is the subtree of $\mathcal{B}$ issued from x) then $r(h(u)) = r(u)$ for every node u of $\mathcal{B}/x$, where h is the isomorphism: $\mathcal{B}/x \to \mathcal{B}/y$.

Lemma 5 *For every parity automaton $\mathcal{PA}$ and every tree $\mathcal{B}$ if $\mathcal{PA}$ accepts $\mathcal{B}$ then there is a regular accepting run of $\mathcal{PA}$ on $\mathcal{B}$.*

Proof

The lemma follows from the results about games with parity conditions considered in [7, 6]. It was shown there that such games have memoryless strategies. We will briefly recall this result here and show how it applies in our case.

Let n be a natural number and let Σ be the set of all the subsets of $\{1,\ldots,n\}$. A game over Σ is given by a bipartite directed graph $\mathcal{G}$ whose set of nodes is partitioned in two sets N_I and N_{II}. From any node of N_I there may be an arbitrary number of edges to nodes of N_{II} each edge is labeled by a letter from Σ. No restrictions are imposed on this edges, there may be several edges with the same label, edges with different labels may have the same source and target. From every node of N_{II} there is exactly one *left* edge and exactly one *right* edge. The graph has designated start node n_0 which belongs to N_I and is equipped with a function $\Omega : N_I \to \mathcal{N}$.

The game is played on an infinite labeled tree $\mathcal{B} \in \mathcal{T}_n$. The starting position of the game is the pair consisting of the root r of $\mathcal{B}$ and the start node n_0 of $\mathcal{G}$. The game proceeds in rounds. In a position (s, m) first player I chooses a node n of N_{II} reachable from m by an edge labeled by the set $\{i : P_i(s)\}$. Then player *II* chooses a direction left or right. The new position of the play consists of a node of $\mathcal{T}$ reachable from s in the chosen direction and a node of $\mathcal{G}$ reachable from n in this direction. From this new position a new round is started. The play may be finite or infinite. The play may end in a finite number of steps only because player I cannot make a move; in this case player *II* is the winner. If a play is infinite we get as the result an infinite sequence $n_0, n_1, \ldots$ of nodes from N_I. Player I is the winner iff this sequence is accepted by condition Ω, i.e., the least number in $\mathsf{Inf}(\Omega(n_0), \Omega(n_1), \ldots)$ is even.

A *strategy* for player I in such a game is a partial function F which assigns nodes from N_{II} to positions. It must be defined for the initial position. Moreover if $F(s, m)$ is defined for some position (s, m) then node $F(s, m)$ must be reachable from m by an edge labeled $\{i : P_i(s)\}$ and for every direction d and nodes t, n reachable in direction d from s and $F(s, m)$ respectively $F(t, n)$ must be defined. A strategy is *winning* iff it guarantees that player I wins the game if only she follows the strategy. A strategy is called *memoryless* iff whenever F is defined for two positions with the same second component, say (s, m) and (t, m), and T/s is isomorphic to T/t then $F(s, m) = F(t, m)$.

Strategies for player *II* are defined similarly. In [7, 6] the following theorem was proved.

Theorem 6 *The parity game described above is determined. If a player has a winning strategy in the game then she has a memoryless strategy.*

It is easy to see that every finite parity automaton $\mathcal{PA}$ can be transformed into a graph of the game by taking N_I to be the set of states of $\mathcal{PA}$ and N_{II} to be the set of its transitions. It is also easy to see that player I has a winning strategy in the game on a tree T iff $\mathcal{PA}$ accepts T. From the above theorem follows that whenever $\mathcal{PA}$ accepts T it has a regular accepting run on T. $\square$

Next we introduce a concept of *quasi-automaton*, it is both an extension and a restriction of the notion of parity automaton. It is an extension because quasi-automata may have infinitely many states. It is a restriction because in this automata moves to the left are independent from moves to the right (there are languages recognized by automata but not by automata with independent moves, see also Lemma 7 below).

A *quasi-automaton* is a pair $\mathcal{A} = (A, \Omega)$ where A is a (possibly infinite) transition system of type $(n, 2)$, for some n, and Ω is a function assigning a natural number from a finite set to every node of A. We require that the image of Ω is finite.

Let $\mathcal{A}$ be as above and let U be a complete deterministic transition system of type $(n, 2)$ (in particular U can be a tree in $\mathcal{T}_n$). A *run of* $\mathcal{A}$ on U is a homomorphism of transition systems $r : U \to \mathcal{A}$. For every infinite path P in U, we let $\mathsf{Inf}_\Omega(P)$ to be the set of natural numbers k such that $\{i : \Omega(r(P_i)) = k\}$ is infinite, where P_i denotes i-th element of P. We say that r is *successful* if for every infinite path P, $\min(\mathsf{Inf}_\Omega(P))$ is even. We say that U is *accepted* by $\mathcal{A}$ if $\mathcal{A}$ has a successful run on U.

We let $L(\mathcal{A})$ denote the set of trees accepted by $\mathcal{A}$ (hence $L(\mathcal{A}) \subseteq \mathcal{T}_n$). Note that we may have $n = 0$; in this case $L(\mathcal{A})$ is either empty or the singleton $\{\mathcal{T}\}$.

Let U be a complete deterministic transition system accepted by $\mathcal{A}$. Then $\mathsf{Un}(U) \in L(\mathcal{A})$. Consider a successful run r of $\mathcal{A}$ on U, it is a homomorphism $U \to A$ and $r \circ h_U : \mathsf{Un}(U) \to A$ is a successful run of $\mathcal{A}$ on $\mathsf{Un}(U)$.

The definition of quasi-automaton departs from the definition of parity automata in the following ways:

1. The transitions "towards the left successor" are independent from the transitions "towards the right successor": transitions are defined in terms of two binary relations on states and not in terms of a single ternary one.

2. The states "contain node labels": if in a run r on a tree, a node x with label $w = (w_1, \ldots, w_n) \in \{0, 1\}^n$ has value $r(x) = s$, then for each $i = 1, \ldots, n$ we have $P_i(s) \Leftrightarrow w_i = 1$; hence w is completely defined by s.

3. Quasi-automaton may have infinitely many states.

The following lemma shows that one can transform every parity automaton into a finite quasi-automaton having more than one starting state.

Lemma 7 *Let n be a natural number. Given a set S together with sets $Start, P_1, \ldots, P_n \subseteq S$, two relations $Q_1, Q_2 \subseteq S \times S$ and a function $\Omega : S \to \mathcal{N}$ with a finite image, we define for every $s \in Start$ the quasi-automaton*

$$\mathcal{A}_s = (\langle S, s, P_1, \ldots, P_n, Q_1, Q_2 \rangle, \Omega)$$

For every parity automaton $\mathcal{PA}$ over an alphabet $\Sigma = \mathcal{P}(\{1, \ldots, n\})$ there exists a finite set S and objects $Start, P_1, \ldots, P_n, Q_1, Q_2, \Omega$ as above such that $L(\mathcal{PA}) = \bigcup_{s \in Start} L(\mathcal{A}_s)$.

We say that a quasi-automaton $\mathcal{A} = (A, \Omega)$ is *complete deterministic* if A is so. We write $\mathcal{A} \subseteq \mathcal{A}'$ if $\mathcal{A} = (A, \Omega)$, $\mathcal{A}' = (A', \Omega')$, A and A' are of the same type, $A \subseteq A'$ and Ω' restricted to $\mathcal{A}$ is equal to Ω. Note that $L(\mathcal{A}) \subseteq L(\mathcal{A}')$ if $\mathcal{A} \subseteq \mathcal{A}'$.

We now give a technical tool. Let R be a finite or infinite transition system where each state has at least two outgoing transitions, one of type 1 (called *left* transition) and one of type 2 (*right* transition).

We make it into a complete deterministic transition system $Bin(R)$ where each state has exactly two outgoing transitions by inserting new states. Hence if a state s has $n \geq 3$ transitions towards $s_1, s_2, \ldots, s_n$, where we assume that transitions towards s_{n-1} and s_n are of different types, we insert new states $u_2, \ldots, u_{n-1}$. We delete transitions $s \to s_i$ for $i = 2, \ldots, n$ and we add transitions $s \to u_2$, $u_i \to s_i$ for $i = 2, \ldots, n-1$, $u_i \to u_{i+1}$ for $i = 2, \ldots, n-2$ and, $u_{n-1} \to s_n$. A new transition to s_i has the same type as the corresponding transition $s \to s_i$. The types of the other added transitions are determined by this choice. If s has infinitely many transitions towards $s_1, s_2, \ldots, s_n, \ldots$ we add similarly infinitely many new states $u_2, u_3, \ldots, u_n, \ldots$ and transitions $s \to u_2, u_i \to s_i, u_i \to u_{i+1}$. (Although $Bin(R)$ is not unique because there is no unique linear ordering on transitions of R, we denote it functionally)

For each state s of R let $New(s)$ be the set of new states inserted to make s binary (that is $u_2, \ldots, u_{n-1}$ from the description above). We denote $\bigcup\{New(s) : s \in S_R\}$ by $New(S_R)$.

Let $\mathcal{A}$ be a quasi-automaton $\mathcal{A} = \langle R, \Omega \rangle$. It follows that $\mathsf{Un}(Bin(R))$ is a binary tree with nodes being sequences of elements from $S_R \cup New(S_R)$. This tree contains in some sense all possible runs of $\mathcal{A}$ on binary trees (see Claim 8). We let $\mathsf{Un}^{\Omega}(Bin(R))$ to be the tree obtained from $\mathsf{Un}(Bin(R))$ by labeling each node p by $*$ if p ends in a new state and by $\Omega(s)$ if p ends in a state $s \notin New(S_R)$.

We shall now describe a finite parity automaton that "extracts" from $Un^{\Omega}(Bin(R))$ the trees of $L(\mathcal{A})$. Without loss of generality we assume that $\Omega : S_R \to \{2,3,\ldots,2N\} = I$ for some $N \in \mathcal{N}$. We now construct an automaton B_Ω and a mapping $\bar{\Omega}$ from states of B_Ω to $\{1,2,\ldots,2N+2\}$ as follows:

The states of B_Ω are:

- $\perp$ and we let $\bar{\Omega}(\perp) = 1$,
- i for every $i \in I$ and we let $\bar{\Omega}(i) = i$,
- n_{lr}, n_l, n_r and $\bar{\Omega}$ assigns $2N+1$ to each of them,
- $\top$ and we let $\bar{\Omega}(\top) = 2N+2$.

We now describe the transitions of B_Ω. Intuitively this automaton should accept nothing from state $\perp$ and should accept everything from $\top$. Visiting some node not in $New(S_R)$ and being in a state $i \in I$ the automaton looks for left and right successors of the node skipping through new nodes. States n_{lr}, n_l, n_r are used for this. In state n_{lr} automaton goes through new nodes looking for both right and left successor. When it chooses, say, right successor it takes some appropriate state $j \in I$ to the right and n_l to the left. In state n_l the automaton looks only for right successor.

Formally the transitions of B_Ω are given by 4-tuples listed in the following table (a denotes any letter; i, j, j' stand for elements of I):

state	letter	state$_1$	state$_2$		state	letter	state$_1$	state$_2$
$\perp$	a	$\perp$	$\perp$		$\top$	a	$\top$	$\top$
i	$a \neq i$	$\perp$	$\perp$		n_{lr}	i	$\perp$	$\perp$
i	i	$\top$	n_{lr}		n_{lr}	$*$	n_{lr}	$\top$
i	i	n_{lr}	$\top$		n_{lr}	$*$	$\top$	n_{lr}
i	i	n_l	j		n_{lr}	$*$	n_l	j
i	i	j	n_r		n_{lr}	$*$	j	n_r
i	i	j	j'		n_{lr}	$*$	j	j'
n_l	i	$\perp$	$\perp$		n_r	i	$\perp$	$\perp$
n_l	$*$	$\top$	n_l		n_r	$*$	$\top$	n_r
n_l	$*$	n_l	$\top$		n_r	$*$	n_r	$\top$
n_l	$*$	$\top$	j		n_r	$*$	j	$\top$

The starting state of B_Ω is $\Omega(r)$ where r is the root of R.

We define as follows a *tree reduction* θ taking as an input $T = \mathsf{Un}^{\Omega}(Bin(R))$ together with an accepting run r of B_{Ω} on T and producing the following tree $\theta(T, r) = T'$:

- $Nodes(T') = \{x \in Nodes(T) : r(x) \in I\}$,
- $root_{T'} = root_T$,
- $x \xrightarrow{i} z$ is an edge of type $i \in \{1, 2\}$ in T' iff there is a path in T of the form
$$x \to y_1 \to y_2 \to \cdots \to y_k \to z$$
where $r(x) \in I$, $r(z) \in I$, $r(y_1), \ldots, r(y_k) \in \{n_{lr}, n_l, n_r\}$, $y_k \to z$ is of type i (if $k = 0$ one takes the condition that the transition $x \to z$ is of type i).

The following claim explains the dependence between automata (R, Ω) and B_{Ω}.

Claim 8 *Every accepting run r of B_{Ω} on $T = \mathsf{Un}^{\Omega}(Bin(R))$ can be transformed into an accepting run of (R, Ω) on $\theta(T, r)$. Conversely every accepting run of (R, Ω) on some tree can be transformed into an accepting run of B_{Ω} on $\mathsf{Un}^{\Omega}(Bin(R))$.*

Proof
Let r be an accepting run of B_{Ω} on $T = \mathsf{Un}^{\Omega}(Bin(R))$. Let σ be the mapping $Nodes(T) \to S_R \cup New(S_R)$ assigning to every node of T, which is a sequence of nodes from $S_R \cup New(S_R)$, the last state of the sequence. Then the restriction of σ to $Nodes(\theta(T, r))$ (which is a subset of $Nodes(T)$) is an accepting run of (R, Ω) on $\theta(T, r)$.

The proof of the other part of the claim is similar. $\square$

Lemma 9 *Let $\mathcal{A}$ be a (possibly infinite) quasi-automaton. If $L(\mathcal{A}) \neq \emptyset$ then there exists a complete deterministic quasi-automaton $\mathcal{A}' \subseteq \mathcal{A}$ such that $L(\mathcal{A}') \neq \emptyset$.*

Proof
Let $\mathcal{A} = (R, \Omega)$. If a state of R has no left transition or no right transition then we can delete it because it cannot appear in a run accepting a binary tree. Hence we can assume that all the states have both left and right transitions. So there exists a system $Bin(R)$.

Since $L(\mathcal{A}) \neq \emptyset$ there exists a run of B_Ω on $T = \mathsf{Un}^\Omega(Bin(R))$ and even a regular run by Lemma 5. Let us denote it by r.

Let σ be as in the proof of Claim 8. Let T' be the complete binary tree $\theta(T, r)$ and σ' be the restriction of σ to its nodes. Note that σ' takes values in S_R. It follows that σ' is an accepting run of (R, Ω) on T'.

Let x, y be two nodes of T such that $\sigma(x) = \sigma(y) \in S_R$ and $r(x), r(y) \in I$. This implies that $r(x) = r(y) = \Omega(\sigma(x)) = \Omega(\sigma(y))$. The subtrees of $\mathsf{Un}_\Omega(Bin(R))$ issued from x and y are isomorphic (by the definition of Un_Ω and since $Bin(R)$ is complete deterministic) and since r is a regular run, it is identical (up to isomorphism) on these subtrees. It follows that the subtrees of T issued from x and y are isomorphic and that σ' is identical on them (via the isomorphism). Hence T can be "folded" into a complete deterministic transition system $R' \subseteq R$, such that $T = \mathsf{Un}(R')$. More precisely, any two nodes x and y with isomorphic corresponding subtrees are made identical. The mapping σ' defines an accepting run of (R', Ω) on T. $\square$

3 Monadic second-order logic

We denote by $STR(R)$ the set of finite or countable structures of type R. Any two isomorphic structures are considered as equal.

In order to express properties of transition systems by monadic second-order (MS in short) formulas, we represent a transition system R of type (n, m) by the relational structure:

$$|R|_2 = \langle S_R \cup T_R, \mathsf{rt}_R, \mathsf{src}_R, \mathsf{tgt}_R, P_{1R}, \ldots, P_{nR}, Q_{1R}, \ldots, Q_{mR} \rangle$$

where $\mathsf{rt}_R = \{root_R\}$. It is clear that R is completely defined (up to isomorphism) by $|R|_2$.

We let $\mathcal{L}_2(n, m)$ be the set of MS formulas written with the relation symbols $\mathsf{rt}, \mathsf{src}, \mathsf{tgt}, Q_1, \ldots, Q_m$ (and of course $=$ and $\in$) and with free variables in $\{X_1, \ldots, X_n\}$.

We define $|R|_2 \models \alpha$ where $\alpha \in \mathcal{L}_2(n, m)$ by taking $P_{1R}, \ldots, P_{nR}$ as respective values of $X_1, \ldots, X_n$.

The properties of the behavior $\mathsf{Un}(R)$ of a system R as above can be expressed in a similar way by formulas of $\mathcal{L}_2(n, m)$ (since $\mathsf{Un}(R)$ is a transition system of type (n, m)). However, we shall use the following simpler representation: For a transition system V of type (n, m) we let

$$|V|_1 = \langle S_V, \mathsf{rt}_V, \mathsf{suc}_{1V}, \ldots, \mathsf{suc}_{mV}, P_{1V}, \ldots, P_{nV} \rangle$$

where $(x, y) \in \mathsf{suc}_{iV}$ iff there is in Q_{iV} a transition from x to y.

We let $\mathcal{L}_1(n, m)$ denote the set of MS formulas written with the symbols $\mathsf{rt}, \mathsf{suc}_1, \ldots, \mathsf{suc}_m$ (in addition to $=$ and $\in$) and having their free variables in $\{X_1, \ldots, X_n\}$. Again, we define $|V|_1 \models \alpha$ for $\alpha \in \mathcal{L}_1(n, m)$ by taking $P_{1V}, \ldots, P_{nV}$ as values of $X_1, \ldots, X_n$ respectively. By the results of Courcelle [5], the same properties of trees can be represented by formulas of $\mathcal{L}_2$ and $\mathcal{L}_1$.

Our objective is to prove the following theorem.

Theorem 10 *Let $n, m \in \mathcal{N}$, $m \geq 1$. For every formula $\varphi \in \mathcal{L}_1(n, m)$ one can construct a formula $\psi \in \mathcal{L}_2(n, m)$ such that, for every transition system R of type (n, m):*

$$|R|_2 \models \psi \Leftrightarrow |\mathsf{Un}(R)|_1 \models \varphi$$

We shall need the notion of an MS-definable transduction of relational structures that we now recall from [4].

Let $\mathcal{R}$ and $\mathcal{Q}$ be two finite ranked sets of relation symbols. Let $\mathcal{W}$ be a finite set of set variables, called here the set of *parameters*. (It is not a loss of generality to assume that all parameters are set variables.) A $(\mathcal{Q}, \mathcal{R})$-*definition scheme* is a tuple of formulas of the form :

$$\Delta = (\varphi, \psi_1, \cdots, \psi_k, (\theta_w)_{w \in Q*k})$$

where

$k > 0$, $\mathcal{R}^*k = \{(q, \vec{\jmath}) |\ q \in \mathcal{Q},\ \vec{\jmath} \in [k]^{\rho(q)}\}$

$\varphi \in MS(\mathcal{R}, \mathcal{W})$,

$\psi_i \in MS(\mathcal{R}, \mathcal{W} \cup \{x_1\})$ for $i = 1, \cdots, k$,

$\theta_w \in MS(\mathcal{R}, \mathcal{W} \cup \{x_1, \cdots, x_{\rho(q)}\})$, for $w = (q, \vec{\jmath}) \in \mathcal{Q}^*k$.

These formulas are intended to define a structure T in $STR(\mathcal{Q})$ from a structure S in $STR(\mathcal{R})$ and will be used in the following way. The formula φ defines the domain of the corresponding transduction; namely, T is defined only if φ holds true in S. Assuming this condition fulfilled, the formulas $\psi_1, \ldots, \psi_k$ define the domain of T as the disjoint union of the sets $D_1, \cdots, D_k$, where D_i is the set of elements in the domain of S that satisfy ψ_i. Finally, the formulas θ_w for $w = (q, \vec{\jmath}), \vec{\jmath} \in [k]^{\rho(q)}$ define the relation q_T. Here are the formal definitions.

Let $S \in STR(\mathcal{R})$, let μ be a $\mathcal{W}$-assignment in S. A $\mathcal{Q}$-structure T with domain $D_T \subseteq D_S \times [k]$ is *defined in* (S, μ) by Δ if :

(i) $(S,\mu) \models \varphi$

(ii) $D_T = \{(d,i) \mid d \in D_S, i \in [k], (S,\mu,d) \models \psi_i\}$

(iii) for each q in $\mathcal{Q}$:

$q_T = \{((d_1,i_1),\cdots,(d_t,i_t)) \in D_T^t \mid (S,\mu,d_1,\cdots,d_t) \models \theta_{(q,\vec{\jmath})}\}$,

where $\vec{\jmath} = (i_1,\cdots,i_t)$ and $t = \rho(q)$.

(By $(S,\mu,d_1,\cdots,d_t) \models \theta_{(q,\vec{\jmath})}$, we mean $(S,\mu') \models \theta_{(q,\vec{\jmath})}$, where μ' is the assignment extending μ, such that $\mu'(x_i) = d_i$ for all $i = 1,\cdots,t$; a similar convention is used for $(S,\mu,d) \models \psi_i$.)

Since T is associated in a unique way with S,μ and Δ whenever it is defined, i.e., whenever $(S,\mu) \models \varphi$, we can use the functional notation $def_\Delta(S,\mu)$ for T.

The *transduction defined by* Δ is the relation $def_\Delta := \{(S,T) \mid T = def_\Delta(S,\mu)$ for some $\mathcal{W}$-assignment μ in $S\} \subseteq STR(\mathcal{R}) \times STR(\mathcal{Q})$. A transduction $f \subseteq STR(\mathcal{R}) \times STR(\mathcal{Q})$ is *MS-definable* if it is equal to def_Δ for some $(\mathcal{Q},\mathcal{R})$-definition scheme Δ. In the case where $\mathcal{W} = \emptyset$, we say that f is *MS-definable without parameters* (note that it is functional). We shall refer to the integer k by saying that def_Δ is k-copying ; if $k = 1$ we say that it is *non copying* and we can write more simply Δ as $(\varphi,\psi,(\theta_q)_{q\in\mathcal{Q}})$. In this case:

$D_T = \{d \in D_S \mid (S,\mu,d) \models \psi\}$

and for each q in $\mathcal{Q}$

$q_T = \{(d_1,\cdots d_t) \in D_T^t \mid (S,\mu,d_1,\cdots d_t) \models \theta_q\}$, where $t = \rho(q)$.

We give an example: the product of a finite-state automaton $\mathcal{A}$ by a *fixed* finite-state automaton $\mathcal{B}$. A finite-state automaton is defined as a 5-tuple $\mathcal{A} = < A,Q,M,I,F >$ where A is the input alphabet, (here we shall take $A = \{a,b\}$), Q is the set of states, M is the transition relation which is here a subset of $Q \times A \times Q$ (because we consider nondeterministic automata without ε-transitions), I is the set of initial states and F is that of final states. The language it recognizes is denoted by $L(\mathcal{A})$. The automaton $\mathcal{A}$ is represented by the relational structure : $\mid \mathcal{A} \mid = < Q, trans_a, trans_b, I, F >$ where $trans_a$ and $trans_b$ are binary relations and :

$trans_a(p,q)$ holds if and only if $(p,a,q) \in M$,

$trans_b(p, q)$ holds if and only if $(p, b, q) \in M$.

Let $\mathcal{B} = < A', Q', M', I', F' >$ be a similar automaton, and $\mathcal{A} \times \mathcal{B} = < A, Q \times Q', M", I \times I', F \times F' >$ be the product automaton intended to define the language $L(\mathcal{A}) \cap L(\mathcal{B})$. We let Q' be $\{1, \cdots, k\}$ (let us recall that $\mathcal{B}$ is fixed). We let Δ be the k-copying definition scheme $(\varphi, \psi_1, \cdots . \psi_k, (\theta_w)_{w \in \mathcal{R}^* k})$, where $\mathcal{R} = \{trans_a, trans_b, I, F\}$ and :

φ is the constant *true* (because every structure in $STR(\mathcal{R})$ represents an automaton which may have inaccessible states and useless transitions),

$\psi_1, \cdots, \psi_k$ are the constant *true*,

$\theta_{(trans_a, i, j)}(x_1, x_2)$ is the formula $trans_a(x_1, x_2)$ if (i, a, j) is a transition of $\mathcal{B}$ and is the constant *false* otherwise,

$\theta_{(trans_b, i, j)}$ is defined similarly,

$\theta_{(I, i)}(x_1)$ is the formula $I(x_1)$ if i is an initial state of $\mathcal{B}$ and is *false* otherwise,

$\theta_{(F, i)}(x_1)$ is defined similarly.

It is not hard to check that $\mid \mathcal{A} \times \mathcal{B} \mid = def_\Delta(\mid \mathcal{A} \mid)$. Note that the language defined by an automaton $\mathcal{A}$ is nonempty if and only if there is a path in $\mathcal{A}$ from some initial state to some final state. This later property is expressible in monadic second-order logic. Hence it follows from Proposition 12 below that, for a fixed rational language K, the set of structures representing an automata $\mathcal{A}$ such that $L(\mathcal{A}) \cap K$ is nonempty is definable. This construction is used systematically in Courcelle [2].

Fact 11 *The domain of an MS-definable transduction is MS-definable.*

Proof: Δ be a definition scheme as in the general definition with $\mathcal{W} = \{X_1, \cdots, X_n\}$. Then $Dom(def_\Delta) = \{S \mid S \models \exists X_1, \cdots, X_n . \varphi\}$. $\square$

The following proposition says that if $S = def_\Delta(T, \mu)$, i.e., if S is defined in (T, μ) by Δ, then the monadic second-order properties of S can be expressed as monadic second-order properties of (T, μ). The usefulness of MS-definable transductions is based on this proposition.

Let $\Delta = (\varphi, \psi_1, \cdots, \psi_k, (\theta_w)_{w \in \mathcal{Q}^* k})$ be a $(\mathcal{Q}, \mathcal{R})$-definition scheme, written with a set of parameters $\mathcal{W}$. Let $\mathcal{V}$ be a set of set variables disjoint from $\mathcal{W}$. For every variable X in $\mathcal{V}$, for every $i = 1, \cdots, k$, we let X_i

be a new variable. We let $\mathcal{V} := \{X_i/X \in \mathcal{V}, i = 1, \cdots, k\}$. For every mapping $\eta : \mathcal{V}' \to \mathcal{P}(D)$, we let $\eta \uparrow k : \mathcal{V} \to \mathcal{P}(D \times [k])$ be defined by $\eta \uparrow k(X) = \eta(X_1) \times \{1\} \cup \cdots \cup \eta(X_k) \times \{k\}$. With these notations we can state :

Proposition 12 *For every formula β in $MS(\mathcal{Q}, \mathcal{V})$ one can construct a formula β' in $MS(\mathcal{R}, \mathcal{V}' \cup \mathcal{W})$ such that, for every T in $STR(\mathcal{R})$, for every assignment $\mu : \mathcal{W} \to T$ for every assignment $\eta : \mathcal{V} \to T$, we have:*

> *$def_\Delta(T, \mu)$ is defined (if it is, we denote it by S), $\eta \uparrow k$ is a $\mathcal{V}$-assignment in S, and $(S, \eta \uparrow k) \models \beta$*
>
> *if and only if*
>
> *$(T, \eta \cup \mu) \models \beta'$.*

Note that, even if S is well-defined, the mapping $\eta \uparrow k$ is not necessarily a $\mathcal{V}$-assignment in S, because $\eta \uparrow k(X)$ is not necessarily a subset of the domain of S which is a possibly proper subset of $D \times [k]$.

From this proposition, we get easily :

Proposition 13

1. *The inverse image of an MS-definable class of structures under an MS-definable transduction is MS-definable.*
2. *The composition of two MS-definable transductions is MS-definable.*

Proposition 14 *Let $k, m \geq 1$, let $n \geq 0$. There exists an MS-definable transduction associating with every transition system R of type (n, m) the set of its k-coverings (where a system R is represented by a structure $|R|_2$).*

Proof

Let R be a transition system of type (n, m) and $h : R' \to R$ be a k-covering.

By choosing an arbitrary linear ordering of each set $h^{-1}(x), x \in S_R$, we can assume that $S_{R'} \subseteq S_R \times [k]$ and $h(x, i) = x$ for every i such that $(x, i) \in S_{R'}$. We can assume that $root_{R'} = (root_R, 1)$.

For each $i \in [k]$, we let $Y_i = \{x \in S_R : (x, i) \in S_{R'}\}$. For $i, j \in [k]$, we let

$$Z_{i,j} = \{t \in T_r : h(t') = t \text{ for some } t' \in T_{R'} \text{ with source } (\mathsf{src}_R(t), i) \text{ and target } (\mathsf{tgt}_R(t), j)\}$$

Since h is a bijection of $out_{R'}(x)$ onto $out_R(h(x))$ for every $x \in S_{R'}$ it follows that for every $t \in Z_{i,j}$, there is a unique $t' \in T_{R'}$, with source $(\mathsf{src}_R(t), i)$ and target $(\mathsf{tgt}_R(t), j)$ such that $h(t') = t$. We shall identify t' with the triple (t, i, j).

Hence

$$S_{R'} = \bigcup\{Y_i \times \{i\} : 1 \leq i \leq k\} \tag{1}$$
$$T_{R'} = \bigcup\{Z_{i,j} \times \{(i,j)\} : i,j \in [k]\} \tag{2}$$

This gives a description of $|R'|$ as the output of a definable transduction taking as input $|R|_2$ and the parameters $Y_1, \ldots, Y_k, Z_{1,1}, \ldots, Z_{k,k}$.

Specifically we have

$$\mathsf{rt}_{R'} = \{(x,1)\} \text{ where } x \text{ is the unique state in } \mathsf{rt}_R \tag{3}$$
$$\mathsf{src}_{R'} = \{((t,i,j),(x,i)) : i,j \in [k], t \in Z_{i,j}, (t,x) \in \mathsf{src}_R\} \tag{4}$$
$$\mathsf{tgt}_{R'} = \{((t,i,j),(x,j)) : i,j \in [k], t \in Z_{i,j}, (t,x) \in \mathsf{tgt}_R\} \tag{5}$$
$$P_{iR'} = \{(x,j) : x \in P_{iR} \cap Y_j, j \in [k]\}, \quad i = 1, \ldots, n \tag{6}$$
$$Q_{iR'} = \{(t,j,j') : x \in Q_{iR} \cap Z_{j,j'}, j,j' \in [k]\}, \quad i = 1, \ldots, m \tag{7}$$

In this construction, we have assumed that the parameters $Y_1, \ldots, Y_k, Z_{1,1}, \ldots, Z_{k,k}$ are defined from a k-covering R' of R. In order to ensure that the constructed transduction *only defines* k-coverings of the input transduction systems we must find a formula $\varphi(Y_1, \ldots, Y_k, Z_{1,1}, \ldots, Z_{k,k})$ that verifies that the structure defined by (1)–(7) is actually of the form $|R'|_2$ for some k-covering R' of R.

We consider the following conditions:

$$S_R = \bigcup\{Y_i : 1 \leq i \leq k\} \tag{8}$$
$$T_R = \bigcup\{Z_{i,j} : i,j \in [k]\} \tag{9}$$

For every $i \in [k]$, every $x \in Y_i$, every transition $t \in \mathsf{out}_R(x)$ there is one and only one $j \in [k]$ such that $t \in Z_{i,j}$ (10)

Every state of R' is accessible by a path from $root_{R'}$. (11)

Conditions (8)–(11) can be written as an MS-formula in parameters $Y_1, \ldots, Y_k, Z_{1,1}, \ldots, Z_{k,k}$ to be evaluated in $|R|_2$. Let us review them: (8)–(9) state that the mapping $h : S_{R'} \cup T_{R'} \to S_R \cup T_R$ defined by

$$\begin{array}{ll} h((x,i)) = x & \text{if } (x,i) \in S_{R'} \text{ and} \\ h((t,(i,j))) = t & \text{if } (t,(i,j)) \in T_{R'} \end{array}$$

is surjective. From its definition it is a homomorphism. Condition 10 states that it is a covering. Condition 11 states that R' is indeed a transition system.

Hence $\varphi(Y_1, \ldots, Y_k, Z_{1,1}, \ldots, Z_{k,k})$ is the desired formula which completes the proof. $\square$

Here is the last definition. Let $\mathcal{S}$ and $\mathcal{S}'$ be two classes of structures with $\mathcal{S} \subseteq STR(R)$ and $\mathcal{S}' \subseteq STR(R')$, and let f be a transduction $\mathcal{S} \to \mathcal{S}'$. We say that f is *MS-compatible* if there exists an algorithm that associates with every MS-formula φ over R' an MS-formula ψ over R such that, for every structure $S \in \mathcal{S}$:

$$S \models \psi \text{ iff } S' \models \varphi \text{ for some } S' \in f(S)$$

It follows from Proposition 12 that every MS-definable transduction is MS-compatible.

Our main result (Theorem 10) says that the transduction $|R|_2 \mapsto |\mathsf{Un}(R)|_1$ is MS-compatible for R ranging over finite and infinite transition systems of type (n, m).

4 A regularization lemma

If R is a transition system of type (n, m) and $Y \subseteq S_R$, we denote by $R * Y$ the system of type $(n+1, m)$ consisting of R augmented with Y as $(n+1)$-st set of states.

The following lemma is a crucial step for the main theorem.

Lemma 15 *Let $n \geq 0$ and $\alpha \in \mathcal{L}_1(n+1, 2)$. One can find an integer k such that, for every (possibly infinite) complete deterministic transition system R of type $(n, 2)$, if $|\mathsf{Un}(R)|_1 \models \exists X_{n+1}.\alpha$, then there exists a k-covering R' of R and a subset Y of $S_{R'}$ such that $|\mathsf{Un}(R' * Y)|_1 \models \alpha$.*

Proof
We let $\mathcal{PA}$ be a parity automaton such that $L(\mathcal{PA}) = \{U \in \mathcal{T}_{n+1} : |U|_1 \models \alpha\}$. By Lemma 7 there exists a finite set S_A and sets $Start, P_{1A}, \ldots, P_{nA} \subseteq S$, two relations $Q_{1A}, Q_{2A} \subseteq S_A \times S_A$ and a function $\Omega : S_A \to \mathcal{N}$ such that $L(\mathcal{PA}) = \bigcup_{s \in Start} L(\mathcal{A}_s)$.

Let Z be a set of nodes of $\mathsf{Un}(R)$ that satisfies α when taken as a value of X_{n+1}. Hence

$$|\mathsf{Un}(R) * Z|_1 \models \alpha \tag{12}$$

Note that $\mathsf{Un}(R) \in \mathcal{T}_n$ and $\mathsf{Un}(R) * Z \in \mathcal{T}_{n+1}$ and by 12, $\mathsf{Un}(R) * Z \in L(\mathcal{PA})$.

Let $r : \mathsf{Un}(R) * Z \to A_s$ be an accepting run of the quasi-automaton $\mathcal{A}_s$ for some $s \in \mathit{Start}$. For every node w of $\mathsf{Un}(R)$ we let

$$\bar{r}(w) = (r(w), h_R(w)) \in S_A \times S_R \tag{13}$$

where h_R is the universal covering $\mathsf{Un}(R) \to R$.

We shall consider $\bar{r}$ as an accepting run of a quasi-automaton $\mathcal{B} = (B, \bar{\Omega})$ that we now construct. We first construct a transition system B.

We let $S_B \subseteq S_A \times S_R$ be the set of pairs (x, y) such that

$$x \in P_{iA} \Leftrightarrow y \in P_{iR} \quad \text{for every } i = 1, \ldots, n \tag{14}$$

We let T_B to be a set of transitions: $(x, y) \to (x', y')$ of type i, $(i = 1, 2)$ such that: $(x, y), (x', y') \in S_B$, $x \to x'$ and $y \to y'$ are transitions of S_A and S_R respectively, both of type i.

We take $(\mathit{root}_A, \mathit{root}_R)$ as a root of B. We let also P_{iB} be defined as follows:

$$x \in P_{iB} \Leftrightarrow x \in P_{iA} \tag{15}$$

for each $i = 1, \ldots, n+1$. We have thus "almost" a transition system of type $(n+1, 2)$: almost because it may be the case that some states of S_B are not accessible. We obtain an actual transition system by restricting S_B to the accessible states and T_B to the transitions having an accessible source. Hence B is now a transition system and $\bar{r}$ is a homomorphism: $\mathsf{Un}(R) * Z \to B$. We make B into a quasi-automaton $\mathcal{B} = (B, \bar{\Omega})$ by defining $\bar{\Omega}((x, y)) = \Omega(x)$.

Claim 16 *$\bar{r}$ is an accepting run of $\mathcal{B} = (B, \bar{\Omega})$.*

Proof: Since $\bar{r}$ is a homomorphism: $\mathsf{Un}(R) * Z \to B$, it is a run of $\mathcal{B}$. It is easy to see that it is accepting. $\square$

By Lemma 9 there exists a complete deterministic quasi-automaton $\mathcal{B}' \subseteq \mathcal{B}$ and an accepting run r' of $\mathcal{B}'$ on some tree $W' \in \mathcal{T}_{n+1}$.

We let B' be the transition system of $\mathcal{B}'$ (of type $(n+1, 2)$) and R' be the transition system of type $(n, 2)$ obtained from $\mathcal{B}'$ by deleting the $(n+1)$-st set of states, $P_{n+1B'}$, that we shall take as the desired set Y.

We have thus $B' = R' * Y$; R' and B' are complete deterministic. We let also $k = \mathit{Card}(S_A)$.

Claim 17 *R' is a k-covering of R*

Proof: Since R' and R are complete deterministic we need only define the desired covering as a mapping of $S_{R'}$ onto S_R. We define it as the projection π_2 that maps $(x, y) \in S_{R'} \subseteq S_A \times S_R$ onto y. We have $\pi_2(root_{R'}) = root_R$ since $root_{R'} = (root_A, root_R)$ and π_2 is a homomorphism from the definitions. The remaining follows from Fact 4 $\square$

Claim 18 $|\mathsf{Un}(B')|_1 \models \alpha$

Proof: The mapping $\pi_1 : S_{B'} \rightarrow S_A$ defined by $\pi_1(x, y) = x$ is a homomorphism of transition systems and even an accepting run of $\mathcal{A}$. It follows that $\mathsf{Un}(B') \in L(\mathcal{A})$ hence that $|\mathsf{Un}(B')|_1 \models \alpha$. $\square$

Since $B' = R' * Y$ we have thus obtained the desired integer k and the proof is complete. $\square$

We consider Lemma 15 as a regularization lemma because it says that if $|\mathsf{Un}(R)|_1$ contains a set Z that satisfies α it contains another one having a special "regular" form, defined from the unfolding of a k-covering of R.

Our next aim is to extend Proposition 15 to transition systems R that are not deterministic. If R is a transition system of type $(n, 1)$, then the nodes of the tree $\mathsf{Un}(R)$ have finite unordered sets of successors. Such trees will be represented by binary trees in way that we now describe.

5 Edge contractions and the proof of the main result

We first consider systems of type $(n, 1)$. We define a transformation that makes a tree $T \in \mathcal{T}_{n+1}$ into a tree $c(T)$ of type $(n, 1)$.

Let $T \in \mathcal{T}_{n+1}$ be defined by an $(n+1)$-tuple of subsets of $\{1, 2\}^*$, namely by $(P_{1T}, \ldots, P_{n+1T})$. We let $c(T)$ be the tree such that:

- $S_{c(T)} = (\{1, 2\}^* \setminus P_{1T}) \cup \{\varepsilon\}$
- $x \rightarrow y$ in $c(T)$ iff there is in T a path of the form $x \rightarrow z_1 \rightarrow z_2 \rightarrow \cdots \rightarrow z_p \rightarrow y$ with $p \geq 0$ and $z_1, z_2, \ldots, z_p \in P_{1T}$ ($x \rightarrow y$ is a shorthand for "there is a transition from x to y").
- $P_{i-1c(T)} = P_{iT} \cap S_{c(T)}$ for $i = 2, \ldots, n + 1$.

Our next aim is to define a similar operation on transition systems so that

$$\mathsf{Un}(c(R)) = c(\mathsf{Un}(R))$$

A *special transition system* is a system R of type $(n+1,2)$, for some n, such that

1. R is complete deterministic,
2. $root_R \notin P_{1R}$,
3. $P_{1R} \cap (P_{2R} \cup \ldots \cup P_{n+1R}) = \emptyset$,

We now define a transformation c that transforms any special transition system R of type $(n+1,2)$ into one of type $(n,1)$. We let $c(R)$ be such that

- $S_{c(R)} = S_R \setminus P_{1R}$,
- $P_{ic(R)} = P_{i+1R} \cap S_{c(R)}$ for $i = 2, \ldots, n$,
- $root_{c(R)} = root_R$,
- $x \to y$ is a transition of $c(R)$ iff we have a path in R of the form $x \to z_1 \to z_2 \to \cdots \to z_p \to y$ with $x, y \notin P_{1R}$, $z_1, z_2, \ldots, z_p \in P_{1R}$, $p \geq 0$.

Fact 19 *If R is special then we have $c(\mathsf{Un}(R)) = \mathsf{Un}(c(R))$*

Proof
Easy verification $\square$

Lemma 20 *For every transition system R of type $(n,1)$ one can construct a special transition system, $Bin(R)$ of type $(n+1,2)$ such that $c(Bin(R)) = R$*

Proof
We let R' be the transition system of type $(n+1,2)$ defined as follows:

1. we add a new "sink" state $\perp$ and two transitions $\perp \to \perp$ of type 1 and 2,
2. for each state $s \in S_R$ we do the following:

(a) if $\mathsf{out}_R(s) = \emptyset$ we add two transitions $s \to \perp$ of types 1 and 2,

(b) if $\mathsf{out}_R(s) = \{t\}$ we add a transition $s \to \perp$ of type 2 (note that the transition t is necessary of type 1).

(c) if $\mathsf{out}_R(s)$ consists of at least two transitions, we let one to be of type 1, and the other one of type 2; they will be transitions of R'.

We let $Bin'(R) = Bin(R')$ where Bin is defined on page 9.

3. We let $P_{1Bin'(R)}$, consist of all "new states" (the state $\perp$ and the states introduced in the construction of $Bin(R')$) and we let $P_{i+1Bin'(R)} = P_{iR}$ for every $i = 1, \ldots, n$.

□

Lemma 21 *If R is a special transition system and K is a k-covering of $Bin(R)$ then K is also special and $c(K)$ is a k-covering of R.*

Proof
We let $h : K \to Bin(R)$ be a k-covering. We first check that K is a special system. Condition 1 of the definition of a special system (saying that K is complete deterministic) holds because every covering of a complete deterministic system is complete deterministic. Conditions 2 and 3 hold easily.

It remains to prove that $c(K)$ is a k-covering of R. Let us consider $h : S_{c(K)} \to S_R$. It is the desired covering. This follows from the observations establishing that K is a special system. □

Proposition 22 *Let $n \geq 0$ and $\alpha \in \mathcal{L}_1(n+1, 1)$. One can find an integer k such that, for every transition system R of type $(n, 1)$, if $|\mathsf{Un}(R)|_1 \models \exists X_{n+1}.\alpha$ then there exists a k-covering R' of R and a subset Y of $S_{R'}$ such that $|\mathsf{Un}(R' * Y)|_1 \models \alpha$.*

Proof
We first construct a formula $\beta \in \mathcal{L}_1(n+2, 2)$ such that for every tree T in $\mathcal{T}_{n+2}$ we have

$$|T|_1 \models \beta \text{ iff } P_{1T} \cap (P_{2T} \cup \ldots \cup P_{n+1T}) = \emptyset \text{ and } |c(T)|_1 \models \alpha$$

This is possible because the mapping from $|T|_1$ to $|c(T)|_1$ is a definable transduction of structures.

We let k be the integer associated with β by Proposition 15. Let R be a transition system of type $(n,1)$ such that $|\mathsf{Un}(R)|_1 \models \exists X_{n+1}.\alpha$. For some set $Z \subseteq S_{\mathsf{Un}(R)}$ we have thus

$$|\mathsf{Un}(R) * Z|_1 \models \alpha$$

Since $\mathsf{Un}(R) = c(\mathsf{Un}(Bin'(R)))$ we have also $Z \subseteq S_{\mathsf{Un}(Bin'(R))}$ and $Z \cap P_{1\mathsf{Un}(Bin'(R))} = \emptyset$. Hence

$$|\mathsf{Un}(Bin'(R)) * Z|_1 \models \beta$$

By Proposition 15 we have some $Y \subseteq S_K$ such that

$$|\mathsf{Un}(K * Y)|_1 \models \beta$$

where K is some k-covering of $Bin'(R)$. It holds in particular that $P_{1K} \cap Y = \emptyset$. By Lemma 21 $c(K)$ is a k-covering of R and $Y \subseteq S_{c(K)}$.

Hence $c(K)$ is the desired system R' since

$$|c(\mathsf{Un}(K * Y))|_1 \models \alpha$$

and

$$c(\mathsf{Un}(K * Y)) = \mathsf{Un}(c(K * Y)) = \mathsf{Un}(c(K) * Y)$$

$\square$

Proof of Theorem 10

Let us first consider the case of the systems of type $(n,1)$. We want to show that for every formula $\varphi \in \mathcal{L}_1(n,1)$ one can construct a formula $\widehat{\varphi} \in \mathcal{L}_2(n,1)$ such that, for every transition system R of type $(n,1)$:

$$|R|_2 \models \widehat{\varphi} \Leftrightarrow |\mathsf{Un}(R)|_1 \models \varphi$$

The proof proceeds by induction on the structure of φ. We assume that φ is a closed formula. This is not a restriction as two formulas are equivalent iff closed formulas obtained by substituting unary relational symbols for free variables are equivalent.

If φ is closed atomic formula then $\widehat{\varphi} = \varphi$. The cases for conjunction and negation are obvious.

Assume $\varphi = \exists X.\alpha(X)$. By Proposition 22 there is an integer k such that for every transitions system of type $(n,1)$:

$|\mathsf{Un}(R)|_1 \models \exists X.\alpha(X)$ iff there exists a k-covering R' of R and a subset Y of $S_{R'}$ such that $|\mathsf{Un}(R' * Y)|_1 \models \alpha[P_{n+1}/X]$.

By induction assumption we have a formula $\widehat{\alpha}[P_{n+1}/X]$ such that for every transition system K of type $(n+1, 1)$:

$$|K|_2 \models \widehat{\alpha}[P_{n+1}/X] \quad \text{iff} \quad |\mathsf{Un}(K)|_1 \models \alpha[P_{n+1}/X]$$

It remains to show that the property:

there exist a k-covering R' of R such that $R' \models \exists X.\widehat{\alpha}(X)$

is MS-definable.

By Proposition 14 we know that the transduction associating with R the set of its k coverings is MS-definable. (This transduction has parameters $Y_1, \ldots, Y_k, Z_{1,1}, \ldots, Z_{k,k}$ each admissible choice of parameters gives us a k-covering). Proposition 12 gives us the desired formula $\widehat{\exists X.\alpha(X)}$.

We now prove the theorem for systems of the general type (n, m) with $m \geq 1$.

We define a transformation α making a transition system R of type (n, m) into a transition system $\alpha(R)$ of type $(n + m, 1)$ such that the transduction $|R|_2 \mapsto |\alpha(R)|_2$ is MS-definable, and a transformation β from transition systems of type $(n + m, 1)$ to transition systems of type (n, m) such that the transduction $|R|_1 \mapsto |\beta(R)|_1$ is MS-definable and

$$\mathsf{Un}(R) = \beta(\mathsf{Un}(\alpha(R))) \tag{16}$$

for every transition system of type (n, m). Clearly such transformations reduce the general case of the Theorem 10 to the case of systems of type $(n, 1)$ which we have just proved.

Definition of α Let R be a transition system of type (n, m) with $m \geq 2$.

The idea of the construction of $\alpha(R)$ is to replace a state x of R by m states $(x, 1), \ldots, (x, m)$ in R' and to replace a transition $y \to x$ of type i by m transitions from $(y, 1), \ldots, (y, m)$ to (x, i) all of type 1. (If there is no transition of type i from y to x then we need not put in $\alpha(R)$ the state (x, i)).

Here is the formal definition of $\alpha(R)$. Suppose

$$R = \langle S_R, T_R, \mathsf{src}_R, \mathsf{tgt}_R, root_R, P_{1R}, \ldots, P_{nR}, Q_{1R}, \ldots, Q_{mR} \rangle$$

Let us denote by $[m]$ the set $\{1, \ldots, m\}$). First we define system R' which is the 5-tuple

$$\langle S_{R'}, T_{R'}, \mathsf{src}_{R'}, \mathsf{tgt}_{R'}, root_{R'}, P_{1R'}, \ldots, P_{nR'}, P'_{1R'}, \ldots, P'_{mR'} \rangle$$

where

$$\begin{array}{rcll}
S_{R'} & = & S_R \times [m] & \\
T_{R'} & = & T_R \times [m] & \\
(s,i) & = & \mathsf{src}_{R'}(t,j) & \text{iff } s = \mathsf{src}_R(t) \text{ and } i = j \\
(s,i) & = & \mathsf{tgt}_{R'}(t,j) & \text{iff } s = \mathsf{tgt}_R(t) \text{ and } t \in Q_{iR} \\
root_{R'} & = & (root_R, 1) & \\
P_{iR'}(s,j) & \Leftrightarrow & s \in S_R \text{ and } P_{iR}(s) & \text{for } i = 1, \ldots, n \\
P'_{iR'}(s,j) & \Leftrightarrow & s \in S_R \text{ and } i = j & \text{for } i = 1, \ldots, n
\end{array}$$

Then R' is "almost" a transition system of type $(n+m, 1)$: "almost" because some sates may be unreachable. One obtains $\alpha(R)$ by restricting R' to the reachable states and transitions. It is clear from this definition that $|\alpha(R)|_2$ is definable from $|R|_2$ by a definable transduction. We omit the details.

Definition of β Let R' be a transition system of the form

$$\langle S_{R'}, T_{R'}, \mathsf{src}_{R'}, \mathsf{tgt}_{R'}, root_{R'}, P_{1R'}, \ldots, P_{nR'}, P'_{1R'}, \ldots, P'_{mR'} \rangle$$

where $P_{1R'}, \ldots, P_{nR'}, P'_{1R'}, \ldots, P'_{mR'}$ are properties of sates. Then we define a transition system $\beta(R)$ iff $(P'_{1R'}, \ldots, P'_{mR'})$ forms a partition of $S_{R'}$. If this is the case we let $\beta(R') = R$ where $S_R = S_{R'}, T_R = T_{R'}, \mathsf{src}_R = \mathsf{src}_{R'}, \mathsf{tgt}_R = \mathsf{tgt}_{R'}, root_R = root_{R'}$, $P_{iR} = P_{iR'}$ for $i = 1, \ldots, n$ and $Q_{iR} = \{t \in T_{R'} | \mathsf{tgt}_R(t) \in P'_{iR'}\}$ for $i = 1, \ldots, n$. It is clear that $|\beta(R)|_1$ is definable from $|R|_1$ by a definable transduction.

It is also clear from the construction that $\beta(\mathsf{Un}(\alpha(R)))$ is well defined for every transition system of type (n, m) and that:

$$\beta(\mathsf{Un}(\alpha(R))) = \mathsf{Un}(R)$$

This completes the proof of Theorem 10.

6 The Shelah-Stupp-Muchnik construction

We recall a construction and a result from Shelah and Stupp [10, 11] extended by Muchnik. The result by Muchnik is stated without a proof in Semenov [9]. We establish that it yields an improvement of our main result. However, this result being unpublished we consider it as a conjecture and not as a proved result.

We let R be a finite set of relational symbols where each symbol r has a finite arity $\rho(r) \in \mathcal{N}_+$. We recall that we denote by $\mathcal{S}(R)$ the class of all R-structures, i.e., of tuples of the form $M = \langle D_M, (r_M)_{r \in R} \rangle$ where D_M is a nonempty set (the domain of M) and $r_M \subseteq D_M^{\rho(r)}$ for every $r \in R$.

We let *son* and *cl* be two relation symbols, binary and unary respectively, which are not in R. We let $R^+ = R \cup \{son, cl\}$.

With $M \in STR(R)$ we associate the R^+-structure

$$M^+ = \langle (D_M)^+, (r_{M^+})_{r \in R}, son_{M^+}, cl_{M^+} \rangle$$

where $D_{M^+} = (D_M)^+$ is the set of nonempty sequences of elements of D_M, and

$$\begin{aligned} r_{M^+} &= \{(wd_1, \ldots, wd_{\rho(r)}) : w \in D_M^*, (d_1, \ldots, d_{\rho(r)}) \in r_M\} \\ son_{M^+} &= \{(w, wd) : w \in D_M^*, d \in D_M\} \\ cl_{M^+} &= \{wdd : w \in D_M^*, d \in D_M\} \end{aligned}$$

We use D_M^* to denote the set of all the sequences of elements of D_M (including the empty sequence).

Intuitively, M^+ is a "tree build over M"; *son* is the corresponding successor relation and *cl* is the set of *clones*, i.e., of elements of M^+ that are "like their fathers" (if $son(x, y)$ we also say that x is the father of y; it is unique).

Conjecture 23 (Semenov [9]) *The mapping $M \mapsto M^+$ is MS-compatible. In words, for every formula φ in $MS(R^+)$ one can construct a formula ψ in $MS(R)$ such that for every $M \in STR(R)$:*

$$M^+ \models \varphi \text{ iff } M \models \psi$$

It is stated in Shelah [10] and Thomas [11] (without a proof) that, if a structure M has a decidable monadic theory then so has the structure M^+

with respect to the language $MS(R^+ - \{cl\})$. This statement weakens Conjecture 23 in two respects: the "clone" relation is omitted and the statement only concerns decidability of theories and not translations of formulas. From Conjecture 23, one gets the following improvement of Theorem 10:

Theorem 24 *If conjecture 23 is true, then, for every $n, m \in \mathcal{N}$ with $m \geq 1$, the transduction:*

$$|R|_1 \mapsto |\mathsf{Un}(R)|_1$$

is MS-compatible where R ranges over simple transition systems of type (n, m).

A transition system is *simple* if no two distinct transitions have the same source, target and type.

Since some properties of simple graphs are MS-expressible with set edge quantifications but not without them, the result of Theorem 24 is an improvement of Theorem 10. (The property that a simple directed graph has a directed spanning tree of out-degree no bigger than some constant is an example of such a property; the existence of a Hamiltonian circuit is another example [5], page 125.)

This theorem is an immediate consequence of

Proposition 25 *For every $n, m \in \mathcal{N}$, $m \geq 1$, the transduction $(|R|_1)^+ \mapsto |\mathsf{Un}(R)|_1$ where R is a simple transition system of type (n, m) is MS-definable.*

Proof
We let $M = |R|_1 = \langle S_R, root_R, \mathsf{suc}_{1R}, \ldots, \mathsf{suc}_{mR}, P_{1R}, \ldots, P_{nR} \rangle$. We define a binary relation W on S_R^+ $(= D_{M^+})$ as follows:

$$W(x, y) \Leftrightarrow W_1(x, y) \vee \ldots \vee W_m(x, y)$$

where for each i:

$$W_i(x, y) \Leftrightarrow \exists z.(son(x, z) \wedge cl(z) \wedge \mathsf{suc}_i(z, y))$$

Note that $W_i(x, y)$ implies that y is a son of x.

We let $N \subseteq S_R^+$ be defined as follows:

$$y \in N \quad \Leftrightarrow \quad \begin{array}{c} \text{there exists } x \in S_{R+} \text{ such that} \\ root_{M^+}(x) \wedge \forall z(\neg son_{M^+}(z, x)) \wedge (x, y) \in W^* \end{array}$$

Note that the first two conjuncts of the above condition define x uniquely since $root_R$ consists of a unique state (x is r where $root_R = \{r\}$). We used W^* to denote the transitive closure of W.

Hence N is the set of elements of S_R^+ that are accessible from this x by a directed path all edges of which are in W.

Claim 26 $|\mathsf{Un}(R)|_1 = \langle N, W'_1, \ldots, W'_m, P'_1, \ldots, P'_n \rangle$, *where* $W'_i = W_i \cap (N \times N)$ *and* $P'_j = P_{jM} \cap N$ *for every* $i = 1, \ldots, m$ *and* $j = 1, \ldots, n$.

Proof: We define a bijection h of $Paths(R)$ (the set of nodes of $\mathsf{Un}(R)$) onto N. Let p be a path in $Paths(R)$, say $p = (t_1, \ldots, t_k)$, $t_1, \ldots, t_k \in T_R$. We let $h(p) = (s_0, \ldots, s_k) \in D_{M^+} = (S_R)^+$ where s_0 is the initial state of R and for each $i = 1, \ldots, k$, s_{i-1} is the source of t_i and s_i is the target of t_i.

Since R is simple, h is one-to-one. It is not hard to see that if $s_i \to s_{i+1}$ is a transition of type j then $W_j((s_0, \ldots, s_i), (s_0, \ldots, s_{i+1})$ holds. Hence h maps $Paths(R)$ onto N

It is then easy to verify that every $y \in N$ is the image by h of some path p (the proof is by induction on the unique integer k such that $(x, y) \in W^k$ where x is the element of D_{M^+} used in the definition of N). Finally, h is an isomorphism. We omit the details. $\square$

It is clear from the definition that N is a definable subset of D_{M^+} (by an MS formula on M^+) and that the relations $W'_1, \ldots, W'_m, P'_1, \ldots, P'_n$ are MS-definable similarly. Hence $|\mathsf{Un}(R)|_1$ can be obtained from $M^+ = |R|_1$ by an definable transduction. $\square$

The proof of this proposition is due to W. Thomas.

Example Let $R = \emptyset$, $M = \langle \{0,1\} \rangle$. Consider $M^+ = \langle \{0,1\}^+, son_{M^+}, cl_{M^+} \rangle$. One can define the complete binary tree $B = \langle N, \mathsf{suc}_1, \mathsf{suc}_2 \rangle$ in M^+ as follows: one lets x be any element of M^+ having no father; one lets N to be the set of elements y of D_{M^+} such that $(x, y) \in (son_{M^+})^*$, one lets then

$$\begin{aligned}
\mathsf{suc}_1(u, v) &\Leftrightarrow son_{M^+}(u, v) \wedge cl_{M^+}(v) \\
\mathsf{suc}_2(u, v) &\Leftrightarrow son_{M^+}(u, v) \wedge \neg cl_{M^+}(v)
\end{aligned}$$

There are only two choices for x and the corresponding structures are both isomorphic to B.

It follows that the monadic theory of B reduces to that of M^+ that is decidable since (trivially) the monadic theory of M is decidable (since M is finite).

7 Graph coverings

We have seen that the mapping from a transition system to its universal covering is MS-compatible (where a system R is represented by $|R|_2$). We ask the same question for graphs. We consider actually two different notions of covering for which the answers are completely different.

7.1 Bidirectional coverings

We consider directed graphs G, defined by means of sets: V_G (vertices), E_G (edges) and the source and target mappings respectively $\mathsf{src}_G : E_G \to V_G$, $\mathsf{tgt}_G : E_G \to V_G$.

For $x \in V_G$ we denote by $in_G(x)$ the set of edges of G with target x and by $out_G(x)$ the set of edges with the source x.

Definition 27 (Bidirectional covering) *Let G, G' be connected graphs. A homomorphism $h : G' \to G$ is a* bidirectional covering *iff it is surjective and for every $x \in V_{G'}$, h is a bijection of $in_{G'}(x)$ onto $in_G(x)$ and of $out_{G'}(x)$ onto $out_G(x)$.*

For short we shall write *b-covering* for bidirectional covering. Unlike coverings, b-coverings treat incoming edges exactly as outgoing edges.

Definition 28 (Walks) *A* walk *in G is a sequence $w = ((e_1, \eta_1), \ldots, (e_k, \eta_k))$ such that $e_1, \ldots, e_k \in E_G$, $\eta_1, \ldots, \eta_k \in \{+, -\}$, for every $i = 1, \ldots, k-1$ we have $t(e_i, \eta_i) = s(e_{i+1}, \eta_{i+1})$ where $t(e, \eta) = \mathsf{tgt}_g(e)$ if $\eta = +$ and $t(e, \eta) = \mathsf{src}_g(e)$ if $\eta = -$. Similarly $s(e, \eta) = t(e, -1 * \eta)$. Moreover we require that whenever $e_i = e_{i+1}$ then $\eta_i = \eta_{i+1}$. This condition means that the edge cannot be traversed twice consecutively in opposite directions. This condition allows to take the same edge successively twice if its source and target are identical.*

We say that w as above is walk from $s(e_1, \eta_1)$ to $t(e_k, \eta_k)$.

Fact 29 *If $h : G \to G'$ is a homomorphism and $((e_1, \eta_1), \ldots, (e_k, \eta_k))$ is a walk from x to y in G then the image of the walk is defined as the sequence $((h(e_1), \eta_1), \ldots, (h(e_k), \eta_k))$; it is a walk in G' from $h(x)$ to $h(y)$.*

Fact 30 *If $h : G \to G'$ is a b-covering, $x \in V_G$, $h(x) = x'$ and w' is a walk from x' to y', then there is a unique walk w in G from x to some y such that $h(w) = w'$; we have $h(y) = y'$.*

We now construct a b-covering of a graph G in terms of finite walks.

Let G be connected, let $s \in V_G$. Denote by $W(s)$ the set of all the walks from s to arbitrary vertices. We put in $W(s)$ the empty walk ε and assume that it goes from s to s.

We let H be the graph such that:

$$V_H = W(s) \quad E_H = \text{a disjoint copy of } W(s) - \{\varepsilon\}$$

If $w.(e,\eta) \in E_H$ for some $e \in E_G$ and $\eta \in \{+,-\}$, we let $\mathsf{src}_H(w.(e,\eta)) = w$ and $\mathsf{tgt}_H(w.(e,\eta)) = w.(e,\eta)$ if $\eta = +$ and $\mathsf{src}_H(w.(e,\eta)) = w.(e,\eta)$ and $\mathsf{tgt}_H(w.(e,\eta)) = w$ otherwise.

We now let $h : H \to G$ to be the homomorphism such that

$$\begin{array}{rcll} h(\varepsilon) & = & s & \\ h(w) & = & x & \text{such that } w \text{ goes from } s \text{ to } x,\ w \in V_H - \{\varepsilon\} \\ h(w) & = & e & \text{where } w \in E_H \text{ is of the form } w'.(e,\eta) \end{array}$$

Fact 31 *$h : H \to G$ is a b-covering.*

Proposition 32 *For every b-covering $k : K \to G$ there is a surjective homomorphism: $m : H \to K$ such that $k \circ m = h$ which is a b-covering. For any two such homomorphisms $m, m' : H \to K$, there is an automorphism i of H such that $m' = m \circ i$*

Proof: Easy consequence of Facts 29 and 30. □

We shall call H the *universal* b-covering of G and denote it by $UBC(G)$.

Theorem 33 *Let $d \in \mathcal{N}$. The transduction mapping $|G|_2$ to $|UBC(G)|_1$ for connected graphs G of degree at most d is MS-compatible.*

Proof

Let G be a graph of degree at most d. By Vizing's theorem (see [1]) there exists an edge-coloring of G with $m = d+1$ colors such that no two adjacent distinct edges have the same color.

The result is proved in [1] for finite graphs but the extension to infinite graphs is an easy application of Koenig's lemma (see [3]).

The coloring can be defined by a partition $X_1, \ldots, X_m$ of E_G in m sets. We let $X_0 = \{s\}$ be a singleton with $s \in V_G$. We now construct from

$(G, X_0, \ldots, X_m)$ deterministic transition system R of type $(0, 2m)$ as follows:

$$\begin{array}{rcl} S_R & = & V_G \\ T_R & = & E_G \times \{+,-\} \\ \mathsf{src}_R(e,+) & = & \mathsf{src}_G(e) \\ \mathsf{src}_R(e,-) & = & \mathsf{tgt}_G(e) \\ \mathsf{tgt}_R(e,+) & = & \mathsf{tgt}_G(e) \\ \mathsf{tgt}_R(e,-) & = & \mathsf{src}_G(e) \\ Q_{iR} & = & X_i \times \{+\} \quad \text{for } i = 1, \ldots, m \\ Q_{m+iR} & = & X_i \times \{-\} \quad \text{for } i = 1, \ldots, m \\ root_R & = & s \quad \text{where } X_0 = \{s\} \end{array}$$

We shall denote R by $R(G, X_0, \ldots, X_m)$. It is clear that the transduction mapping $(|G|_2, X_0, \ldots, X_m) \mapsto |R(G, X_0, \ldots, X_m)|_2$ is MS-definable.

We now define $UBC(G)$ from $\mathsf{Un}(R(G, X_0, \ldots, X_m))$ by an MS-definable transduction.

We let $H = \mathsf{Un}(R)$ and define K as follows:

V_K is the set of vertices of H defined by good paths in R, where we say that a path is *good* if it does not contain two successive edges e and e' such that:

either $\quad e \in Q_{iR}$ and $e' \in Q_{m+iR}$ for $1 \leq i \leq m$
or $\quad e \in Q_{m+iR}$ and $e' \in Q_{iR}$ for $1 \leq i \leq m$

We let $e \in E_K$ iff $e \in T_R$ and its two ends are in V_K. We let e link $u \to v$ in K if e links $u \to v$ in R and $e \in Q_{iR}$, $1 \leq i \leq m$ and we let e link $v \to u$ in K if it links $u \to v$ in R and $e \in Q_{i+m}$ for some $1 \leq i \leq m$.

Fact 34 $K = UBC(G)$

Fact 35 *There exists an MS-definable transduction τ such that*

$$\tau(|\mathsf{Un}(R(G, X_0, \ldots, X_m)|_2)) = |UBC(G)|_1$$

for every connected graph G of degree at most d and every $X_0, \ldots, X_m$ such that $R(G, X_0, \ldots, X_m)$ is well defined.

Proof: From the definition of K it follows that V_K can be defined as a subset of V_H by an MS-formula, because the notion of a good path is MS-expressible. It is easy to see that the relations src_K and tgt_K are also MS-definable. The result follows from the Fact 34. $\square$

We obtain thus that the transduction $|G|_2 \mapsto |\,UBC(G)|_1$ is MS-compatible because it can be written as the following composition:

$$|G|_2 \mapsto |R(G, X_0, \ldots, X_m)|_2 \mapsto |\mathsf{Un}(R(G, X_0, \ldots, X_m))|_2 \mapsto |\,UBC(G)|_2$$

where the first and the third transformations are MS-definable whereas the second is MS-compatible (where of course G is connected and of degree at most d, $m = d + 1$).

This conclues the proof of the Theorem 33.

$\square$

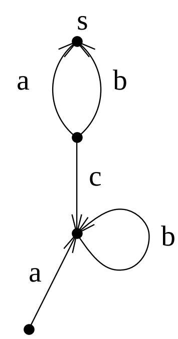


Figure 1: Example graph

Remarks

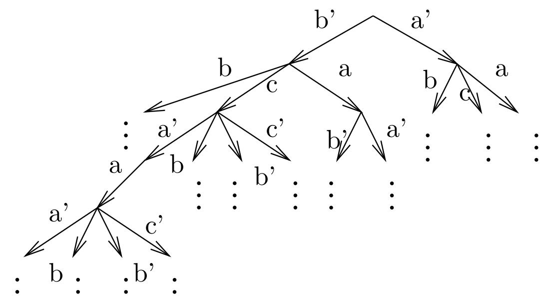
1. Since $UBC(G)$ is a tree, one can replace in Theorem 33 $|\,UBC(G)|_1$ by $|\,UBC(G)|_2$ (and obtain thus a stronger statement) because the mapping $|H|_1 \mapsto |H|_2$ is MS-definable whenever H is a tree. This is proved in Courcelle [5] for finite graphs but the proofs are based on coloring arguments which extend easily from finite to infinite graphs essentially by Koenig's lemma (see [3]).

2. Similarly the transduction $|G|_1 \mapsto |G|_2$ is MS-definable for finite and infinite simple graphs G of degree at most d. It follows that in the statement of Theorem 33, $|G|_2$ can be replaced by $|G|_1$ if G is restricted to be simple.

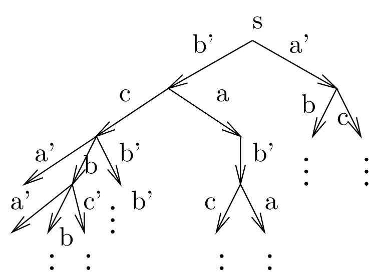
Now we give an example to illustrate the construction of the proof of Theorem 33.

Example We let G be the graph shown in Figure 1. Its edges are colored by a, b, c and s is a distinguished vertex. For each edge e of G of color x, we color by x the transition $(e, +)$ of R and by x' the "opposite" transition $(e, -)$.

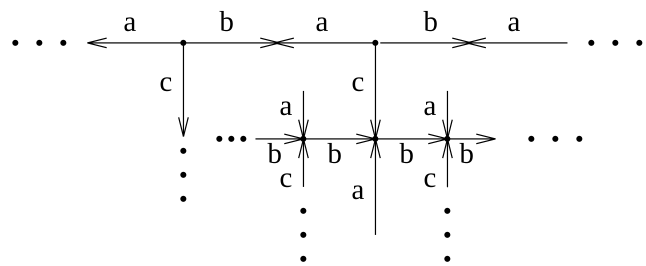
The top part of the tree $H = \mathsf{Un}(R)$ is:



After restriction to the vertices in V_K we obtain



After reversal of the "primed" edges we get $UBC(G)$:



Open question: *Can one waive the restriction to graphs of bounded degree in Theorem 33* ?

Even if we assume Conjecture 23 to be true, we do not know the answer.

We shall conclude this section by a negative result concerning a "stronger notion" of graph covering.

7.2 Definition: Distance-1-coverings

For every graph G and every $x \in V_G$, we denote by $B_G(x)$ the subgraph of G induced by $\{x\} \cup V$, where V is the set of vertices adjacent to x.

A *distance-1-covering* (a d1-covering for short) is a covering $h : G' \to G$ such that for every $y \in V_{G'}$, h is a isomorphism: $B_{G'}(y) \to B_G(h(y))$.

Example

G' is d1-covering of G where G and G' are presented in Figure 2 and h maps x' and x'' to x for $x \in \{a, b, c, d\}$.

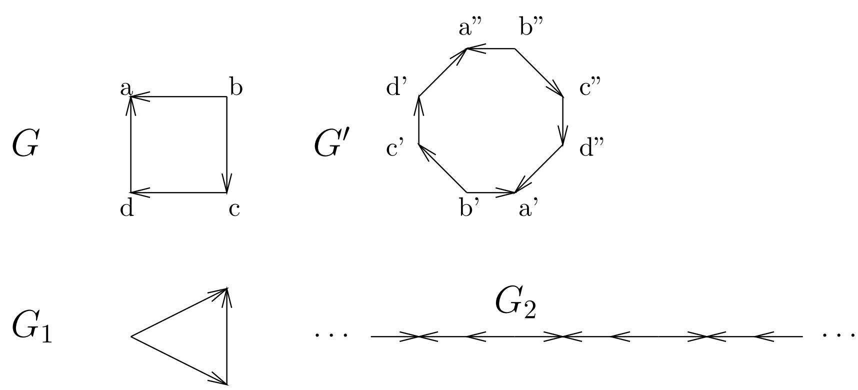


Figure 2: Example of d1-covering

The graph G_2 is a b-covering of the graph G_1 presented in Figure 2. But G_2 is not a d1-covering. Clearly, G is isomorphic to all its d1-coverings since $G = B_G(x)$ for some x. $\square$

We shall now construct a universal d1-covering of a graph G as a quotient of its universal b-covering $UBC(G)$.

We let $H = UBC(G)$ (see Fact 31 above) and $h : H \to G$. We let $E \subseteq (V_H \times V_H) \cup (E_H \times E_H)$ be the equivalence relation defined as:

$$\{(u, v) : \quad h(u) = h(v) \text{ and } u, v \text{ belong to a connected component of } h^{-1}(B_G(x)) \text{ for some } x\}$$

We let H' be the quotient graph $H|E$, we let $k : H \to H'$ be the canonical surjective homomorphism such that $h = h' \circ k$. It is not hard to see that h' is a d1-covering of G and that every d1-covering $m : G' \to G$ factors into $h' \circ m'$ where m' is a surjective homomorphism: $G' \to H'$, and further more a d1-covering. We shall call H' the *universal-d1-covering* of G and denote it by $UDC(G)$.

Proposition 36 *The mapping* $|G|_2 \mapsto |UDC(G)|_1$ *is not MS-compatible even if* G *is restricted to finite connected graphs of degree at most* 6.

Proof

We construct a finite connected graph G of degree 6 such that $UDC(G)$ is the infinite grid (augmented with diagonals on each square). Since the monadic theory of $UDC(G)$ is undecidable (even if MS-formulas do not use edge set quantification), and since the monadic theory of $|G|_2$ is decidable (since G is finite) it follows that MS-formulas expressing properties of $UDC(H)$ cannot be translated into equivalent MS-formulas on $|H|_2$ in a uniform way, for all finite connected graphs H, even of bounded degree at most 6.

The infinite grid with diagonals is the graph H such that:

$$\begin{aligned} V_H &= Int \times Int \\ E_H &= \{((x, y), (x', y')) | x, y, x', y' \in Int, \\ &\quad x \leq x' \leq x + 1, y \leq y' \leq y + 1, (x, y) \neq (x', y')\} \end{aligned}$$

Int denotes the set of integers. Figure 3 shows a portion of H.

For $x, x' \in Int$ we let $x \sim x'$ iff $x - x'$ is a multiple of 4. For $(x, y), (x', y') \in V_H$ we let $(x, y) \sim (x', y')$ iff $x \sim x'$ and $y \sim y'$. For $e, e' \in E_H$ linking respectively z_1 to z_2 and z'_1 to z'_2, we let $e \sim e'$ iff $z_1 \sim z'_1$ and $z_2 \sim z'_2$.

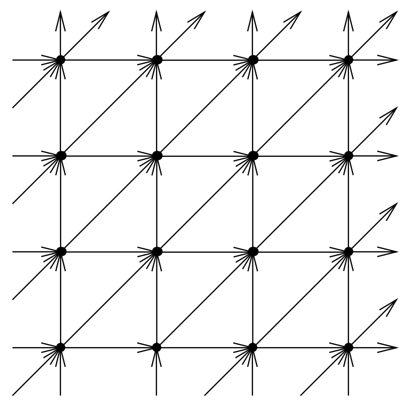


Figure 3: A portion of H

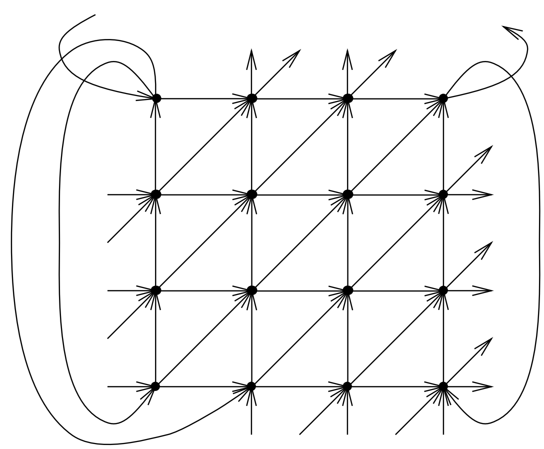


Figure 4: Graph G

We let G be the quotient graph $H|\sim$. It is not hard to see that G is the graph partially shown on Figure 4. We let h be the canonical surjective homomorphism $h : H \to G$.

It is easy to see that h is a d1-covering. In order to prove that $H = UDC(G)$ it is enough to prove that if $k : K \to H$ is a d1-covering then k is an isomorphism.

So let $k : K \to H$ be a d1-covering of H. If k is not an isomorphism, there exist $x, y \in V_K$ such that $x \neq y$ and $k(x) = k(y)$. Let us select such a pair where x and y are at minimal distance, say n. Hence in K there exists a walk from x to y of the form $w = ((e_1, \eta_1), \ldots, (e_n, \eta_n))$. Its image under k is a walk $k(w) = ((k(e_1), \eta_1), \ldots, (k(e_n), \eta_n))$ from $z = k(x)$ to itself.

The intermediate vertices on this walk are pairwise distinct and distinct with z because otherwise, n would not be the distance between x and y or one could find a pair $x', y' \in V_K$ such that $k(x') = k(y')$, $x \neq y'$ and the distance between x' and y' is less than n.

Consider now $k(w)$. It defines a cycle on the planar graph H (where edges can be traversed in either direction). This cycle is simple (it does not cross itself) and has a certain *area* namely, the number of triangles forming its interior part. We shall prove that we can replace w by a walk w' from x to y of the same length and such that the area of $k(w')$ is strictly smaller than that of $k(w)$. This will give us a contradiction and prove that k is an isomorphism.

Let u be the unique vertex of $k(w)$ having a maximal first component among those that have a maximal second component. We first assume that $u \neq k(x) = k(y)$. Let v and v' be the two neighbors of u on the circular walk $k(w)$. Let $u = (u_0, u_1)$. Up to exchanges of v and v' we have the following possible cases (by the maximality conditions on u_0 and u_1):

case 1: $v = (u_0 - 1, u_1)$, $v' = (u_0 - 1, u_1 - 1)$,

case 2: $v = (u_0, u_1 - 1)$, $v' = (u_0 - 1, u_1 - 1)$,

case 3: $v = (u_0 - 1, u_1)$, $v' = (u_0, u_1 - 1)$.

However case 1 cannot happen because w is minimal. Let us check this. Let $\bar{u}$ be the vertex of w with $k(\bar{u}) = u$. Since k is an isomorphism between $B_K(\bar{u})$ and $B_H(u)$ since $v, v' \in B_H(u)$ and are adjacent, so are $\bar{v} = k^{-1}(v)$ and $\bar{v}' = k^{-1}(v')$ in $B_K(\bar{u})$. It follows that w can be replaced by a shorter walk, which connects directly $\bar{v}$ and $\bar{v}'$ and skips $\bar{u}$. This contradicts the hypothesis that w has a minimal length.

Case 2 cannot happen for the similar reason.

In case 3 we cannot connect directly $\bar{v}$ and $\bar{v}'$ but we can link them via the unique vertex $k^{-1}(u_0-1, u_1-1)$ in $B_K(\bar{u})$ (note that v, v' and (u_0-1, u_1-1) belong all to $B_H(u)$). The resulting walk w' is such that $k(w')$ has a smaller area than $k(w)$ (smaller by 2).

If $u = k(x) = k(y)$ we use a similar argument by replacing u by the unique vertex of $k(w)$ having a minimal first component among those that have a minimal second component. The argument goes through with $+1$ instead of -1 everywhere. □

8 Conclusions

We have shown the main conjecture of [2] (see Theorem 10) saying that the unfolding operation is MS-compatible provided graphs (or transition systems) are represented in a way making it possible quantifications on sets of edges (or of transitions).

A stronger form of this result would follow from a conjecture by Muchnik stated in Semenov [9].

We also considered "bidirectional unfolding" of graphs. Although it is very close to the unfolding, we could not extend the main theorem without the additional assumption that degree is uniformly bounded. Whether this restriction can be lifted is also an open question.

These unfoldings have been defined as instances of the very general topological notion of covering (for appropriate notions of neighbourhood). The two notions correspond to neighbourhoods of increasing strengths. For the next step (distance 1-coverings), MS-logic becomes unmanageable.

References

[1] B. Bollobas. *Extremal graph theory.* Academic Press, 1978.

[2] Bruno Courcelle. The monadic second-order logic on graphs IX: machines and behaviours. *Theoretical Computer Science.* to appear.

[3] Bruno Courcelle. On the extension to infinte graphs of properties of finite ones. In preparation.

[4] Bruno Courcelle. Monadic second-order graph transductions: A survey. *Theoretical Computer Science*, 126:53–75, 1994.

[5] Bruno Courcelle. The monadic second-order logic on graphs vi: on several representations of graphs by relational structures. *Disc. Applied Maths*, 54:117–149, 1994.

[6] E.Allen Emerson and C.S. Jutla. Tree automata, mu calculus and determinacy. In *Proc. FOCS 91*, 1991.

[7] Andrzej W. Mostowski. Games with forbidden positions. Technical Report 78, University of Gdansk, 1991.

[8] Damian Niwiński. Fixed points vs. infinite generation. In *Proc. 3rd. IEEE LICS*, pages 402–409, 1988.

[9] A.L. Semenov. Decidability of monadic theories. In *MFCS'84*, volume 176 of *LNCS*, pages 162–175. Springer-Verlag, 1984.

[10] Saharon Shelah. The monadic second order theory of order. *Annals of Mathematics*, 102:379–419, 1975.

[11] Wolfgang Thomas. Automata on infinite objects. In J.van Leeuven, editor, *Handbook of Theoretical Computer Science Vol.B*, pages 995–1072. Elsvier, 1990.

Recent Publications in the BRICS Report Series

RS-95-44 Bruno Courcelle and Igor Walukiewicz. *Monadic Second-Order Logic, Graphs and Unfoldings of Transition Systems*. August 1995. 39 pp. To be presented at CSL '95.

RS-95-43 Noam Nisan and Avi Wigderson. *Lower Bounds on Arithmetic Circuits via Partial Derivatives (Preliminary Version)*. August 1995. 17 pp. To appear in *36th Annual Conference on Foundations of Computer Science*, FOCS '95, IEEE, 1995.

RS-95-42 Mayer Goldberg. *An Adequate Left-Associated Binary Numeral System in the λ-Calculus*. August 1995. 16 pp.

RS-95-41 Olivier Danvy, Karoline Malmkjær, and Jens Palsberg. *Eta-Expansion Does The Trick*. August 1995. 23 pp.

RS-95-40 Anna Ingólfsdóttir and Andrea Schalk. *A Fully Abstract Denotational Model for Observational Congruence*. August 1995. 29 pp.

RS-95-39 Allan Cheng. *Petri Nets, Traces, and Local Model Checking*. July 1995. 32 pp. Full version of paper appearing in Proceedings of AMAST '95, LNCS 936, 1995.

RS-95-38 Mayer Goldberg. *Gödelisation in the λ-Calculus*. July 1995. 7 pp.

RS-95-37 Sten Agerholm and Mike Gordon. *Experiments with ZF Set Theory in HOL and Isabelle*. July 1995. 14 pp. To appear in *Proceedings of the 8th International Workshop on Higher Order Logic Theorem Proving and its Applications*, LNCS, 1995.

RS-95-36 Sten Agerholm. *Non-primitive Recursive Function Definitions*. July 1995. 15 pp. To appear in *Proceedings of the 8th International Workshop on Higher Order Logic Theorem Proving and its Applications*, LNCS, 1995.

RS-95-35 Mayer Goldberg. *Constructing Fixed-Point Combinators Using Application Survival*. June 1995. 14 pp.

RS-95-34 Jens Palsberg. *Type Inference with Selftype*. June 1995. 22 pp.