A Sound, Complete and Terminating Decision Procedure for SLAOP

Gavin Rens[1] Thomas Meyer[1]
Gerhard Lakemeyer[2]

[1]Centre for Artificial Intelligence Research, University of KwaZulu-Natal, and CSIR Meraka, South Africa
[2]RWTH Aachen University, Informatik, Germany

October 2, 2013

Abstract

SLAOP is the Specification Logic of Actions and Observations with Probability. In this report, we provide the syntax and semantics of SLAOP and then provide a decision procedure for checking validity of sentences. The decision procedure is a tableau method which appeals to solving systems of equations. The tableau rules eliminate propositional connectives, then, for all open branches of the tableau tree, systems of equations are generated and checked for feasibility. We prove that the procedure is sound and complete.

1 Introduction

SLAOP is the Specification Logic of Actions and Observations with Probability. The logic is meant to facilitate the specification of domains with agents whose actions and observations are stochastic. SLAOP is based on modal logic [6, 3, 2] and partially observable Markov decision process (POMDP) theory [1, 14, 7]. Modal logic is considered to be well suited to reasoning about beliefs and changing situations. POMDPs have proven to be good for formalizing dynamic stochastic systems. A POMDP model is a tuple $\langle S, A, T, R, \Omega, O, b^0 \rangle$; S is a finite set of states the agent can be in; A is a finite set of actions the agent can choose to execute; T is the function defining the probability of reaching one state from another, for each action; R is a function, giving the expected immediate reward gained by the agent, for any state and agent action; Ω is a finite set of observations the agent can experience of its world; O is a function, giving for each agent action and the resulting state, a probability distribution over observations; and b^0 is the initial probability distribution over all states in S. Further motivation for the logic can be found elsewhere [9, 10, 11].

Here we only present a decision procedure for checking validity of sentences (§3). The decision procedure is a tableau method which appeals to solving systems of equations. The tableau rules eliminate propositional connectives, then, for all open branches of the tableau tree, systems of equations are generated and checked for feasibility. Then we prove that the procedure is sound (§4), complete (§5) and that it always terminates (§6). First, the syntax and semantics of SLAOP are provided (§ 2).

2 Specification Logic of Actions and Observations with Probability

The Specification Logic of Actions and Observations with Probability (SLAOP) extends the Specification Logic of Actions with Probability (SLAP) [12]. SLAP is extended with (i) notions of rewards and action costs, (ii) a notion of equlity between, respectively, actions and observations and (iii) observations for dealing with perception/sensing. First we present the syntax of SLAOP, then we state its semantics.

2.1 Syntax

The vocabulary of our language contains six sorts of objects of interest:

1. a finite set of *propositional variables* (simply, *propositions*) $\mathcal{P} = \{p_1, \ldots, p_n\}$,
2. a finite set of names of atomic *actions* $\mathcal{A} = \{\alpha_1, \ldots, \alpha_n\}$,
3. a finite set of names of atomic *objects* $\Omega = \{\varsigma_1, \ldots, \varsigma_n\}$,
4. all *real numbers* $\mathbb{R}$,
5. a countable set of *action variables* $V_\mathcal{A} = \{v_1^\alpha, v_2^\alpha, \ldots\}$,
6. a countable set of *observation variables* $V_\Omega = \{v_1^\varsigma, v_2^\varsigma, \ldots\}$.

We shall refer to elements of $\mathcal{A} \cup \Omega$ as *constants*. From now on, we denote $\mathbb{R} \cap [0,1]$ as $\mathbb{R}_{[0,1]}$. We are going to work in a multi-modal setting, in which we have modal operators $[\alpha]_q$, one for each $\alpha \in \mathcal{A}$ and $q \in \mathbb{R}_{[0,1]}$, and predicates $(\varsigma \mid \alpha : q$, one for each pair in $\Omega \times \mathcal{A}$ and $q \in \mathbb{R}_{[0,1]}$.

Definition 2.1 *Let $p \in \mathcal{P}$, $\alpha, \alpha' \in \mathcal{A}$, $\varsigma, \varsigma' \in \Omega$, $v \in (V_\mathcal{A} \cup V_\Omega)$ and $q \in \mathbb{R}_{[0,1]}, r, c \in \mathbb{R}$. The language of SLAOP, denoted $\mathcal{L}_{SLAOP}$, is the least set of Ψ defined by the grammar:*

$$\varphi ::= \ p \mid \top \mid \neg\varphi \mid \varphi \wedge \varphi.$$
$$\Phi ::= \ \varphi \mid \alpha = \alpha' \mid \varsigma = \varsigma' \mid \mathit{Reward}(r) \mid \mathit{Cost}(\alpha, c) \mid [\alpha]_q\varphi \mid (\varsigma \mid \alpha : q) \mid (\forall v)\Phi \mid \neg\Phi \mid \Phi \wedge \Phi.$$
$$\Psi ::= \ \Phi \mid \Box\Phi \mid \neg\Psi \mid \Psi \wedge \Psi.$$

Note that formulae with nested modal operators of the form $\Box\Box\Phi$, $\Box\Box\Box\Phi$, $[\alpha]_q[\alpha]_q\varphi$ and $[\alpha]_q[\alpha]_q[\alpha]_q\varphi$ et cetera are not in $\mathcal{L}_{SLAOP}$. 'Single-step' or 'flat' formulae are sufficient to *specify* action transitions probabilities and perception probabilities.

As usual, we treat $\bot, \vee, \rightarrow$ and $\leftrightarrow$ as abbreviations. $\rightarrow$ and $\leftrightarrow$ have the weakest bindings and $\neg$ the strongest; parentheses enforce or clarify the scope of operators conventionally.

The definition of a POMDP reward function $R(a, s)$ may include not only the reward value of state s, but it may deduct the cost of performing a in s. It will be convenient for the person specifying a POMDP using SLAOP to be able to specify action costs independently from the rewards of states, because these two notions are not necessarily connected. To specify rewards and execution costs in SLAOP, we require *Reward* and *Cost* as special predicates. $Reward(r)$ can be read 'The reward for being in the current situation is r units' and we read $Cost(\alpha, c)$ as 'The cost for executing α is c units'.

$\alpha = \alpha'$, $\varsigma = \varsigma'$, $Reward(r)$, $Cost(\alpha, c)$ and their negations are referred to as *erc literals*. $[\alpha]_q\varphi$ and $\neg[\alpha]_q\varphi$ are referred to as *dynamic literals*. Any formula which includes a dynamic literal shall be referred to as *dynamic*. $(\varsigma \mid \alpha : q)$ and $\neg(\varsigma \mid \alpha : q)$ are referred to as *perception literals*. Any formula which includes a perception literal shall be referred to as *perceptual*. A formula may thus be both dynamic and perceptual. $[\alpha]_q\varphi$ is read 'The probability of reaching a φ-world after executing α, is equal to q'. $[\alpha]$ abbreviates $[\alpha]_1$. $(\varsigma \mid \alpha : q)$ is read 'The probability of perceiving ς, given α was performed, is q'.

$\langle\alpha\rangle\varphi$ abbreviates $\neg[\alpha]_0\varphi$ and is read 'It is possible to reach a world in which φ holds after executing α'. Note that $\langle\alpha\rangle\varphi$ does not mean $\neg[\alpha]\neg\varphi$.

One reads $\Box\Phi$ as 'Φ holds in every possible world'. We require the $\Box$ operator to mark certain information (sentences) as holding in *all* possible worlds—essentially, the axioms which model the domain of interest.

$(\forall v^\alpha)$ is to be read 'For all actions' and $(\forall v^\varsigma)$ is to be read 'For all observations'. $(\forall v)\Phi$ (where $v \in (V_\mathcal{A} \cup V_\Omega)$) can be thought of as a syntactic shorthand for the finite conjunction of Φ with the variables replaced by the constants of the right sort (cf. Def. 2.4 for the formal definition). $(\exists v)\Phi$ abbreviated $\neg(\forall v)\neg\Phi$.

Definition 2.2 *A formula* $\Psi \in \mathcal{L}_{SLAOP}$ *is in* conjunctive normal form *(CNF) if and only if it is in the form*

$$\Psi_1 \wedge \Psi_2 \wedge \cdots \wedge \Psi_n,$$

where each of the Ψ_i *is a disjunction of literals.*

A formula $\Psi \in \mathcal{L}_{SLAOP}$ *is in* disjunctive normal form *(DNF) if and only if it is in the form*

$$\Psi_1 \vee \Psi_2 \vee \cdots \vee \Psi_n,$$

where each of the Ψ_i *is a conjunction of literals. Note that if a dynamic literal* $[\alpha]_q\varphi$ *or* $\neg[\alpha]_q\varphi$ *is a disjunct/conjunct of* Ψ_i*,* φ *is allowed to have any form, as long as* $\varphi \in \mathcal{L}_{SLAOP}$*.*

2.2 Semantics

Standard modal logic structures (alias, possible worlds models) are tuples $\langle W, R, V\rangle$, where W is a (possibly infinite) set of states (possibly without internal structure), R is a binary relation on W, and V is a valuation, assigning subsets of W to each atomic proposition. This is the standard Kripke-style semantics (see, e.g., [4, 8, 6]).

SLAOP extends SLAP. SLAP structures are non-standard: They have the form $\langle W, R\rangle$, where W is a *finite* set of worlds such that each world assigns a truth value to each atomic proposition, and R is a binary relation on W. Moreover, SLAOP is multi-modal in that there are multiple accessibility relations.

Intuitively, when talking about some world w, we mean a set of features (*propositions*) that the agent understands and that describes a state of affairs in the world or that describes a possible, alternative world. Let $w : \mathcal{P} \mapsto \{0,1\}$ be a total function that assigns a truth value to each proposition. Let C be the set of all possible functions w. We call C the *conceivable worlds*.

Definition 2.3 *A SLAOP structure is a tuple $\mathcal{S} = \langle W, R, O, N, Q, U\rangle$ such that*

1. *$W \subseteq C$ a non-empty set of* possible worlds*.*
2. *$R : \mathcal{A} \mapsto R_\alpha$, where $R_\alpha : (W \times W) \mapsto \mathbb{R}_{[0,1]}$ is a total function from pairs of worlds into the reals; That is, R is a mapping that provides an accessibility relation R_α for each action $\alpha \in \mathcal{A}$; For every $w^- \in W$, it is required that either $\sum_{w^+\in W} R_\alpha(w^-, w^+) = 1$ or $\sum_{w^+\in W} R_\alpha(w^-, w^+) = 0$.*
3. *O is a nonempty finite set of observations;*
4. *$N : \Omega \mapsto O$ is a bijection that associates to each name in Ω, a unique observation in O;*
5. *$Q : \mathcal{A} \mapsto Q_\alpha$, where $Q_\alpha : (W \times O) \mapsto \mathbb{R}_{[0,1]}$ is a total function from pairs in $W \times O$ into the reals; That is, Q is a mapping that provides a perceivability relation Q_α for each action $\alpha \in \mathcal{A}$; For all $w^-, w^+ \in W$: if $R_\alpha(w^-, w^+) > 0$, then $\sum_{o\in O} Q_\alpha(w^+, o) = 1$, that is, there is a probability distribution over observations in a reachable world; Else if $R_\alpha(w^-, w^+) = 0$, then either $\sum_{o\in O} Q_\alpha(w^+, o) = 1$ or $\sum_{o\in O} Q_\alpha(w^+, o) = 0$;*
6. *U is a pair $\langle Re, Co\rangle$, where $Re : W \mapsto \mathbb{R}$ is a reward function and Co is a mapping that provides a cost function $Co_\alpha : C \mapsto \mathbb{R}$ for each $\alpha \in \mathcal{A}$.*

Note that the set of possible worlds may be the whole set of conceivable worlds.

R_α defines the transition probability $pr \in \mathbb{R}_{[0,1]}$ between worlds w^+ and world w^- via action α. If $R_\alpha(w^-, w^+) = 0$, then w^+ is said to be *inaccessible* or *not reachable* via α performed in w^-, else if $R_\alpha(w^-, w^+) > 0$, then w^+ is said to be *accessible* or *reachable* via action α performed in w^-. If for some w^-, $\sum_{w^+\in W} R_\alpha(w^-, w^+) = 0$, we say that α is *inexecutable* in w^-.

Q_α defines the observation probability $pr \in \mathbb{R}_{[0,1]}$ of observation o perceived in world w^+ after the execution of action α. Assuming w^+ is accessible, if $Q_\alpha(w^+, o) > 0$, then o is said to be *perceivable* in w^+, given α, else if $Q_\alpha(w^+, o) = 0$, then o is said to be *unperceivable* in w^+, given α. The definition of perceivability relations implies that there is always at least one possible observation in any world reached due to an action.

Because N is a bijection, it follows that $|O| = |\Omega|$. (We take $|X|$ to be the cardinality of set X.) The value of the reward function $Re(w)$ is a real number representing the reward an agent gets for being in or getting to the world w. It must be defined for each $w \in C$. The value of the cost function $Co(\alpha, w)$ is a real number representing the cost of executing α in the world w. It must be defined for each action $\alpha \in \mathcal{A}$ and each $w \in C$.

Definition 2.4 (Truth Conditions) *Let $\mathcal{S}$ be a SLAOP structure, with $\alpha, \alpha' \in \mathcal{A}$, $q, pr \in \mathbb{R}_{[0,1]}, r, c \in \mathbb{R}$. Let $p \in \mathcal{P}$ and let Φ be any sentence in $\mathcal{L}_{SLAOP}$. We say Φ is* satisfied *at world w in structure $\mathcal{S}$ (written $\mathcal{S}, w \models \Phi$) if and only if the following holds:*

1. $\mathcal{S}, w \models \top$ for all $w \in W$;
2. $\mathcal{S}, w \models p \iff w(p) = 1$ for $w \in W$;
3. $\mathcal{S}, w \models \neg\Psi \iff \mathcal{S}, w \not\models \Psi$;
4. $\mathcal{S}, w \models \Psi \wedge \Psi' \iff \mathcal{S}, w \models \Psi$ and $\mathcal{S}, w \models \Psi'$;
5. $\mathcal{S}, w \models (\alpha = \alpha') \iff \alpha, \alpha' \in \mathcal{A}$ are the same element;
6. $\mathcal{S}, w \models (\varsigma = \varsigma') \iff \varsigma, \varsigma' \in \Omega$ are the same element;
7. $\mathcal{S}, w \models Reward(r) \iff Re(w) = r$;
8. $\mathcal{S}, w \models Cost(\alpha, c) \iff Co_\alpha(w) = c$;
9. $\mathcal{S}, w \models [\alpha]_q\varphi \iff \sum_{w' \in W, \mathcal{S}, w' \models \varphi} R_\alpha(w, w') = q$;
10. $\mathcal{S}, w \models (\varsigma \mid \alpha : q) \iff Q_\alpha(w, N(\varsigma)) = q$;
11. $\mathcal{S}, w \models \Box\Phi \iff$ for all $w' \in W, \mathcal{S}, w' \models \Phi$;
12. $\mathcal{S}, w \models (\forall v^\alpha)\Phi \iff \Phi|^{v^\alpha}_{\alpha_1} \wedge \ldots \wedge \Phi|^{v^\alpha}_{\alpha_n}$;
13. $\mathcal{S}, w \models (\forall v^\varsigma)\Phi \iff \Phi|^{v^\varsigma}_{\varsigma_1} \wedge \ldots \wedge \Phi|^{v^\varsigma}_{\varsigma_n}$,

where we write $\Psi|^v_c$ to mean the formula Ψ with all variables $v \in (V_\mathcal{A} \cup V_\Omega)$ appearing in it replaced by constant $c \in \mathcal{A} \cup \Omega$ of the right sort.

A formula φ is *valid* in a SLAOP structure (denoted $\mathcal{S} \models \varphi$) if $\mathcal{S}, w \models \varphi$ for every $w \in W$. φ is *SLAOP-valid* (denoted $\models \varphi$) if φ is true in every structure $\mathcal{S}$. If $\models \theta \leftrightarrow \psi$, we say θ and ψ are *semantically equivalent* (abbreviated $\theta \equiv \psi$).

φ is *satisfiable* if $\mathcal{S}, w \models \varphi$ for some $\mathcal{S}$ and $w \in W$. A formula that is not satisfiable is *unsatisfiable* or a *contradiction*. The truth of a propositional formula depends only on the world in which it is evaluated. We may thus write $w \models \varphi$ instead of $\mathcal{S}, w \models \varphi$ when φ is a propositional formula.

Let $\mathcal{K}$ be a finite subset of $\mathcal{L}_{SLAOP}$. We say that ψ is a *local semantic consequence* of $\mathcal{K}$ (denoted $\mathcal{K} \models \psi$) if for all structures $\mathcal{S}$, and all $w \in W$ of $\mathcal{S}$, if $\mathcal{S}, w \models \bigwedge_{\theta \in \mathcal{K}} \theta$ then $\mathcal{S}, w \models \psi$. We shall also say that $\mathcal{K}$ *entails* ψ whenever $\mathcal{K} \models \psi$. If $\{\theta\} \models \psi$ then we simply write $\theta \models \psi$. In fact,

$$\mathcal{K} \models \Psi \quad \text{if and only if} \quad \models \bigwedge_{\theta \in \mathcal{K}} \theta \rightarrow \Psi$$

(i.e., $\mathcal{K}$ entails Ψ iff $\bigwedge_{\theta \in \mathcal{K}} \theta \rightarrow \Psi$ is SLAOP-valid).

Proposition 2.1 *Let $\mathcal{L}^{-\Box}_{SLAOP}$ be all formulae in $\mathcal{L}_{SLAOP}$ such that the formulae contain no $\Box$ operators. For every $\Psi \in \mathcal{L}^{-\Box}_{SLAOP}$, there exists a formula $\Psi' \in \mathcal{L}^{-\Box}_{SLAOP}$ in CNF and there exists a formula $\Psi'' \in \mathcal{L}^{-\Box}_{SLAOP}$ in DNF such that $\Psi \equiv \Psi' \equiv \Psi''$.*

Proof:
Let $A, B, C, D \in \mathcal{L}^{-\Box}_{SLAOP}$. When it was said in § 2.1 that $\vee$ is an abbreviation, it was meant that formulae involving $\vee$ can be converted into formulae not involving $\vee$, by using the De Morgan's laws mentioned below.

To convert to an equivalent formula in CNF or DNF, perform the following steps, each of which preserves semantic equivalence:

1. Push all negations inwards using De Morgan's laws:
$$\begin{aligned} \neg(A \wedge B) &\equiv (\neg A \vee \neg B) \\ \neg(A \vee B) &\equiv (\neg A \wedge \neg B). \end{aligned}$$
2. Eliminate double negation using the equivalence $\neg\neg A \equiv A$.
3. The formula now consists of disjunctions and conjunctions of literals. To obtain CNF, use the distributive laws
$$\begin{aligned} A \vee (B \wedge C) &\equiv (A \vee B) \wedge (A \vee C) \\ (A \wedge B) \vee C &\equiv (A \vee C) \wedge (B \vee C) \end{aligned}$$
to replace (sub)formulae of the LHS form by the appropriate formula in the RHS form. To obtain DNF, use the distributive laws
$$\begin{aligned} A \wedge (B \vee C) &\equiv (A \wedge B) \vee (A \wedge C) \\ (A \vee B) \wedge C &\equiv (A \wedge C) \vee (B \wedge C) \end{aligned}$$
to replace (sub)formulae of the LHS form by the appropriate formula in the RHS form.

■

Proposition 2.2 *Clearly, by definition,*
$$(\forall v^\alpha)\Phi \equiv \Phi|^{v^\alpha}_{\alpha_1} \wedge \ldots \wedge \Phi|^{v^\alpha}_{\alpha_n} \textit{ for } \mathcal{A} = \{\alpha_1, \ldots, \alpha_n\}$$
and
$$(\forall v^\varsigma)\Phi \equiv \Phi|^{v^\varsigma}_{\varsigma_1} \wedge \ldots \wedge \Phi|^{v^\varsigma}_{\varsigma_n} \textit{ for } \Omega = \{\varsigma_1, \ldots, \varsigma_n\}.$$

In the rest of the report, we shall assume that whenever the validity of a sentence is to be checked, a preprocessing step occurs, where all (sub)formulae of the form $(\forall v^\alpha)\Phi$ and $(\forall v^\varsigma)\Phi$ in the sentence are replaced by, respectively, $(\Phi|^{v^\alpha}_{\alpha_1} \wedge \ldots \wedge \Phi|^{v^\alpha}_{\alpha_n})$ and $(\Phi|^{v^\varsigma}_{\varsigma_1} \wedge \ldots \wedge \Phi|^{v^\varsigma}_{\varsigma_n})$. By Proposition 2.2, a sentence before the preprocessing step is valid if and only if the sentence after the preprocessing step is valid. Hence, in the decision procedure, sentences of the form $(\forall v)\Phi$ do not occur. Neither do such sentences need to be dealt with in the proofs of soundness and completeness of the decision procedure.

3 Decision Procedure for SLAOP Entailment

In this section we describe a decision procedure which has two phases: creation of a tableau tree (the *tableau* phase) which essentially eliminates propositional connectives, then a phase which checks for inconsistencies given possible mappings from 'labels' (of the tableau calculus) to worlds (the *label assignment* phase). Particularly, in the label assignment phase, solutions for system of inequalities (equations and disequalities) are sought.

3.1 The Tableau Phase

The necessary definitions and terminology are given next.

Definition 3.1 *A* labeled formula *is a pair* (x, Ψ)*, where* $\Psi \in \mathcal{L}_{SLAOP}$ *is a formula and* x *is an integer called the* label *of* Ψ*.*

Definition 3.2 *A* node Γ^j_k *with superscript* j *(the* branch *index) and subscript* k *(the* node *index), is a set of labeled formulae.*

Definition 3.3 *The initial node, that is,* Γ^0_0*, to which the tableau rules must be applied, is called the* trunk*.*

Definition 3.4 *A* tree T *is a set of nodes. A tree must include* Γ^0_0 *and only nodes resulting from the application of* tableau rules *to the trunk and subsequent nodes. If one has a tree with trunk* $\Gamma^0_0 = \{(0, \Psi)\}$*, we'll say one has* a tree for Ψ*.*

When we say '...where x is a fresh integer', we mean that x is the smallest positive integer of the right sort (formula label or branch index) not yet used in the node to which the incumbent tableau rule will be applied.

A tableau rule applied to node Γ^j_k creates one or more new nodes; its child(ren). If it creates one child, then it is identified as Γ^j_{k+1}. If Γ^j_k creates a second child, it is identified as $\Gamma^{j'}_0$, where j' is a fresh integer. That is, for every child created beyond the first, a new branch is started.

Definition 3.5 *A node* Γ *is a* leaf *node of tree* T *if no tableau rule has been applied to* Γ *in* T*.*

Definition 3.6 *A* branch *is the set of nodes on a path from the trunk to a leaf node.*

Note that nodes with different branch indexes may be on the some branch.

Definition 3.7 Γ *is* higher *on a branch than* Γ' *if and only if* Γ *is an ancestor of* Γ'*.*

Definition 3.8 *A node* Γ *is* closed *if* $(x, \perp) \in \Gamma$ *for any* $x \geq 0$*. It is* open *if it is not closed. A branch is closed if and only if its leaf node is closed. A tree is closed if all of its branches are closed, else it is open.*

A preprocessing step occurs, where all (sub)formulae of the form $(\forall v^{\alpha})\Phi$ and $(\forall v^{\varsigma})\Phi$ are replaced by, respectively, $(\Phi|^{v^{\alpha}}_{\alpha_1} \wedge \ldots \wedge \Phi|^{v^{\alpha}}_{\alpha_n})$ and $(\Phi|^{v^{\varsigma}}_{\varsigma_1} \wedge \ldots \wedge \Phi|^{v^{\varsigma}}_{\varsigma_n})$. The occurrence of $(\exists v^{\varsigma})\neg(v^{\varsigma} \mid \alpha : 0)$ in rule obs (below) is only an abbreviation for the semantically equivalent formula without a quantifier and variables.

The tableau rules for SLAOP follow. A rule may only be applied to an open leaf node. To constrains rule application to prevent trivial re-applications of rules, a rule may not be applied to a formula if it has been applied to that formula higher in the tree, as in Definition 3.7. For example, if rule $\Box$ were applied to $\{(0, \Box p_1), (1, \neg[go]_0 p_2)\} \subset \Gamma^2_3$, then it may not be applied to $\{(0, \Box p_1), (1, \neg[go]_0 p_2)\} \subset \Gamma^2_4$.

Let Γ^j_k be a leaf node.

- rule $\bot$: If Γ^j_k contains (n, Φ) and $(n, \neg\Phi)$, then create node $\Gamma^j_{k+1} = \Gamma^j_k \cup \{(n, \bot)\}$.
- rule $\neg$: If Γ^j_k contains $(n, \neg\neg\Phi)$, then create node $\Gamma^j_{k+1} = \Gamma^j_k \cup \{(n, \Phi)\}$.
- rule $\wedge$: If Γ^j_k contains $(n, \Phi \wedge \Phi')$, then create node $\Gamma^j_{k+1} = \Gamma^j_k \cup \{(n, \Phi), (n, \Phi')\}$.
- rule $\vee$: If Γ^j_k contains $(n, \neg(\Phi \wedge \Phi'))$, then create node $\Gamma^j_{k+1} = \Gamma^j_k \cup \{(n, \neg\Phi)\}$ and node $\Gamma^{j'}_0 = \Gamma^j_k \cup \{(n, \neg\Phi')\}$, where j' is a fresh integer.
- rule $=$: If Γ^j_k contains $(n, c = c')$ and c and c' are distinct constants, or if Γ^j_k contains $(n, \neg(c = c'))$ and c and c' are identical constants, then create node $\Gamma^j_{k+1} = \Gamma^j_k \cup \{(n, \bot)\}$.
- rule $\Diamond\varphi$: If Γ^j_k contains $(0, \neg[\alpha]_0\varphi)$ or $(0, [\alpha]_q\varphi)$ for $q > 0$, then create node $\Gamma^j_{k+1} = \Gamma^j_k \cup \{(n, \varphi)\}$, where n is a fresh integer.
- rule obs: If Γ^j_k contains $(x, \neg[\alpha]_0\varphi)$ or $(x, [\alpha]_q\varphi)$ for $q > 0$ and some x, then create node $\Gamma^j_{k+1} = \Gamma^j_k \cup \{(0, \Box(\delta_1 \rightarrow (\exists v^{\varsigma})\neg(v^{\varsigma} \mid \alpha : 0)) \vee \Box(\delta_2 \rightarrow (\exists v^{\varsigma})\neg(v^{\varsigma} \mid \alpha : 0)) \vee \cdots \vee \Box(\delta_n \rightarrow (\exists v^{\varsigma})\neg(v^{\varsigma} \mid \alpha : 0)))\}$, where $\delta_i \in \mathit{Def}(\varphi)$.
- rule $\Box$: If Γ^j_k contains $(0, \Box\Phi)$ and (n, Φ') for any $n \geq 0$, and if it does not yet contain (n, Φ), then create node $\Gamma^j_{k+1} = \Gamma^j_k \cup \{(n, \Phi)\}$.
- rule $\Diamond$: If Γ^j_k contains $(0, \neg\Box\Phi)$, then create node $\Gamma^j_{k+1} = \Gamma^j_k \cup \{(n, \neg\Phi)\}$, where n is a fresh integer.

Definition 3.9 *A branch is* saturated *if and only if any rule that can be applied to its leaf node has been applied. A tree is* saturated *if and only if all its branches are saturated.*

3.2 Systems of Inequalities

We first need to explain how a system of inequalities (SI) can be generated from a set of dynamic and perception literals, before the remaining phase can be explained.

Definition 3.10 $W(\Gamma, n) \stackrel{def}{=} \{w \in C \mid w \models \ell$ *for all* $(n, \ell) \in \Gamma$ *where* ℓ *is a propositional literal*$\}$.

Definition 3.11 *$W(\Gamma) \overset{def}{=} \bigcup_{x \in \{0,1,\ldots,n'\}} W(\Gamma, x)$, where n' is the largest label mentioned in Γ.*

Let $n = |W(\Gamma)|$. Let $W(\Gamma)^{\#} = (w_1, w_2, \ldots, w_n)$ be an ordering of the worlds in $W(\Gamma)$. With each world $w_k \in W(\Gamma)^{\#}$, we associate a real variable $pr_k^{\alpha} \in \mathbb{Q}_{[0,1]}$. One can generate

$$c_{i,1}pr_1^{\alpha} + c_{i,2}pr_2^{\alpha} + \cdots + c_{i,n}pr_n^{\alpha} = q_i,$$

and

$$c_{i,1}pr_1^{\alpha} + c_{i,2}pr_2^{\alpha} + \cdots + c_{i,n}pr_n^{\alpha} \neq q_i,$$

for a formulae $(x, [\alpha]_{q_i}\varphi_i) \in \Gamma$, respectively $(x, \neg[\alpha]_{q_i}\varphi_i) \in \Gamma$ such that $c_{i,k} = 1$ if $w_k \models \varphi_i$, else $c_{i,k} = 0$, where x represents a label.

Let $\Delta(\alpha)$ be a set of dynamic literals mentioning α, and let

$$\Delta(\alpha)^{\#} = \{[\alpha]_{q_1}\varphi_1, [\alpha]_{q_2}\varphi_2, \ldots, [\alpha]_{q_g}\varphi_g, \neg[\alpha]_{q_{g+1}}\varphi_{g+1}, \neg[\alpha]_{q_{g+2}}\varphi_{g+2}, \ldots, \neg[\alpha]_{q_{g+h}}\varphi_{g+h}\}$$

be an ordering of the members of $\Delta(\alpha)$. With this notation in hand, given some α, we define the system

$$\begin{array}{lr} c_{1,1}pr_1^{\alpha} + c_{1,2}pr_2^{\alpha} + \cdots + c_{1,n}pr_n^{\alpha} & = q_1 \\ c_{2,1}pr_1^{\alpha} + c_{2,2}pr_2^{\alpha} + \cdots + c_{2,n}pr_n^{\alpha} & = q_2 \\ \vdots & \\ c_{g,1}pr_1^{\alpha} + c_{g,2}pr_2^{\alpha} + \cdots + c_{g,n}pr_n^{\alpha} & = q_g \\ c_{g+1,1}pr_1^{\alpha} + c_{g+1,2}pr_2^{\alpha} + \cdots + c_{g+1,n}pr_n^{\alpha} & \neq q_{g+1} \\ c_{g+2,1}pr_1^{\alpha} + c_{g+2,2}pr_2^{\alpha} + \cdots + c_{g+2,n}pr_n^{\alpha} & \neq q_{g+2} \\ \vdots & \\ c_{g+h,1}pr_1^{\alpha} + c_{g+h,2}pr_2^{\alpha} + \cdots + c_{g+h,n}pr_n^{\alpha} & \neq q_{g+h} \\ pr_1^{\alpha} + pr_2^{\alpha} + \cdots + pr_n^{\alpha} & = \lceil pr_1^{\alpha} + pr_2^{\alpha} + \cdots + pr_n^{\alpha} \rceil, \end{array} \quad (1)$$

where each of the first $g+h$ (in)equalities represents a member in $\Delta(\alpha)^{\#}$. The equation

$$pr_1^{\alpha} + pr_2^{\alpha} + \cdots + pr_n^{\alpha} = \lceil pr_1^{\alpha} + pr_2^{\alpha} + \cdots + pr_n^{\alpha} \rceil$$

is to ensure that either $\sum_{(w^-,w^+,pr) \in R_\alpha} pr = 1$ or $\sum_{(w^-,w^+,pr) \in R_\alpha} pr = 0$, as stated in Definition 2.3 on page 4.

Let $m = |\Omega|$. Let $\Omega^{\#} = (\varsigma_1, \varsigma_2, \ldots, \varsigma_m)$ be an ordering of the observations in Ω. With each observation in $\varsigma_j \in \Omega^{\#}$, we associate a real variable pr_j^{ς}.

One can generate

$$pr_j^{\sigma} = q_j \text{ and } pr_j^{\sigma} \neq q_j$$

for a formula $(x, (\sigma_j \mid \alpha : q_j)) \in \Gamma$, respectively, $(x, \neg(\sigma_j \mid \alpha : q_j) \in \Gamma$, where $\sigma_j \in \Omega^{\#}$ and $pr_j^{\sigma} \in \{pr_1^{\varsigma}, \ldots, pr_2^{\varsigma}, \ldots, pr_m^{\varsigma}\}$.

Let $\Omega(\alpha)$ be a set of perception literals involving α, and let

$$\Omega(\alpha)^{\#} = \{(\varsigma_1 \mid \alpha : q_1), \ldots, (\varsigma_t \mid \alpha : q_t), \neg(\varsigma_{t+1} \mid \alpha : q_{t+1}), \ldots, \neg(\varsigma_{t+v} \mid \alpha : q_{t+v})\}$$

be an ordering of the members of $\Omega(\alpha)$.

Then given some α, one can induce the following system.

$$\begin{aligned}
pr_1^\sigma &= q_1 \\
pr_2^\sigma &= q_2 \\
&\vdots \\
pr_t^\sigma &= q_t \\
pr_{t+1}^\sigma &\neq q_{t+1} \\
pr_{t+2}^\sigma &\neq q_{t+2} \\
&\vdots \\
pr_{t+v}^\sigma &\neq q_{t+v} \\
pr_1^\varsigma + pr_2^\varsigma + \cdots + pr_2^\varsigma + \cdots + pr_m^\varsigma &= \lceil pr_1^\varsigma + pr_2^\varsigma + \cdots + pr_2^\varsigma + \cdots + pr_m^\varsigma \rceil.
\end{aligned} \tag{2}$$

where each of the first $t+v$ (in)equalities represents a member in $\Omega(\alpha)^{\#}$. The equation

$$pr_1^\varsigma + pr_2^\varsigma + \cdots + pr_2^\varsigma + \cdots + pr_m^\varsigma = \lceil pr_1^\varsigma + pr_2^\varsigma + \cdots + pr_2^\varsigma + \cdots + pr_m^\varsigma \rceil.$$

is to ensure that either $\sum_{o \in O,(o,w^+,q) \in Q_\alpha} q = 1$ or $\sum_{o \in O,(o,w^+,q) \in Q_\alpha} q = 0$, as stated in Definition 2.3 on page 4.

Definition 3.12 *Let S be either System* (1) *or* (2)*. Let $\mathbf{v}$ be the vector of all variables mentioned in S. $Z(\Delta(\alpha))$ and $Z(\Omega(\alpha))$ denote the solution set for S. It is the set of all solutions of the form $(s_1^\alpha, s_2^\alpha, \ldots, s_n^\alpha)$, respectively, $(s_1^\varsigma, s_2^\varsigma, \ldots, s_m^\varsigma)$, where assigning s_i^α to $pr_i^\alpha \in \mathbf{v}$ for $i = 1, 2, \ldots, n$, respectively, assigning s_j^ς to $pr_j^\varsigma \in \mathbf{v}$ for $j = 1, 2, \ldots, m$ solves all the (in)equalities in S simultaneously. An SI is* feasible *if and only if its solution set is not empty.*

Lemma 3.1 *Determining whether an SI (as defined in this report) is feasible, is decidable.*

Proof:
Alfred Tarski [15] defines the first-order logic theory of elementary (real number) algebra as having an infinite number of variables (representing elements of $\mathbb{R}$), algebraic constants 1, 0, -1, two algebraic operation signs + (addition) and $\cdot$ (multiplication), two algebraic relation symbols = (equals) and > (greater than), (logical) sentential connectives $\sim$ (negation), $\wedge$ (conjunction), $\vee$ (disjunction), the existential quantifier $\exists$, and a set of axioms defining the theory. "If ξ is any variable, then $(\exists\xi)$ is called a *quantifier expression.*[1] The expression $(\exists\xi)$ is to be read "there exists a ξ such that ."

[1]He actually uses the symbol E for existential quantification.

For every SI (as defined in §3.2), the question of whether it has a solution can be represented in the language of first-order elementary algebra as follows.

$$
\begin{aligned}
&(\exists pr_1^\alpha)(\exists pr_2^\alpha)\cdots(\exists pr_n^\alpha)(\exists pr_1^\varsigma)(\exists pr_2^\varsigma)\cdots(\exists pr_m^\varsigma)\\
&c_{1,1}\cdot pr_1^\alpha + c_{1,2}\cdot pr_2^\alpha + \cdots + c_{1,n}\cdot pr_n^\alpha = q_1^\alpha \quad \wedge\\
&c_{2,1}\cdot pr_1^\alpha + c_{2,2}\cdot pr_2^\alpha + \cdots + c_{2,n}\cdot pr_n^\alpha = q_2^\alpha \quad \wedge\\
&\vdots\\
&c_{g,1}\cdot pr_1^\alpha + c_{g,2}\cdot pr_2^\alpha + \cdots + c_{g,n}\cdot pr_n^\alpha = q_g^\alpha \quad \wedge\\
&\sim (c_{g+1,1}\cdot pr_1^\alpha + c_{g+1,2}\cdot pr_2^\alpha + \cdots + c_{g+1,n}\cdot pr_n^\alpha = q_{g+1}^\alpha) \quad \wedge\\
&\sim (c_{g+2,1}\cdot pr_1^\alpha + c_{g+2,2}\cdot pr_2^\alpha + \cdots + c_{g+2,n}\cdot pr_n^\alpha = q_{g+2}^\alpha) \quad \wedge\\
&\vdots\\
&\sim (c_{g+h,1}\cdot pr_1^\alpha + c_{g+h,2}\cdot pr_2^\alpha + \cdots + c_{g+h,n}\cdot pr_n^\alpha = q_{g+h}^\alpha) \quad \wedge\\
&(pr_1^\alpha + pr_2^\alpha + \cdots + pr_n^\alpha = 1 \vee pr_1^\alpha + pr_2^\alpha + \cdots + pr_n^\alpha = 0) \quad \wedge\\
&pr_1^\sigma = q_1^\varsigma \quad \wedge\\
&pr_2^\sigma = q_2^\varsigma \quad \wedge\\
&\vdots\\
&pr_t^\sigma = q_t^\varsigma \quad \wedge\\
&\sim (pr_{t+1}^\sigma = q_{t+1}^\varsigma) \quad \wedge\\
&\sim (pr_{t+2}^\sigma = q_{t+2}^\varsigma) \quad \wedge\\
&\vdots\\
&\sim (pr_{t+v}^\sigma = q_{t+v}^\varsigma) \quad \wedge\\
&(pr_1^\varsigma + pr_2^\varsigma + \cdots + pr_m^\varsigma = 1 \vee pr_1^\varsigma + pr_2^\varsigma + \cdots + pr_m^\varsigma = 0).
\end{aligned}
$$

such that (algebraic constant) $c_{i,k} = 1$ or 0 as described in §3.2, the bracketed subformulae are present only if the corresponding (dis)equations are present in (2), and the pr_k^α, the pr_j^σ and the pr_j^ς are the variables.

Tarski provided a finite method which can always decide whether a sentence in the elementary algebra is in the theory [15]. Hence, feasibility of SIs is decidable. ■

3.3 The Label Assignment Phase

Given two formulae $(x, \Phi), (x', \Phi') \in \Gamma$ such that Φ contradicts Φ', if x and x' represent the same world, then Γ should close. But if $x \neq x'$, one must determine whether x and x' can be made to represent different worlds. In other words, one must check whether there is a 'proper' assignment of worlds to labels such that no contradictions occur.

Informally, x mentioned in Γ could represent any one of the worlds in $W(\Gamma, x)$. Now suppose $(x, \Phi), (x', \Phi') \in \Gamma$ such that Φ contradicts Φ' and $W(\Gamma, x) = \{w_1, w_2\}$ and $W(\Gamma, x') = \{w_2, w_3\}$. Assuming that Φ and Φ' do not involve the $\Box$ operator, it is conceivable that there exists a structure $\mathcal{S}$ such that (i) $\mathcal{S}, w_1 \models \Phi$ and $\mathcal{S}, w_2 \models \Phi'$, (ii) $\mathcal{S}, w_1 \models \Phi$ and $\mathcal{S}, w_3 \models \Phi'$ or (iii) $\mathcal{S}, w_2 \models \Phi$ and $\mathcal{S}, w_3 \models \Phi'$. But to have $\mathcal{S}, w_2 \models \Phi$ and $\mathcal{S}, w_2 \models \Phi'$ is inconceivable. Hence, if it were the case that, for example,

$W(\Gamma, x) = \{w_2\}$ and $W(\Gamma, x') = \{w_2\}$, then we would have found a contradiction and Γ should be made closed.

To formalize the process, some more definitions are required:

Definition 3.13 *$SoLA(\Gamma) \stackrel{def}{=} \{(0{:}w^0, 1{:}w^1, \ldots, x'{:}w^{x'}) \mid w^x \in W(\Gamma, x)\}$, where 0, 1, …, x' are all the labels mentioned in Γ. We shall call an element of $SoLA(\Gamma)$ a* label assignment*. LA denotes an element of $SoLA(\Gamma)$.*

Definition 3.14 *$E(\Gamma, x) \stackrel{def}{=} \{(x, \Phi) \in \Gamma \mid \Phi$ is $Reward(r)$ or $\neg Reward(r)$ or $Cost(\alpha, c)$ or $\neg Cost(\alpha, c)$ for some/any constants r and c and some/any action $\alpha\}$.*

Definition 3.15 *$E(\Gamma, LA, w) \stackrel{def}{=} \bigcup_{x:w \in LA(\Gamma)} E(\Gamma, x)$.*

In natural language, $E(\Gamma, LA, w)$ is a set of formulae (as defined) in Γ with labels x such that the labels could logically represent world w, that is, such that $w \models \ell$ for all $(x, \ell) \in \Gamma$, and LA is one of the ways in which worlds can be assigned to labels mentioned in Γ.

Definition 3.16 *$F(\Gamma, \alpha, x) \stackrel{def}{=} \{[\alpha]_q\varphi \mid (x, [\alpha]_q\varphi) \in \Gamma\} \cup \{\neg[\alpha]_q\varphi \mid (x, \neg[\alpha]_q\varphi) \in \Gamma\}$.*

Definition 3.17 *$F(\Gamma, \alpha, LA, w) \stackrel{def}{=} \bigcup_{x:w \in LA(\Gamma)} F(\Gamma, \alpha, x)$.*

In natural language, $F(\Gamma, \alpha, LA, w)$ is the set of dynamic literals mentioning α in Γ with labels x such that the labels could logically represent world w.

Definition 3.18 *$G(\Gamma, \alpha, x) \stackrel{def}{=} \{(\varsigma \mid \alpha : q) \mid (x, (\varsigma \mid \alpha : q)) \in \Gamma\} \cup \{\neg(\varsigma \mid \alpha : q) \mid (x, \neg(\varsigma \mid \alpha : q)) \in \Gamma\}$.*

Definition 3.19 *$G(\Gamma, \alpha, LA, w) \stackrel{def}{=} \bigcup_{x:w \in LA(\Gamma)} G(\Gamma, \alpha, x)$.*

In natural language, $G(\Gamma, \alpha, LA, w)$ is the set of perception literals mentioning α in Γ with labels x such that the labels could logically represent world w.

After the tableau phase has completed, the label assignment phase begins. For each leaf node Γ^j_k of an open branch, do the following.

Do the following for every $LA \in SoLA(\Gamma^j_k)$. If one of the following three cases holds, then mark LA as "unsat".

- For some $w \in W(\Gamma^j_k)$, $E(\Gamma^j_k, LA, w)$ contains
  - $Reward(r)$ and $Reward(r')$ such that $r \neq r'$, or
  - $Reward(r)$ and $\neg Reward(r)$, or
  - $Cost(\alpha, c)$ and $Cost(\alpha, c')$ (same action α) such that $c \neq c'$, or
  - $Cost(\alpha, c)$ and $\neg Cost(\alpha, c)$ (same action α).
- For some action $\alpha \in \mathcal{A}$ and some $w \in W(\Gamma^j_k)$, $Z(F(\Gamma^j_k, \alpha, LA, w)) = \emptyset$ or $Z(G(\Gamma^j_k, \alpha, LA, w)) = \emptyset$.

If every $LA \in SoLA(\Gamma^j_k)$ is marked as "unsat", then create new leaf node $\Gamma^j_{k+1} = \Gamma^j_k \cup \{(0, \bot)\}$.

That is, if for all logically correct ways of assigning possible worlds to labels (i.e., for all the label assignments in $SoLA(\Gamma^j_k)$), no assignment (LA) satisfies all formulae in Γ^j_k, then Γ^j_k is unsatisfiable.

Definition 3.20 *A tree is called* finished *after the label assignment phase is completed.*

Note that all branches of a finished tree are saturated.

Definition 3.21 *If a tree for $\neg\Psi$ is closed, we write $\vdash \Psi$. If there is a finished tree for $\neg\Psi$ with an open branch, we write $\not\vdash \Psi$.*

4 Soundness

Theorem 4.1 (Soundness) *If $\vdash \Psi$ then $\models \Psi$. (Contrapositively, if $\not\models \Psi$ then $\not\vdash \Psi$.)*

Let $\psi = \neg\Psi$. Then $\not\vdash \Psi$ if and only if the tree for ψ is open. And

$$\begin{aligned}\not\models \Psi &\iff \text{not } (\forall\mathcal{S})\ \mathcal{S} \models \Psi \\ &\iff \text{not } (\forall\mathcal{S}, w)\ \mathcal{S}, w \models \Psi \\ &\iff (\exists\mathcal{S}, w)\ \mathcal{S}, w \models \psi.\end{aligned}$$

For the soundness proof, it thus suffices to show that if there exists a structure $\mathcal{S}$ and w in it such that $\mathcal{S}, w \models \psi$, then the tree rooted at $\Gamma^0_0 = \{(0, \psi)\}$ is open.

Lemma 4.1 *Let Γ be the leaf node of a saturated tree. Suppose there exists a structure $\mathcal{S} = \langle W(\Gamma), R, O, N, Q, U\rangle$ such that for all $(x, \delta\omega) \in \Gamma$, where $\delta\omega$ is a dynamic or perception literal involving α, there exists a $w \in W(\Gamma)$ such that $\mathcal{S}, w \models \delta\omega$. Then there exists an $LA \in SoLA(\Gamma)$ such that for all $w \in W(\Gamma)$, $Z(F(\Gamma, \alpha, LA, w)) \neq \emptyset$ and $Z(G(\Gamma, \alpha, LA, w)) \neq \emptyset$.*

Proof:
We prove the contrapositive of the lemma. Assume that for all $LA \in SoLA(\Gamma)$, there exists a $w \in W(\Gamma)$ such that $Z(FG(\Gamma, \alpha, LA, w)) = \emptyset$.

Let LA be an arbitrary label assignment in $SoLA(\Gamma)$. Let $W(\Gamma)^{\#} = (w_1, w_2, \ldots, w_n)$ be an ordering of the worlds in $W(\Gamma)$, where $n = |W(\Gamma)|$. Let $\Omega^{\#} = (\varsigma_1, \varsigma_2, \ldots, \varsigma_m)$ be an ordering of the observations in Ω, where $m = |\Omega|$. Let w be an arbitrary world in W and let α be an arbitrary action in $\mathcal{A}$.

Let $\{[\alpha]_{q^\alpha_1}\varphi^\alpha_1, [\alpha]_{q^\alpha_2}\varphi^\alpha_2, \ldots, [\alpha]_{q^\alpha_g}\varphi^\alpha_g, \neg[\alpha]_{q^\alpha_{g+1}}\varphi^\alpha_{g+1}, \neg[\alpha]_{q^\alpha_{g+2}}\varphi^\alpha_{g+2}, \ldots, \neg[\alpha]_{q^\alpha_{g+h}}\varphi^\alpha_{g+h}\}$ be an ordered set of the dynamic literals in $F(\Gamma, \alpha, LA, w)$.

Let $\{(\varsigma_1 \mid \alpha : q_1), \ldots, (\varsigma_t \mid \alpha : q_t), \neg(\varsigma_{t+1} \mid \alpha : q_{t+1}), \ldots, \neg(\varsigma_{t+v} \mid \alpha : q_{t+v})\}$ be an ordered set of the perception literals in $G(\Gamma, \alpha, LA, w)$.

If $Z(F(\Gamma, \alpha, LA, w_k) = \emptyset$, there is no solution $(s_1^\alpha, s_2^\alpha, \ldots, s_n^\alpha)$ for which

$$\sum_{\substack{i=1 \\ R_\alpha(w_k,w_i)=s_i^\alpha \\ \mathcal{S},w_i\models\varphi_1^\alpha}}^{n} s_i^\alpha = q_1^\alpha \quad \text{and}$$

$$\sum_{\substack{i=1 \\ R_\alpha(w_k,w_i)=s_i^\alpha \\ \mathcal{S},w_i\models\varphi_2^\alpha}}^{n} s_i^\alpha = q_2^\alpha \quad \text{and}$$

$$\vdots$$

$$\sum_{\substack{i=1 \\ R_\alpha(w_k,w_i)=s_i^\alpha \\ \mathcal{S},w_i\models\varphi_{g+h}^\alpha}}^{n} s_i^\alpha \neq q_{g+h}^\alpha$$

such that $\sum_{w'\in W} R_\alpha(w_k, w') = 1$ or $\sum_{w'\in W} R_\alpha(w_k, w') = 0$.

If $Z(G(\Gamma, \alpha, LA, w_k) = \emptyset$, there is no solution $(s_1^\alpha, s_2^\alpha, \ldots, s_n^\alpha)$ for which

$$\begin{aligned} Q_\alpha(w_k, N(\sigma_1)) &= s_1^\sigma \quad \text{and} \\ Q_\alpha(w_k, N(\sigma_2)) &= s_2^\sigma \quad \text{and} \\ &\vdots \\ Q_\alpha(w_k, N(\sigma_{t+v})) &= s_{t+v}^\sigma, \end{aligned}$$

where $s_1^\sigma, s_2^\sigma, \ldots, s_{t+v}^\sigma \in \{s_1^\varsigma, s_2^\varsigma, \ldots, s_m^\varsigma\}$ and such that if $R_\alpha(w, w') > 0$, then $\sum_{o\in O} Q_\alpha(w', o) = 1$ (due to tableau rule obs), else if $R_\alpha(w, w') = 0$, then either $\sum_{o\in O} Q_\alpha(w', o) = 1$ or $\sum_{o\in O} Q_\alpha(w', o) = 0$.

Thus, there is no way to assign transition and perception probabilities such that R_α and Q_α conform to the definition of a SLAOP structure. That is, there is no SLAOP structure $\mathcal{S}$ such that for all $(x, \delta\omega) \in \Gamma$ there exists a $w \in W$ such that $\mathcal{S}, w \models \delta\omega$. Hence, the lemma holds. ■

Lemma 4.2 *Let T be a finished tree. For every node Γ in T: If there exists a structure $\mathcal{S}$ such that for all $(x, \Phi) \in \Gamma$ there exists a $w \in W$ such that $\mathcal{S}, w \models \Phi$, then the (sub)tree rooted at Γ is open.*

Proof:
(by induction on the height of the node Γ_k)

Base case: Height $h = 0; \Gamma_k$ is a leaf. If there exists a structure $\mathcal{S}$ such that for all $(x, \Phi) \in \Gamma_k$ there exists a $w \in W$ such that $\mathcal{S}, w \models \Phi$, then $(x', \bot) \notin \Gamma_k$ for all x'. Hence, the sub-tree consisting of Γ_k is open.

Induction step: If $h > 0$, then some rule was applied to create the child(ren) $\Gamma_{k'}$ of Γ_k. We abbreviate "there exists a structure $\mathcal{S}^j = \langle W^j, R^j, O^j, N^j, Q^j, U^j\rangle$ such that for all $(x^j, \Phi^j) \in \Gamma_j$ there exists a $w^j \in W^j$ such that $\mathcal{S}^j, w^j \models \Phi^j$" as $A(j)$ and we abbreviate "the (sub)tree rooted at Γ_j is open" as $B(j)$.

We must show the following for every rule/phase. IF: If $A(k')$, then $B(k')$, THEN: If $A(k)$, then $B(k)$. We assume the antecedent (induction hypothesis): If $A(k')$, then $B(k')$. To show the consequent, we must assume $A(k)$ and show that $B(k)$ follows.

Note that if the (sub)tree rooted at $\Gamma_{k'}$ is open, then the (sub)tree rooted at Γ_k is open. That is, if $B(k')$ then $B(k)$. So we want to show $B(k')$. But, by the induction hypothesis, $B(k')$ follows from $A(k')$. Therefore, it will suffice, in each case below, to assume $A(k)$, and prove $A(k')$.

- rule $=$:
  For the rule to have been applied, $(x, (c = c')) \subseteq \Gamma_k$ or $(x, \neg(c = c')) \subseteq \Gamma_k$. The rule is only applied when $(c = c')$, resp., $\neg(c = c')$ is unsatisfiable. Therefore, Γ_k is unsatisfiable. But this contradicts our main assumption $A(k)$. Hence, rule $=$ could not have been applicable to Γ_k.

- rule $\bot$:
  For the rule to have been applied, $\{(x, \Psi), (x, \neg\Psi)\} \subseteq \Gamma_k$, and after its application, $\Gamma_{k'} = \Gamma_k \cup \{(x, \bot)\}$. But there exists no structure $\mathcal{S}^k = \langle W^k, R^k\rangle$ such that there exists a $w^k \in W^k$ such that $\mathcal{S}^k, w^k \models \Psi$ and $\mathcal{S}^k, w^k \models \neg\Psi$. Hence, assumption $A(k)$ is false and this rule could not have been applied.

- rule $\neg$:
  For the rule to have been applied, $\{(x, \neg\neg\Psi)\} \subseteq \Gamma_k$, and after its application, $\Gamma_{k'} = \Gamma_k \cup \{(x, \Psi)\}$. By assumption, $\mathcal{S}^k, w^k \models \neg\neg\Psi$. Hence, $\mathcal{S}^k, w^k \models \Psi$. Thus, $A(k')$.

- rule $\wedge$:
  For the rule to have been applied, $\{(x, \Psi \wedge \Psi')\} \subseteq \Gamma_k$, and after its application, $\Gamma_{k'} = \Gamma_k \cup \{(x, \Psi), (x, \Psi')\}$. By assumption, $\mathcal{S}^k, w^k \models \Psi \wedge \Psi'$. Hence, $\mathcal{S}^k, w^k \models \Psi$ and $\mathcal{S}^k, w^k \models \Psi'$. Thus, $A(k')$.

- rule $\vee$:
  For the rule to have been applied, $\{(x, \Psi \vee \Psi')\} \subseteq \Gamma_k$, and after its application, either $\Gamma_{k'} = \Gamma_k \cup \{(x, \Psi)\}$ or $\Gamma_{k''} = \Gamma_k \cup \{(x, \Psi')\}$. By assumption, $\mathcal{S}^k, w^k \models \Psi \vee \Psi'$. Hence, $\mathcal{S}^k, w^k \models \Psi$ or $\mathcal{S}^k, w^k \models \Psi'$. Thus, $A(k')$ or $A(k'')$. Thus, $B(k')$ or $B(k'')$. Therefore, $B(k)$.

- rule $\Diamond\varphi$:
  For the rule to have been applied, $\{(0, \neg[\alpha]_0\varphi)\} \subseteq \Gamma_k$ or $\{(0, [\alpha]_q\varphi)\} \subseteq \Gamma_k$ for $q > 0$, and after its application, $\Gamma_{k'} = \Gamma_k \cup \{(x, \varphi)\}$ where x is a fresh integer.

  By assumption, there exists a $w \in W$ such that $\mathcal{S}, w \models \neg[\alpha]_0\varphi$ or $\mathcal{S}, w \models [\alpha]_q\varphi$. Then by definition of $\langle\alpha\rangle$, there exists a $w'' \in W$ such that $(w, w'', pr) \in R_\alpha$ for $pr > 0$ and $\mathcal{S}, w'' \models \varphi$. Hence, for all $(x, \Phi') \in \Gamma_{k'}$ there exists a $w' \in W$ such that $\mathcal{S}, w' \models \Phi'$.

- rule obs:
  For the rule to have been applied, $\{(x, \neg[\alpha]_0\varphi)\} \subseteq \Gamma_k$ or $\{(x, [\alpha]_q\varphi)\} \subseteq \Gamma_k$ for $q > 0$ and some x, and after its application, $\Gamma_{k'} = \Gamma_k \cup \{(0, \Box(\delta_1 \to (\exists v^\varsigma)\neg(v^\varsigma \mid \alpha : 0)) \vee \Box(\delta_2 \to (\exists v^\varsigma)\neg(v^\varsigma \mid \alpha : 0)) \vee \cdots \vee \Box(\delta_n \to (\exists v^\varsigma)\neg(v^\varsigma \mid \alpha : 0)))\}$, where $\delta_i \in \mathit{Def}(\varphi)$.

By assumption, there exists a $w \in W$ such that $\mathcal{S}, w \models \neg[\alpha]_0\varphi$ or $\mathcal{S}, w \models [\alpha]_q\varphi$. That is, there exists a $w'' \in W$ such that $R_\alpha(w, w'') > 0$ and $\mathcal{S}, w'' \models \varphi$.

By definition of a SLAOP structure, for all $w^-, w^+ \in W$: if $R_\alpha(w^-, w^+) > 0$, then $\sum_{o \in O}, Q_\alpha(w^+, o) = 1$. Thus, there exists a $w'' \in W$ such that $\sum_{o \in O} Q_\alpha(w'', o) = 1$, where $\mathcal{S}, w'' \models \varphi$. This implies that there exists a $w'' \in W$ such that $w'' \models \varphi$ and there exists at least one observation $\varsigma \in \Omega$ such that it is not the case that $Q_\alpha(w'', N(\varsigma)) = 0$. Hence, for all $w' \in W$ if $\mathcal{S}, w' \models \delta_1$ then $(\exists v^\varsigma)\neg(v^\varsigma \mid \alpha : 0))$ or for all $w' \in W$ if $\mathcal{S}, w' \models \delta_2$ then $(\exists v^\varsigma)\neg(v^\varsigma \mid \alpha : 0))$ or ... or for all $w' \in W$ if $\mathcal{S}, w' \models \delta_n$ then $(\exists v^\varsigma)\neg(v^\varsigma \mid \alpha : 0))$, where $\delta_i \in \mathit{Def}(\varphi)$. Therefore, there exists a $w \in W$ such that $\mathcal{S}, w \models \Box(\delta_1 \rightarrow (\exists v^\varsigma)\neg(v^\varsigma \mid \alpha : 0)) \vee \Box(\delta_2 \rightarrow (\exists v^\varsigma)\neg(v^\varsigma \mid \alpha : 0)) \vee \cdots \vee \Box(\delta_n \rightarrow (\exists v^\varsigma)\neg(v^\varsigma \mid \alpha : 0)$.

- rule $\Box$:
  For the rule to have been applied, $\{(0, \Box\Phi), (x, \Phi'')\} \subseteq \Gamma_k$ for some $x \geq 0$, and after its application, $\Gamma_{k'} = \Gamma_k \cup \{(x, \Phi)\}$.

  By assumption, there exist $w, w' \in W$ such that $\mathcal{S}, w \models \Box\Phi$ and $\mathcal{S}, w' \models \Phi''$. Then by definition of $\Box$, for all $w'' \in W, \mathcal{S}, w'' \models \Phi$. That is, there exists a $w'' \in W$ such that $\mathcal{S}, w'' \models \Phi$. Hence, for all $(x, \Phi') \in \Gamma_{k'}$ there exists a $w''' \in W$ such that $\mathcal{S}, w''' \models \Phi'$.

- rule $\Diamond$:
  For the rule to have been applied, $\{(0, \neg\Box\Phi)\} \subseteq \Gamma_k$, and after its application, $\Gamma_{k'} = \Gamma_k \cup \{(x, \neg\Phi)\}$ where x is a fresh integer.

  By assumption, there exists a $w \in W$ such that $\mathcal{S}, w \models \neg\Box\Phi$. Then by definition of $\Box$, there exists a $w' \in W$ such that $\mathcal{S}, w' \not\models \Phi$. That is, there exists a $w' \in W$ such that $\mathcal{S}, w' \models \neg\Phi$. Hence, for all $(x, \Phi') \in \Gamma_{k'}$ there exists a $w \in W$ such that $\mathcal{S}, w \models \Phi'$.

- Label Assignment Phase:
  If it can be shown that there exists an $LA \in \mathit{SoLA}(\Gamma_k)$ such that

  1. for no $w \in W(\Gamma_k)$, $E(\Gamma_k, LA, w)$ contains
     - $\mathit{Reward}(r)$ and $\mathit{Reward}(r')$ such that $r \neq r'$, or
     - $\mathit{Reward}(r)$ and $\neg\mathit{Reward}(r)$, or
     - $\mathit{Cost}(\alpha, c)$ and $\mathit{Cost}(\alpha, c')$ (same action α) such that $c \neq c'$, or
     - $\mathit{Cost}(\alpha, c)$ and $\neg\mathit{Cost}(\alpha, c)$ (same action α);
  2. for all $\alpha \in \mathcal{A}$ and all $w \in W(\Gamma_k)$, $Z(F(\Gamma_k, \alpha, LA, w)) \neq \emptyset$ and $Z(G(\Gamma_k, \alpha, LA, w)) \neq \emptyset$,

  then no child is created for Γ_k and trivially, for all $(x, \Phi') \in \Gamma_{k'}$ there exists a $w \in W$ such that $\mathcal{S}, w \models \Phi'$.

  Let LA be a member of $\mathit{SoLA}(\Gamma_k)$. By assumption, there exists a structure $\mathcal{S} = \langle W, R, O, N, Q, U\rangle$ such that for all $(x, \Phi) \in \Gamma_k$ there exists a $w \in W$ such that $\mathcal{S}, w \models \Phi$. The label assignment phase occurs only when Γ_k is the leaf node of a saturated tree. Thus, $w \in W(\Gamma_k, x)$. Each of the three cases is considered separately.

(1) For all $(x, \Phi) \in \Gamma_k$, either $x{:}w \in LA$ or $x{:}w \notin LA$. If $x{:}w \in LA$, then by the assumption, the first case must be true. If $x{:}w \notin LA$, then $\Phi \notin E(\Gamma_k, LA, w)$, and the first case is trivially true.

(2) By assumption, there exists a structure $\mathcal{S} = \langle W, R, O, N, Q, U\rangle$ such that for all $(x, \delta\omega) \in \Gamma_k$, where $\delta\omega$ is a dynamic or perception literal involving α, there exists a $w \in W$ such that $\mathcal{S}, w \models \delta\omega$.

Due to $w \in W(\Gamma_k, x)$, if $w \in W$, then $w \in W(\Gamma_k)$. That is, $W \subseteq W(\Gamma_k)$. Thus, a SLAOP structure $\mathcal{S}' = \langle W(\Gamma_k), R', O, N, Q', U\rangle$ can be constructed as follows. For all $\alpha \in \mathcal{A}$, for all $w, w' \in W(\Gamma_k)$, if $R_\alpha(w, w') = pr$, let $R'_\alpha(w, w') = pr$, else let $R'_\alpha(w, w') = 0$. Hence, for all $(x, \delta) \in \Gamma_k$ there exists a $w \in W(\Gamma_k)$ such that $\mathcal{S}', w \models \delta$, where δ is a dynamic literal involving α. And for all $\alpha \in \mathcal{A}$, for all $o \in O$, for all $w' \in W(\Gamma_k)$, if $Q_\alpha(w', o) = pr$, let $Q'_\alpha(w', o) = pr$, else let $Q'_\alpha(w', o) = 0$. Hence, for all $(x', \omega) \in \Gamma_k$ there exists a $w' \in W(\Gamma_k)$ such that $\mathcal{S}', w' \models \omega$, where ω is a perception literal involving α.

Then, by Lemma 4.1, there exists an $LA \in SoLA(\Gamma_k)$ such that for all $w \in W(\Gamma_k)$ and $\alpha \in \mathcal{A}$, $Z(F(\Gamma_k, \alpha, LA, w)) \neq \emptyset$ and $Z(G(\Gamma_k, \alpha, LA, w)) \neq \emptyset$.

Moreover, the systems of inequalities (SIs), as used in this report, can be described in the language of first-order logic elementary real number theory [15, 13, 5] (cf. Lem. 3.1). There exist sound methods for determining whether an SI is feasible [15, 13, 5]. In other words, there is a reliable means of determining whether there exists at least one solution to an SI. Action α is arbitrary; the second case is true.

■

5 Completeness

Theorem 5.1 (Completeness) *If* $\models \Psi$ *then* $\vdash \Psi$. *(Contrapositively, if* $\not\vdash \Psi$ *then* $\not\models \Psi$.*)*

Let $\psi = \neg\Psi$. Then $\not\vdash \Psi$ means that there is an open branch of a finished tree for ψ. And

$$\begin{aligned} \not\models \Psi &\iff (\exists \mathcal{S})\ \mathcal{S} \not\models \Psi \\ &\iff (\exists \mathcal{S}, w)\ \mathcal{S}, w \not\models \Psi \\ &\iff (\exists \mathcal{S}, w)\ \mathcal{S}, w \models \psi. \end{aligned}$$

For the completeness proof, it thus suffices to construct for some open branch of a finished tree for $\psi \in \mathcal{L}_{SLAOP}$, a SLAOP structure $\mathcal{S} = \langle W, R, O, N, Q, U\rangle$ in which there is a world $w \in W$ in $\mathcal{S}$ such that ψ is satisfied in $\mathcal{S}$ at w.

We now start with the description of the construction of a SLAOP structure, given the leaf node Γ of some open branch of a finished tree. $\mathcal{S} = \langle W, R, O, N, Q, U\rangle$ can be constructed as follows:

- Let $W = W(\Gamma)$.

- For every action $\alpha \in \mathcal{A}$, the accessibility relation R_α can be constructed as follows. Let $R_\alpha(w, w_j) = s_j^\alpha \iff$
  - $w \in W$,
  - $w_j \in W(\Gamma)^\#$,
  - $(s_1^\alpha, s_2^\alpha, \ldots, s_n^\alpha) \in Z(F(\Gamma, \alpha, LA, w))$.

  In other words, for every $w \in W$, determine which labels represent it according to the label assignment LA and then use any solution $(s_1^\alpha, s_2^\alpha, \ldots, s_n^\alpha)$ for the SI generated from $F(\Gamma, \alpha, LA, w)$ to assign the transition probabilities, where s_1^α is the probability of reaching world w_1 from world w, s_2^α is the probability of reaching world w_2 from world w, and so on until s_n^α/w_n.

- Let $O = \Omega$.

- Let $N = \{(o, o) \mid o \in O\}$.

- For every action $\alpha \in \mathcal{A}$, the perceivability relation Q_α can be constructed as follows. Recall that $\Omega^\# = (\varsigma_1, \varsigma_2, \ldots, \varsigma_m)$ is an ordering of Ω. Let $Q_\alpha(w_j, N(\varsigma_j)) = s_j^\varsigma \iff$
  - $\varsigma_j \in \Omega^\#$,
  - $w_j \in W(\Gamma)^\#$,
  - $(s_1^\varsigma, s_2^\varsigma, \ldots, s_m^\varsigma) \in Z(G(\Gamma_k, \alpha, LA, w_j))$,

  In other words, for every $w \in W$, determine which labels represent it according to the label assignment LA and then use any solution $(s_1^\varsigma, s_2^\varsigma, \ldots, s_m^\varsigma)$ for the SI generated from $G(\Gamma, \alpha, LA, w)$ to assign the perception probabilities, where s_1^ς is the probability of perceiving ς_1 in world w, s_2^ς is the probability of perceiving ς_2 in w, and so on until s_m^ς.

- If there is $(x, \neg \mathit{Reward}(r)) \in \Gamma$ for some x and r, then let $\mathit{maxRew}(\Gamma) = \max_r$ $(x, \neg \mathit{Reward}(r)) \in \Gamma$, else, let $\mathit{maxRew}(\Gamma) = 0$. For each $(x, \mathit{Reward}(r)) \in \Gamma$, let $\mathit{Re}(w) = r$, where $x{:}w \in X(\Gamma)$. For all $w \in W(\Gamma)$, if it's not the case that $(x, \mathit{Reward}(r)) \in \Gamma$, where $x{:}w \in X(\Gamma)$, then let $\mathit{Re}(w) = \mathit{maxRew}(\Gamma) + 1$. If there is $(x, \neg \mathit{Cost}(\alpha, c)) \in \Gamma$ for some x, α and c, then let $\mathit{maxCost}(\Gamma) = \max_c (x, \neg \mathit{Cost}(\alpha, c)) \in \Gamma$, else, let $\mathit{maxCost}(\Gamma) = 0$. For each $(x, \mathit{Cost}(\alpha, c)) \in \Gamma$, let $\mathit{Co}_\alpha(w) = c$, where $x{:}w \in X(\Gamma)$. For all $w \in W(\Gamma)$, if it's not the case that $(x, \mathit{Cost}(\alpha, c)) \in \Gamma$, where $x{:}w \in X(\Gamma)$, then let $\mathit{Co}_\alpha(w) = \mathit{maxCost}(\Gamma) + 1$. Let $U = \langle \mathit{Re}, \mathit{Co} \rangle$ such that $\mathit{Co} = \{(\alpha, \mathit{Co}_\alpha) \mid \alpha \in \mathcal{A}\}$.

Lemma 5.1 *$\mathcal{S}$ is a SLAOP structure.*

Proof:
The components of the structure are well-formed:

- $W = W(\Gamma) = \bigcup_{x \geq 0} \{w \in C \mid w \models \ell$ for all $(x, \ell) \in \Gamma$ where ℓ is a propositional literal$\}$. That is, $W = \{w \in C \mid$ for all $x, w \models \ell$ for all $(x, \ell) \in \Gamma$ where ℓ is a propositional literal$\}$. Thus, for W to be empty, it must be the case that for all $w \in C$ there exists some $(x, \ell) \in \Gamma$, for which $w \not\models \ell$. But this is a contradiction. Hence, W is not empty.

- Due to Γ being open (and by rule SI), we know that for all $\alpha \in \mathcal{A}$ and all $w \in W(\Gamma)$, there exists a solution in $Z(F(\Gamma, \alpha, LA, w))$.

  By construction, R maps each action $\alpha \in \mathcal{A}$ to R_α such that R_α is a relation in $(W \times W) \times \mathbb{R}_{[0,1]}$. Moreover, by the nature of the SI generated from $F(\Gamma, \alpha, LA, w)$, R_α is a (total) function $R_\alpha : (W \times W) \mapsto \mathbb{R}_{[0,1]}$.

  And by construction, the fact that $pr_1 + pr_2 + \cdots + pr_n = 1$ or $pr_1 + pr_2 + \cdots + pr_n = 0$ is an equation in any SI generated, either $\sum_{w' \in W} R_\alpha(w, w') = 1$ or $\sum_{w' \in W} R_\alpha(w, w') = 0$, for every $w \in W$.

- Assuming Ω is non-empty, O will be non-empty.

- By construction, $N : \Omega \mapsto O$ is a bijection.

- Due to Γ being open (and by rule SI), we know that for all $\alpha \in \mathcal{A}$ and all $w \in W(\Gamma)$, there exists a solution in $Z(G(\Gamma, \alpha, LA, w))$.

  By construction, Q maps each action $\alpha \in \mathcal{A}$ to Q_α such that Q_α is a relation in $(W \times O) \times \mathbb{R}_{[0,1]}$.

  Moreover, by the nature of the SI generated from $G(\Gamma, \alpha, LA, w)$, Q_α is a (total) function $Q_\alpha : (W \times O) \mapsto \mathbb{R}_{[0,1]}$.

  And by construction, the fact that in any SI generated, there is an equation $pr_1^\varsigma + pr_2^\varsigma + \cdots + pr_m^\varsigma = \lceil pr_1^\varsigma + pr_2^\varsigma + \cdots + pr_m^\varsigma \rceil$ and the fact that $\lceil pr_1^\varsigma + pr_2^\varsigma + \cdots + pr_m^\varsigma \rceil$ equals 0 or 1, it must be the case that $\sum_{o \in O} Q_\alpha(w, o)$ equals 0 or 1. Furthermore, due to tableau rule obs, if there exists a w' such that $R_\alpha(w', w) > 0$, then $Q_\alpha(w, o) > 0$, which implies that $\lceil pr_1^\varsigma + pr_2^\varsigma + \cdots + pr_m^\varsigma \rceil = 1$, which implies that: For all $w, w' \in W$: if $R_\alpha(w', w) > 0$ for some w', then $\sum_{o \in O} Q_\alpha, (w, o) = 1$.

- By construction, $U = \langle Re, Co \rangle$, where $Re : W \mapsto \mathbb{R}$ and Co is a mapping from $\mathcal{A}$ to a function $Co_\alpha : C \mapsto \mathbb{R}$. Suppose $x{:}w, x'{:}w \in LA$ where $x \neq x'$. If $(x, \mathit{Reward}(r)), (x', \mathit{Reward}(r')) \in \Gamma$ such that $r \neq r'$, then by construction $Re(w) = r = r'$ which is impossible. But if $(x, \mathit{Reward}(r)), (x', \mathit{Reward}(r'))$ were in Γ, then LA would have caused $\mathit{Reward}(r)$ and $\mathit{Reward}(r')$ to be in $E(\Gamma, LA, w)$ and the branch would have closed. So either (i) $x{:}w, x'{:}w \in LA$ but $\mathit{Reward}(r)$ and $\mathit{Reward}(r')$ are not both in Γ, or (ii) $(x, \mathit{Reward}(r)), (x', \mathit{Reward}(r')) \in \Gamma$ but $x{:}w$ and $x'{:}w$ are not both in LA, or (iii) $x{:}w$ and $x'{:}w$ are not both in LA and $\mathit{Reward}(r)$ and $\mathit{Reward}(r')$ are not both in Γ.

■

W.l.o.g., one can assume that, for every $(x, \Box\Phi) \in \Gamma$, Φ is in DNF.

Lemma 5.2 *Let Γ be the leaf node of a finished tree, where $(0, \Box\Phi) \in \Gamma$, for some $\Box\Phi \in \mathcal{L}_{SLAOP}$. For every label x mentioned in Γ, there exists a term $(\Phi_{k1} \wedge \Phi_{k2} \wedge \cdots \wedge \Phi_{km_k})$ of Φ such that (x, Φ_{k1}), (x, Φ_{k2}), ..., $(x, \Phi_{km_k}) \in \Gamma$.*

Proof:
Let $\Phi := t_1 \vee t_2 \vee \cdots \vee t_z$. Let the labels mentioned in Γ be $\{0, 1, 2, \ldots, x'\}$. Rule $\Box$ is applied to $(0, \Box\Phi)$ for every label $(0, 1, 2, \ldots, x')$. Hence, due to multiple applications of rule $\Box$, the following labeled formulae are in Γ: $(0, t_1 \vee t_2 \vee \cdots \vee t_z)$, $(1, t_1 \vee t_2 \vee \cdots \vee t_z)$, $(2, t_1 \vee t_2 \vee \cdots \vee t_z)$, ... $(x', t_1 \vee t_2 \vee \cdots \vee t_z)$. And due to multiple applications of rule $\vee$, one of the following sets is a subset of Γ.

- $\{(0,t_1),(1,t_1),(2,t_1),\ldots,(x',t_1)\}$,
- $\{(0,t_1),(1,t_1),(2,t_1),\ldots,(x',t_2)\}$,

$\vdots$

- $\{(0,t_1),(1,t_1),(2,t_1),\ldots,(x',t_z)\}$,
- $\{(0,t_2),(1,t_1),(2,t_1),\ldots,(x',t_1)\}$,
- $\{(0,t_2),(1,t_1),(2,t_1),\ldots,(x',t_2)\}$,

$\vdots$

- $\{(0,t_2),(1,t_1),(2,t_1),\ldots,(x',t_z)\}$,

$\vdots$

- $\{(0,t_z),(1,t_z),(2,t_z),\ldots,(x',t_z)\}$.

Now choose any one of these sets T. For every label $x \in \{0,1,2,\ldots,x'\}$, $(x,t_k) \in T \subset \Gamma$ for some term $t_k := \Phi_{k1} \wedge \Phi_{k2} \wedge \cdots \wedge \Phi_{km_k}$ of Φ. Therefore, due to successive applications of rule $\wedge$, (x,Φ_{k1}), (x,Φ_{k2}), ..., $(x,\Phi_{km_k}) \in \Gamma$, for every label x mentioned in Γ. ■

Lemma 5.3 *Let Γ be the leaf node of an open branch of a finished tree. We know that there exists a label assignment $LA \in SoLA(\Gamma)$ such that $Z(F(\Gamma,\alpha,LA,w))$ and $Z(G(\Gamma,\alpha,LA,w))$ are not empty, for all $w \in W(\Gamma)$ and all $\alpha \in \mathcal{A}$. If $\mathcal{S}$ is constructed as described above, then for all $(x,\Psi) \in \Gamma$, $\mathcal{S},w \models \Psi$ for $x{:}w \in LA$.*

Proof:

The proof will be by induction on the structure of a formula.

The induction step will work as follows. Let $\gamma' \subseteq \Gamma$ be added to Γ due to some rule applied to $\gamma \subseteq \Gamma$. Thus, we need to prove that IF for all $(x',\Psi') \in \gamma'$, $\mathcal{S},w' \models \Psi'$ for $x'{:}w' \in LA$, THEN for all $(x,\Psi) \in \gamma$, $\mathcal{S},w \models \Psi$ for $x{:}w \in LA$.

We assume the antecedent (induction hypothesis).

Base case:

- Ψ is a propositional literal. By definition, $\mathcal{S},w' \models \Psi$ for all $w' \in W(\Gamma,x)$. But if $x{:}w \in LA$, then $w \in W(\Gamma,x)$. Thus $\mathcal{S},w \models \Psi$.
- Ψ is $c = c'$. Because $(x,\bot) \notin \Gamma$ for some label x, rule $=$ was not applied. Hence, c is identical to c', and $\mathcal{S},w \models c = c'$.
- Ψ is $\neg(c = c')$. Because $(x,\bot) \notin \Gamma$ for some label x, rule $=$ was not applied. Hence, c is not identical to c', and $\mathcal{S},w \models \neg(c = c')$.
- Ψ is $Reward(r)$. By construction, $(w,r) \in Re$ for $(x, Reward(r)) \in \Gamma$ and $x{:}w \in LA$. Hence, $\mathcal{S},w \models Reward(r)$.
- Ψ is $\neg Reward(r)$. By construction, $(w,r) \notin Re$. Hence, $\mathcal{S},w \models \neg Reward(r)$.

- Ψ is $Cost(\alpha, c)$. By construction, $(w, c) \in Co_\alpha$ for $(x, Cost(\alpha, c)) \in \Gamma$ and $x{:}w \in LA$. Hence, $\mathcal{S}, w \models Cost(\alpha, c)$.
- Ψ is $\neg Cost(\alpha, c)$. By construction, $(w, c) \notin Co_\alpha$. Hence, $\mathcal{S}, w \models \neg Cost(\alpha, c)$.
- Ψ is $[\alpha]_q\varphi$. Then as a direct consequence of the construction of $\mathcal{S}$, $\mathcal{S}, w \models [\alpha]_q\varphi$.
- Ψ is $\neg[\alpha]_q\varphi$. Then as a direct consequence of the construction of $\mathcal{S}$, $\mathcal{S}, w \models \neg[\alpha]_q\varphi$.
- Ψ is $(\varsigma \mid \alpha : q)$. Then as a direct consequence of the construction of $\mathcal{S}$, $\mathcal{S}, w \models (\varsigma \mid \alpha : q)$.
- Ψ is $\neg(\varsigma \mid \alpha : q)$. Then as a direct consequence of the construction of $\mathcal{S}$, $\mathcal{S}, w \models \neg(\varsigma \mid \alpha : q)$.

Induction step:

- Ψ is $\neg\neg\psi$. By rule $\neg$, $(x, \psi) \in \Gamma$. By induction hypothesis, $\mathcal{S}, w \models \psi$ for $x{:}w \in LA$. By the definition of $\neg$, $\mathcal{S}, w \models \neg\neg\psi$.
- Ψ is $\psi \wedge \psi'$. By rule $\wedge$, $(x, \psi), (x, \psi') \in \Gamma$. By induction hypothesis, $\mathcal{S}, w \models \psi$ and $\mathcal{S}, w \models \psi'$ for $x{:}w \in LA$. By the definition of $\wedge$, $\mathcal{S}, w \models \psi \wedge \psi'$.
- Ψ is $\neg(\psi \wedge \psi')$. By rule $\vee$, $(x, \neg\psi) \in \Gamma$ or $(x, \neg\psi') \in \Gamma$ for $x{:}w \in LA$. By induction hypothesis, $\mathcal{S}, w \models \neg\psi$ or $\mathcal{S}, w \models \neg\psi'$. By the definition of $\vee$, $\mathcal{S}, w \models \neg(\psi \wedge \psi')$.
- Ψ is $\Box\Phi$. Let $X(\Gamma) = \{0, 1, 2, \ldots, x'\}$. Due to successive applications of rule $\Box$, $(0, \Phi), (1, \Phi), \ldots, (x', \Phi) \in \Gamma$. Then, by induction hypothesis, $\mathcal{S}, w_0 \models \Phi$ for $0{:}w_0 \in LA$ and $\mathcal{S}, w_1 \models \Phi$ for $1{:}w_1 \in LA$ and $\cdots$ and $\mathcal{S}, w_{x'} \models \Phi$ for $x'{:}w_{x'} \in LA$.

  We need to show that $\mathcal{S}, w \models \Box\Phi$ for $0{:}w_0 \in LA$, that is, that for all $w' \in W(\Gamma)$, $\mathcal{S}, w' \models \Phi$. This will be the case if: For every $w' \in W(\Gamma)$, there exists a term $t := \Phi_1 \wedge \Phi_2 \wedge \cdots \wedge \Phi_m$ of Φ such that $\mathcal{S}, w' \models t$. Note that for $w_i, w_j \in W(\Gamma)$, if $w_i \neq w_j$, it is sufficient that there exist terms t_i and t_j of Φ such that $\mathcal{S}, w_i \models t_i$ and $\mathcal{S}, w_j \models t_j$, even if $t_i \neq t_j$.

  By Lemma 5.2, for every label $x \in X(\Gamma)$, there exists a term $t := \Phi_1 \wedge \Phi_2 \wedge \cdots \wedge \Phi_m$ of Φ such that (x, Φ_1), (x, Φ_2), $\ldots$, $(x, \Phi_m) \in \Gamma$.

  Then we define the set

  $$L(t) := \{\Phi_i \mid \Phi_i \text{ is a propositional literal conjunct of } t\},$$

  the set

  $$ERC(t) := \{\Phi_i \mid \Phi_i \text{ is an erc literal conjunct of } t\}$$

  the set

  $$\Delta(t) := \{\Phi_i \mid \Phi_i \text{ is a dynamic literal conjunct of } t\}$$

  and the set

  $$\Omega(t) := \{\Phi_i \mid \Phi_i \text{ is a or perception literal conjunct of } t\}.$$

  Note that $t \equiv \bigwedge_{\ell \in L(t)} \ell \wedge \bigwedge_{\rho \in ERC(t)} \rho \wedge \bigwedge_{\delta\omega \in \Delta\Omega(t)} \delta\omega$.

Let $\ell \in L(t)$. Then by induction hypothesis, $\mathcal{S}, w'' \models \ell$ for $x{:}w'' \in LA$. Note that if $w'' \models \ell$ for some $w'' \in W(\Gamma, x)$, then $w^* \models \ell$ for all $w^* \in W(\Gamma, x)$. Thus,

$$\mathcal{S}, w^* \models \bigwedge_{\ell \in L(t)} \ell \text{ (for all } w^* \in W(\Gamma, x)),$$

and by definition of $W(\Gamma)$,

$$\mathcal{S}, w' \models \bigwedge_{\ell \in L(t)} \ell \text{ (for all } w' \in W(\Gamma)). \tag{3}$$

Let $\rho \in ERC(t)$. Then by induction hypothesis, $\mathcal{S}, w'' \models \rho$ for $x{:}w'' \in LA$. If ρ is $(b = b')$ or $\neg(b = b')$, then

$$\mathcal{S}, w''' \models \rho \text{ (for all } w''' \in W(\Gamma)). \tag{4}$$

By construction (see base case), for all $w' \in W(\Gamma)$,

$$\mathcal{S}, w' \models \bigwedge_{\rho \in E(\Gamma, LA, w)} \rho \tag{5}$$

and

$$\mathcal{S}, w' \models \bigwedge_{\delta \in F(\Gamma, LA, w)} \delta. \tag{6}$$

and

$$\mathcal{S}, w' \models \bigwedge_{\omega \in G(\Gamma, LA, w)} \omega. \tag{7}$$

By (4), (5) and Lemma 5.2, for all $w' \in W(\Gamma)$,

$$\mathcal{S}, w' \models \bigwedge_{\rho \in ERC(t)} \rho.$$

By (6) and Lemma 5.2, for all $w' \in W(\Gamma)$,

$$\mathcal{S}, w' \models \bigwedge_{\delta \in \Delta(t)} \delta.$$

By (7) and Lemma 5.2, for all $w' \in W(\Gamma)$,

$$\mathcal{S}, w' \models \bigwedge_{\omega \in \Omega(t)} \omega.$$

Hence, for all $w' \in W(\Gamma)$, there exists a $x \in X(\Gamma)$ such that

$$\mathcal{S}, w' \models \bigwedge_{\substack{i=1 \\ (x, \Phi_i) \in \Gamma}}^{m} \Phi_i,$$

(where $\bigwedge^{m}_{\substack{i=1 \\ (x, \Phi_i) \in \Gamma}} \Phi_i$ is a term of Φ) which implies that for all $w' \in W(\Gamma)$,

$$\mathcal{S}, w' \models \Phi,$$

which concludes the proof.

- Ψ is $\neg\Box\Phi$. By rule $\Diamond$, $(x', \neg\Phi) \in \Gamma$ for some $x' > x$. By induction hypothesis, $\mathcal{S}, w' \models \neg\Phi$ for $x':w' \in LA$. That is, it is not the case that for all $w'' \in W(\Gamma), \mathcal{S}, w'' \models \Phi$. Hence, $\mathcal{S}, w \not\models \Box\Phi$, if and only if $\mathcal{S}, w \models \neg\Box\Phi$.

■

Corollary 5.1 *By Lemma 5.3, given the leaf node Γ of an open branch of a finished tree, there exists a structure $\mathcal{S}$ such that for all $(x, \Psi) \in \Gamma$, $\mathcal{S}, w \models \Psi$ for x:$w \in LA$. But $(0, \psi) \in \Gamma$. Thus, if there is a finished open tableau for ψ, then ψ is satisfiable.*

Theorem 5.1 follows directly from Corollary 5.1.

6 Termination

Definition 6.1 *Let Φ' be a strict sub-part of Φ. A tableau rule has the* subformula property *if and only if the new node(s) (Γ') created by the application of the rule, contains (x, Φ'), respectively, $(x, \neg\Phi')$ for some x, where $(x, \Phi') \notin \Gamma$, respectively, $(x, \neg\Phi') \notin \Gamma$.*

Lemma 6.1 *A tree for any formula $\Phi \in \mathcal{L}_{SLAOP}$ becomes saturated. That is, the tableau phase terminates.*

Proof:
We can divide all the tableau rules into three categories: (i) those which add $\bot$ to the new node, (ii) those with the subformula property and (iii) rule obs. Category-(i) rules never cause rules to become applicable later. As a direct consequence of sentences being finite and their subformula property, every category-(ii) rule must eventually become inapplicable. Rule obs is reproduced here: If Γ_k^j contains $(x, \neg[\alpha]_0\varphi)$ or $(x, [\alpha]_q\varphi)$ for $q > 0$ and some x, then create node $\Gamma_{k+1}^j = \Gamma_k^j \cup \{(0, \Box(\delta_1 \rightarrow (\exists v^\varsigma)\neg(v^\varsigma \mid \alpha : 0)) \vee \Box(\delta_2 \rightarrow (\exists v^\varsigma)\neg(v^\varsigma \mid \alpha : 0)) \vee \cdots \vee \Box(\delta_n \rightarrow (\exists v^\varsigma)\neg(v^\varsigma \mid \alpha : 0)))\}$, where $\delta_i \in Def(\varphi)$. Note that $\Box(\delta_1 \rightarrow (\exists v^\varsigma)\neg(v^\varsigma \mid \alpha : 0)) \vee \Box(\delta_2 \rightarrow (\exists v^\varsigma)\neg(v^\varsigma \mid \alpha : 0)) \vee \cdots \vee \Box(\delta_n \rightarrow (\exists v^\varsigma)\neg(v^\varsigma \mid \alpha : 0))$ is not dynamic; it can thus not make rule obs applicable. That is, rule obs can only cause category-(i) and category-(ii) rules to become applicable.

Therefore, all rules eventually become inapplicable, and it follows that any tree (for any formula) would become saturated. ■

Theorem 6.1 *The decision procedure for SLAOP terminates.*

Proof:
Due to Lemma 6.1, the tableau phase terminates (with a finite number of branches).

In the SI phase: for each open branch of a tree for Φ, a solution set for an SI is sought once for each action in $\mathcal{A}$ for each world in $W(\Gamma)$, where Γ is the leaf node of an open branch. Hence, a solution set for an SI is sought a finite number of times in the label assignment phase.

By Lemma 3.1, finding the solution set for an SI is decidable as used in the label assignment phase and the process thus terminates in this phase. ■

Corollary 6.1 *The validity problem for the SLAOP is decidable.*

Proof:
Because the procedure is sound and complete, it will be decidable if it always terminates, which, by Theorem 6.1, it does. ■

7 Conclusion

A logic for specifying partially observable Markov decision processes (minus belief-states) was presented. We proved that the logic is decidable—with respect to validity of sentences—by showing that an appropriate decision procedure is sound, complete and always terminating.

This work forms a step towards defining a decidable logic for reasoning with incompletely specified POMDPs, including the ability to posed queries about (i) the degree of belief in a propositional sentence after an arbitrary number of actions and observations and (ii) the utility of a sequence of actions after an arbitrary number of actions and observations.

References

[1] K. Aström. Optimal control of Markov decision processes with incomplete state estimation. *J. Math. Anal. Appl.*, 10:174–205, 1965.

[2] P. Blackburn, M. De Rijke, and Y. Venema. *Modal Logic*. Cambridge University Press, Cambridge, UK, 2001.

[3] A. Chagrov and M. Zakharyaschev. *Modal Logic (Oxford Logic Guides, Vol. 35)*. Oxford University Press, Oxford, England, 1997.

[4] B. Chellas. *Modal Logic: an introduction*. Cambridge University Press, Cambridge, MA, 1980.

[5] G. Collins. Quantifier elimination for real closed fields by cylindrical algebraic decomposition. In *Second GI Conf on Automata Theory and Formal Languages*, pages 515–532, 1975.

[6] G. Hughes and M. Cresswell. *A New Introduction to Modal Logic*. Routledge, New York, NY, 1996.

[7] G. E. Monahan. A survey of partially observable Markov decision processes: Theory, models, and algorithms. *Management Science*, 28(1):1–16, 1982.

[8] S. Popkorn. *First Steps in Modal Logic*. Cambridge University Press, 1994.

[9] G. Rens. From an agent logic to an agent programming language for partially observable stochastic domains. In T. Walsh, editor, *Proc. of 22nd Intl. Joint Conf. on Artif. Intell. (IJCAI-11)*, pages 2842–2843, Menlo Park, CA, July 2011. AAAI Press.

[10] G. Rens, G. Lakemeyer, and T. Meyer. A logic for specifying agent actions and observations with probability. In K. Kersting and M. Toussaint, editors, *Proc. of 6th Starting AI Researchers' Symposium (STAIRS 2012)*, volume 241 of *Frontiers in Artificial Intelligence and Applications*, pages 252–263. IOS Press, 2012. url: http://www.booksonline.iospress.nl/Content/View.aspx?piid=31509.

[11] G. Rens, T. Meyer, and G. Lakemeyer. A logic for specifying stochastic actions and observations. In *Proc. of 19th Intl. Conf. on Logic for Programming, Artificial Intelligence and Reasoning (LPAR-19)*, Heidelberg, Germany, December 2013. Springer-Verlag.

[12] G. Rens, T. Meyer, and G. Lakemeyer. On the logical specification of probabilistic transition models. In *Proc. of 11th Intl. Symposium on Logical Formalizations of Commonsense Reasoning (COMMONSENSE 2013)*, May 2013.

[13] A. Seidenberg. A new decision method for elementary algebra. *Annals of Mathematics*, 60:365–374, 1954.

[14] R. Smallwood and E. Sondik. The optimal control of partially observable Markov processes over a finite horizon. *Operations Research*, 21:1071–1088, 1973.

[15] A. Tarski. A decision method for elementary algebra and geometry. Technical report, The RAND Corporation, Santa Monica, Calif., 1957.