The Microsoft Research - University of Trento
Centre for Computational
and Systems Biology

Technical Report CoSBi 22/2008

Decision problems for Spatial Logics revisited

Radu Mardare
The Microsoft Research-University of Trento Centre for Computational and Systems Biology, Trento, Italy
mardare@cosbi.eu

Alberto Policriti
Department of Mathematics and Computer Science, University of Udine
policriti@dimi.uniud.it

Decision problems for Spatial Logics revisited

Radu Mardare[1], Alberto Policriti[2,3]

[1]The Microsoft Research-University of Trento, Italy

[2]Department of Mathematics and Computer Science,University of Udine

[3]Applied Genomics Institute, Udine, Italy

September 30, 2009

Abstract

Spatial Logics are modal logics developed for process-algebraic semantics. They have been proposed for specifying concurrent properties of dynamic systems and have been proved useful in a wide range of applications. Their expresivity often comes with the price of undecidability, however, even against finite fragments of process calculi. This paper investigates the decidability of satisfiability, validity, and model checking for various Spatial Logics against semantics based on a fragment of CCS that embodies the core features of concurrent behaviors. We prove some decidability and undecidability properties for (combinations of) basic modal operators of spatial logics that entail some of the already known results in the field and provide a taxonomy for this class of problems.

1 Introduction

The success of Process Algebras [2] in modelling a wide class of concurrent and distributed systems from Computer Science and Artificial Intelligence to Systems Biology and Biochemistry, raises the necessity to develop analysis techniques for studying and predicting the behaviour of modelled systems. This is the origin of the idea of defining complex query languages specifically designed to express temporal and structural properties of the systems. The dual nature of these calculi—algebraic/equational syntax versus coalgebraic operational semantics, makes them particularly appropriate for a modal logic-based approach. The same idea is also sustained by the important results obtained by using temporal logics for specifying properties of dynamic systems.

In this context were proposed the process semantics for modal logics, which can be considered as a special case of Kripke semantics: it involves structuring a class of processes as a Kripke model by endowing it with accessibility relations and then using the standard clauses of Kripke semantics. The most obvious accessibility relations on processes are the ones induced by action transitions $\alpha.P\overset{\alpha}{\textbf{to}}P$ and thus the corresponding (Hennessy-Milner) logic [11] was the first process-based modal logic to be developed. Later, temporal [18], mobile, concurrent [9, 16] and dynamic-epistemic [12, 13] features were added.

A relatively new type of process logics are the *spatial logics* [7, 3], which are particularly tailored for capturing spatial and concurrent properties of processes. These are *intensional logics*, [17], able to differentiate bisimilar processes with different structures. Various versions of spatial logics for CCS, pi-calculus, ambient calculus, and other process algebras have been developed. The peculiarity of this class of logics consists in the presence of the *spatial operators*, which are logical counterparts of the program constructors of process calculi. Thus, we have a *parallel operator* meant to specify properties of complementary (parallel) modules of a program; a process P has the property $\phi|\psi$, if it can be split in two disjoint parts $P \equiv Q|R$ s.t. Q satisfies ϕ and R satisfies ψ. The *guarantee operator* $\triangleright$ is the adjoint of parallel and contains an implicit quantification over the possible contexts (characterized by some property) in which a process can evolve. The spatial operators are further combined with temporal or different types of dynamic operators expressing the transitions of the process. Thus, we can define operators for simple action-transitions or for communications. Some spatial logics defined for semantics on calculi with locations, such as ambient logic[1] [7], also contain operators for naming or quantifying over locations. Name passing and name restrictions in process calculi can be characterized logically too, by means of special operators [3].

The operators of spatial logics are similar to modal operators studied in other contexts. The parallel operator, for instance, is just a modal operator of arity 3 that satisfies the axioms of associativity, commutativity, and modal distribution [14]. Operators such as this have been studied, e.g., in the context of *Arrow Logic* [1] where it entails undecidability for Kripke semantics [10]. The parallel and the guarantee operators are also similar to operators of *Relevant* and *Substructural Logics* [19]—the *intentional conjunction* and *relevant implication* respectively. Similarities with operators of linear and intensional logics are also discussed in the literature [7].

In spite of the similarities with other logics, the particular combination of modal operators proposed by spatial logics raises genuinely new problems concerning decidability and complexity for satisfiability, validity, and model checking against the semantics on process algebras. The utility in applications of the analysis involving spatial logic-based specifications depends directly on the decidability and the computational costs of the analysis. In the literature the problems have been already approached, the state of the art in the field being more or less defined on the lines of [4, 8].

This paper reconsiders the problem of decidability for spatial logics reporting new results, some of them improving the state of art. The spatial logics we approach in this paper are given for semantics based on a fragment of CCS [15] that embodies the core features of finite concurrent behaviors. Spatial logics for the same semantics have been studied in [4] where it is proved that combining the parallel and the guarantee operators with a modality $\diamond\phi$ that encodes the communication-based transitions and/or with second order constructs (involving dynamic operators indexed by variables over actions, $\langle x\rangle\phi$, and quantifiers for them, $\exists x.\phi$) generate undecidability for both validity/satisfiability and model checking.

[1]Ambient logic is a spatial logic defined over ambient calculus.

We first prove that the undecidability of satisfiability for spatial logics with second order quantifiers derives from the undecidability of a more basic logic that contains second order quantifiers but does not contain the guarantee or the parallel operators. Only the expressive power of $\top|\phi$ is sufficient to generate, in this context, undecidability for satisfiability ($\top$ standing for the Boolean constant "true"). Moreover, the model-checking problem remains undecidable for any such logic, as long as it can express at least $\top \triangleright \phi$ (which is less expressive than the guarantee operator). On the other hand, we prove that the complete absence of guarantee makes the model checking decidable.

Concerning logics without second order constructions, we prove that, in the logic studied in [4], by replacing the operator $\diamond$ expressing a communication-based transition with a class of modal operators of type $\langle \alpha, \overline{\alpha} \rangle$, that expresses a communication by action α and its coaction, we obtain a decidable logic that includes the parallel and the guarantee operators. The same result is ensured by the replacing the $\diamond$ with the class $\langle \alpha \rangle$ of dynamic operators encoding atomic actions. We also show that, in absence of guarantee operator, the logic combining parallel and $\diamond$ has the model-checking problem decidable.

All these results are collected in Table 1 that provides a taxonomy for this class of problems. Before the results reported in this paper, only the cases PSL_4 and SOL_4 were known, [4]. For PSL_1 and SOL_0 the decidability of satisfiability against process semantics remain open problems.

Name	**Signature**	**Model checking**	**Satisfiability**
PSL_1	$\phi := 0, 1 \mid \neg\phi \mid \phi \wedge \phi \mid \phi\|\phi \mid \ \diamond \phi$	decidable	unknown
PSL_2	$\phi := 0, 1 \mid \neg\phi \mid \phi \wedge \phi \mid \phi\|\phi \mid \langle \alpha \rangle \phi \mid \phi \triangleright \phi$	decidable	decidable
PSL_3	$\phi := 0, 1 \mid \neg\phi \mid \phi \wedge \phi \mid \phi\|\phi \mid \langle \alpha, \overline{\alpha} \rangle \phi \mid \phi \triangleright \phi$	decidable	decidable
PSL_4	$\phi := 0, 1 \mid \neg\phi \mid \phi \wedge \phi \mid \phi\|\phi \mid \ \diamond \phi \mid \phi \triangleright \phi$	undecidable	undecidable
SOL_0	$\phi := 0, 1 \mid \neg\phi \mid \phi \wedge \phi \mid \ \exists x.\phi \mid \langle x \rangle \phi$	decidable	unknown
SOL_1	$\phi := 0, 1 \mid \neg\phi \mid \phi \wedge \phi \mid \top\|\phi \mid \exists x.\phi \mid \langle x \rangle \phi$	decidable	undecidable
SOL_2	$\phi := 0, 1 \mid \neg\phi \mid \phi \wedge \phi \mid \phi\|\phi \mid \exists x.\phi \mid \langle x \rangle \phi$	decidable	undecidable
SOL_3	$\phi := 0, 1 \mid \neg\phi \mid \phi \wedge \phi \mid \phi\|\phi \mid \exists x.\phi \mid \langle x \rangle \phi \mid \top \triangleright \phi$	undecidable	undecidable
SOL_4	$\phi := 0, 1 \mid \neg\phi \mid \phi \wedge \phi \mid \phi\|\phi \mid \exists x.\phi \mid \langle x \rangle \phi \mid \langle \overline{x} \rangle \phi \mid \ \diamond \phi \mid \phi \triangleright \phi$	undecidable	undecidable

Table 1: The decidability problems for spatial logics

2 Preliminaries on Process Algebra

In this section we recall a number of basic notions of process algebra, mainly to establish some basic terminology and notations. We introduce a finite fragment of CCS calculus that will later be used as semantics for spatial logics. This fragment is particularly interesting as it embodies the core features of finite concurrent behaviors (it does not contain recursion) and, as semantics for spatial logics, it is complex enough to cause undecidability for satisfiability/validity and model checking with respect to some relevant spatial logics.

Definition 2.1 (CCS processes) *Let Σ be a denumerable set of elements called* actions *and $0 \notin \Sigma$ a special object called the* null process. *The class $\mathbb{P}$ of* CCS processes *is introduced inductively for arbitrary $\alpha \in \Sigma$, as follows.*

$$P := 0 \mid \alpha.P \mid P|P.$$

Definition 2.2 (Structural congruence) *The structural congruence is the smallest congruence relation on $\mathbb{P}$ closed to the algebraical constructures such that $(\mathbb{P}, |, 0)$ is an abelian monoid with respect to $\equiv$, i.e.*

1. $(P|Q)|R \equiv P|(Q|R)$ 2. $P|0 \equiv P$ 3. $P|Q \equiv Q|P$
4. *If $P \equiv P'$, then for any $\alpha \in \Sigma$ and $Q \in \mathbb{P}$, $\alpha.P \equiv \alpha.P'$ and $P|Q \equiv P'|Q$.*

Definition 2.3 (Operational semantics) *Let $\tau \notin \Sigma \cup \mathbb{P}$ and consider a function on Σ that associates to each $\alpha \in \Sigma$ its* complementary action *$\overline{\alpha}$, such that $\overline{\overline{\alpha}} = \alpha$ and $\overline{\alpha} \neq \alpha$. The operational semantics on $\mathbb{P}$ defines a labeled transition system $\mathbb{T} : \mathbb{P} \to (\Sigma \cup \{\tau\}) \times \mathbb{P}$ by means of the rules in Table 2, where $\mathbb{T}(P) = (\alpha, Q)$ is denoted by $P\overset{\alpha}{\textbf{\textit{to}}}Q$ for any $\alpha \in \Sigma$, $\mathbb{T}(P) = (\tau, Q)$ is denoted by $P\overset{\tau}{\textbf{\textit{to}}}Q$, and μ is used to denote arbitrary elements in $\Sigma \cup \{\tau\}$.*

$$\alpha.P\overset{\alpha}{\textbf{to}}P \text{ , } \alpha \in \Sigma \qquad\qquad \alpha.P|\overline{\alpha}.Q\overset{\tau}{\textbf{to}}P|Q \text{ , } \alpha \in \Sigma$$

$$\begin{matrix} \text{P}\equiv Q \\ P\overset{\mu}{\textbf{to}}P' \; Q\overset{\mu}{\textbf{to}}P', \mu \in \Sigma \cup \{\tau\} \end{matrix} \qquad\qquad \text{P}\overset{\mu}{\textbf{to}}\,P' \; P|Q\overset{\mu}{\textbf{to}}P'|Q, \mu \in \Sigma \cup \{\tau\}$$

Table 2: The transition system

In this paper we consider, in addition, the transitions labeled by pairs of complementary actions $(\alpha, \overline{\alpha})$ and defined by $\alpha.P'|\overline{\alpha}.P''|P'''\overset{\alpha,\overline{\alpha}}{\textbf{to}}P'|P''|P'''$. This transition is still a communication but without the "anonymous status" that characterizes the τ transition. We will show this transition particularly interesting from a logical perspective.

Hereafter, we call a process P *guarded* if $P \equiv \alpha.Q$ for some $\alpha \in \Sigma$ and we use the notation $P^k \overset{def}{=} \underbrace{P|...|P}_{k}$ for $k \leq 1$.

Definition 2.4 *The* set of actions *$Act(P) \subset \Sigma$ of an arbitrary process $P \in \mathbb{P}$ is defined, inductively, as follows.*
$1)Act(0) \overset{def}{=} \emptyset$ $\;2)Act(\alpha.P) \overset{def}{=} \{\alpha\} \cup Act(P)$ $\;3)Act(P|Q) \overset{def}{=} Act(P) \cup Act(Q)$.

For a set $\Omega \subseteq \Sigma$ and a pair h, w of nonnegative integers we define the class $\mathbb{P}^{\Omega}_{(h,w)}$ of processes having the actions from Ω and the syntactic trees bound by two dimensions: the

depth h of the tree and the width w that represents the maximum number of structural congruent processes that can be found in a node of the tree. $\mathbb{P}^{\Omega}_{(h,w)}$ is introduced inductively on h.
$\mathbb{P}^{\Omega}_{(0,w)} = \{0\}$;
$\mathbb{P}^{\Omega}_{(h+1,w)} = \{(\alpha_1.P_1)^{k_1}|...|(\alpha_i.P_i)^{k_i},\text{ for } k_j \leq w, \alpha_j \in \Omega, P_j \in \mathbb{P}^{\Omega}_{(h,w)}, \forall j = 1..i\}$.

If $\Omega \subseteq \Sigma$ is a finite set, then $\mathbb{P}^{\Omega}_{(h,w)}$ is a finite set of processes.

Hereafter, we introduce *structural bisimulation*, a relation on processes similar to the *pruning relation* proposed for trees (static ambient processes) in [5]. This relation will play an essential role in establishing the bounded model property for some spatial logics. The structural bisimulation is indexed by a class $\Omega \subseteq \Sigma$ of actions and by two nonnegative integers h, w. Intuitively, two processes are Ω-*structural bisimilar* on size (h, w) if they look indistinguishable for an external observer that sees only the actions in Ω, does not follow a process for more than h transition steps, and cannot distinguish more than w cloned parallel subprocesses of an observed process.

Definition 2.5 (Ω-Structural Bisimulation) *Let $\Omega \subseteq \Sigma$ and h, w two nonnegative integers. The Ω-structural bisimulation on $\mathbb{P}$, denoted by $\approx^{\Omega}_{(h,w)}$, is defined inductively as follows.*
If $P \equiv Q \equiv 0$, then $P \approx^{\Omega}_{(h,w)} Q$;
If $P \not\equiv 0$ and $Q \not\equiv 0$, then
$P \approx^{\Omega}_{(0,w)} Q$ always.
$P \approx^{\Omega}_{(h+1,w)} Q$ iff for any $i \in 1..w$ and any $\alpha \in \Omega$:

- *$P \equiv \alpha.P_1|...|\alpha.P_i|P'$ implies $Q \equiv \alpha.Q_1|...|\alpha.Q_i|Q'$, $P_j \approx^{\Omega}_{(h,w)} Q_j$, $j = 1..i$;*
- *$Q \equiv \alpha.Q_1|...|\alpha.Q_i|Q'$ implies $P \equiv \alpha.P_1|...|\alpha.P_i|P'$, $Q_j \approx^{\Omega}_{(h,w)} P_j$, $j = 1..i$.*

We emphasize further some properties of Ω-structural bisimulation. The proofs of these results can be found in Appendix.

[Equivalence] For a set $\Omega \subseteq \Sigma$ and nonnegative integers h, w, $\approx^{\Omega}_{(h,w)}$ is an equivalence relations on $\mathbb{P}$.

[Congruence] Let $\Omega \subseteq \Sigma$ be a set of actions.
1. If $P \approx^{\Omega}_{(h,w)} Q$, then $\alpha.P \approx^{\Omega}_{(h+1,w)} \alpha.Q$.
2. If $P \approx^{\Omega}_{(h,w)} P'$ and $Q \approx^{\Omega}_{(h,w)} Q'$, then $P|Q \approx^{\Omega}_{(h,w)} P'|Q'$.

For nonnegative integers h, h', w, w' we convey to write $(h', w') \leq (h, w)$ iff $h' \leq h$ and $w' \leq w$.

Let $\Omega' \subseteq \Omega \subseteq \Sigma$ and $(h', w') \leq (h, w)$. If $P \approx^{\Omega}_{(h,w)} Q$, then $P \approx^{\Omega'}_{(h',w')} Q$.

[Split] If $P'|P'' \approx^{\Omega}_{(h,w_1+w_2)} Q$ for some $\Omega \subseteq \Sigma$, then there exists $Q, Q' \in \mathbb{P}$ such that $Q \equiv Q'|Q''$ and $P' \approx^{\Omega}_{(h,w_1)} Q'$, $P'' \approx^{\Omega}_{(h,w_2)} Q''$.

[Step-wise propagation] If $P \approx^{\Omega}_{(h,w)} Q$ and $P\overset{\alpha}{\textbf{to}}P'$ for some $\alpha \in \Omega \subseteq \Sigma$, then there exists a transition $Q\overset{\alpha}{\textbf{to}}Q'$ such that $P' \approx^{\Omega}_{(h-1,w-1)} Q'$; if $P\overset{\alpha,\overline{\alpha}}{\textbf{to}}P'$, then there exists a transition $Q\overset{\alpha,\overline{\alpha}}{\textbf{to}}Q'$ such that $P' \approx^{\Omega}_{(h-2,w-2)} Q'$.

As Σ is a denumerable set, assume a lexicographic order $\ll \subseteq \Sigma \times \Sigma$ on it. Then, any element $\alpha \in \Sigma$ has a successor denoted by $succ(\alpha)$ and any finite subset $\Omega \subset \Sigma$ has a maximum element denoted by $sup(\Omega)$. We define $\Omega^+ = \Omega \cup \{succ(sup(\Omega))\}$.

The next theorem states that for any finite set Ω of actions and any nonnegative integers h, w, the equivalence relation $\approx^{\Omega}_{(h,w)}$ partitions the class $\mathbb{P}$ of processes in equivalence classes such that each equivalence class has a representative in the set $\mathbb{P}^{\Omega+}_{(h,w)}$. This set, by Lemma 2, is finite.

[Representation Theorem] For any finite set $\Omega \subseteq \Sigma$, any nonnegative integers h, w and any process $P \in \mathbb{P}$, there exists a process $Q \in \mathbb{P}^{\Omega+}_{(h,w)}$ such that $P \approx^{\Omega}_{(h,w)} Q$.

3 Spatial Logic

In this section we introduce spatial logics. In the literature two classes of such logics have been studied. One class contains the *propositional spatial logics* (PSLs) that extend classic propositional logic with modal-spatial and dynamic operators. The other class consists of *second order spatial logics* (SOLs) that are equipped with variables and quantifiers over modalities which, by their nature, are second order variables and quantifiers. In this paper we study both classes.

Definition 3.1 (Syntax of Spatial Logics) *Let Σ and X be two disjoint denumerable sets. Consider the modal logics defined for the set $\{0,1\}$ of atomic proposition, for arbitrary $\alpha \in \Sigma$ and $x, y \in X$ as follows.*

$$\begin{array}{ll}
PSL_1 & \phi := 0, 1 \mid \neg\phi \mid \phi \wedge \phi \mid \phi|\phi \mid \diamond\phi \\
PSL_2 & \phi := 0, 1 \mid \neg\phi \mid \phi \wedge \phi \mid \phi|\phi \mid \langle\alpha\rangle\phi \mid \phi \triangleright \phi \\
PSL_3 & \phi := 0, 1 \mid \neg\phi \mid \phi \wedge \phi \mid \phi|\phi \mid \langle\alpha, \overline{\alpha}\rangle\phi \mid \phi \triangleright \phi \\
PSL_4 & \phi := 0, 1 \mid \neg\phi \mid \phi \wedge \phi \mid \phi|\phi \mid \diamond\phi \mid \phi \triangleright \phi \\
 & \\
SOL_0 & \phi := 0, 1 \mid \neg\phi \mid \phi \wedge \phi \mid \exists x.\phi \mid \langle x\rangle\phi \\
SOL_1 & \phi := 0, 1 \mid \neg\phi \mid \phi \wedge \phi \mid \top|\phi \mid \exists x.\phi \mid \langle x\rangle\phi \\
SOL_2 & \phi := 0, 1 \mid \neg\phi \mid \phi \wedge \phi \mid \phi|\phi \mid \exists x.\phi \mid \langle x\rangle\phi \\
SOL_3 & \phi := 0, 1 \mid \neg\phi \mid \phi \wedge \phi \mid \phi|\phi \mid \exists x.\phi \mid \langle x\rangle\phi \mid \top \triangleright \phi \\
SOL_4 & \phi := 0, 1 \mid \neg\phi \mid \phi \wedge \phi \mid \phi|\phi \mid \exists x.\phi \mid \langle x\rangle\phi \mid \langle\overline{x}\rangle\phi \mid \diamond\phi \mid \phi \triangleright \phi
\end{array}$$

The semantics of spatial logics is given over the class $\mathbb{P}$ of CCS processes taken as a frame. In particular, a definition of the satisfiability operator, $P \models \phi$ that relates a process $P \in \mathbb{P}$ with the property ϕ written in the syntax of $PSLs$, is given.

Definition 3.2 (Semantics of PSLs) *Let $P \in \mathbb{P}$ and ϕ a formula of PSL_i, $i = 1..4$. The relation $P \models \phi$ is defined inductively as follows.*

$P \models 0$ *iff* $P \equiv 0$.

$P \models 1$ *iff there exists* $\alpha \in \Sigma$ *such that* $P \equiv \alpha.P$.

$P \models \neg\phi$ *iff* $P \not\models \phi$.

$P \models \phi \wedge \psi$ *iff* $P \models \phi$ *and* $P \models \psi$.

$P \models \phi|\psi$ *iff* $P \equiv Q|R$, $Q \models \phi$ *and* $R \models \psi$.

$P \models \top|\phi$ *iff* $P \equiv Q|R$ *and* $R \models \phi$.

$P \models \diamond\phi$ *iff there exists a transition* $P\overset{\tau}{\textbf{\textit{to}}}P'$ *and* $P' \models \phi$.

$P \models \langle\alpha\rangle\phi$ *iff there exists a transition* $P\overset{\alpha}{\textbf{to}}P'$ *and* $P' \models \phi$.

$P \models \langle\alpha,\overline{\alpha}\rangle\phi$ *iff there exists a transition* $P\overset{\alpha,\overline{\alpha}}{\textbf{to}}P'$ *and* $P' \models \phi$.

$P \models \phi \triangleright \psi$ *iff for any* Q, $Q \models \phi$ *implies* $P|Q \models \psi$.

$P \models \top \triangleright \phi$ *iff for any* Q, $P|Q \models \psi$.

Observe that, equivalently, we can introduce the semantics in the modal logic fashion by defining a frame for PSLs as the structure

$\mathcal{M} = (\mathbb{P}, i, (\mathcal{R}_\alpha)_{\alpha\in\Sigma}, (\mathcal{R}_{(\alpha,\overline{\alpha})})_{\alpha\in\Sigma}, \mathcal{R}_\tau, \mathcal{R}_|, \mathcal{R}_\triangleright)$ where

$i : \mathbb{P} \to 2^{\{0\}}$ is the interpretation function defined by $i(P) = \{0\}$ for $P \equiv 0$ and $i(P) = \emptyset$ else;

$(\mathcal{R}_\alpha)_{\alpha\in\Sigma}$ is a class of accessibility relations $\mathcal{R}_\alpha \subseteq \mathbb{P} \times \mathbb{P}$ indexed by actions and defined by $(P, Q) \in \mathcal{R}_\alpha$ iff $P\overset{\alpha}{\textbf{to}}Q$.

$(\mathcal{R}_{(\alpha,\overline{\alpha})})_{\alpha\in\Sigma}$ is a class of accessibility relations indexed by pairs of complementary actions and defined by $(P, Q) \in \mathcal{R}_{(\alpha,\overline{\alpha})}$ iff $P\overset{\alpha,\overline{\alpha}}{\textbf{to}}Q$.

$\mathcal{R}_\tau$ is an accessibility relations $\mathcal{R}_\tau$ defined by $(P, Q) \in \mathcal{R}_\alpha$ iff $P\overset{\tau}{\textbf{to}}Q$.

$\mathcal{R}_| \subseteq \mathbb{P} \times \mathbb{P} \times \mathbb{P}$ is a relation defined by $(P, Q, R) \in \mathcal{R}_|$ iff $P \equiv Q|R$

$\mathcal{R}_\triangleright \subseteq \mathbb{P} \times \mathbb{P} \times \mathbb{P}$ is a relation defined by $(P, Q, R) \in \mathcal{R}_\triangleright$ iff $R \equiv P|Q$.

In this presentation $\langle\alpha\rangle$, $\langle\alpha,\overline{\alpha}\rangle$ and $\diamond$ are modal operators of arity 2, while $|$ and $\triangleright$ are modal operators of arity 3 having the semantics given as follows.

$\mathcal{M}, P \models \langle\alpha\rangle\phi$ iff there exists $P' \in \mathbb{P}$ s.t. $(P, P') \in \mathcal{R}_\alpha$ and $\mathcal{M}, P' \models \phi$.

$\mathcal{M}, P \models \langle\alpha,\overline{\alpha}\rangle\phi$ iff there exists $P' \in \mathbb{P}$ s.t. $(P, P') \in \mathcal{R}_{(\alpha,\overline{\alpha})}$ and $\mathcal{M}, P' \models \phi$.

$\mathcal{M}, P \models \langle\alpha\rangle\phi$ iff there exists $P' \in \mathbb{P}$ s.t. $(P, P') \in \mathcal{R}_\tau$ and $\mathcal{M}, P' \models \phi$.

$\mathcal{M}, P \models \phi|\psi$ iff there exist $Q, R \in \mathbb{P}$ s.t. $(P, Q, R) \in \mathcal{R}_|$, $\mathcal{M}, Q \models \phi$, $\mathcal{M}, R \models \psi$.

$\mathcal{M}, P \models \phi \triangleright \psi$ iff for $Q, R \in \mathbb{P}$ s.t. $(P, Q, R) \in \mathcal{R}_\triangleright$, $\mathcal{M}, Q \models \phi$ implies $\mathcal{M}, R \models \psi$.

In this interpretation the fact that $|$ and $\triangleright$ are adjoint operators is revealed by the fact that $(P, Q, R) \in \mathcal{R}_|$ if $(R, P, Q) \in \mathcal{R}_\triangleright$.

Before introducing the semantics of second order spatial logics (SOLs), we should stress the fact that in our syntax X is a set of variables that will be interpreted over Σ. As usual, we call an occurrence of a variable $x \in X$ in a formula ϕ (written in the syntax of SOL_i, $i = 0, \ldots, 4$) a *free occurrence* if it is not in the scope of a quantifier $\exists x$. We call a variable x a *free variable* in a formula if it has at least one free occurrence[2]. A formula ϕ is *closed* if it contains no free variables; else, we call it *open*. A valuation $v : X \hookrightarrow \Sigma$ is a partial function that associates values in Σ to some variables in X. If v is a valuation, $x \in X$ is a variable that is not in the domain of v, and $\alpha \in \Sigma$, we denote by $v_{\{x\to\alpha\}}$ the valuation v' that extends v with $v'(x) = \alpha$.

The semantics of second order spatial logics (SOLs) is given by the satisfiability operator, $P, v \models \phi$ that relates a process $P \in \mathbb{P}$ and valuation $v : X \to \Sigma$ interpreting the free variable of ϕ, to a well formed formula ϕ of SOL_i, $i = 0, \ldots, 4$.

Definition 3.3 (Semantics) *The relation* $P, v \models \phi$ *is defined inductively as follows.*

$P, v \models 0$ *iff* $P \equiv 0$.

[2]As usual, we assume that variables occurring under different boundaries or both bound and free do not clash, even if the same (meta) symbol $x \in X$ is used to name them.

$P, v \models 1$ *iff there exists* $\alpha \in \Sigma$ *such that* $P \equiv \alpha.P$.
$P, v \models \neg\phi$ *iff* $P, v \not\models \phi$.
$P, v \models \phi \wedge \psi$ *iff* $P, v \models \phi$ *and* $P, v \models \psi$.
$P, v \models \phi|\psi$ *iff* $P \equiv Q|R$, $Q, v \models \phi$ *and* $R, v \models \psi$.
$P, v \models \top|\phi$ *iff* $P \equiv Q|R$, $Q, v \models \phi$.
$P, v \models \diamond\phi$ *iff* $P \overset{\tau}{\textbf{to}} P'$ *and* $P', v \models \phi$.
$P, v \models \langle x\rangle\phi$ *iff* $P \overset{v(x)}{\textbf{to}} P'$ *and* $P', v \models \phi$.
$P, v \models \langle \overline{x}\rangle\phi$ *iff* $P \overset{\overline{v(x)}}{\textbf{to}} P'$ *and* $P', v \models \phi$.
$P, v \models \phi \triangleright \psi$ *iff for any process* $P', v \models \phi$ *implies* $P'|P, v \models \phi$.
$P, v \models \top \triangleright \phi$ *iff for any process* P', $P'|P, v \models \phi$.
$P, v \models \exists x.\phi$ *iff there exists* $\alpha \in \Sigma$ *such that* $P, v_{\{\alpha \to x\}} \models \phi$.

In addition to the boolean operators we also introduce the next derived operators that will be used both with PSLs and SOLs.

$$\top \overset{def}{=} 0 \vee \neg 0 \qquad \bot \overset{def}{=} \neg\top \qquad \phi \parallel \psi \overset{def}{=} \neg(\neg\phi|\neg\psi)$$
$$\circ\phi \overset{def}{=} (\neg\phi) \triangleright \bot \qquad \phi^{\forall} \overset{def}{=} \phi \parallel \top \qquad \alpha.\phi \overset{def}{=} 1 \wedge \langle\alpha\rangle\phi$$

In the light of the previous definition, $\top$ and $\bot$ are just the boolean constants, hence $\top|\phi$ and $\top \triangleright \phi$ are just particular cases of $\psi|\phi$ and $\psi \triangleright \phi$ respectively, defined for $\psi = \top$. Notice that in the logics where $\phi|\psi$ is a legal construction, 1 can be defined from 0 by $1 \overset{def}{=} \neg 0 \wedge (0 \parallel 0)$. Observe also that the operator $\circ$, definable in the logics where $\phi \triangleright \psi$ is a legal construction and is a universal modality, as $\circ\phi$ encodes the validity of ϕ over $\mathbb{P}$.

Definition 3.4 *A formula* ϕ *of PSLs is* satisfiable *if there exists a process* $P \in \mathbb{P}$ *such that* $P \models \phi$*; it is valid (a* validity*) if for any process* $P \in \mathbb{P}$, $P \models \phi$*. A closed formula* ϕ *of SOLs is* satisfiable *if there exists a process* $P \in \mathbb{P}$ *such that* $P, \emptyset \models \phi$*, where* $\emptyset$ *is the empty valuation; it is valid (a* validity*) if for any process* $P \in \mathbb{P}$ $P, \emptyset \models \phi$.

We denote the fact that ϕ is a validity by $\models \phi$. Hereafter, we call the **satisfiability problem** (**validity problem**) for a logic against a given semantics the problem of deciding if an arbitrary formula is satisfiable (valid). The **model checking problem** for PSLs consists in deciding, for an arbitrary formula ϕ and an arbitrary process P, if $P \models \phi$. The same problem for SOLs consists in deciding, for an arbitrary closed formula ϕ and an arbitrary process P, if $P, \emptyset \models \phi$.

Observe that Definition 3.4 implies that ϕ is a validity iff $\neg\phi$ is not satisfiable and reverse, ϕ is satisfiable iff $\neg\phi$ is not valid. Consequently, satisfiability and validity are dual problems implying that once one has been proved decidable/undecidable, the other shares the same property.

4 Decision problems for Second Order Spatial Logics

In [4] it is proved that SOL_4 is undecidable. The proof is based on the method proposed previously in [8] where it is shown that the second order quantifiers (over ambient names) in ambient logic, in combination with the parallel operator, can induce undecidability for satisfiability. A corollary of this result is the undecidability of SOL_2. In what follows, we use the same method for proving a stronger result, i.e. that satisfiability for SOL_1 is undecidable. This result shows that even in absence of the parallel operator (as in SOL_1, parallel can only appear in constructions of type $\top|\phi$) second order quantification produces undecidability and implies the undecidability of satisfiability for SOL_2, SOL_3 and SOL_4.

In the second part of this section we will also approach the model checking problem and prove that the situation is different. For SOL_2 the model-checking problem is decidable (implying decidability of model checking for both SOL_1 and SOL_0), while for SOL_3 model checking is undecidable (implying the undecidability of model checking for SOL_4). This shows that we do not need the entire expressivity of the guarantee operator in order to reach undecidability of model checking: the presence of $\top \triangleright \phi$ is sufficient. Notice that $P, v \models \top \triangleright \phi$ is equivalent to say that all processes having P as subprocess have the property ϕ under the evaluation v, i.e. we face a universal quantification on the class of upper processes of P. The satisfiability problem for SOL_0 remains as an open problem.

4.1 The satisfiability problem

In what follows, we prove that the satisfiability problem of SOL_1 is equivalent with the satisfiability problem of a fragment of first order logic known to be undecidable for finite domains[3]. This fragment is FOL introduced inductively, for a single binary predicate $p(x, y)$ and for $x, y \in X$, by:

$$f := p(x, y) \mid \neg f \mid f \wedge f \mid \exists x.f.$$

The semantics of FOL is defined for a finite domains $D \subseteq \Sigma$ and for an interpretation $I \subseteq D \times D$ of the predicate. The satisfiability relation is given as follows (for valuations $v : X \to D$).

$(D, I), v \models p(x, y)$ iff $(v(x), v(y)) \in I$

$(D, I), v \models \neg f$ iff $(D, I), v \not\models f$

$(D, I), v \models f \wedge g$ iff $(D, I), v \models f$ and $(D, I), v \models g$

$(D, I), v \models \exists x.f$ iff there exists $\alpha \in D$ and $(D, I), v_{\{x \to \alpha\}} \models f$.

It is known that satisfiability for FOL is undecidable. We will prove further that satisfiability of FOL is equivalent with satisfiability for SOL_1.

We begin by describing a special class $\mathcal{P} \subseteq \mathbb{P}$ of processes that can be characterized by the formulas of SOL_1.

Consider the following derived operators in SOL_1:

$D(x) = \langle x \rangle 0$, $R(x, y) = 1 \wedge \langle x \rangle \langle y \rangle 0$ and

$$Model = [(1 \to (\exists x D(x) \vee \exists x \exists y R(x, y)))|\top] \wedge [\forall x \forall y((R(x, y)|\top) \to (D(x)|\top \wedge D(y)|\top))]$$

We prove that the formula $Model$ characterizes all processes of type $\alpha_1.0|...|\alpha_k.0 \mid \alpha_{i_1}.\alpha_{j_1}.0|...|\alpha_{i_l}.\alpha_{j_l}.0$ for $i_1, ..., i_l, j_1, .., j_l \in \{1, ..k\}$ and the process 0. This class, which we denote by $\mathcal{P}$, will play a major role in our proof.

[3] The same fragment of first order logic is used in [4] for proving the undecidability of SOL_4.

$P, v \models Model$ iff either $P \equiv 0$, or there exist actions $\alpha_1, ...\alpha_k \in \Sigma$, and $i_1, ..., i_l, j_1, .., j_l \in \{1, ..k\}$ s.t. $P \equiv \alpha_1.0|...|\alpha_k.0 \mid \alpha_{i_1}.\alpha_{j_1}.0|...|\alpha_{i_l}.\alpha_{j_l}.0$

Proof $P, v \models (1 \to (\exists x D(x) \vee \exists x \exists y R(x, y)))|\top$ iff for any Q s.t. $P \equiv Q|R$ we have that if $Q \models 1$ (i.e. $Q \equiv \alpha.Q'$ for some α), then $Q' \equiv 0$ or $Q' \equiv \beta.0$. Hence, $Q \models 1$ implies $Q \equiv \alpha.0$ or $Q \equiv \alpha.\beta.0$ for some $\alpha, \beta \in \Sigma$. Moreover, $P, v \models \forall x \forall y((R(x, y)|\top) \to (D(x)|\top \wedge D(y)|\top))$ iff $P \equiv \alpha.\beta.0|Q$ implies $P \equiv \alpha.0|\beta.0|P'$. □

Now, we describe a method for associating to each pair (D, I) used in the semantics of FOL, a process $P_D^I \in \mathcal{P}$.

Let $D \subseteq \Sigma$ be a finite set and $I \subset D \times D$ a relation on D. Suppose that $D = \{\alpha_1, ...\alpha_k\}$ with $k \geq 1$, and $I = \{(\alpha_{i_1}, \alpha_{j_1}), (\alpha_{i_2}, \alpha_{j_2}), ..., (\alpha_{i_l}, \alpha_{j_l})\}$, with $i_1, ..., i_l, j_1, .., j_l \in \{1, ..k\}$. We denote by $Dom(\Sigma)$ the class of these pairs (D, I). We associate to each pair $(D, I) \in Dom(\Sigma)$ the process $P_D^I \in \mathcal{P}$ defined by

$$P_D^I \equiv \alpha_1.0|...|\alpha_k.0 \mid \alpha_{i_1}.\alpha_{j_1}.0|...|\alpha_{i_l}.\alpha_{j_l}.0.$$

Reverse, consider a process $P \in \mathcal{P}$ for which there exists $\alpha_1, ...\alpha_k \in \Sigma$, not necessarily distinct, and $i_1, ..., i_l, j_1, .., j_l \in \{1, ..k\}$ s.t.

$$P \equiv \alpha_1.0|...|\alpha_k.0 \mid \alpha_{i_1}.\alpha_{j_1}.0|...|\alpha_{i_l}.\alpha_{j_l}.0.$$

We take $D = \{\alpha_1, ...\alpha_k\}$ and $I = \{(\alpha_{i_1}, \alpha_{j_1}), (\alpha_{i_2}, \alpha_{j_2}), ..., (\alpha_{i_l}.\alpha_{j_l})\}$ and this is the pair we associate to P. Notice that, by construction, if $\alpha_i = \alpha_j$ then it appears in D only once and similarly, if $(\alpha_{i_s}, \alpha_{j_s}) = (\alpha_{i_t}, \alpha_{j_t})$ for some $s \neq t$, then it is taken only once in I.

For proving the equivalence between the two decidability problems, we define the encoding [ ] that associates each formula of FOL to a formula of SOL_1, inductively as follows.

$[p(x, y)] = R(x, y)|\top$
$[\neg f] = \neg[f]$
$[f \wedge g] = [f] \wedge [g]$
$[\exists x.f] = \exists x.((D(x)|\top) \wedge [f])$
$(D, I), v \models f$ iff $P_I^D, v \models [f]$.

Proof We prove it by induction on $f \in FOL$.

The case $f = \neg g$**:** $(D, I), v \models \neg g$ iff $(D, I), v \not\models g$. Then the inductive hypothesis gives $P_I^D, v \not\models [g]$ which is equivalent with $P_I^D, v \models \neg[g]$.

The case $f = g \wedge h$**:** $(D, I), v \models g \wedge h$ iff $(D, I), v \models g$ and $(D, I), v \models h$. Using the inductive hypothesis, we obtain $P_I^D, v \models [g]$ and $P_I^D, v \models [h]$, i.e. $P_I^D, v \models [g] \wedge [h]$, meaning that $P_I^D, v \models [f]$.

The case $f = \exists x.g$**:** $(D, I), v \models \exists x.g$ iff there exists $\alpha \in D$ s.t. $(D, I), v_{\{x \to \alpha\}} \models g$. Further, the inductive hypothesis gives $P_I^D, v_{\{x \to \alpha\}} \models [g]$. But because $P_I^D \equiv \alpha.0|P'$, we obtain that $P_I^D, v_{\{x \to \alpha\}} \models D(x)|\top$. Hence $P_I^D, v_{\{x \to \alpha\}} \models D(x)|\top \wedge [g]$ that implies $P_I^D, v \models \exists x.(D(x)|\top \wedge [g])$ that is equivalent with $P_I^D, v \models [f]$.

The case $f = p(x, y)$**:** $(D, I), v \models p(x, y)$ iff $(v(x), v(y)) \in I$. But this is equivalent with $P_I^D \equiv v(x).v(y).0|P'$ that implies $P_I^D, v \models (1 \wedge \langle x \rangle \langle y \rangle.0)|\top$, i.e. $P_I^D, v \models [f]$. □

Let f be a closed formula of FOL. Then f is satisfiable in FOL iff $Model \wedge [f]$ is satisfiable in SOL_1.

Proof $Model$ characterizes the class $\mathcal{P}$ of processes. So, if there exists a model $(D, I) \in Dom(\Sigma)$ such that $(D, I), \emptyset \models f$, then $P_I^D, \emptyset \models Model \wedge [f]$, where $\emptyset$ is the empty valuation. Reverse, if there is a process $P \in \mathbb{P}$ that satisfies $Model \wedge [f]$, then $P, \emptyset \models Model$, i.e. $P \in \mathcal{P}$ meaning that there exists $(D, I) \in Dom(\Sigma)$ such that $P \equiv P_I^D$. Then $P_I^D, \emptyset \models [f]$ that implies $(D, I), \emptyset \models f$. $\Box$

Theorem 4.1 *For SOL_1 validity and satisfiability are undecidable.*

This result implies the undecidability of satisfiability for all the more expressive logics.

Corollary 4.1 *The satisfiability is undecidable for SOL_2 and SOL_3.*

4.2 The model-checking problem

With model checking the situation is different. The simple presence of second order quantification does not imply undecidability, as for the case of satisfiability.

Theorem 4.2 *For SOL_2 model checking is decidable.*

Proof For the beginning, observe that for arbitrary P, ϕ, v' and $\alpha, \beta \notin Act(P) \cup dom(v')$, we have $P, v'_{\{x \to \alpha\}} \models \phi$ iff $P, v'_{\{x \to \beta\}} \models \phi$. Due to this property, it can be proved that in deciding $P, v \models \phi$ it is sufficient to consider only valuations assigning values from $Act(P) \cup \{\alpha_1, ..., \alpha_k\}$ to free variables of ϕ, where $\alpha_i \notin Act(P)$ for $i = 1..k$ and k is the number of distinct variables[4] that appear in ϕ. Further, it is easy to prove that operators can be eliminated, inductively, and the model-checking problem can be reduced, at each step, to a finite number of model-checking problems involving subprocesses of P (which are finitely many, modulo structural congruence) and subformulas of ϕ. $\Box$

Corollary 4.2 *The model-checking problems for SOL_1 and SOL_0 are decidable.*

The presence in a logic of the operator $\top \triangleright \phi$ is sufficient to turn the model-checking problem undecidable. Notice that $P, v \models \top \triangleright \phi$ involves a universal quantification over the class of upper processes of P.

Theorem 4.3 *For SOL_3 model checking is undecidable.*

Proof The proof is based on the observation, emphasized also in [8], that in any logic which can express $\top \triangleright \phi$, the decidability of model-checking problem implies the decidability of satisfiability. Hence, the undecidability of satisfiability implies undecidability of model checking. Indeed, for an arbitrary formula ϕ in SOL_3, it is trivial to verify that $\models \phi$ iff $0, \emptyset \models \top \triangleright \phi$. As for SOL_3 validity is undecidable, we obtain undecidability for model checking. $\Box$

[4]This number can be defined inductively on the syntax of ϕ.

5 Decision problems for Propositional Spatial Logics

In this section we focus on propositional spatial logics. In [4] it has been proved that for PSL_4 satisfiability, validity and model checking are both undecidable against the semantics presented in Section 3. The proof is based on an equivalence between the satisfiability problem for SOL_5 and satisfiability problem of PSL_4. The result reveals that the combination of the modality $\diamond$ based on τ-transition with the spatial operators $|$ and $\triangleright$, generates undecidability.

In this section we show that the combination of the two spatial operators and a transition-based modality does not always produce undecidability. We will show that for PSL_2 and PSL_3 both the satisfiability/validity and model-checking problems are decidable. PSL_2 contains dynamic operators indexed by actions $\langle\alpha\rangle$ that reflect the interleaving semantics of CCS. PSL_3 is closer to PSL_4 as it expresses communications by the dynamic operators $\langle\alpha,\overline{\alpha}\rangle$. But while the communications reflected by $\diamond$ have an anonymous status, the communications expressible in PSL_3 can also specify the pairs of actions involved. Observe that $\diamond$ can be seen as an existential quantifier over the class of $\langle\alpha,\overline{\alpha}\rangle$ and in this sense PSL_4 has a second-order nature that might explain its undecidability.

In the second part of the section we consider the logic PSL_1 which combines the communication-based modal operator $\diamond$ with the parallel operator. We prove that for this logic the model-checking problem is decidable. The satisfiability for PSL_1 remains as an open problem.

5.1 Decidability of PSL_2 and PSL_3

In this subsection we prove that for PSL_2 and PSL_3 satisfiability/validity problems are decidable. The proofs are based on the *bounded model property* technique which consists in showing that, given a formula ϕ of PSL_2 or PSL_3, we can identify a finite class of processes $\mathbb{P}_\phi$, bound by the dimension of ϕ, such that if ϕ has a model in $\mathbb{P}$, then it has a model in $\mathbb{P}_\phi$. Thus, the satisfiability problem in $\mathbb{P}$ is equivalent with the satisfiability in $\mathbb{P}_\phi$. This result can be further used to prove the decidability of satisfiability for the two logics. Indeed, as $\mathbb{P}_\phi$ is finite, checking the satisfiability of a formula can be done by exhaustively verifying it for all the processes in $\mathbb{P}_\phi$.

The method adapted for spatial logics was first proposed in [6] and reused in [5] for the case of static ambient logic. It consists in identifying a structural equivalence on processes, sensitive to the dimension of the logical formulas, that relates two processes whenever they satisfy the same formulas of a given size. In our case this relation is the structural bisimulation defined in Section 2.

Definition 5.1 (Size of a formula) *The* sizes of a formula *of PSL_3, denoted by $\phi = (h, w)$, is defined inductively on the structure of a formula. In what follows, suppose that $\phi = (h, w)$ and $\psi = (h', w')$.*

1. $0 \stackrel{def}{=} (1, 1)$.
2. $\neg\phi \stackrel{def}{=} \phi$.
3. $\phi \wedge \psi \stackrel{def}{=} (max(h, h'), max(w, w'))$.
4. $\langle\alpha\rangle\phi \stackrel{def}{=} (h + 1, w + 1)$.
5. $\phi \triangleright \psi \stackrel{def}{=} (max(h, h'), w + w')$.
6. $\phi|\psi \stackrel{def}{=} (max(h, h'), w + w')$.
7. $\langle\alpha, \overline{\alpha}\rangle\phi \stackrel{def}{=} (h + 2, w + 2)$.

Definition 5.2 *The set of actions of a formula ϕ, $act(\phi) \subseteq \Sigma$ is given by:*

1. $act(0) \stackrel{def}{=} \emptyset$ 2. $act(\neg\phi) = act(\phi)$
3. $act(\phi \wedge \psi) \stackrel{def}{=} act(\phi) \cup act(\psi)$ 4. $act(\langle\alpha\rangle\phi) \stackrel{def}{=} \{\alpha\} \cup act(\phi)$
5. $act(\phi \triangleright \psi) \stackrel{def}{=} act(\phi) \cup act(\psi)$ 6. $act(\phi|\psi) \stackrel{def}{=} act(\phi) \cup act(\psi)$
7. $act(\langle\alpha, \overline{\alpha}\rangle\phi) \stackrel{def}{=} \{\alpha, \overline{\alpha}\} \cup act(\phi)$.

The next Lemma states that a formula ϕ of PSL_2 or PSL_3 expresses a property of a process P up to $\approx_{\phi}^{act(\phi)}$. A sketch of its proof can be found in Appendix.

The next assertion is true for PSL_2 and PSL_3.
If $P \approx_{\phi}^{act(\phi)} Q$, then $P \models \phi$ iff $Q \models \phi$.

This result guarantees the bounded model property for both PSL_2 and PSL_3.

Theorem 5.1 (Bounded model property) *The next assertion is true for PSL_2 and PSL_3. If $P \models \phi$, then there exists $Q \in \mathbb{P}_{\phi}^{act(\phi)^+}$ such that $Q \models \phi$.*

Proof The results are direct consequences of Lemma 2 and Lemma 5.1. □

Theorem 5.2 (Decidability) *For PSL_2 and PSL_3 validity and satisfiability are decidable against process semantics.*

Proof The decidability of satisfiability derives, for both logics, from the bounded model property. Indeed, if ϕ has a model, by Lemma 5.1, it has a model in $\mathbb{P}_{\phi}^{act(\phi)^+}$. As $act(\phi)$ is finite, by Lemma 2, $\mathbb{P}_{\phi}^{act(\phi)^+}$ is finite. Hence, checking for membership is decidable.

The decidability of validity derives from the fact that ϕ is valid iff $\neg\phi$ is not decidable. □

5.2 The model-checking problems

We focus now on the model-checking problems. We start by stating the decidability of model checking for PSL_2 and PSL_3.

Theorem 5.3 *For PSL_2 and PSL_3 model checking is decidable against process semantics.*

Proof Given the process P and the formula ϕ, we show inductively on the structure of ϕ that $P \models \phi$ is decidable, by showing that the problem can be reduced, step by step, to a finite number of model checking problems involving subformulas of ϕ. The only interesting case is $\phi = \phi_1 \triangleright \phi_2$. Due to the bounded model property, $P \models \phi_1 \triangleright \phi_2$ iff for any $Q \in \mathbb{P}_{\phi_1}^{act(\phi_1)^+}$ we have that $Q \models \phi_1$ implies $P|Q \models \phi_2$. As there are only a finite number of processes $Q \in \mathbb{P}_{\phi_1}^{act(\phi_1)^+}$, we are done. □

Theorem 5.4 *For PSL_1 model checking is decidable against process semantics.*

Proof As before, we reduce the problem $P \models \phi$ to a finite number of model checking problems involving subprocesses of P (as in this case we do not have $\triangleright$) and subformulas of ϕ. The only difference w.r.t. PSL_2 or PSL_3 is case $\phi = \diamond\psi$. We have $P \models \diamond\psi$ iff there exists a transition $P\overset{\tau}{\textbf{to}}P'$ such that $P' \models \psi$. But the number of processes P' such that $P\overset{\tau}{\textbf{to}}P'$ is finite modulo structural congruence. Hence, also in this case, the problem can be reduced to a finite number of model checking problems that refers to ψ. □

6 Conclusive remarks

Our original goal was to complete Table 1 and, save the decidability of satisfiability for PSL_1 and SOL_0, we succeeded. The results reported here improve some known results, e.g. the undecidability of satisfiability of SOL_1 explains the known undecidability of SOL_4 reported in [4] (and of SOL_4 anticipated in [8]) and the undecidability of model checking of SOL_3 is liked to the undecidability of model checking for SOL_4.

The results on the decidability of propositional spatial logics proved in this paper are, to the best of our knowledge, original. The fact that PSL_3 is decidable shows that the communication in combination with spatial operators can still be used without losing decidability. The decidability of SOL_2 is useful for applications in which interleaving semantics is relevant.

Notice that in light of Table 1, the undecidability of satisfiability seems generated either by the combination of second order quantifiers with $\top|\phi$, or by the combination of $\diamond$ and $\triangleright$. Undecidability of model checking seems generated by the presence of $\top \triangleright \phi$ in the context of undecidable satisfiability.

References

[1] J. van Benthem, Language in action. Categories, Lambdas and Dynamic Logic, Elsevier Science Publisher, 1991

[2] J.A. Bergstra, A. Ponse, S.A. Smolka (eds.), Handbook of Process Algebra, North Holland, Elsevier, 2001.

[3] L. Caires and L. Cardelli. A Spatial Logic for Concurrency (Part I), Information and Computation vol. 186/2, 2003

[4] L. Caires and E. Lozes, Elimination of Quantifiers and Decidability in Spatial Logics for Concurrency, In Proc. of CONCUR'2004, LNCS vol.3170, 2004

[5] C. Calcagno, L. Cardelli and A. D. Gordon, Deciding validity in a spatial logic for trees, Journal of Functional Programming, Vol. 15, 2005

[6] C.Calcagno, et al. Computability and complexity results for a spatial assertion language for data structures, In Proc. of FSTTCS01, LNCS, vol. 2245, 2001

[7] L. Cardelli and A. D. Gordon. Anytime, Anywhere: Modal Logics for Mobile Ambients, In Proc. 27th ACM Symposium on Principles of Programming Languages, 2000

[8] W. Charatonik, J.M. Talbot, The decidability of model checking mobile ambients, In Proc. of the 15th International Workshop on Computer Science Logic, LNCS vol.2142, 2001

[9] M. Dam, Model checking mobile processes, Information and Computation vol.129(1), 1996

[10] V. Gyuris, Associativity does not imply undecidability without the axiom of Modal Distribution, In M. Marx, et.al eds., Arrow Logic and Multi-Modal Logic, CSLI and FOLLI, 1996

[11] M. Hennessy and R. Milner, Algebraic laws for Nondeterminism and Concurrency, Journal of JACM vol. 32(1), 1985

[12] R. Mardare, Observing distributed computation. A dynamic-Epistemic approach, In Proc. CALCO'07, LNCS vol.4624, 2007

[13] R. Mardare and C. Priami, Decidable extensions of Hennessy-Milner Logic, In Proc. FORTE'06, LNCS vol.4229, 2006

[14] R. Mardare, A. Polocriti, Towards a complete axiomatization for Spatial Logics, TechRep. CoSBi, TR-03-2008, reachable from www.cosbi.eu

[15] R. Milner, A Calculus of Communicating Systems, Springer-Verlag New York, Inc., 1982

[16] R. Milner, J. Parrow and D. Walker, Modal logics for mobile processes, TCS vol.114, 1993

[17] D. Sangiorgi, Extensionality and Intensionality of the Ambient Logics, In Proc. of the 28th ACM Annual Symposium on Principles of Programming Languages, 2001

[18] C. Stirling, Modal and temporal properties of processes, Springer-Verlag New York, Inc., 2001

[19] A. Urquhart, Semantics for Relevant Logics, Journal of Symbolic Logic, 37(1), 1972

Appendix

In this appendix we present some of the proofs of the main lemmas presented in the paper.

Proof [Proof of Lemma 2]

2. We prove it by induction on h. The case $h = 0$ is immediate.
For the case $h+1$, suppose that $P \approx^{\Omega}_{(h+1,w)} P'$ and $Q \approx^{\Omega}_{(h+1,w)} Q'$.

Consider any $i = 1..w$, and any $\alpha \in \Omega$ such that $P|Q \equiv \alpha.R_1|...|\alpha.R_i|R_{i+1}$. Suppose, without loss of generality, that R_j are ordered in such a way that there exist $k \in 1..i$, P'', Q'' such that $P \equiv \alpha.R_1|...|\alpha.R_k|P''$, $Q \equiv \alpha.R_{k+1}|...|\alpha.R_i|Q''$ and $R_{i+1} \equiv P''|Q''$. Because $k \in 1..w$, from $P \approx^{\Omega}_{(h+1,w)} P'$ we have $P' \equiv \alpha.P'_1|...|\alpha.P'_k|P_0$ such that $R_j \approx^{\Omega}_{(h,w)} P'_j$ for $j = 1..k$. Similarly, from $Q \approx^{\Omega}_{(h+1,w)} Q'$ we have $Q' \equiv \alpha.Q'_{k+1}|...|\alpha.Q'_i|Q_0$ such that $Rj \approx^{\Omega}_{(h,w)} Q'_j$ for $j = (k+1)..i$. Hence, $P'|Q' \equiv \alpha.P'_1|...|\alpha.P'_k|\alpha.Q'_{k+1}|...|\alpha.Q'_i|P_0|Q_0$ with $R_j \approx^{\Omega}_{(h,w)} P'_j$ for $j = 1..k$ and $R_j \approx^{\Omega}_{(h,w)} Q'_j$ for $j = (k+1)..i$.

□

Proof [Proof of Lemma 2]

We prove it by induction on h. The case $h = 0$ is trivial.
The case $h+1$: Suppose that $P'|P'' \approx^{\Omega}_{(h+1,w)} Q$. Let $w = w_1 + w_2$.

Following an idea proposed in [5], we say that a process P is in $\Omega_{(h,w)}$*-normal form* if whenever $P \equiv \alpha_1.P_1|\alpha_2.P_2|P_3$ for $\alpha_1, \alpha_2 \in \Omega$ and $P_1 \approx^{\Omega}_{(h,w)} P_2$ then $P_1 \equiv P_2$. Note that $P \approx^{\Omega}_{(h+1,w)} \alpha_1.P_1|\alpha_2.P_1|P_3$. This shows that for any P, any Ω and any (h,w) we can find a P_0 such that P_0 is in (h,w)-normal form and $P \approx^{\Omega}_{(h+1,w)} P_0$.

We can suppose, without loosing generality, that the canonical representations of P', P'' and Q are[5]: $P' \equiv (\alpha_1.P_1)^{k'_1}|...|(\alpha_n.P_n)^{k'_n}|P_1$, $P'' \equiv (\alpha_1.P_1)^{k''_1}|...|(\alpha_n.P_n)^{k''_n}|P_2$ and $Q \equiv (\alpha_1.P_1)^{l_1}|...|(\alpha_n.P_n)^{l_n}|Q_1$, where P_1, P_2, Q_1 have all the guarded subprocesses prefixed by actions that are not in Ω. For each $i \in 1..n$, we split $l_i = l'_i + l''_i$ in order to obtain a splitting of Q. We define the splitting of l_i such that $(\alpha_i.P_i)^{k'_i} \approx_{h+1,w_1} (\alpha_i.P_i)^{l'_i}$ and $(\alpha_i.P_i)^{k''_i} \approx_{h+1,w_2} (\alpha_i.P_i)^{l''_i}$. We do this as follows:

If $k'_i + k''_i < w_1 + w_2$ then $P'|P'' \approx^{w}_{h+1} Q$ implies $l_i = k'_i + k''_i$, so we can choose $l'_i = k'_i$ and $l''_i = k''_i$.

If $k'_i + k''_i \geq w_1 + w_2$ then $P'|P'' \approx^{w}_{h+1} Q$ implies $l_i \geq w_1 + w_2$. We meet the following subcases:

- $k'_i \geq w_1$ and $k''_i \geq w_2$. We choose $l'_i = w_1$ and $l''_i = l_i - w_1$ (note that as $l_i \geq w_1 + w_2$, we have $l''_i \geq w_2$).
- $k'_i < w_1$, then we must have $k''_i \geq w_2$. We choose $l'_i = k'_i$ and $l''_i = l_i - k'_i$. So $l''_i \geq w_2$ as $l_i \geq w_1 + w_2$ and $l'_i < w_1$.
- $k''_i < w_2$ is similar with the previous one. We choose $l''_i = k''_i$ and $l'_i = l_i - k''_i$.

Now, for $Q' \equiv (\alpha_1.P_1)^{l'_1}|...|(\alpha_n.P_n)^{l'_n}$ and $Q'' \equiv (\alpha_1.P_1)^{l''_1}|...|(\alpha_n.P_n)^{l''_n}$, the result is verified.

□

[5] Else we can replace P', P'' with $(h+1,w)$-related processes having the same (h,w)-normal forms

Proof [Proof of Lemma 2]

Because $P \approx^{\Omega}_{(h,w)} Q$, $\alpha \in \Omega$ and $P \equiv \alpha.P'|P''$, we obtain that $Q \equiv \alpha.Q'|Q''$ with $P' \approx^{\Omega}_{(h-1,w)} Q'$. We prove that $P'|P'' \approx^{\Omega}_{(h-1,w-1)} Q'|Q''$.

Consider $\beta \in \Omega$ and $i = 1..w-1$ such that: $P'|P'' \equiv \beta.P_1|...|\beta.P_i|P^{\star}$. We can suppose that, for some $k \leq i$, we have $P' \equiv \beta.P_1|...|\beta.P_k|P^{+}$, $P'' \equiv \beta.P_{k+1}|...|\beta.P_i|P^{-}$ and $P^{\star} \equiv P^{+}|P^{-}$. Because $P' \approx^{\Omega}_{(h-1,w)} Q'$ and $k \leq i \leq w-1$, we obtain that $Q' \equiv \beta.Q_1|...|\beta.Q_k|Q^{+}$ with $P_j \approx^{\Omega}_{(h-2,w)} Q_j$ for $j = 1..k$. Further we distinguish two cases.

1. If $\alpha \neq \beta$, then we have $P \equiv \beta.P_{k+1}|...|\beta.P_i|(P^{-}|\alpha.P')$ and because $P \approx^{\Omega}_{(h,w)} Q$, we obtain $Q \equiv \beta.R_{k+1}|...|\beta.R_i|R^{\star}$ with $R_j \approx^{\Omega}_{(h-1,w)} P_j$ for $j = k+1..i$. But $Q \equiv \alpha.Q'|Q''$ and because $\alpha \neq \beta$, we obtain $Q'' \equiv \beta.R_{k+1}|...|\beta.R_i|R^{+}$ that gives us in the end $Q'|Q'' \equiv \beta.Q_1|...|\beta.Q_k|\beta.R_{k+1}|...|\beta.R_i|(R^{+}|Q^{+})$, with $P_j \approx^{\Omega}_{(h-2,w)} Q_j$ for $j = 1..k$ (hence, $P_j \approx^{\Omega}_{(h-2,w-1)} Q_j$) and $P_j \approx^{\Omega}_{(h-1,w)} R_j$ for $j = k+1..i$ (hence, $P_j \approx^{\Omega}_{(h-2,w-1)} R_j$).

2. If $\alpha = \beta$, then we have $P \equiv \alpha.P_{k+1}|...|\alpha.P_i|\alpha.P'|P^{-}$ and as $P \approx^{\Omega}_{(h,w)} Q$ and $i \leq w-1$, we obtain $Q \equiv \alpha.R_{k+1}|...|\alpha.R_i|\alpha.R'|R^{\star}$, with $R_j \approx^{\Omega}_{(h-1,w)} P_j$ for $j = k+1..i$ and $R' \approx^{\Omega}_{(h-1,w)} P'$. Because $P' \approx^{\Omega}_{(h-1,w)} Q'$ and $\approx^{\Omega}_{(h,w)}$ is an equivalence relation, we can suppose that[6] $R' \equiv Q'$. Consequently, $Q \equiv \alpha.R_{k+1}|...|\alpha.R_i|\alpha.Q'|R^{\star}$ that gives $Q'' \equiv \alpha.R_{k+1}|...|\alpha.R_i|R^{\star}$, which entails further $Q'|Q'' \equiv \alpha.Q_1|...|\alpha.Q_k|\alpha.R_{k+1}|...|\alpha.R_i|(R^{\star}|Q^{+})$ with $P_j \approx^{\Omega}_{(h-2,w)} Q_j$ for $j = 1..k$ (hence, $P_j \approx^{\Omega}_{(h-2,w-1)} Q_j$) and $P_j \approx^{\Omega}_{(h-1,w)} R_j$ for $j = k+1..i$ (hence, $P_j \approx^{\Omega}_{(h-2,w-1)} R_j$). All these prove that $P'|P'' \approx^{\Omega}_{(h-1,w-1)} Q'|Q''$.

The communication case goes similarly. □

Proof [Proof of Lemma 2] We construct Q inductively on h. For the case $P \equiv 0$ we take $Q \equiv P$, as $0 \in \mathbb{P}^{\Omega+}_{(h,w)}$.

Suppose $P \not\equiv 0$. Let $\beta = succ(sup(\Omega))$. In the case $h = 0$ we just take $Q \equiv \beta.0$.

The case $h+1$. Suppose, without loss of generality, that

$$P \equiv (\alpha_1.P_1)^{k_1}|...|(\alpha_n.P_n)^{k_n}|(\gamma_{n+1}.P_{n+1})^{k_{n+1}}|...|(\gamma_{n+m}.P_{n+m})^{k_{n+m}}$$

where $\alpha_1, ..\alpha_n \in \Omega$ with $\alpha_i.P_i \not\equiv \alpha_j.P_j$ for $i \neq j$, and $\gamma_{n+1}, ..\gamma_{n+m} \in \Sigma \setminus \Omega$ with $\gamma_i.P_i \not\equiv \gamma_j.P_j$ for $i \neq j$.

Let P'_j for $j = 1..n$ be the processes constructed at the previous inductive step such that $P_j \approx^{\Omega}_{(h,w)} P'_j$ with $P'_j \in \mathbb{P}^{\Omega+}_{(h,w)}$ - their existence is guaranteed by the inductive hypothesis. Let $l_i = min(k_i, w)$ and consider the process $P' \equiv (\alpha_1.P'_1)^{l_1}|...|(\alpha_n.P'_n)^{l_n}|\beta.0$. It is trivial to verify that P' is a process that fulfills the requirements of the lemma, i.e. $P \approx^{\Omega}_{(h,w)} P'$ and $P' \in \mathbb{P}^{\Omega+}_{(h,w)}$. □

Proof [Proof of Lemma 5.1] Induction on the structure of ϕ. We show here only the nontrivial cases.

The case $\phi = \langle\alpha\rangle\psi$**:** $P \models \langle\alpha\rangle\psi$ iff $P \overset{\alpha}{\textbf{to}} P'$ and $P' \models \psi$. Suppose that $\psi = (h, w)$. Then $\phi = (h+1, w+1)$. Because $\alpha \in act(\phi)$ and $P \approx^{act(\phi)}_{(h+1,w+1)} Q$, we obtain applying Lemma

[6]Indeed, if $\alpha.Q'$ is a subprocess of $R^{\star}$ then we can just substitute R' with Q'; if $\alpha.Q' \equiv \alpha.R_s$, then $Q' \approx^{\Omega}_{(h-1,w)} P_s$ and as $Q' \approx^{\Omega}_{(h-1,w)} P'$ and $P' \approx^{\Omega}_{(h-1,w)} R'$ we derive $R' \approx^{\Omega}_{(h-1,w)} P_s$ and $Q' \approx^{\Omega}_{(h-1,w)} P'$, so we can consider this correspondence.

2 that $Q\overset{\alpha}{\textbf{to}}Q'$ and $P' \approx^{act(\phi)}_{(h,w)} Q'$. We can apply the inductive hypothesis, as $P' \models \psi$ and we obtain $Q' \models \psi$. Then $Q \models \phi$.

The case $\phi = \langle \alpha, \overline{\alpha} \rangle \psi$**:** can be prove as the previous one using the second part of Lemma2.

The case $\phi = \psi_1 | \psi_2$**:** $P \models \psi_1 | \psi_2$ iff $P \equiv S|R$, $S \models \phi_1$ and $R \models \psi_2$. Suppose that $\psi_1 = (h_1, w_1)$ and $\psi_2 = (h_2, w_2)$. Then $\phi = (max(h_1, h_2), w_1 + w_2)$. Applying Lemma 2 for $P \approx^{act(\phi)}_{(max(h_1,h_2),w_1+w_2)} Q$, we obtain that $Q \equiv S'|R'$ such that $S \approx^{act(\phi)}_{(max(h_1,h_2),w_1)} S'$ and $R \approx^{act(\phi)}_{(max(h_1,h_2),w_2)} R'$. Further Lemma 2 gives $S \approx^{act(\psi_1)}_{(h_1,w_1)} S'$ and $R \approx^{act(\psi_2)}_{(h_2,w_2)} R'$. Further, the inductive hypothesis gives $S' \models \psi_1$ and $R' \models \psi_2$, i.e. $Q \models \psi_1 | \psi_2$.

The case $\phi = \psi_1 \triangleright \psi_2$**:** $P \models \psi_1 \triangleright \psi_2$ iff any $R \models \psi_1$ implies $P|R \models \psi_2$. But $P \approx^{act(\phi)}_{\phi} Q$ and $R \approx^{act(\phi)}_{\phi} R$ implies, by Lemma 2, that $P|R \approx^{act(\phi)}_{\phi} Q|R$. Further $P|R \approx^{act(\psi_2)}_{\psi_2} Q|R$ and because $P|R \models \psi_2$, we can apply the inductive hypothesis deriving $Q|R \models \psi_2$. Hence, $R \models \psi_1$ implies $Q|R \models \psi_2$, i.e., $Q \models \psi_1 \triangleright \psi_2$. □