Petri Net Reachability Graphs: Decidability Status of FO Properties

Philippe Darondeau[1], Stéphane Demri[2], Roland Meyer[3], and Christophe Morvan[4]

1 **IRISA/INRIA, Campus de Beaulieu, Rennes, France**
`philippe.darondeau@inria.fr`
2 **LSV, ENS Cachan, CNRS, INRIA, France** `Stephane.Demri@lsv.ens-cachan.fr`
3 **University of Kaiserslautern, Germany** `meyer@cs.uni-kl.de`
4 **Université Paris-Est, Marne-La-Vallée, France**
`christophe.morvan@univ-paris-est.fr`

Abstract

We investigate the decidability and complexity status of model-checking problems on unlabelled reachability graphs of Petri nets by considering first-order, modal and pattern-based languages without labels on transitions or atomic propositions on markings. We consider several parameters to separate decidable problems from undecidable ones. Not only are we able to provide precise borders and a systematic analysis, but we also demonstrate the robustness of our proof techniques.

1998 ACM Subject Classification F.3.1 Specifying and Verifying and Reasoning about Programs, F.4.1 Mathematical Logic, D.2.4 Software/Program Verification

Keywords and phrases Petri nets, First order logic, Reachability graph

Digital Object Identifier 10.4230/LIPIcs.FSTTCS.2011.140

1 Introduction

Decision problems for Petri nets. Much effort has been dedicated to decision problems about Petri nets such as reachability or equivalence, or model checking logical fragments. Reachability is decidable [20] but this is a hard problem. Language equality is, by contrast, undecidable for labelled Petri nets [11, 1]. Many important problems have received decision procedures, e.g., boundedness [16], deadlock-freeness and liveness [10] (by reduction to reachability), semilinearity [12], etc. Hack's thesis [10] provides a comprehensive overview of problems equivalent to reachability. Hack showed that equality of reachability sets of two Petri nets with identical places is undecidable [11]. As our main contribution, we link this result to first-order logic expressing properties of general Petri net reachability graphs.
Our motivations. For Petri nets, model checking CTL formulae with atomic propositions expressing that a place contains at least one token is known to be undecidable [7]. This result carries over to all fragments of CTL containing the modalities EF or AF. Model checking CTL without atomic propositions but with next-time modalities indexed by action labels is undecidable too [7]. In contrast, LTL model-checking over VASS is ExpSpace-complete [9] (atomic propositions are control states). These negative results do not compromise the search for decidable fragments of first-order logic that describe only purely graph-theoretically the reachability graphs. Our intention is to deliberately discard edge labels and atomic propositions on markings. As an example, we consider the structure $(\mathbb{N}^n, \rightarrow)$ derived from a Petri net N with n places such that $M \rightarrow M'$ iff M evolves to M' by firing a transition of N. Since $(\mathbb{N}^n, \rightarrow, =)$ is an automatic structure, its first-order theory is decidable, see e.g. [5].

© P. Darondeau, S. Demri, R. Meyer, and C. Morvan;
licensed under Creative Commons License NC-ND
31st Int'l Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2011).
Editors: Supratik Chakraborty, Amit Kumar; pp. 140–151
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

However, it is unclear what happens if we consider the first-order theory of $\rightarrow$ over the more interesting structure $(\mathrm{Reach}(N), \rightarrow)$. Here, $\mathrm{Reach}(N)$ denotes the set of all markings *reachable* from the initial marking of N. This paper investigates this question and therefore investigates the decidability status of first-order logic with a bit of MSO (via quantification over reachable markings) on $\mathbb{N}^n$, sharing with [21] a common motivation. We study properties of the Petri net reachability graph that are *purely graph-theoretical*; they do not refer to tokens or transition labels and they are mostly *local* in that they can often be expressed in terms of $\rightarrow$ instead of its transitive closure. For instance, this contrasts with logics in [3] that state quantitative properties on markings and transitions, and evaluate formulae on runs.

Our contributions. We investigate the model-checking problem over structures of the form $(\mathrm{Reach}(N), \rightarrow, \xrightarrow{*})$ generated from Petri nets N with first-order languages including predicate symbols for $\rightarrow$ and/or $\xrightarrow{*}$. As it is a classical fragment of first-order logic, we also consider the modal language $\mathrm{ML}(\Box, \Box^{-1})$ with forward and backward modalities. To conclude the study, we consider an alternative framework where the structures are *reachability sets*, subsets of $\mathbb{N}^n$ when the underlying net has n places. For these structures, we study satisfiability of properties defined by *patterns*. Patterns are bounded n-dimensional sets of points that are colored black, white, or grey to mean "reachable", "non-reachable", or "don't care", respectively. Let us mention prominent features of our investigation. (1) Undecidability proofs are obtained by reduction from the equality problem (or the inclusion problem) between reachability sets defined by Petri nets, shown undecidable in [11]. We demonstrate that our proof schema is robust and can be adapted to numerous formalisms specifying local properties as in first-order logic. (2) To determine the cause of undecidability, we investigate logical fragments. At the same time, we strive for maximally expressive decidable fragments. (3) For decidable problems, we assess the computational complexity — either relative to standard complexity classes or by establishing a reduction from the reachability problem for Petri nets. Our main findings are as follows: Model-checking $(\mathrm{Reach}(N), \rightarrow)$ [resp. $(\mathrm{Reach}(N), \xrightarrow{*})$, $(\mathrm{Reach}(N), \xrightarrow{+})$] is undecidable for the appropriate first-order language with one binary predicate symbol. Undecidability is also shown for the positive fragment of $\mathrm{FO}(\rightarrow)$, the forward fragment of $\mathrm{FO}(\rightarrow)$ and $\mathrm{FO}(\rightarrow)$ augmented with $\xrightarrow{*}$ even if the reachability sets are effectively semilinear. We prove that model-checking the existential fragment of $\mathrm{FO}(\rightarrow)$ is decidable, but as hard as the reachability problem for Petri nets. As far as $\mathrm{ML}(\Box, \Box^{-1})$ is concerned, the global model-checking on $(\mathrm{Reach}(N), \rightarrow)$ is undecidable but it becomes decidable when restricted to $\mathrm{ML}(\Box)$ (even if extended with Presburger-definable predicates on markings); the latter problem is also as hard as the reachability problem for Petri nets. The satisfiability of properties defined by bounded patterns is undecidable.

2 Preliminaries

We recall basics on Petri nets and semilinear sets; we introduce Petri net reachability graphs as first-order structures. We define first-order logic and modal logic interpreted on these graphs. Finally, we present decidability results about model-checking problems.

2.1 Petri nets

A *Petri net* is a bi-partite graph $N = (P, T, F, M_0)$, where P and T are *finite* disjoint sets of *places* and *transitions*, and $F : (P \times T) \cup (T \times P) \rightarrow \mathbb{N}$. A *marking* of N is a function $M : P \rightarrow \mathbb{N}$. M_0 is the *initial marking* of N. A transition $t \in T$ is *enabled at* a marking

M, written $M[t\rangle$, if $M(p) \geq F(p,t)$ for all places $p \in P$. If t is enabled at M then it can *be fired*. This leads to the marking M' defined by $M'(p) = M(p) + F(t,p) - F(p,t)$ for all $p \in P$, in notation: $M[t\rangle M'$. The definitions are extended to transition sequences $s \in T^*$ in the expected way. A marking M' is *reachable* from a marking M if $M[s\rangle M'$ for some $s \in T^*$. A transition t is *in self-loop* with a place p iff $F(p,t) = F(t,p) > 0$. A transition is *neutral* if it has null effect on all places. The *reachability set* $\mathrm{Reach}(N)$ of N is the set of all markings that are reachable from the initial marking.

▶ **Theorem 2.1.** *(I) [20] Given a Petri net N and two markings M and M', it is decidable whether M' is reachable from M. (II) [11] Given two Petri nets N and N', it is not decidable whether* $\mathrm{Reach}(N) = \mathrm{Reach}(N')$ *[resp.* $\mathrm{Reach}(N) \subseteq \mathrm{Reach}(N')$*].*

A Petri net N induces several standard structures. The *unlabelled reachability graph* of N is the structure $\mathrm{URG}(N) = (D, init, \to, \xrightarrow{*}, \xrightarrow{+}, =)$ where $D = \mathrm{Reach}(N)$, $init = \{M_0\}$, and $\to$ is the binary relation on D defined by $M \to M'$ if $M[t\rangle M'$ for some $t \in T$. The relations $\xrightarrow{*}$ and $\xrightarrow{+}$ are the iterative and strictly iterative closures of $\to$, respectively. The *reachability graph* $\mathrm{RG}(N)$ of N is $(\mathrm{Reach}(N), \to)$. The *unlabelled transition graph* of N is the structure $\mathrm{UG}(N) = (D, init, \to, \xrightarrow{*}, \xrightarrow{+}, =)$ with $D = \mathbb{N}^P$. *Note that reachability of markings is not taken into account in* $\mathrm{UG}(N)$. In the sequel, by default $\mathrm{card}(P) = n$ and we identify $\mathbb{N}^P$ and $\mathbb{N}^n$. We also call 1-*loop* an edge $M \to M'$ with $M = M'$.
Semilinear subsets of $\mathbb{N}^n$ form an effective Boolean algebra and they coincide with sets definable in Presburger arithmetic (decidable first-order theory of natural numbers with addition). Hence, herein we use equally semilinearity or definability in Presburger arithmetic. Note that in [8], Ginsburg and Spanier gave an effective correspondence between semilinear subsets and subsets of $\mathbb{N}^n$ definable in Presburger arithmetic.We know that given a Petri net N and a semilinear set $E \subseteq \mathbb{N}^{|P|}$ one can decide whether $\mathrm{Reach}(N) \cap E \neq \emptyset$ [11, L. 4.3].

2.2 First-order languages

We introduce a first-order logic FO with atomic predicates $\mathsf{x} \to \mathsf{y}$, $\mathsf{x} \xrightarrow{*} \mathsf{y}$, $\mathsf{x} \xrightarrow{+} \mathsf{y}$ and $init(\mathsf{x})$. Formulae in FO are defined by $\mathsf{x} \to \mathsf{y} \mid \mathsf{x} \xrightarrow{*} \mathsf{y} \mid \mathsf{x} \xrightarrow{+} \mathsf{y} \mid init(\mathsf{x}) \mid \mathsf{x} = \mathsf{y} \mid \neg\varphi \mid \varphi \wedge \varphi \mid \exists\, \mathsf{x}\ \varphi \mid \forall\, \mathsf{x}\ \varphi$. Given a set $\mathtt{P}$ of predicate symbols from the above signature, we denote the *restriction of* FO *to the predicates in* $\mathtt{P}$ by $\mathrm{FO}(\mathtt{P})$. Formulae are interpreted either on $\mathrm{URG}(N)$ or on $\mathrm{UG}(N)$. Observe that FO on $\mathrm{UG}(N)$ enables, using *init* and reachability predicates, to relativize formulae to $\mathrm{URG}(N)$. We omit the standard definition of the satisfaction relation $\mathcal{U}, \mathbf{v} \models \varphi$ with $\mathcal{U}$ a structure ($\mathrm{URG}(N)$, $\mathrm{RG}(N)$ or $\mathrm{UG}(N)$) and $\mathbf{v}$ a valuation of the free variables in φ. Typically, $\forall\, \mathsf{x}\ \varphi$ holds true whenever the formula φ holds true for all elements (markings) of the considered structure. *Sentences* are closed formulae, i.e. without free variables. If $\mathcal{U} \models \varphi$ then $\mathcal{U}$ is called a model of φ.

In the sequel, we consider several model-checking problems. The model-checking problem $\mathrm{MC}^{\mathrm{URG}}(\mathrm{FO})$ [resp. $\mathrm{MC}^{\mathrm{UG}}(\mathrm{FO})$] is stated as follows: given a Petri net N and a sentence $\varphi \in \mathrm{FO}$, does $\mathrm{URG}(N) \models \varphi$ [resp. $\mathrm{UG}(N) \models \varphi$] ? The logics $\mathrm{FO}(\mathtt{P})$ induce restricted model checking problems $\mathrm{MC}^{\mathrm{URG}}(\mathrm{FO}(\mathtt{P}))$ and $\mathrm{MC}^{\mathrm{UG}}(\mathrm{FO}(\mathtt{P}))$, respectively. Formulae in FO can express standard structural properties, like deadlock-freeness ($\forall\, \mathsf{x}\ \exists \mathsf{y}\ \mathsf{x} \to \mathsf{y}$) or cyclicity ($\forall \mathsf{x} \forall \mathsf{y}\ \mathsf{x} \xrightarrow{*} \mathsf{y} \Rightarrow \mathsf{y} \xrightarrow{*} \mathsf{x}$). Semilinear sets and relations are known to be automatic (may be generated by finite synchronous automata [5]). In particular, it means that $(\mathbb{N}^n, \to, =)$ is automatic. By [5], ($\star$) $\mathrm{MC}^{\mathcal{S}}(\mathrm{FO})$ is decidable for each automatic structure $\mathcal{S}$. Proposition 2.2 below, consequence of ($\star$), is our current state of knowledge.

▶ **Proposition 2.2.** (I) $\mathrm{MC}^{\mathrm{UG}}(\mathrm{FO}(\rightarrow,=))$ is decidable. (II) Let $\mathcal{C}$ be a class of Petri nets N for which the restriction on $\mathrm{Reach}(N)$ of the reachability relation $\mathsf{x} \xrightarrow{*} \mathsf{y}$ is effectively semilinear. Then, $\mathrm{MC}^{\mathrm{URG}}(\mathrm{FO})$ restricted to $\mathcal{C}$ is decidable. (III) Let $\mathcal{C}$ be a class of Petri nets N for which $\mathrm{Reach}(N)$ is effectively semilinear. Then, $\mathrm{MC}^{\mathrm{URG}}(\mathrm{FO}(\rightarrow,=))$ restricted to $\mathcal{C}$ is decidable.

Here are some classes of Petri nets for which the reachability relation $\xrightarrow{*}$ is effectively semilinear: cyclic Petri nets [2], communication-free Petri nets [6], vector addition systems with states of dimension 2 [18], single-path Petri nets [14], etc.

Note that given φ in $\mathrm{FO}(\rightarrow,=)$, one can effectively build a Presburger formula that characterizes exactly the valuations satisfying φ in $\mathrm{UG}(N)$. However, having $\mathbb{N}^n$ as a domain does not always guarantee decidability, see the undecidability result in [21, Theorem 2] about a structure with domain $\mathbb{N}^n$ but equipped with successor relations for each dimension and with regularity constraints on them.

2.3 Standard first-order fragments: modal languages

By moving along edges, modal languages provide a local view for graph structures. Note the constrast to first-order logic in which one quantifies over any element of the structure. Applications of modal languages include modelling temporal and epistemic reasoning, and they are central for designing logical specification languages. The modal language $\mathrm{ML}(\Box,\Box^{-1})$ (or simply ML) defined below has no propositional variable (like Hennessy-Milner modal logic) and no label on modal operators. This allows us to interpret modal formulae on directed graphs of the form $(\mathrm{Reach}(N),\rightarrow)$. The modal formulae in ML are defined by the grammar $\bot \mid \top \mid \neg\varphi \mid \varphi\wedge\psi \mid \Box\varphi \mid \Box^{-1}\varphi$. We write $\mathrm{ML}(\Box)$ to denote the restriction of ML to $\Box$ and we use the standard abbreviations $\Diamond\varphi \overset{\text{def}}{=} \neg\Box\neg\varphi$ and $\Diamond^{-1}\varphi \overset{\text{def}}{=} \neg\Box^{-1}\neg\varphi$. We interpret modal formulae on directed graphs $(\mathrm{Reach}(N),\rightarrow)$. We provide the definition of the satisfaction relation $\models$ relatively to an arbitrary directed graph $\mathcal{M} = (W,R)$ and $w \in W$ (clauses for Boolean connectives and logical constants are omitted):

- ⋆ $\mathcal{M}, w \models \Box\varphi \overset{\text{def}}{\Leftrightarrow}$ for every $w' \in W$ such that $(w,w') \in R$, we have $\mathcal{M}, w' \models \varphi$.
- ⋆ $\mathcal{M}, w \models \Box^{-1}\varphi \overset{\text{def}}{\Leftrightarrow}$ for every $w' \in W$ such that $(w',w) \in R$, we have $\mathcal{M}, w' \models \varphi$.

Model-checking problem $\mathrm{MC}^{\mathrm{URG}}(\mathrm{ML})$ is the following: given a Petri net N and $\varphi \in \mathrm{ML}$, does $(\mathit{Reach}(N),\rightarrow), M_0 \models \varphi$? Let $\mathrm{MC}^{\mathrm{URG}}(\mathrm{ML}(\Box))$ denote $\mathrm{MC}^{\mathrm{URG}}(\mathrm{ML})$ restricted to $\mathrm{ML}(\Box)$.

▶ **Proposition 2.3.** $\mathrm{MC}^{\mathrm{URG}}(\mathrm{ML}(\Box))$ is decidable and PSPACE-complete.

Adding $\Box^{-1}$ to $\mathrm{ML}(\Box)$, often does not change the computational complexity of model checking, see e.g. [4]. When it comes to Petri net reachability graphs $\mathrm{RG}(N)$, adding $\Box^{-1}$ preserves decidability but at the cost of performing reachability checks. With a hardness result in Section 3.4, we argue that such checks cannot be avoided.

▶ **Proposition 2.4.** $\mathrm{MC}^{\mathrm{URG}}(\mathrm{ML}(\Box,\Box^{-1}))$ is decidable.

We introduce another decision problem about ML that is related to $\mathrm{MC}^{\mathrm{URG}}(\mathrm{FO}(\rightarrow))$. The *validity problem* $\mathrm{VAL}^{\mathrm{URG}}(\mathrm{ML})$, is stated as follows: given a Petri net N and $\varphi \in \mathrm{ML}$, does $(\mathrm{Reach}(N),\rightarrow), M \models \varphi$ for every marking $M \in \mathrm{Reach}(N)$? As observed earlier, formulae from $\mathrm{ML}(\Box,\Box^{-1})$ can be viewed as first-order formulae in $\mathrm{FO}(\rightarrow)$. Therefore, using modal languages in specifications is a way to consider fragments of $\mathrm{FO}(\rightarrow)$. Indeed, given φ in $\mathrm{ML}(\Box,\Box^{-1})$, one can compute in linear time a first-order formula φ' with only two individual variables (see e.g. [4]) that satisfies: for every Petri net N we have $\mathrm{RG}(N) \models \varphi'$ iff $\mathrm{RG}(N), M \models \varphi$ for every M in $\mathrm{Reach}(N)$. Hence, the validity problem $\mathrm{VAL}^{\mathrm{URG}}(\mathrm{ML})$ appears as a natural counterpart to $\mathrm{MC}^{\mathrm{URG}}(\mathrm{FO}(\rightarrow))$.

3 Structural Properties of Unlabelled Net Reachability Graphs

We study the decidability status of model checking unlabelled reachability graphs of Petri nets against the first-order and modal logics defined in the previous section.

3.1 A proof schema for the undecidability of FO(→)

To establish undecidability of $\mathrm{MC}^{\mathrm{URG}}(\mathrm{FO}(\rightarrow))$, we provide a reduction of the equality problem for reachability sets, see Theorem 2.1(II). Given two Petri nets N_1 and N_2 with the same places, we build $\overline{N}$ and φ in FO(→) such that $\mathrm{Reach}(N_1) = \mathrm{Reach}(N_2)$ iff $\mathrm{RG}(\overline{N}) \models \varphi$. Interestingly, φ shall be independent of N_1 and N_2.

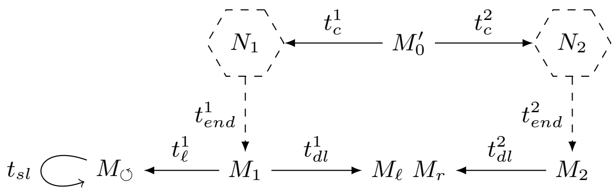

Figure 3.1 Reachability graph of $\overline{N}$

In $\overline{N}$, the nets N_1 and N_2 to be compared for equality of reachability sets share all places except two added control places p_1 and p_2 (set in self-loop with the respective transitions of N_1 and N_2). The Petri net $\overline{N}$ has one more extra place p initially marked. Two concurrent transitions t_c^1 and t_c^2 compete to consume the initial token and mark either p_1 and all places marked in the initial configuration of N_1 or p_2 and all places marked in the initial configuration of N_2 (see Figure 3.1). The first step in the execution of $\overline{N}$ implements an arbitrary choice between simulating N_1 or N_2.

Once the simulation of N_1 or N_2 has started, it may be stopped at any time. This is done by two transitions t_{end}^1 and t_{end}^2 that move the control token from p_1 or p_2 to a new control place p_1' or p_2', thus leading to the marking M_1 or M_2 shown in Figure 3.1. After this, the token count on the places of N_1 and N_2 is not changed any more. Moving the token to p_1' or p_2' switches control to reporting subnets N_1' or N_2' that behave as indicated in Figure 3.1 starting from markings M_1 and M_2.

By just emptying the control place p_1' or p_2', N_1' and N_2' may forget the index 1 or 2 of the net N_1 or N_2 that was simulated and enter a deadlock marking M, that reflects the last marking of N_1 or N_2 in the simulation. For this purpose, $\overline{N}$ is provided with two transitions t_{dl}^1 and t_{dl}^2 (in Figure 3.1, M is denoted M_ℓ and M_r indicating whether it emerged from the simulation of N_1 (*left*) or N_2 (*right*)). $\mathrm{Reach}(N_1) = \mathrm{Reach}(N_2)$ iff every simulation result or deadlock marking M can be obtained from N_1 *and* N_2. But inspecting M in isolation does not reveal whether it stemmed from N_1 or N_2.

Deadlock markings (M) and their immediate predecessor markings (M_1 and/or M_2) are easily characterized by first-order formulae. In order to express in FO(→) that every simulation result M has exactly two direct ancestor markings M_1 and M_2 (such that $M_1[t_{dl}^1\rangle M$ and $M_2[t_{dl}^2\rangle M$), it is necessary that the behaviours of N_1' and N_2' from M_1 or M_2 can be distinguished by FO(→) formulae. For this purpose, one gives to N_1' *but not* to N_2' the possibility to avoid the deadlock state $M_\ell = M$ by firing from M_1 a special transition t_ℓ^1 that leads to a marking ($M_\circlearrowleft$) with a 1-loop $t_\circlearrowleft$ (no new deadlock is introduced thus). In $\overline{N}$, t_ℓ^1 competes with t_{dl}^1 to move the token from the control place p_1' to another control place

$p_{\circlearrowleft}$, controlling the 1-loop $t_{\circlearrowleft}$. In this way, the formula $\varphi_\ell(\mathsf{x}) \stackrel{\text{def}}{=} \exists\, \mathsf{y}\ (\mathsf{x} \rightarrow \mathsf{y} \wedge \mathsf{y} \rightarrow \mathsf{y})$ holds in markings M_1 and does not hold in markings M_2.

A formula φ expressing that N_1 and N_2 have equal reachability sets is then: $\forall\, \mathsf{z}\ (\neg\exists \mathsf{z}'\ \mathsf{z} \rightarrow \mathsf{z}') \Rightarrow (\exists \mathsf{z}_1\ \mathsf{z}_1 \rightarrow \mathsf{z} \wedge \varphi_\ell(\mathsf{z}_1)) \wedge (\exists \mathsf{z}_2\ \mathsf{z}_2 \rightarrow \mathsf{z} \wedge \neg\varphi_\ell(\mathsf{z}_2))$. The formula φ requires that for any simulation result M, both logical experiments witnessing for N_1 and N_2 succeed. It is important to observe that the only deadlock markings of $\overline{N}$ are the markings reached by the transitions t^1_{dl} and t^2_{dl}. Lemma 3.1 below, based on this remark, shows that the formula φ expresses in fact the equality of the reachability sets of N_1 and N_2.

The strength of the construction stems from the combination of two ideas. A Petri net can (i) store choices over arbitrary long histories and (ii) reveal this propagated information by finite back and forth experiments determining local structures characterised by first-order formulae. The experiments consist here of one backward transition, reconstructing the initial choice, and some forward transitions checking the presence of a 1-loop.

▶ **Lemma 3.1.** $\mathrm{Reach}(N_1) = \mathrm{Reach}(N_2)$ *if and only if* $\mathrm{RG}(\overline{N}) \models \varphi$.

For the implication from left to right, consider a deadlock marking M. M is only reachable via t^1_{dl} or t^2_{dl}, say $M'_1[t^1_{dl}\rangle M$. Then marking M'_1 satisfies φ_ℓ and stems from a marking $M_1[t^1_{end}\rangle M'_1$ of N_1. The hypothesis on equal reachability sets then yields a marking M_2 of N_2 that leads by transition t^2_{end} to a marking M'_2 satisfying $\neg\varphi_\ell$ as required.

In turn, if φ holds, then we prove two inclusions. To show $\mathrm{Reach}(N_1) \subseteq \mathrm{Reach}(N_2)$, consider marking M_1 reachable via sequence s_1 in N_1. In $\overline{N}$, the marking can be prolonged to a deadlock M with $M'_0[t^1_c\rangle M^1_0[s_1\rangle M_1[t^1_{end}\rangle M'_1[t^1_{dl}\rangle M$. Here, M'_1 satisfies φ_ℓ. But φ yields another predecessor M'_2 of M with $M'_2 \neq M'_1$. To avoid the 1-loop, it has to result from a sequence $M'_0[t^2_c\rangle M^2_0[s_2\rangle M_2[t^2_{end}\rangle M'_2[t^2_{dl}\rangle M$. It is readily checked that M_1 and M_2 coincide up to the token on the control place. This means $M_1 \in \mathrm{Reach}(N_2)$ as required.

By recycling variables in φ above, we get a sharp result that marks the undecidability border of model checking against $\mathrm{FO}(\rightarrow)$ by two variables. Model checking $\mathrm{FO}(\rightarrow)$ restricted to a one variable is decidable.

▶ **Theorem 3.2.** *There exists a formula φ in* $\mathrm{FO}(\rightarrow)$ *with two individual variables such that* $\mathrm{MC}^{\mathrm{URG}}(\mathrm{FO}(\rightarrow))$ *restricted to φ is undecidable.*

The above undecidability result can be further sharpened since it is shown in [15] that the undecidability of the equality problem holds already for Petri nets with 5 unbounded places.

3.2 Robustness of the proof schema

Based on the previous proof schema, we present undecidability results for subproblems of $\mathrm{MC}^{\mathrm{URG}}(\mathrm{FO}(\rightarrow))$. We consider the positive fragment, the forward fragment, the restriction when the direction of edges is omitted, and $\mathrm{ML}(\Box, \Box^{-1})$. Let $\lambda(\mathsf{x}, \mathsf{x}') \stackrel{\text{def}}{=} (\mathsf{x} \rightarrow \mathsf{x}') \vee (\mathsf{x}' \rightarrow \mathsf{x})$. Expressing properties about $\mathrm{RG}(N)$ in $\mathrm{FO}(\lambda)$ amounts to getting rid of the direction of edges of this graph. Despite this weakening, undecidability is still present. To instantiate the above argumentation, we have to identify deadlock markings and analyse their environment. In $\mathrm{FO}(\lambda)$, we augment markings encountered during the simulation by 3-cycles. Then, the absence of 3-cycles and an environment without such cycles characterises deadlock markings.

▶ **Proposition 3.3.** $\mathrm{MC}^{\mathrm{URG}}(\mathrm{FO}(\lambda))$ is undecidable.

Proposition 3.4 below is proved by adapting the construction depicted in Figure 3.1.

▶ **Proposition 3.4.** $\mathrm{VAL}^{\mathrm{URG}}(\mathrm{ML}(\Box, \Box^{-1}))$ is undecidable.

This undecidability result is tight (see Section 3.3). Translating formulae in $\mathrm{ML}(\Box, \Box^{-1})$ to $\mathrm{FO}(\to)$ with two individual variables gives another evidence that $\mathrm{MC}^{\mathrm{URG}}(\mathrm{FO}(\to))$ with two variables is undecidable. Although $\mathrm{VAL}^{\mathrm{URG}}(\mathrm{ML}(\Box, \Box^{-1}))$ and $\mathrm{MC}^{\mathrm{URG}}(\mathrm{FO}(\to))$ are undecidable, we have identified decidable fragments of modal logic in Section 2.3. By analogy, one may expect to find decidable fragments of first-order logic. We prove that this is not the case. We consider here positive $\mathrm{FO}(\to)$ and forward $\mathrm{FO}(\to)$. In a *positive formula*, atomic propositions occur only under the scope of an even number of negations. Let $\mathrm{FO}^{+}(\mathtt{P})$ denote the positive fragment of $\mathrm{FO}(\mathtt{P})$. A *forward formula* is a formula in which every occurrence $\mathtt{x} \to \mathtt{y}$ is in the scope of a quantifier sequence of the form $Q_1\, \mathtt{x} \ldots Q_2\, \mathtt{y}$ where $\mathtt{x}$ is bound before $\mathtt{y}$. Let $\mathrm{FO}_f(\mathtt{P})$ denote the forward fragment of $\mathrm{FO}(\mathtt{P})$.

▶ **Proposition 3.5.** $\mathrm{MC}^{\mathrm{URG}}(\mathrm{FO}^{+}(\to))$ is undecidable.

▶ **Proposition 3.6.** $\mathrm{MC}^{\mathrm{URG}}(\mathrm{FO}_f(\to))$ is undecidable.

While forward formulae can well identify the deadlock markings used in the proof schema, the difficulty is in the description of the local environment witnessing the simulation results.

3.3 Taming undecidability with fragments

In this section, we present the restrictions of $\mathrm{FO}(\to)$ that we found to have decidable model checking or validity problems. We write $\exists\mathrm{FO}$ for the fragment of FO whose formulae use only existential quantification when written in prenex normal form.

▶ **Proposition 3.7.** $\mathrm{MC}^{\mathrm{URG}}(\exists\mathrm{FO}(\to, =))$ is decidable.

Decidability of $\mathrm{MC}^{\mathrm{URG}}(\exists\mathrm{FO}(\to, =))$ is obtained by checking the existence of reachable markings satisfying Presburger constraints. As a corollary, $\mathrm{MC}^{\mathrm{URG}}(\mathrm{FO}(\to, =))$ restricted to Boolean combinations of existential formulae is decidable, and so is the subgraph isomorphism problem as follows: given a finite directed graph $\mathcal{G}$ and a Petri net N, is there a subgraph of $(\mathrm{Reach}(N), \to)$ isomorphic to $\mathcal{G}$? Section 3.2 proves that $\mathrm{VAL}^{\mathrm{URG}}(\mathrm{ML}(\Box, \Box^{-1}))$ is undecidable. To our surprise, and in contrast to the negative result on model checking the forward fragment of FO, this undecidability depends on the backward modality, see Proposition 3.8 below (it can be extended to allow labels on edges). We write $\mathrm{PAML}(\Box)$ to denote the extension of $\mathrm{ML}(\Box)$ by allowing as atomic formulae quantifier-free Presburger formulae about the number of tokens in places.

▶ **Proposition 3.8.** The validity problem $\mathrm{VAL}^{\mathrm{URG}}(\mathrm{PAML}(\Box))$ is decidable.

Decidability mainly holds because (non-)satisfaction of formulae in $\mathrm{PAML}(\Box)$ requires the existence of finite tree-like patterns and if the root is in $\mathrm{Reach}(N)$, so are all its nodes (unlike with $\mathrm{ML}(\Box, \Box^{-1})$).

3.4 On the hardness of the decidable problems

Some of our decision procedures call subroutines for solving reachability in Petri nets. As this problem is not known to be primitive recursive, we provide here some complexity-theoretic justification for these costly invocations: we reduce the reachability problem for Petri nets to the decidable problems $\mathrm{MC}^{\mathrm{URG}}(\mathrm{ML}(\Box, \Box^{-1}))$ and to $\mathrm{MC}^{\mathrm{URG}}(\exists\mathrm{FO}(\to))$. Besides reachability, we gave decision procedures that exploit the semilinearity of reachability sets or relations (see e.g. Proposition 2.2), but already for bounded Petri nets, $\mathrm{MC}^{\mathrm{URG}}(\mathrm{FO}(\to))$ is of high complexity.

▶ **Proposition 3.9.** $\mathrm{MC}^{\mathrm{URG}}(\mathrm{FO}(\to))$ restricted to bounded Petri nets is decidable but this problem has nonprimitive recursive complexity.

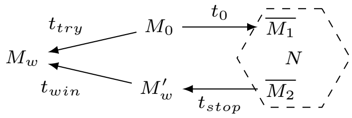

Figure 3.2 Reachability graph in the hardness proof of ML($\Box, \Box^{-1}$)-model checking

▶ **Proposition 3.10.** There is a logarithmic-space reduction from the reachability problem for Petri nets to $\mathrm{MC}^{\mathrm{URG}}(\mathrm{ML}(\Box, \Box^{-1}))$.

We reduce reachability of marking M_2 from marking M_1 in a Petri net N to an instance of $\mathrm{MC}^{\mathrm{URG}}(\mathrm{ML}(\Box, \Box^{-1}))$ for a larger net $\overline{N}$. The idea is to introduce a marking M_w (see Figure 3.2) such that the existence of a path to M_w of length greater than 1 witnesses for the existence of some path from M_1 and M_2 in $RG(N)$. To reach M_w by an ML formula, we place it close to the new initial marking. We sketch the argumentation. The inital marking M_0 of $\overline{N}$ contains a single marked place p_i on which compete two transitions t_{try} and t_0. Transition t_{try} moves the unique token from p_i to another place p_w and thus produces the marking M_w *where no other place is marked*. Transition t_0 loads M_1 in the places of N and moves the control token from p_i to another control place p_c set in self-loop with all transitions of N. This starts the simulation of N from M_1. The simulation may be interrupted whenever it reaches a marking of N greater than or equal to M_2. Then, transition t_{stop} consumes M_2 from the places of N and moves the control token from p_c to a place $p_{w'}$. The control token is finally moved from $p_{w'}$ to p_w by firing t_{win}. M_w is reached, after firing $t_{stop}\, t_{win}$, iff $\overline{M_2}$ is reached. Therefore M_2 is reachable from M_1 iff M_w is reachable from $\overline{M_1}$ (its restriction to places of N equals M_1). This is equivalent to stating that M_w has a predecessor different from M_0. The shape of the reachability graph enables to formulate the latter as a local property in $\mathrm{ML}(\Box, \Box^{-1})$: $\varphi \;:=\; \Diamond(\Box \perp \wedge \Diamond^{-1}\Diamond^{-1}\top)$. Without loss of generality, we can assume that M_1 is no deadlock and $M_2 \neq M_1$. Formula φ requires that M_0 has a deadlock successor and has an incoming path of length two. That the successor is a deadlock means it is not $\overline{M_1}$ but M_w obtained by firing t_{try}. The path from M_0 to M_w is of length one and M_0 has no predecessor. So the path of length two to M_w is not via t_{try} but stems from t_{win}. This means M_w is reachable from $\overline{M_1}$, which means M_2 is reachable from M_1 in N.

The proof of Proposition 3.10 can be adapted to $\exists\mathrm{FO}(\rightarrow)$ for which we also have shown decidability of the model-checking by reduction to the reachability problem for Petri nets.

▶ **Proposition 3.11.** There is a logarithmic-space reduction from the reachability problem for Petri nets to $\mathrm{MC}^{\mathrm{URG}}(\exists\mathrm{FO}(\rightarrow))$.

4 FO with Reachability Predicates

We consider several first-order languages with reachability relations $\xrightarrow{*}$ or $\xrightarrow{+}$, mainly without the one-step relation $\rightarrow$. Undecidability does not follow from Theorem 3.2 since we may exclude $\rightarrow$. Nonetheless we follow the same proof schema. Besides, we distinguish the case when reachability sets are semilinear leading to a surprising undecidability result (Proposition 4.4). Finally, we show that $\mathrm{MC}^{\mathrm{UG}}(\mathrm{FO}(\rightarrow, \xrightarrow{*}))$ is undecidable too.

4.1 FO with reachability relations

The decidability status of $\mathrm{MC}^{\mathrm{URG}}(\mathrm{FO}(\xrightarrow{+}))$ is not directly dependent upon the decidability status of $\mathrm{MC}^{\mathrm{URG}}(\mathrm{FO}(\rightarrow))$. Still we are able to adapt the construction of Section 3.1 but using now a formula φ in FO($\xrightarrow{+}$). The Petri net $\overline{N}$ is the one depicted on Figure 3.1. The formula φ is defined as follows: $\varphi \stackrel{\text{def}}{=} \forall\ \mathsf{z}\ dl(\mathsf{z}) \Rightarrow (\exists\ \mathsf{z}_1\ (\mathsf{z}_1 \xrightarrow{+} \mathsf{z}) \wedge \varphi_2(\mathsf{z}_1)) \wedge (\exists\ \mathsf{z}_2\ (\mathsf{z}_2 \xrightarrow{+} \mathsf{z}) \wedge \psi_2(\mathsf{z}_2))$ where $dl(\mathsf{z}) \stackrel{\text{def}}{=} \neg\exists \mathsf{z}'\ \mathsf{z} \xrightarrow{+} \mathsf{z}'$, $sl(\mathsf{y}) \stackrel{\text{def}}{=} \mathsf{y} \xrightarrow{+} \mathsf{y}\ \wedge\ \forall \mathsf{w}\ [\mathsf{y} \xrightarrow{+} \mathsf{w} \Rightarrow \mathsf{w} \xrightarrow{+} \mathsf{y}]$, $\varphi_2(\mathsf{z}) \stackrel{\text{def}}{=} [\exists\ \mathsf{y}\ \mathsf{z} \xrightarrow{+} \mathsf{y} \wedge sl(\mathsf{y})] \wedge [\forall \mathsf{y}\ \mathsf{z} \xrightarrow{+} \mathsf{y} \Rightarrow (sl(\mathsf{y}) \vee dl(\mathsf{y}))]$, and $\psi_2(\mathsf{z}) \stackrel{\text{def}}{=} [\exists \mathsf{y}\ \mathsf{z} \xrightarrow{+} \mathsf{y} \wedge \forall \mathsf{y}\ \mathsf{z} \xrightarrow{+} \mathsf{y} \Rightarrow dl(\mathsf{y})]$. One can show that $\text{Reach}(N_1) = \text{Reach}(N_2)$ iff $\text{RG}(\overline{N}) \models \varphi$.

▶ **Proposition 4.1.** $\text{MC}^{\text{URG}}(\text{FO}(\xrightarrow{+}))$ is undecidable. Furthermore this results holds for the fixed formula φ defined earlier.

In order to prove undecidability of $\text{MC}^{\text{URG}}(\text{FO}(\xrightarrow{*}))$ we have to adapt our usual proof schema, since, in contrast with FO($\xrightarrow{+}$), we are no longer able to identify 1-loops.

▶ **Proposition 4.2.** $\text{MC}^{\text{URG}}(\text{FO}(\xrightarrow{*}))$ is undecidable.

Even though $\text{MC}^{\text{UG}}(\text{FO}(\rightarrow, =))$ is decidable (see Proposition 2.2), replacing $\rightarrow$ by $\xrightarrow{*}$ and adding *init* leads to undecidability.

▶ **Corollary 4.3.** $\text{MC}^{\text{UG}}(\text{FO}(init, \xrightarrow{*}))$ *is undecidable.*

Indeed, $\text{MC}^{\text{URG}}(\text{FO}(\xrightarrow{*}))$ reduces to $\text{MC}^{\text{UG}}(\text{FO}(init, \xrightarrow{*}))$ by relativization: $\text{URG}(N) \models \varphi$ iff $\text{UG}(N) \models \exists \mathsf{x}_0\ \ init(\mathsf{x}_0) \wedge f(\varphi)$ where φ and $f(\varphi)$ are in FO($\xrightarrow{*}$), f is homomorphic for Boolean connectives and $f(\forall\ \mathsf{x}\ \psi) \stackrel{\text{def}}{=} \forall\ \mathsf{x}\ (\mathsf{x}_0 \xrightarrow{*} \mathsf{x}) \Rightarrow f(\psi)$.

4.2 When semilinearity enters into the play

We saw that $\text{MC}^{\text{URG}}(\text{FO}(\rightarrow, =))$ restricted to Petri nets with effectively semilinear reachability sets is decidable (see Proposition 2.2), but it is unclear what happens if the relation $\xrightarrow{*}$ is added. We establish that $\text{MC}^{\text{URG}}(\text{FO}(\rightarrow, \xrightarrow{*}))$ restricted to Petri nets with semilinear reachability sets is undecidable, by a reduction from $\text{MC}^{\text{URG}}(\text{FO}(\rightarrow))$. Given a Petri net N and a sentence $\varphi \in \text{FO}(\rightarrow)$, we reduce the truth of φ in RG(N) to the truth of a formula $\overline{\varphi}$ in RG($\overline{N}$) with a semilinear reachability set. The Petri net $\overline{N}$ is defined from N by adding the new places p_0, p_1 and p_2; each transition from N is in self-loop with p_1. Moreover, we add a new set of transitions that are in self-loop with p_2 and that consist in adding or removing tokens from the original places of N (thus modifying its content arbitrarily). These transitions form a subnet denoted by Br. Three other transitions are added; see Figure 4.1 for a schematic representation of $\overline{N}$ (initial marking M_0' of $\overline{N}$ restricted to places in N is M_0 with $M_0'(p_0) = M_0'(p_1) = 1$ and $M_0'(p_2) = 0$). Our intention is to enforce Reach($\overline{N}$) to be semilinear while being able to identify a subset from Reach($\overline{N}$) that is in bijection with Reach(N); this is a way to drown Reach(N) into Reach($\overline{N}$). Indeed, Reach($\overline{N}$) contains all the markings such that the sum of p_1 and p_2 is 1 and p_0 is at most 1. Moreover, if the transition t is fired first, then the subsequently reachable markings are precisely those of N; RG(N) embeds isomorphically into RG($\overline{N}$). Until t is fired, one may always come back to M_0', using the brownian subnet Br, but this is impossible afterwards.

▶ **Proposition 4.4.** $\text{MC}^{\text{URG}}(\text{FO}(\rightarrow, \xrightarrow{*}))$ restricted to Petri nets with semilinear reachability sets is undecidable.

Proof. In a first stage, we use *init* although this predicate cannot be expressed in FO($\rightarrow, \xrightarrow{*}$). Let $\overline{\varphi}$ be the formula $\exists\ \mathsf{x}_0\ \mathsf{x}_1\ init(\mathsf{x}_0) \wedge \mathsf{x}_0 \rightarrow \mathsf{x}_1 \wedge \neg(\mathsf{x}_1 \xrightarrow{*} \mathsf{x}_0) \wedge f(\varphi)$ where $f(\cdot)$ is homomorphic for Boolean connectives and $f(\forall\ \mathsf{x}\ \psi) \stackrel{\text{def}}{=} \forall\ \mathsf{x}\ (\mathsf{x}_1 \xrightarrow{*} \mathsf{x}) \Rightarrow f(\psi)$ (relativization). In $\overline{\varphi}$, x_0 is interpreted as the initial marking, and x_1 is interpreted as a successor of x_0 from which x_0 cannot be reached again. This may only happen by firing t from M_0'. Now the relativization of every other variable to x_1 in $\overline{\varphi}$ ensures that $\text{RG}(N) \models \varphi$ iff $\text{RG}(\overline{N}) \models \overline{\varphi}$. To remove

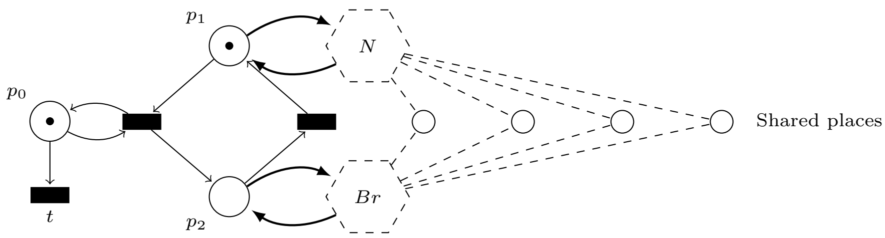

Figure 4.1 Petri net $\overline{N}$

init, we construct a Petri net $\overline{N}'$ similar to $\overline{N}$. $\overline{N}'$ has an extra place p_0', initially marked with one token, and a new transition that consumes this token and produces two tokens in p_0 and p_1, which were initially empty. By construction, the initial marking of $\overline{N}'$ is the sole marking in RG($\overline{N}'$) with no incoming edge and one outgoing edge. We use the formula $\overline{\varphi}' = \exists\, \mathsf{x}_0'\, \mathsf{x}_0\, \mathsf{x}_1\, (\neg\exists\, \mathsf{y}\, \mathsf{y} \to \mathsf{x}_0') \wedge \mathsf{x}_0' \to \mathsf{x}_0 \wedge \mathsf{x}_0 \to \mathsf{x}_1 \wedge (\neg \mathsf{x}_1 \xrightarrow{*} \mathsf{x}_0) \wedge f(\varphi)$. For the same reasons as above, RG(N) $\models \varphi$ iff RG($\overline{N'}$) $\models \overline{\varphi'}$. ◀

4.3 The reachability relation and the structure UG

Corollary 4.3 states a first undecidable result for UG. In this section we examine two other cases where the model checking of formulas in FO($\to, \xrightarrow{*}$) are undecidable for this structure.

▶ **Proposition 4.5.** $\mathrm{MC}^{\mathrm{UG}}(\mathrm{FO}(\to, \xrightarrow{*}))$ is undecidable.

Proposition 4.5 holds even when the reachable set of the net is effectively semilinear.

▶ **Proposition 4.6.** $\mathrm{MC}^{\mathrm{UG}}(\mathrm{FO}(\to, \xrightarrow{*}))$ is undecidable for classes of Petri nets having an effective semilinear reachability set.

In this section we have examined several first-order sublanguages involving the reachability predicate. We obtained undecidability results, even when the reachable markings form a semilinear set, and even when UG(N) is considered instead of URG(N)

5 Pattern Matching Problem

In this section, we do not consider the reachability graphs of Petri nets but their reachability sets (Reach(N)), plain subsets of $\mathbb{N}^n$ where n is the number of places of the net. In [17] the author characterizes such sets as *almost-semilinear* sets, a global property. On the opposite, we focus here on the shape of local neighborhoods by determining the existence of markings in $\mathbb{N}^n$ whose surrounding satisfies a specific pattern of reachable and non-reachable positions.

Using such patterns, one may check for instance whether there exist two reachable markings that differ only on a fixed place and by exactly one token.

A *pattern* $\mathcal{P}$ is defined as a map $[0, N_1] \times \cdots \times [0, N_n] \to \{\{\circ\}, \{\bullet\}, \{\circ, \bullet\}\}$ (values 'unreachable', 'reachable', 'dontcare'). A *constrained* position for $\mathcal{P}$ is an element of $[0, N_1] \times \cdots \times [0, N_n]$ with $\mathcal{P}$-image different from $\{\circ, \bullet\}$. Observe that patterns have the full dimension of the state space of the net. Each Petri net N with n (ordered) places induces a map $f_N : \mathbb{N}^n \to \{\{\circ\}, \{\bullet\}\}$ such that $f_N(M) = \{\bullet\}$ iff $M \in$ Reach(N). Given a Petri net N, a pattern $\mathcal{P}$ is *matched* by the net N at a point $\vec{v} \in \mathbb{N}^n$ if, for all $\vec{a} \in [0, N_1] \times \cdots \times [0, N_n]$, $f_N(\vec{v} + \vec{a}) \subseteq \mathcal{P}(\vec{a})$. A pattern $\mathcal{P}$ is *matched* by a Petri net N if it is matched by N at some point $\vec{v} \in \mathbb{N}^n$ (that may not be a reachable marking). The *Pattern Matching Problem* (PMP) is defined as follows: given a Petri net N and a pattern $\mathcal{P}$, is $\mathcal{P}$ matched by N?

Table 1 Summary

Problem	♯	Arbitrary	Effectively semilinear Reach(N)
MC$^\sharp$(FO($\to$))	URG UG	**UNDEC** (Theo. 3.2) DEC	**DEC** DEC
MC$^\sharp$(FO($\xrightarrow{+}$))	URG	**UNDEC** (Prop. 4.1)	open
MC$^\sharp$(FO($\xrightarrow{*}$))	URG	**UNDEC** (Prop. 4.2)	open
MC$^\sharp$(FO($\to, \xrightarrow{*}$))	URG UG	UNDEC **UNDEC** (Prop. 4.5)	**UNDEC** (Prop. 4.4) **UNDEC** (Prop. 4.6)
MC$^\sharp$(FO$^+$($\to$))	URG	**UNDEC** (Prop. 3.5)	DEC
MC$^\sharp$(FO$_f$($\to$))	URG	**UNDEC** (Prop. 3.6)	DEC
MC$^\sharp$($\exists$FO($\to, =$))	URG	**DEC** (Prop. 3.7)	DEC
MC$^\sharp$(FO($\to, =$))	UG	**DEC** (Prop. 2.2)	DEC
MC$^\sharp$(ML($\Box$))	URG	**PSpace-complete**	PSPACE-complete
MC$^\sharp$(ML($\Box, \Box^{-1}$))	URG	**DEC** (Prop. 2.4)	DEC
VAL$^\sharp$(ML($\Box, \Box^{-1}$))	URG	**UNDEC** (Prop. 3.4)	DEC
VAL$^\sharp$(PAML($\Box$))	URG	**DEC** (Prop. 3.8)	DEC
PMP	$\mathbb{N}^n$	**UNDEC** (Proposition 5.1)	**DEC** (Proposition 5.1)

▶ **Proposition 5.1.** (1) Let $\mathcal{C}$ be a class of Petri nets with effectively semilinear reachability sets. Then, PMP restricted to Petri nets in $\mathcal{C}$ is decidable. (2) PMP restricted to patterns with at most two constrained positions is undecidable.

Proposition 5.1(1) follows from the semilinearity of the set of marking satisfying patterns. To prove (2) we embed the reachable sets of two nets into two hyperplanes. Then these sets do not match iff there are two markings one reachable, the other not which may be encoded into a pattern. We use, here, a pattern with 2 adjacent, reachable and non-reachable, positions. It seems uneasy to prove this result using patterns having a single kind of constraints.

6 Concluding Remarks

We investigated mainly the model-checking problem over unlabelled reachability graphs of Petri nets with FO($\to$). The robustness of our main undecidability proof has been tested against standard fragments of FO($\to$), modal fragments, patterns and against the additional assumption that reachability sets are effectively semilinear. Table 1 provides a summary of the main results (observe that whenever the reachability relation is effectively semilinear, each problem is decidable). Results in bold are proved in the paper, whereas unbold ones are their consequences. Despite the quantity of results, a few rules of thumb can be synthesized: (1) undecidability of MC(FO($\to$)) is robust for several fragments of FO($\to$); (2) decidability results with simple restrictions such as considering bounded Petri nets or $\exists$FO($\to$) lead to computationally difficult problems (see Section 3.4); (3) the above points are still relevant for modal languages and patterns. Let us conclude by mentioning possible continuations of this work. Firstly, our taxonomy of results is partially incomplete.

New directions can also be followed. First, one could check geometrical properties of the reachability set Reach(N), e.g., the existence of an homogeneous ball around some reachable marking. Second, one could ask decidability questions about infinite unfoldings of nets in place of net reachability graphs. Such unfoldings may be shaped as trees if they may be local event structures [13]. With tree-unfoldings, labelling arcs (or nodes) is required if one wants to be able to express non-trivial properties, but then markings can be encoded to trees in which each arc represents one token being removed from a place identifies by the label of the arc. With event structure unfoldings, labelling an event e by a (sufficiently large) number k may always be simulated by adding k events triggered by e and in direct conflict with one another. In both cases, for obtaining decidable fragments of FO, one must avoid introducing any relation that would allow comparing for isomorphism two subtrees of two substructures triggered by two different events (like t^1_{dl} and t^2_{dl} in Fig. 3.1). The situation is different with *regular* trace event structures, although the substructure triggered by an event is characterized here by the label of this event. The decidability of FO over regular trace event structures has indeed been shown in [19]. However, *regular* trace event structures model safe Petri nets, whereas the model checking questions studied in this paper bear upon general and thus unbounded Petri nets.

Acknowledgments: We would like to thank the anonymous referees for helpful remarks and suggestions.

References

1. T. Araki and T. Kasami. Some decision problems related to the reachability problem for Petri nets. *TCS*, 3:85–104, 1976.
2. T. Araki and T. Kasami. Decidability problems on the strong connectivity of Petri net reachability sets. *TCS*, 4:99–119, 1977.
3. M. F. Atig and P. Habermehl. On Yen's path logic for Petri nets. *International Journal of Foundations of Computer Science*, 22(4):783–799, 2011.
4. P. Blackburn, M. de Rijke, and Y. Venema. *Modal Logic*. CUP, 2001.
5. A. Blumensath and E. Grädel. Automatic structures. In *LICS'00*, pages 51–62, 2000.
6. J. Esparza. Petri nets, commutative context-free grammars, and basic parallel processes. *Fundamenta Informaticae*, 31(13):13–26, 1997.
7. J. Esparza. Decidability and complexity of Petri net problems — an introduction. In *Advances in Petri Nets 1998*, volume 1491 of *LNCS*, pages 374–428. Springer, 1998.
8. S. Ginsburg and E. Spanier. Semigroups, Presburger formulas, and languages. *Pacific Journal of Mathematics*, 16:285–296, 1966.
9. P. Habermehl. On the complexity of the linear-time mu-calculus for Petri nets. In *ICATPN'97*, volume 1248 of *LNCS*, pages 102–116. Springer, 1997.
10. M. Hack. *Decidability Questions for Petri nets*. PhD thesis, MIT, 1975.
11. M. Hack. The equality problem for vector addition systems is undecidable. *TCS*, 2:77–96, 1976.
12. D. Hauschildt. Semilinearity of the reachability set is decidable for Petri nets. Technical Report FBI-HH-B-146/90, University of Hamburg, 1990.
13. P. W. Hoogers, H. C. M. Kleijn, and P. S. Thiagarajan. An event structure semantics for general Petri nets. *TCS*, 153:129–170, 1993.
14. R. Howell, P. Jančar, and L. Rosier. Completeness results for single-path Petri nets. *I & C*, 106(2):253–265, 1993.
15. P. Jančar. Undecidability of bisimilarity for Petri nets and some related problems. *TCS*, 148:281–301, 1995.
16. R. M. Karp and R. E. Miller. Parallel program schemata. *JCSS*, 3:147–195, 1969.
17. J. Leroux. Vector Addition System Reachability Problem (A Short Self-Contained Proof). In *POPL'11*, pages 307–316, 2011.
18. J. Leroux and G. Sutre. On Flatness for 2-Dimensional Vector Addition Systems with States. In *CONCUR'04*, volume 3170 of *LNCS*, pages 402–416. Springer, 2004.
19. P. Madhusudan. Model-checking trace event structures. In *LICS'03*, pages 371–380, 2003.
20. E. Mayr. An algorithm for the general Petri net reachability problem. *SIAM Journal of Computing*, 13(3):441–460, 1984.
21. S. Schulz. First-order logic with reachability predicates on infinite systems. In *FST&TCS'10*, pages 493–504. LIPICS, 2010.