(Co)inductive Proof Systems for Compositional Proofs in Reachability Logic

Vlad Rusu
Inria
Lille, France

David Nowak
CRIStAL*
Lille, France

Reachability Logic is a formalism that can be used, among others, for expressing partial-correctness properties of transition systems. In this paper we present three proof systems for this formalism, all of which are sound and complete and inherit the coinductive nature of the logic. The proof systems differ, however, in several aspects. First, they use induction and coinduction in different proportions. The second aspect regards compositionality, broadly meaning their ability to prove simpler formulas on smaller systems, and to reuse those formulas as lemmas for more complex formulas on larger systems. The third aspect is the difficulty of their soundness proofs. We show that the more induction a proof system uses, and the more specialised is its use of coinduction (with respect to our problem domain), the more compositional the proof system is, but the more difficult its soundness proof becomes. We also briefly present mechanisations of these results in the Isabelle/HOL and Coq proof assistants.

1 Introduction

Reachability Logic (RL) [18] has been introduced as a language-parametric program logic: a formalism for specifying the functional correctness of programs, which may belong to any programming language whose operational semantics is also specified in RL. The functional correctness of a program is stated as the validity of a set of RL formulas (specifying the program's expected properties) with respect to another set of RL formulas (specifying the operational semantics of the language containing the program).

Such statements are proved by means of a proof system, which has adequate meta-properties with respect to validity: soundness (i.e., only valid RL formulas can be proved) and relative completeness (all valid RL formulas can, in principle, be proved, modulo the existence of "oracles" for auxiliary tasks). The proof of meta-properties for the RL proof system is highly nontrivial, but it only needs to be done once.

Program logics already have a half-century history between them, from the first occurrence of Hoare logic [5] to contemporary separation logics [11]. However, all those logics depend on a language's syntax and therefore have to be defined over and over again, for each new language (or even, for each new language version). In particular, the meta-properties of the corresponding proof systems should be reproved over and over again, a tedious task that is often postponed to an indeterminate future.

Despite being language-parametric, Reachability Logic does not come in only one version. Several versions of the logic have been proposed over the years [13, 19, 18]. The formalism has been generalised from programming languages to more abstract models: rewriting logic [8, 17] and transition systems [14], which can be used for specifying designs, and verifying them before they are implemented in program code. This does not replace code verification, just as code verification does not replace the testing of the final running software; but it enables the early catching of errors and the early discovery of key functional-correctness properties, all of which are known to have practical, cost-effective benefits.

*Univ. Lille, CNRS, Centrale Lille, UMR 9189 - CRIStAL - Centre de Recherche en Informatique Signal et Automatique de Lille, F-59000 Lille, France

To appear in EPTCS.

Contributions. We further study RL on transition systems (TS). We propose three proof systems for RL, and formalise them in the Coq [1] and Isabelle/HOL [10] proof assistants. One may naturally ask: why having several proof systems and proof assistants - why not one of each? The answer is manyfold:

- the proof systems we propose have some common features: the soundness and completenes meta-properties, and the coinductiveness nature inherited from RL. However, they do differ in others aspects: (i) the "amount" of induction they contain; (ii) their degree of compositionality (i.e., their ability to prove local formulas on "components" of a TS, and then to use those formulas as lemmas in proofs of global formulas on the TS); and (iii) the difficulty level of their soundness proofs.
- we show that the more induction a proof system uses, and the closest its coinduction style to our problem domain of proving reachability-logic formulas, the more compositional the proof system is, but the more difficult its soundness proof. There is a winner: the most compositional proof system of the three, but we found that the other ones exhibit interesting, worth-presenting features as well.
- Coq and Isabelle/HOL have different styles of coinduction: Knaster-Tarski style vs. Curry-Howard style. Experiencing this first-hand with the nontrivial examples constituted by proof systems suggested a spinoff project, which amounts to porting some of the features of one proof assistant into the other one. For the moment, porting Knaster-Tarski features into the Curry-Howard coinduction of Coq produced promising results, with possible practical impact for a broader class of Coq users.

Related Work. Regarding RL, most papers in the above-given list of references mention its coinductive nature, but do not actually use it. Several Coq mechanisations of soundness proofs for RL proof systems are presented, but Coq's coinduction is absent from them. In [3, 7] coinduction is used for formalising RL and for proving RL properties for programs and for term-rewriting systems, but their approach is not mechanised in a proof assistant. More closely related work to ours is reported in [9]; they attack, however, the problem exactly in the opposite way: they develop a general theory of coinduction in Coq and use it to verify programs directly based on the semantics of programming languages, i.e., without using a proof system. They do show that a proof system for RL is an instance of their approach for theoretical reasons, in order to give a formal meaning to the completeness of their approach.

Regarding coinduction in Isabelle/HOL, which is based on the Knaster-Tarski fixpoint theorems, we used only a small portion of what is available: coinductive predicates, primitive coinductive datatypes and primitive corecursive functions. More advanced developments are reported in [2]. Regarding coinduction in Coq, it is based on the Curry-Howard isomorphism that views proofs as programs, hence, coinductive proofs are well-formed corecursive programs [4]. An approach that bridges the gap between this and the Knaster-Tarski style of coinduction is [6]. A presentation of our own results on porting Knaster-Tarski style coinduction to Coq and a detailed comparison with the above is left for future work.

Regarding coinduction in formal methods, we note that it is mostly used for proving bisimulations. The book [16] serves as introduction to both these notions and explores the relationships between them.

Regarding compositional verification, most existing techniques decompose proofs among parallel composition operators. Various compositional methods for various parallel composition operators (rely-guarantee for variable-based composition, assumption-commitment for synchronisation-based composition, ...) are presented in the book [12]. We employ compositionality in a different sense - structural, for transition systems, and logical, for formulas. We note, however, that many of the techniques presented in [12] have a coinductive nature, which could perhaps be exploited in future versions of RL proof systems.

Organisation. The next section recaps preliminary notions: Knaster-Tarski style induction and coinduction, transition systems, and RL on transition systems. A first compositionality result, of RL-validity with respect to certain sub-transition systems, is given. The three following sections present our three proof systems in increasing order of complexity. Soundness and completeness results are given and a notion of compositionality with respect to formulas, in two versions: asymmetrical and symmetrical, is introduced and combined with the compositionality regarding sub-transition systems. The three proof systems are shown to have increasingly demanding compositionality features. We then briefly discuss the mechanisations of the proof systems in the Coq and Isabelle/HOL proof assistants before we present future work and conclude. Most proofs are placed in a separated Appendix, for better readability. The Coq and Isabelle/HOL formalisations are available at `http://project.inria.fr/from2019`.

2 Preliminaries

2.1 Induction and Coinduction

Consider a complete lattice $(L,\sqsubseteq,\sqcup,\sqcap,\bot,\top)$ and a monotone function $F : L \to L$. According to the Knaster-Tarski fixpoint theorem, F has a least fixpoint μF (respectively, greatest fixpoint νF), which is the least (respectively, greatest) element of L such that $F(x) \sqsubseteq x$ (respectively, $x \sqsubseteq F(x)$). From this one deduces Tarski's induction and coinduction principles: $F(x) \sqsubseteq x$ implies $\mu F \sqsubseteq x$, and $x \sqsubseteq F(x)$ implies $x \sqsubseteq \nu F$.

Those principles can be used to define inductive and coinductive datatypes and recursive and corecursive functions. For example, the type of natural numbers is defined as the least fixpoint of the function $F(X) = \{0\} \cup \{Suc(x) \mid x \in X\}$. The greatest fixpoint of F is the type of natural numbers with infinity.

As another example, let $\mathcal{S} = (S,\to)$ be a transition system where S is the set of states and $\to\subseteq S \times S$ is the transition relation. A state s is *final*, and we write $\bullet\, s$, if there exists no s' such that $s \to s'$. A path is a nonempty, possibly infinite sequence of states. More formally, the set *Paths* of paths is the greatest fixpoint νF, where $F(X) = \{s \mid \bullet s\} \cup \{s\tau \mid s \in S \wedge \tau \in X \wedge s \to (hd\,\tau)\}$, with $hd : Paths \to S$ being simultaneously defined as $hd(s) = s$ and $hd(s\tau) = s$ for all $s \in S$ and $\tau \in X$. One can then corecursively define the length of a path as a value in the natural numbers with infinity: $len\, s = 0$ and $len(s\tau) = Suc(len\,\tau)$.

Hereafter, whenever necessary, we emphasise the fact that certain notions are relative to a transition system $\mathcal{S}$ by postfixing them with $_{\mathcal{S}}$. We omit this subscript when it can be inferred from the context.

A complete lattice associated to a transition system $\mathcal{S} = (S,\to)$, is the set of state predicates Π defined as the set of functions from S to the set of Booleans $\mathbb{B} = \{\mathrm{F},\mathrm{T}\}$. Its operations are defined by $p \sqsubseteq q \triangleq \forall s, p\,s \Rightarrow q\,s$, $(p \sqcup q)\,s \triangleq p\,s \vee q\,s$, $(p \sqcap q)\,s \triangleq p\,s \wedge q\,s$, $\bot\, s \triangleq \mathrm{F}$, $\top\, s \triangleq \mathrm{T}$. We also extend the transition relation $\to$ of $\mathcal{S}$ into a *symbolic transition function* $\partial : \Pi \to \Pi$, defined by $\partial p \triangleq \lambda s\,.\,\exists s'\,.\,p\,s' \wedge s' \to s$.

It is sometimes convenient to use a stronger variant of Tarski's coinduction principle: $X \sqsubseteq F(X \sqcup \nu F)$ iff $X \sqsubseteq \nu F$. Regarding induction, it is sometimes convenient to use *continuous* functions, i.e., functions F such that $F(\bigsqcup_{i\in I} x_i) = \bigsqcup_{i\in I}(F\,x_i)$, and use Kleene's fixpoint theorem: μF exists and is equal to $\bigsqcup_{n=0}^{\infty} F^n(\bot)$.

2.2 Reachability Formulas

We adapt Reachability Logic to transition systems. Assume a transition system $\mathcal{S} = (S,\to)$. Syntactically, a reachability formula (or, simply, a formula) over $\mathcal{S}$ is a pair $p \Rightarrow\!\!\Diamond\, q$ with $p,q \in \Pi$. We let $lhs(p \Rightarrow\!\!\Diamond\, q) \triangleq p$ and $rhs(p \Rightarrow\!\!\Diamond\, q) \triangleq q$. We denote by $\Phi_{\mathcal{S}}$ the set of all reachability formulas over the transition system $\mathcal{S}$.

Example 1 *Figure 1 depicts an extended finite-state machine having three natural-number variables: i,s, and m, and three control nodes: c_0, c_1, and c_2. Arrows connect the nodes and are possibly decorated with a Boolean guard and a set of parallel assignments of the variables. The variable m is never assigned, thus, it stays constant. The purpose of the machine is to compute in s the sum of the first m natural numbers.*

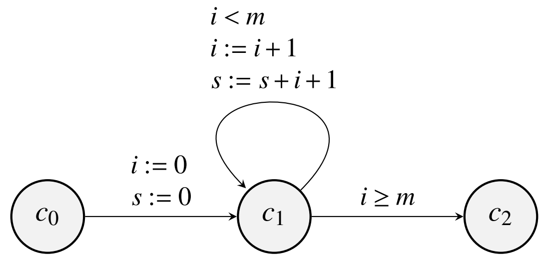

Figure 1: Sum up to m

The machine is also a finite representation of an infinite-state transition system whose set of states is the Cartesian product $\{c_0, c_1, c_2\} \times \mathbb{N} \times \mathbb{N} \times \mathbb{N}$. *and whose transition relation is* $\bigcup_{i,s,m \in \mathbb{N}} \{((c_0, i, s, m), (c_0, 0, 0, m))\} \cup \bigcup_{i,s,m \in \mathbb{N}, i<m} \{((c_1, i, s, m), (c_1, i+1, s+i+1, m))\} \cup \bigcup_{i,s,m \in \mathbb{N}, i \geq m} \{((c_1, i, s, m), (c_2, i, s, m))\}$. *A formula expressing the transition systems's functional correctness is* $(c = c_0) \Rightarrow\Diamond (c = c_2 \wedge s = m \times (m+1)/2)$.

In order to define the semantics of reachability formulas we first introduce the following relation.

Definition 1 $\rightsquigarrow$ *is the largest set of pairs* $(\tau, r) \in Paths \times \Pi$ *such that: (i)* $\tau = s$ *for some* $s \in S$, *and* $r\, s$*; or (ii)* $\tau = s\tau'$, *for some* $s \in S$, $\tau' \in Paths$, *and* $r\, s$*; or (iii)* $\tau = s\tau'$ *for some* $s \in S$, $\tau' \in Paths$, *and* $(\tau', r) \in \rightsquigarrow$.

We write $\tau \rightsquigarrow r$ for $(\tau, r) \in \rightsquigarrow$. Tarski's principle induces the following coinduction principle for $\rightsquigarrow$:

Lemma 1 *For* $R \subseteq Paths \times \Pi$, *if for all* $(\tau, r) \in R$, *it holds that either* $(\exists s. \tau = s \wedge r\, s)$, *or* $(\exists s. \exists \tau'. \tau = s\tau' \wedge r\, s)$ *or* $(\exists s. \exists \tau'. \tau = s\tau' \wedge (\tau', r) \in R)$, *then* $R \subseteq \rightsquigarrow$.

Definition 2 (Validity) *A formula* $\varphi \in \Phi_{\mathcal{S}}$ *is* valid *over* $\mathcal{S}$, *denoted by* $\mathcal{S} \models \varphi$, *whenever for all* $\tau \in Paths_{\mathcal{S}}$ *such that* $(lhs\, \varphi)(hd\, \tau)$ *holds, it also holds that* $\tau \rightsquigarrow_{\mathcal{S}} (rhs\, \varphi)$.

Example 2 *The formula* $(c = c_0) \Rightarrow\Diamond (c = c_2 \wedge s = m \times (m+1)/2)$ *is valid over the transition system denoted by the state-machine depicted in Figure 1. Intuitively, this means that all finite paths "starting" in the control node* c_0 *"eventually reach"* c_2 *with* $s = m \times (m+1)/2$ *holding. The "eventually reach" expression justifies the* $\Rightarrow\Diamond$ *notation borrowed from Linear Temporal Logic (LTL). Indeed, reachability formulas are essentially LTL formulas for a certain version of LTL interpreted over finite paths.*

We close the section with a simple notion of *component* of a transition system, and show that, if a formula is valid on a component, then it is valid on the whole transition system.

Definition 3 (Component) *A transition system* $(S', \rightarrow')$ *is a* component *of* $(S, \rightarrow)$ *if*

- $S' \subseteq S$ *and* $\rightarrow' \subseteq \rightarrow$;
- *for all* $s', s \in S'$, $s' \rightarrow s$ *implies* $s' \rightarrow' s$;
- *for all* $s' \in S'$, $s \in S \setminus S'$, $s' \rightarrow s$ *implies* $s' \in \bullet_{\mathcal{S}'}$.

We write $\mathcal{S}' \lhd \mathcal{S}$ *when* $\mathcal{S}'$ *is a component of* $\mathcal{S}$.

That is, $\mathcal{S}'$ is a full sub-transition system of $\mathcal{S}$, and one may only "exit" from $\mathcal{S}'$ via its final states. We often interchangeably use sets of states and their characteristic predicates, like we did for $\bullet_{\mathcal{S}'}$ above.

Theorem 1 (Compositionality of $\models$ w.r.t transition systems) $\mathcal{S}' \lhd \mathcal{S}$ *and* $\mathcal{S}' \models \varphi$ *imply* $\mathcal{S} \models \varphi$.

Example 3 *In Figure 1, the self-loop on the control node* c_1 *denotes a transition system* $\mathcal{S}'$ *that is a component of the transition system* $\mathcal{S}$ *denoted by the whole state machine. Let* $\varphi \triangleq (c = c_1 \wedge i = 0 \wedge s = 0) \Rightarrow\Diamond (c = c_1 \wedge i = m \wedge s = i \times (i+1)/2)$. *One can show that* $\mathcal{S}' \models \varphi$, *thus,* φ *is also valid over* $\mathcal{S}$.

One could, in principle, prove the validity of reachability formulas directly from the semantical definitions. However, this has several disadvantages: lack of a methodology - each formula is proved in its own ad-hoc way, and lack of a notion of completeness - is there a uniform way for proving every valid formula? These issues are addressed by the proof systems presented by increasing order of complexity in the next sections.

$$[\mathsf{Stp}]\ \frac{\mathcal{S} \vdash \partial l' \Rightarrow\!\Diamond\, r}{\mathcal{S} \vdash l \Rightarrow\!\Diamond\, r}\nu \ \ \textit{if } l \sqsubseteq l' \sqcup r, l' \sqcap \bullet \sqsubseteq \bot$$

Figure 2: One-rule proof system.

3 A One-Rule Proof System

Our first proof system is depicted as the one-rule inference system in Figure 2. It is parameterised by a transition system $\mathcal{S}$, and everything therein depends on it; we omit $_\mathcal{S}$ subscripts for simplicity. Intuitively, an application of the [Stp] rule can be seen as a symbolic execution step, taking a formula $l \Rightarrow\!\Diamond\, r$ and "moving" l "one step closer" to r - specifically, taking an over-approximation l' of the "difference" between l and r (encoded in the side-condition $l \sqsubseteq l' \sqcup r$) that contains no final states ($l' \sqcap \bullet \sqsubseteq \bot$) and performing a symbolic execution step from l' (encoded in the ∂ symbolic transition function). The rule is applicable infinitely many times, hence the ν symbol next to it. Note that there are no hypotheses in the proof system: those would be reachability formulas in the left-hand side of the $\vdash$ symbol, not allowed here.

For a more formal definition, consider the function $F : \mathcal{P}(\Phi) \to \mathcal{P}(\Phi)$ defined by

$$F(X) = \bigcup_{l, l' r \in \Pi,\, l \sqsubseteq l' \sqcup r,\, l' \sqcap \bullet \sqsubseteq \bot,\, \partial l' \Rightarrow\Diamond\, r \in X} \{l \Rightarrow\!\Diamond\, r\}$$

F is monotone, and, by Knaster-Tarski's theorem, F has a greatest fixpoint νF. We now define $\mathcal{S} \vdash \varphi$ by $\varphi \in \nu F$. Tarski's coinduction principle then induces the following coinduction principle for $\vdash$:

Lemma 2 *For all set $X \subseteq \Phi$ of hypotheses and $\varphi \in X$, if* for all $l \Rightarrow\!\Diamond\, r \in X$, there is $l' \in \Pi$ such that $l \sqsubseteq l' \sqcup r$, $l' \sqcap \bullet \sqsubseteq \bot$ and $\partial l' \Rightarrow\!\Diamond\, r \in X$, *then $\mathcal{S} \vdash \varphi$.*

Soundness. Soundness means that only valid formulas are proved:

Theorem 2 (Soundness of $\vdash$) *$\mathcal{S} \vdash \varphi$ implies $\mathcal{S} \models \varphi$.*

The proof uses the coinduction principle of the $\leadsto$ relation (Lemma 1), which occurs in the definition of validity, instantiated with the relation $R \subseteq \mathit{Paths} \times \Pi$ defined by $R \triangleq \lambda(\tau, r).\exists l.(\mathcal{S} \vdash l \Rightarrow\!\Diamond\, r \wedge l(hd\tau))$. As a general observation, all proofs by coinduction use a specific coinduction principle instantiated with a specific predicate/relation. The instantiation step is where the user's creativity is most involved.

Completeness. Completeness is the reciprocal to soundness: any valid formula is provable. It is based on the following lemma, which essentially reduces reachability to a form of inductive invariance.

Lemma 3 *If $l \sqsubseteq q \sqcup r$, $q \sqcap \bullet \sqsubseteq \bot$, and $\partial q \sqsubseteq q \sqcup r$ then $\mathcal{S} \vdash l \Rightarrow\!\Diamond\, r$.*

The proof of this lemma uses Lemma 2 with an appropriate instantion of the set X therein.

Example 4 *In order to establish $\mathcal{S}' \models (c = c_1 \wedge i = 0 \wedge s = 0) \Rightarrow\!\Diamond\, (c = c_1 \wedge i = m \wedge s = i \times (i+1)/2)$ - which has been claimed in Example 3 - one can use Lemma 3 with $q \triangleq (c = c_1 \wedge i < m \wedge s = i \times (i+1)/2)$.*

Theorem 3 (Completeness of $\vdash$) *$\mathcal{S} \models \varphi$ implies $\mathcal{S} \vdash \varphi$.*

The proof of completeness is constructive: it uses the predicate $q \triangleq \lambda s.\neg rs \wedge \forall \tau \in \mathit{Paths}.(s = hd\,\tau \Rightarrow \tau \leadsto r)$ that, for valid formulas $l \Rightarrow\!\Diamond\, r$, is shown to satisfy the three inclusions of Lemma 3. One may wonder: even when one does not know whether a formula $l \Rightarrow\!\Diamond\, r$ is valid, can one still use the above-defined q

and Lemma 3 in order to prove it? The answer is negative: proving the first implication $l \sqsubseteq q \sqcup r$ with the above-defined q amounts to proving validity directly from the semantics of formulas, thus losing any benefit of having a proof system. Hence, completeness is a theoretical property; the practically useful property is Lemma 3, which users have to provide with a suitable q that satisfy the three inclusions therein. In [15] we use this lemmma for verifying an infinite-state transition-system specification of a hypervisor.

Looking back at the proof system $\vdash$, we note that it is purely coinductive - no induction is present at all. This is unlike the proof systems in forthcoming sections. Regarding compositionality (with respect to transition systems) our proof system has it, since, by soundness and completeness and Theorem 1, one has that $\mathcal{S}' \lhd \mathcal{S}$ and $\mathcal{S}' \vdash \varphi$ implies $\mathcal{S} \vdash \varphi$. However, we show below that $\vdash$ does not have another, equally desirable compositionality feature: *asymmetrical compositionality* with respect to formulas.

Asymmetrical compositionality with respect to formulas. A proof system with this feature decomposes a proof of a formula φ into a proof of a formula φ' and one of φ assuming φ'. The asymmetry between the formulas involed suggested the property's name. In Definition 4 below, $\Vdash$ is a binary relation - a subset of $\mathcal{P}(\Phi) \times \Phi$ (equivalently, a predicate of type $\mathcal{P}(\Phi) \to \Phi \to \mathbb{B}$), parameterised by a transition system $\mathcal{S}$. For *hypotheses* $\mathcal{H} \subseteq \Phi$ and $\varphi \in \Phi$, we write $\mathcal{S}, \mathcal{H} \Vdash \phi$ for $(\mathcal{H}, \phi) \in \Vdash$ and $\mathcal{S} \Vdash \phi$ for $\mathcal{S}, \emptyset \Vdash \phi$.

Definition 4 (Asymmetrical compositionality with respect to formulas) *A proof system $\Vdash$ is asymmetrically compositional with respect to formulas if $\mathcal{S} \Vdash \varphi'$ and $\mathcal{S}, \{\varphi'\} \Vdash \varphi$ imply $\mathcal{S} \Vdash \varphi$.*

The proof system $\vdash$ is not asymmetrically compositional w.r.t. formulas, because that requires hypotheses, which $\vdash$ does not have. One could add hypotheses to it, and a new rule to prove a formula if it is found among the hypotheses. However, note that, unlike the [Stp] rule, the new rule has an inductive nature: it can only occur a finite number of times in a $\vdash$ proof (specifically, at most once, at the end of a finite proof).

4 An Asymmetrically-Compositional Proof System

In this section we propose another proof system $\Vdash$ and show that it is compositional with respect to transition systems and asymmetrically compositional with respect to formulas. These gains are achieved thanks to a the introduction of inductive rules in the proof system, enabling a better distribution of roles between these rules and the remaining coinductive rule; all at the cost of a more involved soundness proof.

Our second proof system is depicted in Figure 3. It is a binary relation - a subset of $\mathcal{P}(\Phi) \times \Phi$ (or equivalently, a binary predicate of type $\mathcal{P}(\Phi) \to \Phi \to \mathbb{B}$), parameterised by a transition system $\mathcal{S}$. Intuitively, the rule [Stp], labelled with ν, is coinductive, i.e., it can be applied infinitely many times, and the rules [Hyp], [Trv], [Str], [Spl], and [Tra], labelled by μ are inductive, i.e., they can only be applied finitely many times between two consecutive applications of [Stp]. Stated differently, a proof in $\Vdash$ is a possibly infinite series of *phases*, and in each phase there are finitely many applications of [Hyp], [Trv], [Str], [Spl], and [Tra] and, except in the last phase (if such a last phase exists), one application of [Stp].

Note that making the inductive rules coinductive would compromise soundness, because, e.g., the [Str] rule could forever reduce a proof of any formula to itself, thus proving any formula, valid or not.

The roles of the rules are the following ones. [Hyp] allows one to prove a formula if it is among the hypotheses. [Trv] is in charge of proving trivially valid formulas. [Str] is a general principle that amounts to strengthening a formula before proving it. [Spl] is used for getting rid of disjunctions in left-hand sides of formulas, which occur when several, alternative symbolic behaviours are explored in a proof search. [Tra] is a transitivity rule, used for proving facts about sequential symbolic behaviour. Note also the asymmetry in hypotheses of the rule [Tra]: for one formula validity is required, while for the other

$$\text{[Hyp]}\ \frac{}{\mathcal{S},\mathcal{H} \Vdash \varphi}\mu \quad \mathit{if}\ \varphi \in \mathcal{H}$$

$$\text{[Trv]}\ \frac{}{\mathcal{S},\mathcal{H} \Vdash r \Rightarrow\Diamond\, r}\mu$$

$$\text{[Str]}\ \frac{\mathcal{S},\mathcal{H} \Vdash l' \Rightarrow\Diamond\, r}{\mathcal{S},\mathcal{H} \Vdash l \Rightarrow\Diamond\, r}\mu \quad \mathit{if}\ l \sqsubseteq l'$$

$$\text{[Spl]}\ \frac{\mathcal{S},\mathcal{H} \Vdash l_1 \Rightarrow\Diamond\, r \qquad \mathcal{S},\mathcal{H} \Vdash l_2 \Rightarrow\Diamond\, r}{\mathcal{S},\mathcal{H} \Vdash (l_1 \sqcup l_2) \Rightarrow\Diamond\, r}\mu$$

$$\text{[Tra]}\ \frac{\mathcal{S} \models l \Rightarrow\Diamond\, m \qquad \mathcal{S},\mathcal{H} \Vdash m \Rightarrow\Diamond\, r}{\mathcal{S},\mathcal{H} \Vdash l \Rightarrow\Diamond\, r}\mu$$

$$\text{[Stp]}\ \frac{\mathcal{S},\mathcal{H} \Vdash \partial l \Rightarrow\Diamond\, r}{\mathcal{S},\mathcal{H} \Vdash l \Rightarrow\Diamond\, r}\nu \ \mathit{if}\ l \sqcap \bullet \sqsubseteq \bot$$

Figure 3: Mixed inductive-coinductive proof system.

one, it is provability. This asymmetry is used to avoid technical difficulties that arise when proving the soundness of $\Vdash$, but, as we shall see, it generates difficulties of its own. Finally, [Stp] makes the connection between the concrete paths and the symbolic ones, which the proof system explores during proof search.

For a formal definition: consider the following functions from $\mathcal{P}(\Phi)$ to $\mathcal{P}(\Phi)$ defined by

- $\vdash^{\text{[Hyp]}}_{\mathcal{S},\mathcal{H},Y}(X) = \mathcal{H}$
- $\vdash^{\text{[Trv]}}_{\mathcal{S},\mathcal{H},Y}(X) = \bigcup_{r\in\Pi}\{r \Rightarrow\Diamond\, r\}$
- $\vdash^{\text{[Str]}}_{\mathcal{S},\mathcal{H},Y}(X) = \bigcup_{l,l',r\in\Pi,\, l' \Rightarrow\Diamond\, r \in X,\, l\sqsubseteq l'}\{l \Rightarrow\Diamond\, r\}$
- $\vdash^{\text{[Spl]}}_{\mathcal{S},\mathcal{H},Y}(X) = \bigcup_{l_1,l_2,r\in\Pi,\, \{l_1 \Rightarrow\Diamond\, r,\, l_2 \Rightarrow\Diamond\, r\}\subseteq X}\{(l_1 \sqcup l_2) \Rightarrow\Diamond\, r\}$
- $\vdash^{\text{[Tra]}}_{\mathcal{S},\mathcal{H},Y}(X) = \bigcup_{l,r,m\in\Pi,\, \mathcal{S}\models l \Rightarrow\Diamond\, m,\, m \Rightarrow\Diamond\, r\in X}\{l \Rightarrow\Diamond\, r\}$
- $\vdash^{\text{[Stp]}}_{\mathcal{S},\mathcal{H},Y}(X) = \bigcup_{l,r\in\Pi,\, l\sqcap\bullet\sqsubseteq\bot,\, \partial l \Rightarrow\Diamond\, r\in Y}\{l \Rightarrow\Diamond\, r\}$

Let $\vdash_{\mathcal{S},\mathcal{H},Y}(X) = \vdash^{\text{[Hyp]}}_{\mathcal{S},\mathcal{H},Y}(X) \cup \vdash^{\text{[Trv]}}_{\mathcal{S},\mathcal{H},Y}(X) \cup \vdash^{\text{[Str]}}_{\mathcal{S},\mathcal{H},Y}(X) \cup \vdash^{\text{[Spl]}}_{\mathcal{S},\mathcal{H},Y}(X) \cup \vdash^{\text{[Tra]}}_{\mathcal{S},\mathcal{H},Y}(X) \cup \vdash^{\text{[Stp]}}_{\mathcal{S},\mathcal{H},Y}(X)$. It is not hard to show that $\vdash_{\mathcal{S},\mathcal{H},Y} : \mathcal{P}(\Phi) \to \mathcal{P}(\Phi)$ is continuous, thus, by the Knaster Tarski and Kleene fixpoint theorems it has a smallest fixpoint $\mu\vdash_{\mathcal{S},\mathcal{H},Y} = \bigcup_{n=1}^{\infty} \vdash^{n}_{\mathcal{S},\mathcal{H},Y}(\emptyset)$. Now, we define the function $F_{\mathcal{S},\mathcal{H}} : \mathcal{P}(\Phi) \to \mathcal{P}(\Phi)$ by $F_{\mathcal{S},\mathcal{H}}(Y) = \mu\vdash_{\mathcal{S},\mathcal{H},Y}$. $F_{\mathcal{S},\mathcal{H}}$ is monotone, thus, it has a greatest fixpoint $\nu F_{\mathcal{S},\mathcal{H}} = \nu(\lambda Y.\mu\vdash_{\mathcal{S},\mathcal{H},Y}) = \nu\mu\vdash_{\mathcal{S},\mathcal{H}}$.

We define the proof system $\Vdash$ as follows : for all $\mathcal{H} \subseteq \Phi$ and $\varphi \in \Phi$, $\mathcal{S},\mathcal{H} \Vdash \phi$ iff $\varphi \in \nu\mu\vdash_{\mathcal{S},\mathcal{H}}$. The inductive-coiductive nature of $\Vdash$ is visible from its definition. It admits the following coinduction principle:

Lemma 4 *If $X \subseteq \mu\vdash_{\mathcal{S},\mathcal{H},\,X}$ then for all $\varphi \in X$ it holds that $\mathcal{S},\mathcal{H} \Vdash \varphi$.*

Using the coinduction principle. For proving statements of the form $\mathcal{S},\mathcal{H} \Vdash \varphi$, one can:

- find a sequence $X = X_0, \cdots X_n = \emptyset$ of sets such that $X_i \subseteq \vdash_{\mathcal{S},\mathcal{H},X}(X_{i+1})$, for $i = 0,\ldots,n-1$, and $\varphi \in X$;
- since $\mu\vdash_{\mathcal{S},\mathcal{H},X} = \bigcup_{n=1}^{\infty} \vdash^{n}_{\mathcal{S},\mathcal{H},X}(\emptyset)$, we obtain by induction on n that $X_i \subseteq \mu\vdash_{\mathcal{S},\mathcal{H},X}$ for $i = 0,\ldots,n-1$ and in particular $X \subseteq \mu\vdash_{\mathcal{S},\mathcal{H},X}$. By Lemma 4, $\mathcal{S},\mathcal{H} \Vdash \varphi$.

We illustrate the above approach by proving a key lemma for the completeness of $\Vdash$.

Lemma 5 *If $l \sqsubseteq q \sqcup r$, $q \sqcap \bullet \sqsubseteq \bot$, and $\partial q \sqsubseteq q \sqcup r$ then $\mathcal{S} \Vdash l \Rightarrow\Diamond\, r$.*

Proof We apply the general approach described above. Note that $\mathcal{H} = \emptyset$. We choose $X = X_0 = \{l \Rightarrow\!\Diamond\, r, q \Rightarrow\!\Diamond\, r, \partial q \Rightarrow\!\Diamond\, r\}$.

- Let $X_1 = \{(q \sqcup r) \Rightarrow\!\Diamond\, r, q \Rightarrow\!\Diamond\, r, \partial q \Rightarrow\!\Diamond\, r\}$; using the hypothesis $l \sqsubseteq q \sqcup r$, $X_0 \subseteq \vdash^{[\mathsf{Str}]}_{\mathcal{S},\emptyset,X} (X_1) \subseteq \vdash_{\mathcal{S},\emptyset,X} (X_1)$;
- Let $X_2 = \{q \Rightarrow\!\Diamond\, r, r \Rightarrow\!\Diamond\, r, \partial q \Rightarrow\!\Diamond\, r\}$; we obtain $X_1 \subseteq \vdash^{[\mathsf{Spl}]}_{\mathcal{S},\emptyset,X} (X_2) \subseteq \vdash_{\mathcal{S},\emptyset,X} (X_2)$;
- Let $X_3 = \{q \Rightarrow\!\Diamond\, r, \partial q \Rightarrow\!\Diamond\, r\}$; we obtain $X_2 \subseteq \vdash^{[\mathsf{Trv}]}_{\mathcal{S},\emptyset,X} (X_3) \subseteq \vdash_{\mathcal{S},\emptyset,X} (X_3)$;
- Let $X_4 = \{\partial q \Rightarrow\!\Diamond\, r\}$; using the second hypothesis $q \sqcap \bullet \sqsubseteq \perp$ and the fact that $\partial q \Rightarrow\!\Diamond\, l \in X$ we obtain $X_3 \subseteq \vdash^{[\mathsf{Stp}]}_{\mathcal{S},\emptyset,X} (X_4) \subseteq \vdash_{\mathcal{S},\emptyset,X} (X_4)$;
- Let $X_5 = \{(q \sqcup r) \Rightarrow\!\Diamond\, r\}$; using the hypothesis $\partial q \sqsubseteq q \sqcup r$, we obtain $X_4 \subseteq \vdash^{[\mathsf{Str}]}_{\mathcal{S},\emptyset,X} (X_5) \subseteq \vdash_{\mathcal{S},\emptyset,X} (X_5)$;
- Let $X_6 = \{q \Rightarrow\!\Diamond\, r, r \Rightarrow\!\Diamond\, r\}$; we obtain $X_5 \subseteq \vdash^{[\mathsf{Spl}]}_{\mathcal{S},\emptyset,X} (X_6) \subseteq \vdash_{\mathcal{S},\emptyset,X} (X_6)$;
- Let $X_7 = \{q \Rightarrow\!\Diamond\, r\}$; we obtain $X_6 \subseteq \vdash^{[\mathsf{Trv}]}_{\mathcal{S},\emptyset,X} (X_7) \subseteq \vdash_{\mathcal{S},\emptyset,X} (X_7)$;
- Let $X_8 = \emptyset$; using the second hypothesis $q \sqcap \bullet \sqsubseteq \perp$ and the fact that $\partial q \Rightarrow\!\Diamond\, l \in X$, we obtain $X_7 \subseteq \vdash^{[\mathsf{Stp}]}_{\mathcal{S},\emptyset,X} (X_8) \subseteq \vdash_{\mathcal{S},\emptyset,X} (X_8)$.

Hence, by basic properties of inclusion, $X \subseteq \bigcup_{n=0}^{7} \vdash^{n}_{\mathcal{S},\emptyset,X} (\emptyset) \subseteq \bigcup_{n=0}^{\infty} \vdash^{n}_{\mathcal{S},\emptyset,X} (\emptyset) = \mu \vdash_{\mathcal{S},\emptyset,X}$, and from $l \Rightarrow\!\Diamond\, r \in X$ and Lemma 4 we obtain $\mathcal{S} \vdash l \Rightarrow\!\Diamond\, r$. ◻

Soundness. We define the recursive function $\mathit{suf} : \{\tau \in \mathit{Paths}\} \to \{i : \mathbb{N} \mid i \leq (\mathit{len}\,\tau)\} \to \mathit{Paths}$ by $\mathit{suf}\,\tau\,0 = \tau$ and $\mathit{suf}\,(s\,\tau)(i+1) = \mathit{suf}\,\tau\,i$. Intuitively, $\mathit{suf}\,\tau\,i$ is the sequence obtained by removing $i \leq (\mathit{len}\,\tau)$ elements from the "beginning" of τ. This is required in the definition of the following relation and is used hereafter.

Definition 5 *$\hookrightarrow \subseteq \mathit{Paths} \times \Pi$ is the largest set of pairs (τ, r) such that: (i) $\tau = s$ for some $\in S$ such that $r\,s$; or (ii) $\tau = \ s\tau'$, for some $s \in S$, $\tau' \in \mathit{Paths}$ such that $r\,s$; or (iii) $\tau = \ s\tau'$ for some $s \in S$, $\tau' \in \mathit{Paths}$ and $n \leq (\mathit{len}\,\tau')$ such that $((\mathit{suf}\,\tau'\,n), r) \in \hookrightarrow$.*

We write $\tau \hookrightarrow r$ instead of $(\tau, r) \in \hookrightarrow$. By analogy with Lemma 1 (coinduction principle for the $\rightsquigarrow$ relation), but using Tarski's strong induction principle, we obtain:

Lemma 6 *Let $R \subseteq \mathit{Paths} \times \Pi$ be s.t. $(\tau, r) \in R \Rightarrow (\exists s.\tau = s \wedge r\,s) \vee (\exists s.\exists \tau'.\tau = s\tau' \wedge r\,s) \vee (\exists s.\exists \tau'.\exists n.\exists \tau''.\tau = s\tau' \wedge \tau'' = (\mathit{suf}\,\tau'\,n) \wedge (\tau'', r) \in R \vee \tau'' \hookrightarrow r))$. Then $R \subseteq \hookrightarrow$.*

The following lemma is easily proved, by instantiating the parameter relation R, which occurs in both the coinduction principles of the relations $\rightsquigarrow, \hookrightarrow$, with the other relation:

Lemma 7 ($\rightsquigarrow$ equals $\hookrightarrow$) *For all $\tau \in \mathit{Paths}$ and $r \in \Pi$, $\tau \rightsquigarrow r$ if and only if $\tau \hookrightarrow r$.*

Using the coinduction principle for $\hookrightarrow$ and the above equality, as well as the induction principle for the functional $\vdash_{\mathcal{S},\mathcal{H},\nu\mu\vdash_{\mathcal{S},\mathcal{H}}}$ we obtain, in a rather involved proof mixing induction and coinduction:

Theorem 4 (Soundness of $\Vdash$) *If for all $\varphi' \in \mathcal{H}$, $\mathcal{S} \models \varphi'$, then $\mathcal{S}, \mathcal{H} \Vdash \varphi$ implies $\mathcal{S} \models \varphi$.*

Example 5 *We sketch a proof of the fact that the transition system $\mathcal{S}$ denoted by the state machine in Figure 1 meets its functional correcteness property: (i) $\mathcal{S} \models (c = c_0) \Rightarrow\!\Diamond\, (c = c_2 \wedge s = m \times (m+1)/2)$. We first show (ii) $\mathcal{S} \models (c = c_0) \Rightarrow\!\Diamond\, (c = c_1 \wedge i = 0 \wedge s = 0)$, which can be done using in sequence the rules [Stp], [Str], and [Trv] of the $\Vdash$ proof system together with its soundness. Using (ii) and the [Tra] rule, (i) reduces to proving (iii) $\mathcal{S} \Vdash (c = c_1 \wedge i = 0 \wedge s = 0) \Rightarrow\!\Diamond\, (c = c_2 \wedge s = m \times (m+1)/2)$. Next, in Examples 3 and 4 we established[1] $\mathcal{S} \models (c = c_1 \wedge i = 0 \wedge s = 0) \Rightarrow\!\Diamond\, (c = c_1 \wedge i = m \wedge s = i \times (i+1)/2)$, hence, using this and the [Tra] rule, (iii) reduces to proving $\mathcal{S} \Vdash (c = c_1 \wedge i = m \wedge s = i \times (i+1)/2) \Rightarrow\!\Diamond\, (c = c_2 \wedge s = m \times (m+1)/2)$. This is performed by applying in sequence the rules [Stp], [Str], and [Trv], which concludes the proof.*

[1]Example 4 used the proof system $\Vdash$ and its Lemma 3, but $\Vdash$ and its corresponding Lemma 5 can be used just as well.

Completeness. By analogy with Theorem 3, but using Lemma 5 instead of Lemma 3:

Theorem 5 (Completeness of $\Vdash$) *$\mathcal{S} \models \varphi$ implies $\mathcal{S} \Vdash \varphi$.*

Compositionality. Remembering Definition 4 of asymmetrical compositionality w.r.t formulas:

Theorem 6 *$\Vdash$ is asymetrically compositional with respect to formulas.*

Proof We have to show that if (i) $\mathcal{S} \Vdash \varphi'$ and (ii) $\mathcal{S}, \{\varphi'\} \Vdash \varphi$ then $\mathcal{S} \Vdash \varphi$. Now, (i) and (ii) and the soundness of $\Vdash$ imply $\mathcal{S} \models \varphi'$ and then $\mathcal{S} \models \varphi$, and then the conclusion $\mathcal{S} \Vdash \varphi$ holds by the completeness of $\Vdash$. □

Note that the statement (i) can be replaced by a weaker $\mathcal{S}' \Vdash \varphi'$ for components $\mathcal{S}' \lhd \mathcal{S}$, thanks to the soundness and completenesss of $\Vdash$ and of Theorem 1. This allows us to mix the compositionality of $\Vdash$ with respect to transition systems and the asymetrical one with respect to formulas.

The $\Vdash$ proof system is thus better at compositionality than $\vdash$, thanks to the inclusion of inductive rules, in particular, of the rule [Hyp], but at the cost of a more involved soundness proof. It still has a problem: the asymmetry of the [Tra] rule, required by the soundness proof, is not elegant since the rule mixes semantics $\models$ and syntax $\Vdash$. This is not only an issue of elegance, but a practical issue as well.

Example 6 *We attempt to prove the property (i) from Example 5 using the asymmetrical compositionality of $\Vdash$ w.r.t formulas. The first step, similar to that of Example 5, is proving (ii') $\mathcal{S} \Vdash (c = c_0) \Rightarrow\!\Diamond (c = c_1 \wedge i = 0 \wedge s = 0)$ by using in sequence the rules [Stp], [Str], and [Trv] of $\Vdash$. Then, Theorem 6 reduces (i) to (iii') $\mathcal{S}, \{(c = c_0) \Rightarrow\!\Diamond (c = c_1 \wedge i = 0 \wedge s = 0)\} \Vdash (c = c_0) \Rightarrow\!\Diamond (c = c_2 \wedge s = m \times (m+1)/2)$. The natural next step would be to use the [Tra] rule of $\Vdash$, splitting (iii') in two parts: $\mathcal{S}, \{(c = c_0) \Rightarrow\!\Diamond (c = c_1 \wedge i = 0 \wedge s = 0)\} \Vdash (c = c_0) \Rightarrow\!\Diamond (c = c_1 \wedge i = 0 \wedge s = 0)$, discharged by [Hyp], and then $\mathcal{S}, \{(c = c_0) \Rightarrow\!\Diamond (c = c_1 \wedge i = 0 \wedge s = 0)\} \Vdash (c = c_1 \wedge i = 0 \wedge s = 0) \Rightarrow\!\Diamond \Rightarrow\!\Diamond (c = c_1 \wedge i = 0 \wedge s = 0)$. But the [Tra] rule of $\Vdash$, as it is, does not allow this. Hence, when one uses compositionality, one may get stuck in proofs because of technical issues with [Tra].*

These issues are solved in the third proof system, which incorporates even more induction that the second one, and specialises its coinduction even closer to our problem domain. The third proof system also has better compositionality features. These gains come at the cost of an even more involved soundness proof.

5 A Symmetrically-Compositional Proof System

Our third proof system is depicted in Figure 4. A first difference with the previous one is that hypotheses and conclusions are pairs of a Boolean tag and a formula. We call them tagged formulas, or simply formulas when there is no risk of confusion. The role of the tags is to avoid unsoundness.

The second difference is that the proof system is essentially inductive, i.e., there are no more infinite proofs, and no coinduction principle any more; whatever coinduction remains is tailored to our problem and emulated by the proof system, as can be seen seen below in the description of the proof system's rules.

Another difference, especially with the second proof system $\Vdash$, is that the hypotheses set is not constant. The following rules change the hypotheses set. First, the [Cut] rule, which says that in order to prove (b, φ) under hypotheses $\mathcal{H}$, it is enough to prove (F, φ') - for some formula φ' - under hypotheses $\mathcal{H}$, and to prove (b, φ) under $\mathcal{H} \cup \{(\mathrm{F}, \varphi')\}$. This resembles a standard cut rule, but it is taylored to our specific setting. Second, the [Cof] rule adds a "copy" of the conclusion in the hypotheses, but tagged with F - and the new conclusion is also tagged with F. It is called this way in reference to the Coq `cofix` tactic that builds coinductive proofs in Coq also by copying a conclusion in the hypotheses; hence, we emulate in our proof system's hypotheses a certain existing coinduction mechanism, and taylor it to proving reachability formulas. Note that, without the tags, one could simply assume any formula by [Cof] and prove it by [Hyp],

$$\text{[Hyp]}\ \frac{}{\mathcal{S},\mathcal{H} \Vdash (\mathrm{T},\varphi)}\mu \ \ \textit{if}\ (\mathrm{F},\varphi) \in \mathcal{H}$$

$$\text{[Trv]}\ \frac{}{\mathcal{S},\mathcal{H} \Vdash (b, r \Rightarrow\!\Diamond\, r)}\mu$$

$$\text{[Str]}\ \frac{\mathcal{S},\mathcal{H} \Vdash (b, l' \Rightarrow\!\Diamond\, r)}{\mathcal{S},\mathcal{H} \Vdash (b, l \Rightarrow\!\Diamond\, r)}\mu \ \ \textit{if}\ l \sqsubseteq l'$$

$$\text{[Spl]}\ \frac{\mathcal{S},\mathcal{H} \Vdash (b, l_1 \Rightarrow\!\Diamond\, r) \qquad \mathcal{S},\mathcal{H} \Vdash (b, l_1 \Rightarrow\!\Diamond\, r)}{\mathcal{S},\mathcal{H} \Vdash (b, (l_1 \sqcup l_2)) \Rightarrow\!\Diamond\, r}\mu$$

$$\text{[Tra]}\ \frac{\mathcal{S},\mathcal{H} \Vdash (b, l \Rightarrow\!\Diamond\, m) \qquad \mathcal{S},\mathcal{H} \Vdash (b, m \Rightarrow\!\Diamond\, r)}{\mathcal{S},\mathcal{H} \Vdash (b, l \Rightarrow\!\Diamond\, r)}\mu$$

$$\text{[Stp]}\ \frac{\mathcal{S},\mathcal{H} \Vdash (\mathrm{T}, \partial l \Rightarrow\!\Diamond\, r)}{\mathcal{S},\mathcal{H} \Vdash (b, l \Rightarrow\!\Diamond\, r)}\mu \ \textit{if}\ l \sqcap \bullet \sqsubseteq \bot$$

$$\text{[Cut]}\ \frac{\mathcal{S},\mathcal{H} \Vdash (\mathrm{F},\varphi') \qquad \mathcal{S},\mathcal{H} \cup \{(\mathrm{F},\varphi')\} \Vdash (b,\varphi)}{\mathcal{S},\mathcal{H} \Vdash (b,\varphi)}\mu$$

$$\text{[Cof]}\ \frac{\mathcal{S},\mathcal{H} \cup \{(\mathrm{F},\varphi)\} \Vdash (\mathrm{F},\varphi)}{\mathcal{S},\mathcal{H} \Vdash (b,\varphi)}\mu$$

$$\text{[Clr]}\ \frac{\mathcal{S},\mathcal{H} \Vdash (b,\varphi)}{\mathcal{S},\mathcal{H} \cup \{(b',\varphi')\} \Vdash (b,\varphi)}\mu$$

Figure 4: Inductive proof system, with coinduction managed in hypotheses.

which would be unsound since it would prove any formula, valid or not. Third, the [Clr] rule removes a formula from the hypotheses. Note that the [Stp] rule, when applied bottom to top, switches the Boolean from whatever value b it has to T. Hence, it is [Stp] that makes "progress" in our setting, enabling the use of [Hyp] in a sound way. The other rules have the same respective roles as their homonyms in $\Vdash$.

Soundness. We present the soundness proof of $\Vdash$ at a higher level of abstraction than for the other proof systems. For example, we define $\Vdash$-proofs as finite trees, and assume that finite trees are known to the readers. For the other proof systems we adopted a more formal approach because the proofs in those systems were certain kinds of possibly infinite trees, whose a priori knowledge cannot be assumed.

Definition 6 (Proof) *A* proof *of a tagged formula (b,φ) for a transition system $\mathcal{S}$ and under hypotheses $\mathcal{H}$ - for short, a proof of $\mathcal{S},\mathcal{H} \Vdash (b,\varphi)$ - is a finite tree, whose root is labelled by the sequent $\mathcal{S},\mathcal{H} \Vdash (b,\varphi)$, whose nodes are also labelled by sequents, obtained by applying bottom-up the rules depicted in Figure 4.*

We sometimes just write $\mathcal{S},\mathcal{H} \Vdash (b,\varphi)$ for "there is a proof of $\mathcal{S},\mathcal{H} \Vdash (b,\varphi)$" as defined above. The following definition introduces the sets of all hypotheses and of all conclusions occuring in a proof.

Definition 7 (All hypotheses and conclusions occuring in proof) *Assume a proof Θ of $\mathcal{S},\mathcal{H} \Vdash (b,\varphi)$. The set* Hyp *is the union of all sets $\mathcal{H}'$ of formulas, for all the node-labels $\mathcal{S},\mathcal{H}' \Vdash (b',\varphi')$ in the tree Θ. The set* Con *is the set of all formulas (b',φ'), for all the node-labels $\mathcal{S},\mathcal{H}' \Vdash (b',\varphi')$ occuring in Θ.*

Hereafter in the currenct subsection about soundness we assume a proof (tree) Θ of $\mathcal{S},\mathcal{H} \Vdash (b,\varphi)$ with corresponding sets Hyp and Con. The following technical lemma is proved by structural induction on such trees. It says that tagged fomulas in Hyp are among the hypotheses $\mathcal{H}$ present at the root of Θ, plus the conclusions Con, and, except perhaps for those in $\mathcal{H}$, the formulas in Hyp are tagged with F.

Lemma 8 $\mathsf{Hyp} \subseteq \mathcal{H} \cup \mathsf{Con}$*, and, if $(b',\varphi') \in \mathsf{Hyp} \setminus \mathcal{H}$, then $b' = \mathrm{F}$.*

Some more notions need to be defined. First, a *pad* in a tree is a sequence of consecutive edges, and the length of a pad is the number of nodes on the pad. Hence, the length of a pad is strictly positive.

Definition 8 *The* last occurence of a tagged formula $(b',\varphi') \in \mathsf{Con}$ in Θ *is the maximal length of a pad from the root* $\mathcal{S},\mathcal{H} \Vdash (b,\varphi)$ *of* Θ *to some node labelled by* $\mathcal{S},\mathcal{H}' \Vdash (b',\varphi')$*. For formulas* $(b',\varphi') \notin \mathsf{Con}$ *we define by convention their last occurence in* Θ *to be* 0*. This defines a total function last* : $\mathbb{B} \times \Phi \to \mathbb{N}$.

Let also *fPaths* denote the set of finite paths of the transition system under consideration. We now define the set $\mathcal{D} \triangleq \{(\tau', b', \varphi') \in \mathit{fPaths} \times \mathbb{B} \times \Phi \mid (\mathit{lhs}\,\varphi')(\mathit{hd}\,\tau') \wedge (b',\varphi') \in \mathsf{Con}\}$ on which we shall reason by well-founded induction. We equip $\mathcal{D}$ with a well-founded order, namely, with the restriction to $\mathcal{D}$ of the lexicographic-product order on $\mathit{fPaths} \times \mathbb{B} \times \Phi$ defined by $(\tau_1, b_1, \varphi_1) \prec (\tau_2, b_2, \varphi_2)$ iff

1. $\mathit{len}\,\tau_1 < \mathit{len}, \tau_2$, or
2. $\mathit{len}\,\tau_1 = \mathit{len}, \tau_2$ and $b_1 < b_2$, with $<$ on Booleans is defined by $\textsc{f} < \textsc{t}$, or
3. $\mathit{len}\,\tau_1 = \mathit{len}, \tau_2$ and $b_1 = b_2$, and $\mathit{last}(b_1,\varphi_1) > \mathit{last}(b_2,\varphi_2)$.

The first two orders in the product, on natural numbers and on Booleans, are well-founded. For the third one, since the order $\prec$ on $\mathit{fPaths} \times \mathbb{B} \times \Phi$ is restricted to $\mathcal{D}$, all last occurences are bounded by the height of Θ, ensuring that the inequality $\mathit{last}(b_1,\varphi_1) > \mathit{last}(b_2,\varphi_2)$ induces a well-founded order. Hence, the restriction of $\prec$ on $\mathcal{D}$ (also denoted by $\prec$) is a well-founded order as well. The following lemma uses this.

Lemma 9 *Assume* $\mathcal{S},\mathcal{H} \Vdash (\textsc{f}, l \Rightarrow\!\Diamond\, r)$ *and for all* $(b',\varphi') \in \mathcal{H}$, $b' = \textsc{f}$ *and* $\mathcal{S} \models \varphi'$*. Let* $\mathcal{D}$ *be the domain corresponding to* $\mathcal{S},\mathcal{H} \Vdash (\textsc{f}, l \Rightarrow\!\Diamond\, r)$*. Then, for all* $(\tau, b, \varphi) \in \mathcal{D}$*, there is* $k \le \mathit{len}\,\tau$ *such that* $(\mathit{rhs}\,\varphi)(\tau\, k)$*.*

As a corollary to Lemma 9 we obtain:

Theorem 7 (Soundness of $\Vdash$) *If for all* $(b',\varphi') \in \mathcal{H}$, $b' = \textsc{f}$ *and* $\mathcal{S} \models \varphi'$*, then* $\mathcal{S},\mathcal{H} \Vdash (\textsc{f},\varphi)$ *implies* $\mathcal{S} \models \varphi$*.*

Completeness. Proving the completeness of $\Vdash$ is the same as for the other proof system: prove a lemma reducing reachability to an invariance property and then show that for valid formulas that property holds.

Lemma 10 *If* $l \sqsubseteq q \sqcup r$, $q \sqcap \bullet \sqsubseteq \bot$, *and* $\partial q \sqsubseteq q \sqcup r$ *then* $\mathcal{S} \Vdash (\textsc{f}, l \Rightarrow\!\Diamond\, r)$*.*

Proof We build a proof (tree) for $\mathcal{S} \Vdash (\textsc{f}, l \Rightarrow\!\Diamond\, r)$. The root of the tree is a node N_0 labelled $\mathcal{S} \Vdash (\textsc{f}, l \Rightarrow\!\Diamond\, r)$. N_0 has one successor N_1, generated by the [Str] rule, thanks to the hypothesis $l \sqsubseteq q \sqcup r$, and labelled $\mathcal{S} \Vdash (\textsc{f}, (q \sqcup r) \Rightarrow\!\Diamond\, r)$. N_1 has two successors $N_{2,1}$ and $N_{2,2}$, generated by the [Spl] rule, and labelled $\mathcal{S} \Vdash (\textsc{f}, q \Rightarrow\!\Diamond\, r)$ and $\mathcal{S} \Vdash (\textsc{f}, r \Rightarrow\!\Diamond\, r)$, respectively. Usinng the [Trv] rule, $N_{2,2}$ has no succesors. $N_{2,1}$ has one successor N_3, generated by the [Cof] rule, labelled $\mathcal{S}, \{(\textsc{f}, q \Rightarrow\!\Diamond\, r)\} \Vdash (\textsc{f}, q \Rightarrow\!\Diamond\, r)$. N_3 has one successor N_4, generated by the [Stp] rule, thanks to the hypothesis $q \sqcap \bullet \sqsubseteq \bot$, and labelled $\mathcal{S}, \{(\textsc{f}, q \Rightarrow\!\Diamond\, r)\} \Vdash (\textsc{t}, \partial q \Rightarrow\!\Diamond\, r)$. Note that the Boolean has switched from $\textsc{f}$ to $\textsc{t}$, which enables us to later use the [Hyp] rule. The node N_4 has one successor, generated by the [Str] rule thanks to the hypothesis $\partial q \sqsubseteq q \sqcup r$: $\mathcal{S}, \{(\textsc{f}, q \Rightarrow\!\Diamond\, r)\} \Vdash (\textsc{t}, (q \sqcup r) \Rightarrow\!\Diamond\, r)$. N_4 has two successors $N_{5,1}$ and $N_{5,2}$, labelled $\mathcal{S}, \{(\textsc{f}, q \Rightarrow\!\Diamond\, r)\} \Vdash (\textsc{t}, q \Rightarrow\!\Diamond\, r)$ and $\mathcal{S}, \{(\textsc{f}, q \Rightarrow\!\Diamond\, r)\} \Vdash (\textsc{t}, r \Rightarrow\!\Diamond\, r)$, respectively. Neither has any successor: $N_{5,1}$, by the [Hyp] rule, and $N_{5,2}$, by the [Trv] rule. □

By analogy with Theorems 3 and 5 but using Lemma 10 (instead of 3 and 5, respectively) :

Theorem 8 (Completeness of $\Vdash$) $\mathcal{S} \models \varphi$ *implies* $\mathcal{S} \Vdash \varphi$*.*

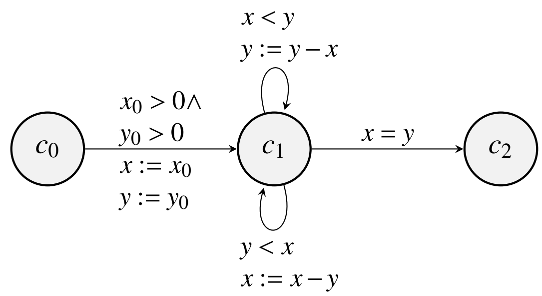

Figure 5: Computing a greatest common divisor

Compositionality w.r.t. Formulas $\Vvdash$ has a symmetrical version of compositionality w.r.t. formulas:

Theorem 9 *$\mathcal{S},\mathcal{H} \cup \{(\text{F},\varphi_1)\} \Vvdash (\text{F},\varphi_2)$ and $\mathcal{S},\mathcal{H} \cup \{(\text{F},\varphi_2)\} \Vvdash (\text{F},\varphi_1)$ imply $\mathcal{S},\mathcal{H} \Vvdash (\text{F},\varphi_1)$ and $\mathcal{S},\mathcal{H} \Vvdash (\text{F},\varphi_2)$.*

Proof The statement is symmetrical in φ_1,φ_2; we prove it for the first formula. The rule[Cof] generates one successor for the root N_0 labelled $\mathcal{S},\mathcal{H} \Vvdash (\text{F},\varphi_1)$: N_1, labelled $\mathcal{S},\mathcal{H} \cup \{(\text{F},\varphi_1)\} \Vvdash (\text{F},\varphi_1)$. From N_1, the rule [Cut] generates two successors, $N_{2,1}$ labelled $\mathcal{S},\mathcal{H} \cup \{(\text{F},\varphi_1)\} \Vvdash (\text{F},\varphi_2)$, which we assumed as a hypothesis, and $N_{2,2}$, labelled $\mathcal{S},\mathcal{H} \cup \{(\text{F},\varphi_1),(\text{F},\varphi_2)\} \Vvdash (\text{F},\varphi_1)$. From $N_{2,2}$ the rule [Clr] removes the first hypothesis and generates a node labelled $\mathcal{S},\mathcal{H} \cup \{(\text{F},\varphi_2)\} \Vvdash (\text{F},\varphi_1)$, which we assumed as a hypothesis as well. □

Example 7 *In Example 6 we tried to prove $\mathcal{S} \models (c = c_0) \Rightarrow\Diamond (c = c_2 \wedge s = m \times (m+1)/2)$ using the asymmetrical compositionality of $\Vdash$, and noted that a certain proof step was impossible because of the asymmetry of the [Tra] rule of $\Vdash$. We show that $\Vvdash$ does not suffer from the same issue. The problem, reformulated in terms of $\Vdash$, was to start the sequent (iii') $\mathcal{S},\{(\text{F},(c = c_0) \Rightarrow\Diamond (c = c_1 \wedge i = 0 \wedge s = 0))\} \Vdash (\text{F},(c = c_0) \Rightarrow\Diamond (c = c_2 \wedge s = m \times (m+1)/2))$ and to use the [Tra] rule in order to split this sequent in two: $\mathcal{S},\{(\text{F},(c = c_0) \Rightarrow\Diamond (c = c_1 \wedge i = 0 \wedge s = 0))\} \Vdash (\text{F},(c = c_0) \Rightarrow\Diamond (c = c_1 \wedge i = 0 \wedge s = 0))$ and then $\mathcal{S},\{(\text{F},(c = c_0) \Rightarrow\Diamond (c = c_1 \wedge i = 0 \wedge s = 0))\} \Vdash (\text{F},(c = c_1 \wedge i = 0 \wedge s = 0) \Rightarrow\Diamond \Rightarrow\Diamond (c = c_1 \wedge i = 0 \wedge s = 0))$. This inference step, which we have just performed above, was not a problem for the $\Vvdash$ proof system.*

Finally, we show how to combine compositionality w.r.t. transition systems and w.r.t. formulas. The following lemma says that $\Vvdash$ is compositional w.r.t. transition systems even in the presence of hypotheses.

Lemma 11 *If $\mathcal{S}',\mathcal{H} \Vvdash (b,\varphi)$ and $\mathcal{S}' \lhd \mathcal{S}$ then $\mathcal{S},\mathcal{H} \Vvdash (b,\varphi)$.*

Combining Theorem 9 and Lemma 11 we obtain as a corollary the following theorem, which combines symmetrical compositionality w.r.t. formulas and compositionality w.r.t. transition systems.

Theorem 10 *If, for $i \in \{0,1\}$, $\mathcal{S}_i \lhd \mathcal{S}$ and $\mathcal{S}_i,\mathcal{H} \cup \{(\text{F},\varphi_{1-i})\} \Vvdash (\text{F},\varphi_i)$, then, for $i \in \{0,1\}$, $\mathcal{S},\mathcal{H} \Vvdash (\text{F},\varphi_i)$.*

Example 8 *We sketch the verification of another infinite-state transition system, denoted by the state machine in Figure 5, which computes the greatest common divisor of two strictly positive natural numbers. The obtained proof is not, by far, the simplest; for such simple systems a global (non-compositional) proof is much shorter. Our goal here is to use all the compositionality features of $\Vvdash$ embodied in Theorem 10.*

The state machine has three control nodes and operates with four natural-number variables: x, y, x_0 and y_0. The last two variables are "symbolic constants", not modified by the transitions of the state machine, whose greatest-common divisor the machine is supposed to compute. On the leftmost transition x and y are initialised to x_0 and y_0, provided that the guard $x_0 > 0 \wedge y_0 > 0$ holds. On the *upper self-loop arrow, x is substracted from y provided the guard $x < y$ holds. The lower self-loop arrow inverses the roles of x and y. The rightmost arrow is taken provided its guard $x = y$ holds. The state-machine denotes an infinite-state transition system $\mathcal{S}$ with state-set $\{c_0, c_1, c_2\}\times\mathbb{N}^4$ and transition relation* $\bigcup_{x,y,x_0,y_0\in\mathbb{N},x_0>0,y_0>0}\{((c_0,x,y,x_0,y_0),(c_1,x_0,y_0,x_0,y_0))\}\cup\bigcup_{x,y,x_0,y_0\in\mathbb{N},x<y}\{((c_1,x,y,x_0,y_0),(c_1,x,y-x,x_0,y_0))\}\cup\bigcup_{x,y,x_0,y_0\in\mathbb{N},y<x}\{((c_1,x,y,x_0,y_0),(c_1,x-y,y,x_0,y_0))\}\cup\bigcup_{x,y,x_0,y_0\in\mathbb{N},x=y}\{((c_1,x,y,x_0,y_0),(c_2,x,y,x_0,y_0))\}$.

We identify two components of this transition system: $\mathcal{S}_1$, encoded by the upper self-loop and rightmost arrow, and $\mathcal{S}_2$, encoded by the lower self-loop and rightmost arrow. Their state-spaces are both $\{c_1,c_2\}\times\mathbb{N}^4$. Their transition relations are $\bigcup_{x,y,x_0,y_0\in\mathbb{N},x<y}\{((c_1,x,y,x_0,y_0),(c_1,x,y-x,x_0,y_0))\}\cup\bigcup_{x,y,x_0,y_0\in\mathbb{N},x=y}\{((c_1,x,y,x_0,y_0),(c_2,x,y,x_0,y_0))\}$ *and* $\bigcup_{x,y,x_0,y_0\in\mathbb{N},y<x}\{((c_1,x,y,x_0,y_0),(c_1,x-y,y,x_0,y_0))\}\cup\bigcup_{x,y,x_0,y_0\in\mathbb{N},x=y}\{((c_1,x,y,x_0,y_0),(c_2,x,y,x_0,y_0))\}$*, induced by their respective arrow subsets. We will show*

(1) $\mathcal{S}\models(c=c_0\wedge x_0>0\wedge y_0>0)\Rightarrow\Diamond(c=c_2\wedge x=y\wedge x=gcd(x_0,y_0))$.

which is the functional correctness of the system. Using the soundness of $\Vdash$ *then the* [Tra] *rule, the latter reduces to (2)* $\mathcal{S}\Vdash(\mathrm{F},(c=c_0\wedge x_0>0\wedge y_0>0)\Rightarrow\Diamond(c=c_1\wedge x=y_0\wedge y=y_0\wedge x_0>0\wedge y_0>0))$ *and (3)* $\mathcal{S}\Vdash(\mathrm{F},(c=c_1\wedge x=y_0\wedge y=y_0\wedge x_0>0\wedge y_0>0)\Rightarrow\Diamond(c=c_2\wedge x=y\wedge x=gcd(x_0,y_0)))$*. Now, (2) is discharged by the sequence of rules* [Stp], [Str] *and* [Trv]*, thus, we focus on (3). Using several times* [Str] *and* [Spl]*, and also* $(x=x_0\wedge y=y_0)\sqsubseteq(gcd(x,y)=gcd(x_0,y_0))$*, (3) reduces to proving the subgoals*

(4) : $\mathcal{S}\Vdash(\mathrm{F},(c_1,gcd(x,y)=gcd(x_0,y_0)\wedge x_0>0\wedge y_0>0\wedge x<y)\Rightarrow\Diamond(c=c_2\wedge x=y\wedge x=gcd(x_0,y_0)))$;

(5) : $\mathcal{S}\Vdash(\mathrm{F},(c_1,gcd(x,y)=gcd(x_0,y_0)\wedge x_0>0\wedge y_0>0\wedge x=y)\Rightarrow\Diamond(c=c_2\wedge x=y\wedge x=gcd(x_0,y_0)))$;

(6) : $\mathcal{S}\Vdash(\mathrm{F},(c_1,gcd(x,y)=gcd(x_0,y_0)\wedge x_0>0\wedge y_0>0\wedge y<x)\Rightarrow\Diamond(c=c_2\wedge x=y\wedge x=gcd(x_0,y_0)))$.

The subgoal (5) is immediately discharged by applying the sequence of rules [Stp], [Str] *and* [Trv].

The two other ones we prove by reducing them, thanks to Theorem 10 to the two following subgoals, with $\varphi_1\triangleq(c=c_1\wedge gcd(x,y)=gcd(x_0,y_0)\wedge x_0>0\wedge y_0>0\wedge x<y)\Rightarrow\Diamond(c=c_2\wedge x=y\wedge x=gcd(x_0,y_0))$ *and* $\varphi_2\triangleq(c=c_1\wedge gcd(x,y)=gcd(x_0,y_0)\wedge x_0>0\wedge y_0>0\wedge y<x)\Rightarrow\Diamond(c=c_2\wedge x=y\wedge x=gcd(x_0,y_0))$*:*

(7) : $\mathcal{S}_1,\{(\mathrm{F},\varphi_2)\}\Vdash(\mathrm{F},\varphi_1)$ *and (8) :* $\mathcal{S}_2,\{(\mathrm{F},\varphi_1)\}\Vdash(\mathrm{F},\varphi_2)$*. We prove (7), the proof of (8) is similar. Using* [Tra]*, (7) reduces to (9) :* $\mathcal{S}_1,\{(\mathrm{F},\varphi_2)\}\Vdash(\mathrm{F},(\phi\wedge x<y)\Rightarrow\Diamond(\phi\wedge y\le x)))$ *and (10) :* $\mathcal{S}_1,\{(\mathrm{F},\varphi_2)\}\Vdash(\mathrm{F},(\phi\wedge y\le x)\Rightarrow\Diamond(c=c_2\wedge x=y\wedge x=gcd(x_0,y_0)))$ *where* $\phi\triangleq(c=c_1\wedge gcd(x,y)=gcd(x_0,y_0)\wedge x_0>0\wedge y_0>0)$.

The subgoal (9) is proved after simplification by [Clr] *using Lemma 10 with* $q\triangleq(\phi\wedge x<y)$.

For the subgoal (10), it is first decomposed using [Str] *then* [Spl] *into (11) :* $\mathcal{S}_1,\{(\mathrm{F},\varphi_2)\}\Vdash(\mathrm{F},(\phi\wedge y=x)\Rightarrow\Diamond(c=c_2\wedge x=y\wedge x=gcd(x_0,y_0)))$ *- which is easily discharged by* [Stp], [Str] *then* [Trv] *- and*

(12) : $\mathcal{S}_1,\{(\mathrm{F},\varphi_2)\}\Vdash(\mathrm{F},(\phi\wedge y<x)\Rightarrow\Diamond(c=c_2\wedge x=y\wedge x=gcd(x_0,y_0)))$*. Using* [Cof]*, (12) becomes*

(13) : $\mathcal{S}_1,\{(\mathrm{F},\varphi_2),(\mathrm{F},(\phi\wedge y<x)\Rightarrow\Diamond\psi)\}\Vdash(\mathrm{F},(\phi\wedge y<x)\Rightarrow\Diamond\psi)$ *with* $\psi\triangleq(c=c_2\wedge x=y\wedge x=gcd(x_0,y_0))$.

We now apply [Stp] *followed by* [Str] *to (13) and get (14) :* $\mathcal{S}_1,\{(\mathrm{F},\varphi_2),(\mathrm{F},(\phi\wedge y<x)\Rightarrow\Diamond\psi)\}\Vdash(\mathrm{T},\phi\Rightarrow\Diamond\psi)$.

After several applications of [Str] *and* [Spl] *(14) is reduced to proving the three last following subgoals:*

(15) : $\mathcal{S}_1,\{(\mathrm{F},\varphi_2),(\mathrm{F},(\phi\wedge y<x)\Rightarrow\Diamond\psi)\}\Vdash(\mathrm{T},(\phi\wedge y<x)\Rightarrow\Diamond\psi)$*, discharged using* [Hyp];

(16) : $\mathcal{S}_1,\{(\mathrm{F},\varphi_2),(\mathrm{F},(\phi\wedge y=x)\Rightarrow\Diamond\psi)\}\Vdash(\mathrm{T},(\phi\wedge y<x)\Rightarrow\Diamond\psi)$*, discharged using* [Stp], [Str], *and* [Trv];

(17) : $\mathcal{S}_1,\{(\mathrm{F},\varphi_2),(\mathrm{F},(\phi\wedge x<y)\Rightarrow\Diamond\psi)\}\Vdash(\mathrm{T},(\phi\wedge y<x)\Rightarrow\Diamond\psi)$*, discharged using* [Hyp] *by noting that* φ_2 *is* $(\phi\wedge y<x)\Rightarrow\Diamond\psi$*. All the subgoals have been discharged, and the proof of (1) is complete.*

6 Implementations in Isabelle/HOL and Coq

We have implemented all the proof systems in Coq and (currently) the first two ones in Isabelle/HOL as well. Our initial goal was to use only Coq, and the reason we also tried Isabelle/HOL (learning it in the process) was that we wanted a “second opinion” when faced with difficulties using Coq’s coinduction.

The Isabelle/HOL implementation for proof systems $\vdash$ and $\Vdash$ is essentially the same as the one described in the paper. The tool automatically generates and proves induction and coinduction principles from inductive and coinductive datatypes or predicates. Proof commands `induction` resp. `coinduction` apply an induction (resp., a coinduction principle) by instantiating the predicate therein via unification with the conclusion, possibly generalised by universally quantifying some variables, (resp., with a conjunction of hypotheses, possibly generalised by existentially quantifying some variables). The overall level of automation is high, which is pleasant to use in practice, the only down side being that users might not understand what is going on. Overall, the proofs in this paper are sketches of the formal Isabelle/HOL proofs, which we did with a lower automation level in order to be able to understand and describe them.

The Coq implementation for the proof systems $\vdash$ and $\Vdash$ is rather different from the above, because support for coinduction in Coq is also rather different. The standard way to perform a proof by coinduction in Coq is to use the `cofix` tactic, which (like the [Cof] rule in our third proof system that emulates it), copies the current goal's conclusion as a new hypothesis, which can only be used after appropriate "progress" has been made in the interactive proof. A proof by coinduction in Coq is ultimately a well-formed corecursive function, where well-formedness is defined as a syntactical guardedness condition, which is quite complex in the theory [4], and even more so in the implementation. We have nonetheless managed to prove the soundness and completeness of $\vdash$ using this tactic: cofix-style proofs of soundness and completeness for $\vdash$, described in standard mathematical notation, are reported in [15]. For $\Vdash$, however, cofix became useless because, for some reason, it does not accept to be mixed in a proof by induction. Fortunately, there is a better version, `pcofix`, part of a Coq package called Paco, based on an extenstion of Knaster-Tarski coinduction called *parameterised* coinduction [6]. Even though the theory is an extension of Knaster-Tarski, anything related to fixpoints of functionals is hidden from the user; a set of tactics, including pcofix, leaves the user with the impression that they are using cofix but without its issues.

The soundness proof of $\Vvdash$, only in Coq for now, generally follows the lines shown in this paper. It is also completely different from the corresponding proofs for the two other proof systems: it does not use general (co)induction principles, but one well-founded induction principle specific to our problem.

7 Conclusions and Future Work

We have presented three proof systems for Reachability Logic on Transition Systems, which use coinduction and induction in different proportions. We have proved their soundness and completeness, and have noted that the more inductive a proof system is, and the more specialised its coinduction style is with respect to our problem domain, the more compositional the proof system is, but the harder its soundness proof. Mechanisations of the proof systems in Isabelle/HOL and Coq have also been briefly presented.

In future work we shall make the formal proof of compositionality with respect to transition systems; and prove the third proof system (currently only proved in Coq) in Isabelle/HOL. We are also planning to port Knaster-Tarski coinduction to Coq, and redo the proofs in this paper in that style, in order to obtain Coq proofs closer in spirit to those in the paper and in Isabelle/HOL. A medium-term project is to use the most compositional proof system, among the three proposed ones, for verifying monadic code, a sizeable amount of which is available to us from earlier projects; and, in the longer term, to enrich our proof system with assume-guarantee-style compositional reasoning related to parallel composition.

Acknowledgment. We would like to thank Andrei Popescu for his help with coinduction in Isabelle/HOL. We acknowledge the support of the CNRS-JSPS Joint Research Project "FoRmal tools for IoT sEcurity" (PRC2199), and thank all the participants of this project for fruitful discussions.

References

[1] Yves Bertot & Pierre Castéran (2004): *Interactive Theorem Proving and Program Development - Coq'Art: The Calculus of Inductive Constructions*. Texts in Theoretical Computer Science. An EATCS Series, Springer.

[2] Jasmin Christian Blanchette, Aymeric Bouzy, Andreas Lochbihler, Andrei Popescu & Dmitriy Traytel (2017): *Friends with Benefits - Implementing Corecursion in Foundational Proof Assistants*. In: *ESOP*, *Lecture Notes in Computer Science* 10201, Springer, pp. 111–140.

[3] Ştefan Ciobâcă & Dorel Lucanu (2018): *A Coinductive Approach to Proving Reachability Properties in Logically Constrained Term Rewriting Systems*. In: *IJCAR*, *Lecture Notes in Computer Science* 10900, Springer, pp. 295–311.

[4] Eduardo Giménez (1994): *Codifying Guarded Definitions with Recursive Schemes*. In: *TYPES*, *Lecture Notes in Computer Science* 996, Springer, pp. 39–59.

[5] C. A. R. Hoare (1969): *An Axiomatic Basis for Computer Programming*. *Commun. ACM* 12(10), pp. 576–580.

[6] Chung-Kil Hur, Georg Neis, Derek Dreyer & Viktor Vafeiadis (2013): *The power of parameterization in coinductive proof*. In: *POPL*, ACM, pp. 193–206.

[7] Dorel Lucanu, Vlad Rusu & Andrei Arusoaie (2017): *A generic framework for symbolic execution: A coinductive approach*. *J. Symb. Comput.* 80, pp. 125–163.

[8] Dorel Lucanu, Vlad Rusu, Andrei Arusoaie & David Nowak (2015): *Verifying Reachability-Logic Properties on Rewriting-Logic Specifications*. In: *Logic, Rewriting, and Concurrency*, *Lecture Notes in Computer Science* 9200, Springer, pp. 451–474.

[9] Brandon M. Moore, Lucas Peña & Grigore Rosu (2018): *Program Verification by Coinduction*. In: *ESOP*, *Lecture Notes in Computer Science* 10801, Springer, pp. 589–618.

[10] Tobias Nipkow, Lawrence C. Paulson & Markus Wenzel (2002): *Isabelle/HOL - A Proof Assistant for Higher-Order Logic*. *Lecture Notes in Computer Science* 2283, Springer.

[11] Peter W. O'Hearn (2019): *Separation logic*. *Commun. ACM* 62(2), pp. 86–95.

[12] Willem P. de Roever, Frank S. de Boer, Ulrich Hannemann, Jozef Hooman, Yassine Lakhnech, Mannes Poel & Job Zwiers (2001): *Concurrency Verification: Introduction to Compositional and Noncompositional Methods*. *Cambridge Tracts in Theoretical Computer Science* 54, Cambridge University Press.

[13] Grigore Rosu, Andrei Stefanescu, Ştefan Ciobâcă & Brandon M. Moore (2013): *One-Path Reachability Logic*. In: *LICS*, IEEE Computer Society, pp. 358–367.

[14] Vlad Rusu, Gilles Grimaud & Michaël Hauspie (2018): *Proving Partial-Correctness and Invariance Properties of Transition-System Models*. In: *TASE*, IEEE Computer Society, pp. 60–67.

[15] Vlad Rusu, Gilles Grimaud & Michaël Hauspie (2019): *Proving Partial-Correctness and Invariance Properties of Transition-System Models*. Available at `https://hal.inria.fr/hal-01962912`.

[16] Davide Sangiorgi (2011): *Introduction to Bisimulation and Coinduction*. Cambridge University Press, New York, NY, USA.

[17] Stephen Skeirik, Andrei Stefanescu & José Meseguer (2017): *A Constructor-Based Reachability Logic for Rewrite Theories*. In: *LOPSTR*, *Lecture Notes in Computer Science* 10855, Springer, pp. 201–217.

[18] Andrei Stefanescu, Ştefan Ciobâcă, Radu Mereuta, Brandon M. Moore, Traian-Florin Serbanuta & Grigore Rosu (2019): *All-Path Reachability Logic*. *Logical Methods in Computer Science* 15(2).

[19] Andrei Stefanescu, Daejun Park, Shijiao Yuwen, Yilong Li & Grigore Rosu (2016): *Semantics-based program verifiers for all languages*. In: *OOPSLA*, ACM, pp. 74–91.

Appendix: Additional Lemmas and Proofs

Lemma 1. *For $R \subseteq Paths \times \Pi$, if for all $(\tau, r) \in R$, it holds that either $(\exists s.\tau = s \wedge r\, s)$, or $(\exists s.\exists \tau'.\tau = s\tau' \wedge r\, s)$ or $(\exists s.\exists \tau'.\tau = s\tau' \wedge (\tau', r) \in R)$, then $R \subseteq \rightsquigarrow$.*

Proof (sketch) Consider the function $F : \mathcal{P}(Paths \times \Pi) \to \mathcal{P}(Paths \times \Pi)$ defined by:

$F(X) = \{(\tau, r) \mid \exists (\tau', r') \in X.(\tau = \tau' = s \wedge r\, s) \vee (\tau = s\tau' \wedge r\, s) \vee (\tau = s\tau' \wedge \tau' \rightsquigarrow r)\}$ Then, F is monotone and by Knaster Tarski's theorem, it has a greatest fixpoint νF, which coincides with the relation $\rightsquigarrow$. The theorem also says that for any $R \subseteq Paths \times \Pi$, if $R \subseteq F(R)$ then $R \subseteq \rightsquigarrow$. Now, let R be the relation in our lemma's hypotheses, and note that the hypothesis "*for all $(\tau, r) \in R$, it holds that either $(\exists s.\tau = s \wedge r\, s)$ or $(\exists s.\exists \tau'.\tau = s\tau' \wedge r\, s)$ or $(\exists s.\exists \tau'.\tau = s\tau' \wedge (\tau', r) \in R)$*" is just the expansion of the inclusion $R \subseteq F(R)$ for F defined as above. Hence the conclusion $R \subseteq \rightsquigarrow$. ◻

The following alternative characterisation of validity will sometimes be useful. We hereafter denote by $fPaths_{\mathcal{S}} \subseteq Paths_{\mathcal{S}}$ the subset of finite paths of a transition system $\mathcal{S}$.

Lemma 12 *$\mathcal{S} \models \varphi$ iff for all $\tau \in fPaths_{\mathcal{S}}$, $(lhs\, \varphi)(hd\, \tau)$ implies $(rhs\, \varphi)(\tau\, n)$ for some $n \leq len\, \tau$.*

Proof (sketch) We first make the following observations. If τ is finite, $\tau \rightsquigarrow_{\mathcal{S}} r$ is equivalent to the existence of $n \leq len\, \tau$ such that $r(\tau\, n)$. This is proved by induction on the finiteness property of the sequence τ. By contrast, if τ is infinite, $\tau \rightsquigarrow_{\mathcal{S}} r$ holds for any r, because, informally, by item (iii) of Definition 1, $\tau \rightsquigarrow r$ can be reduced to $\tau' \rightsquigarrow_{\mathcal{S}} r$ where τ' is the "tail" of τ, and then $\tau' \rightsquigarrow_{\mathcal{S}} r$ can be reduced to $\tau'' \rightsquigarrow_{\mathcal{S}} r$ where τ'' is the "tail" of τ', and so on, *ad infinitum*. The lemma follows from the above observations and from Definition 2. ◻

Lemma 13 (Additional properties of validity) *For all predicates $l, l', l_1, l_2, m, r \in \Pi_{\mathcal{S}}$:*

- *(trivial) : $\mathcal{S} \models r \Rightarrow\!\Diamond\, r$;*
- *(strengthening) : $l \sqsubseteq l'$ and $\mathcal{S} \models l' \Rightarrow\!\Diamond\, r$ imply $\models l \Rightarrow\!\Diamond\, r$;*
- *(splitting) : $\mathcal{S} \models l_1 \Rightarrow\!\Diamond\, r$ and $\mathcal{S} \models l_2 \Rightarrow\!\Diamond\, r$ imply $\mathcal{S} \models (l_1 \sqcup l_2) \Rightarrow\!\Diamond\, r$;*
- *(transitivity) : $\mathcal{S} \models l \Rightarrow\!\Diamond\, m$ and $\mathcal{S} \models m \Rightarrow\!\Diamond\, r$ imply $\mathcal{S} \models l \Rightarrow\!\Diamond\, r$;*
- *(step) : $\mathcal{S} \models \partial l \Rightarrow\!\Diamond\, r$ and $l \sqcap \bullet \sqsubseteq \bot$ imply $\mathcal{S} \models l \Rightarrow\!\Diamond\, r$.*

Proof (sketch) For all items except (transitivity) we use Definition 2 of validity; for (transitivity) it is more convenient tu use the alternative characterisation of validity given by Lemma 12

- (trivial) : Consider any path τ such that $(lhs(r \Rightarrow\!\Diamond\, r))(hd\, \tau)$. Then $(rhs(r \Rightarrow\!\Diamond\, r))(hd\, \tau)$, thus, by Definition 1, $\tau \rightsquigarrow r$. The conclusion $\mathcal{S} \models r \Rightarrow\!\Diamond\, r$ follows by Definition 2.
- (strengthening): Consider any path τ such that $(lhs(l \Rightarrow\!\Diamond\, r))(hd\, \tau)$, i.e., $l\,(hd\, \tau)$. From $l \sqsubseteq l'$ we obtain that $l'\,(hd\, \tau)$, i.e., $(lhs(l' \Rightarrow\!\Diamond\, r))(hd\, \tau)$. From $\mathcal{S} \models l' \Rightarrow\!\Diamond\, r$ we obtain by Definition 2 that $\tau \rightsquigarrow r$. The conclusion $\mathcal{S} \models l \Rightarrow\!\Diamond\, r$ follows by Definition 2.
- (splitting) : Consider any path τ such that $(lhs((l_1 \sqcup l_2) \Rightarrow\!\Diamond\, r))(hd\, \tau)$, i.e., $(l_1 \sqcup l_2)(hd\, \tau)$. Hence, $l_1\,(hd\, \tau)$ or $l_1\,(hd\, \tau)$. We consider the first case, the other one is symetrical. From $l_1\,(hd\, \tau)$ and $\mathcal{S} \models l_1 \Rightarrow\!\Diamond\, r$ we obtain $r \rightsquigarrow r$, and conclusion $\mathcal{S} \models ((l_1 \sqcup l_2) \Rightarrow\!\Diamond\, r)$ follows by Definition 2.
- (transitivity) : Consider any *finite* path τ such that $(lhs(l \Rightarrow\!\Diamond\, r))(hd\, \tau)$, i.e., $l\,(hd\, \tau)$. Hence, $(lhs(l \Rightarrow\!\Diamond\, m))(hd\, \tau)$, and from $\mathcal{S} \models l \Rightarrow\!\Diamond\, m$, using Lemma 12 we obtain $k \leq len\, \tau$ such that $m\,(\tau\, k)$. Let τ' be the suffix of τ starting at $\tau\, k$. Then, τ' is a finite path, and $m\,(\tau\, k)$ means $m\,(hd\, \tau')$, i.e., $(lhs(m \Rightarrow\!\Diamond\, r))(hd\, \tau')$, which, together with $\mathcal{S} \models m \Rightarrow\!\Diamond\, r$ and Lemma 12 gives us $k' \leq len\, \tau'$ such that $r\,(\tau'\, k')$. Let $k'' = k + k'$, hence, $k'' \leq len\, \tau$, and the conclusion $\models l \Rightarrow\!\Diamond\, r$ follows by Lemma 12.

- (step) Consider any path τ such that $(lhs(l \Rightarrow\!\Diamond\, r))(hd\,\tau)$, i.e., $l\,(hd\,\tau)$. Assume first that $\tau = s$ for some $s \in S$. Since τ is a path, s is final, i.e., $\bullet\, s$, which, together with $l\,s$ gives $(\text{ł} \sqcap \bullet)\, s$, in contradiction with the hypothesis $l \sqcap \bullet \sqsubseteq \bot$. Hence, $\tau = s\,\tau'$ for some $s \in S$ and $\tau' \in Paths$, with $s \to (hd\,\tau')$. We now show $(\partial \text{ł})(hd\,\tau')$. Indeed, by the definition of the ∂ function, the latter statement amounts to the existence of some state $s' \in S$ such that $l\,s'$ and $s' \to (hd\,\tau')$; taking $s' = s$ satisfies this. Hence, $lhs(\partial l \Rightarrow\!\Diamond\, r)(hd\,\tau')$ and from $\mathcal{S} \models \partial l \Rightarrow\!\Diamond\, r$ we obtain by Definition 2 that $\tau' \rightsquigarrow r$, which, by Definition 1, implies $\tau \rightsquigarrow r$. The conclusion $\mathcal{S} \models l \Rightarrow\!\Diamond\, r$ follows by Definition 2.

$\square$

Theorem 1 (Compositionality of $\models$ w.r.t transition systems). *$\mathcal{S}' \lhd \mathcal{S}$ and $\mathcal{S}' \models \varphi$ imply $\mathcal{S} \models \varphi$.*

Proof (sketch) Let $\mathcal{S} = (S, \to)$, $\mathcal{S}' = (S', \to')$, $\varphi = l \Rightarrow\!\Diamond\, r$. Note that the hypothesis $\mathcal{S}' \models l \Rightarrow\!\Diamond\, r$ implies $l, r \in \Pi_{\mathcal{S}'}$, which by $\mathcal{S}' \lhd \mathcal{S}$ also implies $l, r \in \Pi_{\mathcal{S}}$. Let $\tau \in fPaths_{\mathcal{S}}$ be arbitrarily chosen such that $l\,(hd\,\tau)$. Since $l \in \Pi_{\mathcal{S}'}$, he have $(hd\,\tau) \in S'$. Hence, the set T'_τ of prefixes of τ is nonempty.

Since all sequences in T'_τ are finite, there is one τ'_m with maximal length $k = len\,\tau'_m$. Let $s_m = \tau'_m\,k$. Since $\tau'_m \in T'_\tau$, in order to show $\tau'_m \in Paths_{\mathcal{S}'}$ we only need (†): $s_m \in \bullet_{\mathcal{S}'}$. The are two cases:

- if $k = len\,\tau$ then s_m is the last state on $\tau \in fPaths_{\mathcal{S}}$, hence, $s_m \in \bullet_{\mathcal{S}} \subseteq \bullet_{\mathcal{S}'}$;
- if $k < len\,\tau$ then $\tau\,(k+1) \in S \setminus S'$, otherwise, from $s_m \in S'$ and $s_m \to \tau\,(k+1)$ and hypotheses one has $s_m \to' \tau\,(k+1)$, hence, $\tau'_m \to' \tau\,(k+1) \in T'_\tau$, in contradiction with the maximal length of of τ'_m in T'_τ. From $\tau\,(k+1) \in S \setminus S'$ and $s_m \to \tau\,(k+1)$ we obtain from the lemma's hypotheses that $s' \in \bullet_{\mathcal{S}'}$.

Hence, (†) is proved, and per the above reasoning, so is $\tau'_m \in Paths_{\mathcal{S}'}$. Observe also that, since $\tau'_m \in T'_\tau$, we have that τ'_m is a prefix of τ, thus, for all $j \le k = len\,\tau'_m$, $\tau'_m\ j = \tau\ j$. In particular, $hd\,\tau'_m = hd\,\tau'$ and since we assumed $l\,(hd\,\tau)$ at the beginning, we also have $l\,(hd\,\tau'_m)$. From the latter and $\tau'_m \in Paths_{\mathcal{S}'}$ and $\mathcal{S}' \models l \Rightarrow\!\Diamond\, r$ and Lemma 12 we obtain $j \le len\,\tau'_m = k \le len\,\tau$ such that $r(\tau'_m\ j)$, hence, $r(\tau\ j)$.

Recapitulating, we started with $\tau \in fPaths_{\mathcal{S}}$ arbitrarily chosen such that $l\,(hd\,\tau)$, and obtained $j \le len\,\tau$ such that $r(\tau\ j)$. By Lemma 12, this means $\mathcal{S}' \models l \Rightarrow\!\Diamond\, r$, which proves the theorem. $\square$

Lemma 2. *For all set $X \subseteq \Phi$ of hypotheses and $\varphi \in X$, if* for all $l \Rightarrow\!\Diamond\, r \in X$, there is $l' \in \Pi$ such that $l \sqsubseteq l' \sqcup r$, $l' \sqcap \bullet \sqsubseteq \bot$ and $\partial l' \Rightarrow\!\Diamond\, r \in X$ *, then $\mathcal{S} \vdash \varphi$.*

Proof (sketch) Choose an arbitrary $X \subseteq \varphi$. The hypothesis '*for all $l \Rightarrow\!\Diamond\, r \in X$, there is $l' \in \Pi$ such that $l \sqsubseteq l' \sqcup r$, $l' \sqcap \bullet \sqsubseteq \bot$ and $\partial l' \Rightarrow\!\Diamond\, r \in X$* " is the expansion of $X \subseteq F(X)$ with F defined as above. By Tarski's principle, $X \subseteq \nu F$. We obtain that for all $\varphi \in X$, it holds that $\mathcal{S} \vdash \varphi$, which proves the lemma. $\square$

Ttheorem 2 (Soundness of $\vdash$). *$\mathcal{S} \vdash \varphi$ implies $\mathcal{S} \models \varphi$.*

Proof (sketch) We first prove the following fact (†): the relation $R \subseteq Paths \times \Pi$ defined as follows: $R \triangleq \lambda(\tau, r).\exists l.(\mathcal{S} \vdash l \Rightarrow\!\Diamond\, r \wedge l\,(hd\tau))$ satisfies $R \subseteq \rightsquigarrow$. We use the coinduction principle for $\rightsquigarrow$, i.e., Lemma 1.

That lemma, in turn, requires us to prove that, assumming $(\tau, r) \in R$, it holds that (i) $(\exists s.\tau = s \wedge r\,s)$ or (ii) $(\exists s.\exists \tau'.\tau = s\,\tau' \wedge r\,s)$ or (iii) $(\exists s.\exists \tau'.\tau = s\,\tau' \wedge (\tau', r) \in R)$. We proceed by case analysis:

- first, assume $\tau = s$ for some $s \in S$. Since $(\tau, r) \in R$, there is $l \in \Pi$ such that $\vdash l \Rightarrow\!\Diamond\, r$ and $l\,s$. Now, $\vdash l \Rightarrow\!\Diamond\, r$ implies that there is $l' \in \Pi$ with $l \sqsubseteq l' \sqcup r$, $l' \sqcap \bullet \sqsubseteq \bot$, and $\vdash \partial l' \Rightarrow\!\Diamond\, r$. Next, $l\,s$ and $l \sqsubseteq l' \sqcup r$ imply $l'\,s$ or $r\,s$. Assume first $l'\,s$. Since τ is the (singleton) path s, the state s is final, hence, $\bullet\, s$, which together with $l'\,s$ contradict $l' \sqcap \bullet \sqsubseteq \bot$. Hence, $l'\,s$ is impossible, and therefore $r\,s$; then, (i) is proved in this case. Note that here we did not use the fact $\mathcal{S} \vdash \partial l' \Rightarrow\!\Diamond\, r$ - it will be used below.
- then, assume $\tau = s\,\tau'$ for some $s \in S$ and $\tau' \in Paths$. We have two subcases:
  - if $r\,s$ then (ii) holds;

- otherwise $\neg r s$. We show $(\tau', r) \in R$. From $(\tau, r) \in R$ we know that there is $l \in \Pi$ such that $\mathcal{S} \vdash l \Rightarrow\Diamond r$ and $l\, s$. Now, $\mathcal{S} \vdash l \Rightarrow\Diamond r$ implies that there is $l' \in \Pi$ with $l \sqsubseteq l' \sqcup r$, $l' \sqcap \bullet \sqsubseteq \bot$, and $\mathcal{S} \vdash \partial l' \Rightarrow\Diamond r$. We prove $(\partial l')(hd\,\tau')$: from $l\, s$ and $\neg r\, s$ and $l \sqsubseteq l' \sqcup r$ we obtain $l'\, s$. Then, since τ is a path, then so is τ', and we have the transition $s \rightarrow (hd\,\tau')$. Since $\partial l' = \lambda s'. \exists s. l'\, s \wedge s \rightarrow s'$ we obtain that $(\partial l')(hd\,\tau')$. The existence of $\partial l'$ such that $\mathcal{S} \vdash \partial l' \Rightarrow\Diamond r$ and $(\partial l')(hd\,\tau')$ ensures $(\tau', r) \in R$, hence, (iii) holds. Note also that we did not use $l' \sqcap \bullet \sqsubseteq \bot$ here, but this inclusion was used in an earlier case.

Summarising, in all possible cases, $(\tau, r) \in R$ implies either statements (i), (ii), or (iii) from the coinduction principle - Lemma 1. Hence, the lemma ensures $R \subseteq \rightsquigarrow$, and (†) is proved. Coming back to our theorem: consider an arbitrary $\varphi \triangleq l' \Rightarrow\Diamond r \in \Phi$ such that $\mathcal{S} \vdash l' \Rightarrow\Diamond r$. In order to show $\mathcal{S} \models l' \Rightarrow\Diamond r$, i.e., the conclusion of the theorem, we only need to show that for all paths τ such that $l'\,(hd\,\tau)$, it holds that $\tau \rightsquigarrow r$. We do this by showing $(\tau, r) \in R$ and using $R \subseteq \rightsquigarrow$ from above. We have defined $R = \lambda(\tau, r). \exists l. (\mathcal{S} \vdash l \Rightarrow\Diamond r \wedge l(hd\tau))$ and for our (τ, r) there does indeed exist $l \triangleq l'$ such that $\mathcal{S} \vdash l \Rightarrow\Diamond r$ (hypothesis of the theorem) and $l\,(hd\,\tau)$ (from above). Hence $(\tau, r) \in R$, which concludes the proof. □

Lemma 3. *If $l \sqsubseteq q \sqcup r$, $q \sqcap \bullet \sqsubseteq \bot$, and $\partial q \sqsubseteq q \sqcup r$ then $\mathcal{S} \vdash l \Rightarrow\Diamond r$.*

Proof (sketch) We use the coinduction principle for $\vdash$ (Lemma 2). Consider the set $X = \{l' \Rightarrow\Diamond r \mid l' \sqsubseteq q \sqcup r \,\wedge\, q \sqcap \bullet \sqsubseteq \bot \,\wedge\, \partial q \sqsubseteq q \sqcup r\}$. We show the premise of the lemma with the coinductive hypothesis X: (†) *for all $l' \Rightarrow\Diamond r \in X$, there exists $l'' \in \Pi$ such that $l' \sqsubseteq l'' \sqcup r'$ and $l'' \sqcap \bullet \sqsubseteq \bot$ and $\partial l'' \Rightarrow\Diamond r \in X$.*

Let then $l' \Rightarrow\Diamond r$ be an arbitrary element in X. Hence, (i) $l' \sqsubseteq q \sqcup r'$ and (ii) $q \sqcap \bullet \sqsubseteq \bot$ and (iii) $\partial q \sqsubseteq q \sqcup r$. Moreover, $\partial q \Rightarrow\Diamond r \in X$, because of the hypothesis $\partial q \sqsubseteq q \sqcup r$ of out lemma and (ii) and (iii). By choosing in (†) to instantiate the existentially quantified l'' to q, for any formula in X, the (†) statement is proved. Using the coinduction principle for $\vdash$ (Lemma 2), for all $\varphi \in X$, it holds that $\mathcal{S} \vdash \varphi$. Finally, the formula $l \Rightarrow\Diamond r$ in our lemma's conclusion does belong to X since, by the lemma's hypotheses it satisfies all the conditions of membership in X. Hence, $\mathcal{S} \vdash l \Rightarrow\Diamond r$, which concludes the proof. □

Theorem 3 (Completeness of $\vdash$). *$\mathcal{S} \models \varphi$ implies $\mathcal{S} \vdash \varphi$.*

Proof Let $\varphi \triangleq l \Rightarrow\Diamond r$. We find a state predicate q that, in the case φ is valid, satisfies the three inclusions in the hypothesis of Lemma 3 and implies $\vdash \varphi$. We define $q \triangleq \lambda s. \neg r s \wedge \forall \tau \in Paths. (\, s = hd\,\tau \Rightarrow \tau \rightsquigarrow r)$.

1. $l \sqsubseteq q \sqcup r$: let s be any state such that $l\, s$; we have to prove $(q \sqcup r)\, s$. If $r\, s$ the proof is done. Thus, assume $\neg r\, s$, and consider any path τ such that $s = hd\,\tau$. From $l\, s$ and $s = hd\,\tau$ and $\models l \Rightarrow\Diamond r$ we obtain by Definition 2, that $\tau \rightsquigarrow r$, and by definition of q we have $q\, s$: the first inclusion is proved.
2. $q \sqcap \bullet \sqsubseteq \bot$: let s be any state such that $q\, s$; we prove that $\bullet\, s$ is impossible. By the above definition of q, $\neg r s$. Consider an arbitrary path τ such that $s = hd\,\tau$; again, by definition of q, $\tau \rightsquigarrow r$. Now, the only way $\tau \rightsquigarrow r$ can hold when $\neg r s$ holds is (cf. Definition 1) when $\tau = s\,\tau'$ for some path τ'. Hence, s is not final, thus no state satisfies $q \sqcap \bullet$, and our second inclusion is also proved.
3. $\partial q \sqsubseteq q \sqcup r$: let s' be a state such that $(\partial q)\, s'$; we have to prove $(q \sqcup r)\, s'$. By the definition of the symbolic transition function ∂, there exists s such that $s \rightarrow s'$ and $q\, s$. By the definition of q, $\neg r\, s$ and for each $\tau \in Paths$ such that $s = hd\,\tau$, it holds that $\tau \rightsquigarrow r$. There are two subcases:
   - if $r\, s'$ then $(q \sqcup r)\, s'$, and our third inclusion is proved;
   - if $\neg r\, s'$: consider any path τ' such that $s' = hd\,\tau'$. Then, the sequence $\tau \triangleq s\, s'\, \tau$ is a path and is such that $s = hd\,\tau$, and, per the above, $\tau \rightsquigarrow r$. We also have $\neg r\, s$, and then the only way $\tau \rightsquigarrow r$ may hold is via $\tau' \rightsquigarrow r$ (cf. Definition1). Summarising, in the case $\neg r\, s'$, we get that any path τ' such that $s' = hd\,\tau'$ satisfies $\tau' \rightsquigarrow r$. Hence, $q\, s'$ by the definition of q, and therefore also $(q \sqcup r)\, s'$, which completes the proof of the third inclusion and of the theorem.

□

Lemma 4. *If $X \subseteq \mu\vdash_{\mathcal{S},\mathcal{H},X}$ then for all $\varphi \in X$ it holds that $\mathcal{S},\mathcal{H} \Vdash \phi$.*

Proof (sketch) The hypothesis $X \subseteq \mu\vdash_{\mathcal{S},\mathcal{H},X}$ is $X \subseteq F_{\mathcal{S},\mathcal{H}}(X)$, the conclusion is $X \subseteq \nu F_{\mathcal{S},\mathcal{H}}$, and the lemma follows from Tarski's coinduction principle. □

Theorem 4 (Soundness of $\Vdash$). *If for all $\varphi' \in \mathcal{H}$, $\mathcal{S} \models \varphi'$, then $\mathcal{S},\mathcal{H} \Vdash \varphi$ implies $\mathcal{S} \models \varphi$.*

Proof(sketch) Let $Q \triangleq \lambda(\tau, r).\exists l.(\mathcal{S},\mathcal{H} \Vdash l \Rightarrow\Diamond r \wedge l(hd\,\tau) \wedge \tau \in fPaths)$, where *fPaths* $\subseteq$ *Paths* is the subset of finite paths. We first show that one can apply the coinduction principle for $\hookrightarrow$ (Lemma 6) with the parameter R therein set to Q. This amounts to showing that for all $l \Rightarrow\Diamond r \in \Phi$ and $\tau \in fPaths$, if $\mathcal{S},\mathcal{H} \Vdash l \Rightarrow\Diamond r$ and $l(hd\,\tau)$ then either (i) or (ii) or (iii) holds, where (i) is $(\exists s.\tau = s \wedge r\,s)$, (ii) is $(\exists s.\exists\tau'.\tau = s\,\tau' \wedge r\,s)$, and (iii) is $(\exists s.\exists\tau'.\exists n.\exists\tau''.\tau = s\,\tau' \wedge \tau'' = (suf\,\tau'\,n) \wedge ((\tau'', r) \in Q \vee \tau'' \hookrightarrow r))$.

Now, remember that $\mathcal{S},\mathcal{H} \Vdash \varphi$ is just $\varphi \in \nu\mu\vdash_{\mathcal{S},\mathcal{H}}$. Since the latter is a fixpoint of the functional $Y \mapsto \mu\vdash_{\mathcal{S},\mathcal{H},Y}$ we have that $\nu\mu\vdash_{\mathcal{S},\mathcal{H}} = \mu\vdash_{\mathcal{S},\mathcal{H},\nu\mu\vdash_{\mathcal{S},\mathcal{H}}}$. Using Knaster-Tarski's theorem for *smallest* fixpoints: for any $P \subseteq \Phi$, if

1. $\mathcal{H} \subseteq P$;
2. for all $r \in \Pi$, $r \Rightarrow\Diamond r \in P$;
3. for all $l, l', r \in \Pi$, $l \sqsubseteq l'$ and $l' \Rightarrow\Diamond r \in P$ implies $l \Rightarrow\Diamond r \in P$;
4. for all $l_1, l_2, r \in \Pi$, $l_1 \Rightarrow\Diamond r \in P$ and $l_2 \Rightarrow\Diamond r \in P$ imply $(l_1 \sqcup l_2) \Rightarrow\Diamond r \in P$;
5. for all $l, m, r \in \Pi$, $\models l \Rightarrow\Diamond m$ and $m \Rightarrow\Diamond r \in P$ imply $l \Rightarrow\Diamond r \in P$;
6. for all $l, r \in \Pi$, $l \sqcap \bullet \sqsubseteq \bot$ and $\partial l \Rightarrow\Diamond r \in \nu\mu\vdash_{\mathcal{S},\mathcal{H}}$ implies $l \Rightarrow\Diamond r \in P$;

then $\mu\vdash_{\mathcal{S},\mathcal{H},\nu\mu\vdash_{\mathcal{S},\mathcal{H}}} \subseteq P$. We choose $P = \{l \Rightarrow\Diamond r \in \Phi \mid \forall\tau \in fPaths, l(hd\tau) \Rightarrow O(\tau, r)\}$, where $O(\tau, r)$ is either

- (i) $(\exists s.\tau = s \wedge r\,s)$, or
- (ii) $(\exists s.\exists\tau'.\tau = s\,\tau' \wedge r\,s)$, or
- (iii) $(\exists s.\exists\tau'.\exists n.\exists\tau''.\tau = s\,\tau' \wedge \tau'' = (suf\,\tau'\,n) \wedge ((\tau'', r) \in Q \vee \tau'' \hookrightarrow r))$.

Now, <u>if</u> the chosen P satisfies the constraints (1)-(6) above, then we obtain $\mu\vdash_{\mathcal{S},\mathcal{H},\nu\mu\vdash_{\mathcal{S},\mathcal{H}}} \subseteq P$, and hence $\nu\mu\vdash_{\mathcal{S},\mathcal{H}} \subseteq P$. This means that the coinduction principle for $\hookrightarrow$ (Lemma 6) with the parameter R therein set to Q can be applied, hence, $Q \subseteq \hookrightarrow$. As a consequence, any $\varphi \in \Phi$ that satisfies the hypotheses of the theorem, in particular, such that $\mathcal{S},\mathcal{H} \Vdash \varphi$, and any $\tau \in fPaths$ such that $(lhs\,\varphi)(hd\,\tau)$, have the property that $(\tau, (rhs\,\varphi)) \in Q$, hence, $\tau \hookrightarrow (rhs\,\varphi)$, thus, by Lemmas 12 and 7, $\mathcal{S} \models \varphi$, i.e., the theorem's conclusion. Since φ, which was chosen arbitrarily to satisfy the hypotheses of the theorem, also satisfies its conclusion, the theorem is proved. There remains to prove that P thus defined satisfies the constraints (1)-(6) above.

- Constraints 1 and 2 refer to valid formulas. For such formulas, say, $l \Rightarrow\Diamond r$, it holds by definition of validity and Lemmas 12 and 7 that for all $\tau \in fPaths$ such that $l(hd\,\tau)$, either $(\exists s.\tau = s \wedge r\,s)$ or $(\exists s.\exists\tau'.\tau = s\,\tau' \wedge r\,s)$ or $(\exists s.\exists\tau'.\exists n.\exists\tau''.\tau = s\,\tau' \wedge \tau'' = (suf\,\tau'\,n) \wedge \tau'' \hookrightarrow r)$. This implies $l \Rightarrow\Diamond r \in P$;
- For constraint 3, assume $l \sqsubseteq l'$ and $l' \Rightarrow\Diamond r \in P$, and consider any $\tau \in fPaths$ such that $l(hd\,\tau)$. Then, we also have $l'(hd\,\tau)$, and from $l' \Rightarrow\Diamond r \in P$ we obtain (i) or (ii) or (iii), which implies $l \Rightarrow\Diamond r \in P$;
- For constraint 4, consider any $\tau \in fPaths$ such that $(l_1 \sqcup l_2)(hd\,\tau)$, i.e., $l_1(hd\,\tau)$ or $l_2(hd\,\tau)$. If $l_1(hd\,\tau)$, then $l_1 \Rightarrow\Diamond r \in P$ implies (i) or (ii) or (iii), hence, $(l_1 \sqcup l_2) \Rightarrow\Diamond r \in P$. The case $l_2(hd\,\tau)$ is similar;

- For constraint 5: consider any $\tau \in fPaths$ such that $l(hd\,\tau)$. From $\models l \Rightarrow\!\Diamond m$ we obtain thanks to Lemma 13 (*transitivity* item) some $k \leq (len\,\tau)$ such that $m(\tau\,k)$. Let $\tau' = (suf\,\tau\,k)$, then $m(\tau\,k)$ means $m(hd\,\tau')$, and from $m \Rightarrow\!\Diamond r \in P$ we obtain that either (a) $(\exists s.\tau' = s \wedge r\,s)$ or (b) $(\exists s.\exists\tau''.\tau' = s\,\tau'' \wedge r\,s)$ or (c) $(\exists s.\exists\tau''.\exists n.\exists\tau'''.\tau' = s\,\tau'' \wedge \tau''' = (suf\,\tau''\,n) \wedge ((\tau''', r) \in Q \vee \tau''' \hookrightarrow r))$. Cases (a) and (b) imply either conditions (i) or (ii) for $l \Rightarrow\!\Diamond r \in P$, hence, we focus on case (c), in which there exist s, τ'', n, τ''' such that $\tau' = s\,\tau'' \wedge \tau''' = (suf\,\tau''\,n) \wedge ((\tau''', r) \in Q \vee \tau''' \hookrightarrow r))$. We obtain that there do exist $s_0 = (hd\,\tau)$, $\tau''_0 = suf\,\tau\,1$, $n_0 = n + k$, $\tau'''_0 = suf\,\tau''\,n_0$ such that $\tau = s_0\,\tau''_0 \wedge \tau'''_0 = (suf\,\tau''\,n_0) \wedge ((\tau'''_0, r) \in Q \vee \tau'''_0 \hookrightarrow r))$. [Specifically, $(\tau''', r) \in Q$ implies $(\tau'''_0, r) \in Q$, and $\tau''' \hookrightarrow r$ implies $\tau'''_0 \hookrightarrow r$ due to the definitions of Q and $\hookrightarrow$.] The existence of $s_0, \tau''_0, n_0, \tau'''_0$ with the above properties is just condition (iii) for $l \Rightarrow\!\Diamond r \in P$. Note that the asymmetry in the [Tra] rule of our proof system gave us the hypothesis $\models l \Rightarrow\!\Diamond m$, which is essential in this case: without it, $l \Rightarrow\!\Diamond r \in P$ cannot be proved.
- For constraint 6: consider any $\tau \in fPaths$ such that $l(hd\,\tau)$. Assume $\tau = s$ for some $s \in S$. Thus, $l\,s$ and s is final, contradicting the hypothesis $l \sqcap \bullet \sqsubseteq \bot$. Hence, $\tau = s\,\tau'$ for some $s \in S$ and $\tau' \in fPaths$. From $\partial l \Rightarrow\!\Diamond r \in \nu\mu\vdash_{\mathcal{S},\mathcal{H}}$ we obtain $\mathcal{S},\mathcal{H} \Vdash \partial l \Rightarrow\!\Diamond r$. Moreover, from the definition of the symbolic transition function ∂ and $s \to (hd\,\tau')$ and $l\,s$ we obtain $(\partial l)(hd\,\tau')$. From the definition of Q, with the existentially quantified variable therein set to ∂l, we obtain $(\tau', r) \in Q$. Hence, there do exist $s_0 = s$, $\tau''_0 = \tau'$, $n_0 = 0$, $\tau'''_0 = suf\,\tau''\,n_0$ such that $\tau = s_0\,\tau''_0 \wedge \tau'''_0 = (suf\,\tau''\,n_0) \wedge (\tau'''_0, r) \in Q$, implying condition (iii) for $l \Rightarrow\!\Diamond r \in P$, which concludes the proof of this last case and of the theorem.

$\square$

Lemma 9. *Assume $\mathcal{S},\mathcal{H} \Vdash (\text{F}, l \Rightarrow\!\Diamond r)$ and for all $(b', \varphi') \in \mathcal{H}$, $b' = \text{F}$ and $\mathcal{S} \models \varphi'$. Let $\mathcal{D}$ be the domain corresponding to $\mathcal{S},\mathcal{H} \Vdash (\text{F}, l \Rightarrow\!\Diamond r)$. Then, for all $(\tau, b, \varphi) \in \mathcal{D}$, there is $k \leq len\,\tau$ such that $(rhs\,\varphi)(\tau\,k)$.*

Proof (sketch) Let Θ be a proof of $\mathcal{S},\mathcal{H} \Vdash (\text{F}, l \Rightarrow\!\Diamond r)$ and consider any $(\tau, b, \varphi) \in \mathcal{D}$; let $\varphi = l_\varphi \Rightarrow\!\Diamond r_\varphi$, hence, $l_\varphi \Rightarrow\!\Diamond r_\varphi \in \mathsf{Con}$. Thus, the last occurence $last(b, l_\varphi \Rightarrow\!\Diamond r_\varphi)$ of $(b, l_\varphi \Rightarrow\!\Diamond r_\varphi)$ in Θ is a strictly positive natural number. In particular, there is $\mathcal{H}'$ and a node N labelled $\mathcal{S},\mathcal{H}' \Vdash (b, l_\varphi \Rightarrow\!\Diamond r_\varphi)$ that is on a pad of length $last(b, l_\varphi \Rightarrow\!\Diamond r_\varphi)$ from the root of Θ.

We shall be using the following observation several times hereafter: (†) for any direct successor labelled $\mathcal{S},\mathcal{H}'' \Vdash (b'', l'' \Rightarrow\!\Diamond r'')$ of the above node N, $last(b'', l'' \Rightarrow\!\Diamond r'') > last(b, l_\varphi \Rightarrow\!\Diamond r_\varphi)$. Indeed, there are instances of $(b'', l'' \Rightarrow\!\Diamond r'')$ occuring further from the root of Θ than the furthest instance of $(b, l_\varphi \Rightarrow\!\Diamond r_\varphi)$, which is in the node N labelled $\mathcal{S},\mathcal{H}' \Vdash (b, l_\varphi \Rightarrow\!\Diamond r_\varphi)$, at distance $last(b, l_\varphi \Rightarrow\!\Diamond r_\varphi)$ from the root of Θ. In particular, no direct successor of the node N has the conclusion $(b, l_\varphi \Rightarrow\!\Diamond r_\varphi)$.

- if the node N is a leaf, then, the leaf results from applying either [Hyp] or [Trv].
  - if the leaf results from applying [Hyp], then $b = \text{T}$ and $(\text{F}, \varphi) \in \mathcal{H}' \subseteq \mathsf{Hyp}$. Using Lemma 8:
    * either $(\text{F}, \varphi) \in \mathcal{H}$, where $\mathcal{H}$ is the set of initial hypotheses. Hence, $\mathcal{S} \models \varphi$, and using Lemma 12 we obtain $k \leq len\,\tau$ such that $(rhs\,\varphi)(\tau\,k)$, which proves the lemma in this case.
    * or $(\text{F}, \varphi) \notin \mathcal{H}$, which, again by Lemma8, implies $(\text{F}, \varphi) \in \mathsf{Con}$. It follows that $(\tau, \text{F}, \varphi) \in \mathcal{D}$, and, since $b = \text{T}$, $(\tau, \text{F}, \varphi) \prec (\tau, b, \varphi)$. Using the well-founded induction hypothesis, we obtain $k \leq len\,\tau$ such that $(rhs\,\varphi)(\tau\,k)$, which proves the lemma in this case.
  - if the leaf results from applying [Trv]), then $l_\varphi = r_\varphi$, and from $(\tau, b, l_\varphi \Rightarrow\!\Diamond r_\varphi) \in \mathcal{D}$ we have $(l_\varphi)(hd\,\tau)$, hence, $(r_\varphi)(hd\,\tau)$, thus with $k = 0$ the lemma is proved in this case.
- if the node N is not a leaf, then it has one or two successors in Θ generated by applying some rule of our proof system except [Hyp] and [Trv]. Depending on the rule:
  - if the rule is [Str], then N has one succesor labelled $\mathcal{S},\mathcal{H}' \Vdash (b, l' \Rightarrow\!\Diamond r_\varphi)$ with $l_\varphi \sqsubseteq l'$. It follows that $(b, l' \Rightarrow\!\Diamond r_\varphi) \in \mathsf{Con}$ and from $(l_\varphi)(hd\,\tau)$ and $l_\varphi \sqsubseteq l'$ we get $l'(hd\,\tau)$, thus, $(\tau, b, l' \Rightarrow\!\Diamond r_\varphi) \in \mathcal{D}$.

Moreover, using (†), $last(b, l' \Rightarrow\Diamond r_\varphi) > last(b, l_\varphi \Rightarrow\Diamond r_\varphi)$. Hence, $(\tau, b, l' \Rightarrow\Diamond r_\varphi) \prec (\tau, b, l_\varphi \Rightarrow\Diamond r_\varphi)$ and then using the well-founded induction hypothesis we obtain $k \le len\,\tau$ such that $(rhs\,\varphi)(\tau\,k)$, proving the lemma in this case.

- if the rule is [Spl] then $l_\varphi \triangleq l_1 \sqcup l_2$ and the node N has two successors, labelled $\mathcal{S}, \mathcal{H}' \Vdash (b, l_1 \Rightarrow\Diamond r_\varphi)$ and $\mathcal{S}, \mathcal{H}' \Vdash (b, l_1 \Rightarrow\Diamond r_\varphi)$, respectively. Hence, $(b, l_1 \Rightarrow\Diamond r_\varphi) \in \mathsf{Con}$ and $(b, l_2 \Rightarrow\Diamond r_\varphi) \in \mathsf{Con}$. From $(l_\varphi)(hd\,\tau)$ we obtain $l_1\,(hd\,\tau)$ or $l_2\,(hd\,\tau)$. We first consider the subcase $l_1\,(hd\,\tau)$. Then, $(\tau, b, l_1 \Rightarrow\Diamond r_\varphi) \in \mathcal{D}$. Using (†), $last(b, l_1 \Rightarrow\Diamond r_\varphi) > last(b, l_\varphi \Rightarrow\Diamond r_\varphi)$. Hence, $(\tau, b, l_1 \Rightarrow\Diamond r_\varphi) \prec (\tau, b, l_\varphi \Rightarrow\Diamond r_\varphi)$ and using the well-founded induction hypothesis we obtain $k \le len\,\tau$ such that $(rhs\,\varphi)(\tau\,k)$, proving the lemma in this subcase. The subcase $l_2\,(hd\,\tau)$ is identical.
- if the rule is [Tra] then the node N has two successors, labelled by $\mathcal{S}, \mathcal{H}' \Vdash (b, l_\varphi \Rightarrow\Diamond m)$ and by $\mathcal{S}, \mathcal{H}' \Vdash (b, m \Rightarrow\Diamond r_\varphi)$, hence, $(b, l_1 \Rightarrow\Diamond m) \in \mathsf{Con}$ and $(b, m \Rightarrow\Diamond r_\varphi) \in \mathsf{Con}$. It follows that $(\tau, b, l_\varphi \Rightarrow\Diamond m) \in \mathcal{D}$. Using (†), $last(b, l_\varphi \Rightarrow\Diamond m) > last(b, l_\varphi \Rightarrow\Diamond r_\varphi)$. Hence, $(\tau, b, l_\varphi \Rightarrow\Diamond m) \prec (\tau, b, l_\varphi \Rightarrow\Diamond r_\varphi)$. B the well-founded induction hypothesis we get $k_1 \le len\,\tau$ such that $m\,(\tau\,k_1)$.
  * if $k_1 = 0$ then $m\,(hd\,\tau)$. It follows that $(\tau, b, m \Rightarrow\Diamond r_\varphi) \in \mathcal{D}$, and, using (†), $last(b, m \Rightarrow\Diamond r_\varphi) > last(b, l_\varphi \Rightarrow\Diamond r_\varphi)$. Hence, $(\tau, b, m \Rightarrow\Diamond r_\varphi) \prec (\tau, b, l_\varphi \Rightarrow\Diamond r_\varphi)$. Using the well-founded induction hypothesis we obtain $k \le len\,\tau$ such that $r_\varphi\,(\tau\,k)$, proving the lemma in this subcase.
  * if $k_1 > 0$ then let τ' be the prefix of τ starting at k_1. Hence, $len\,\tau' < len\,\tau$ and $m\,(hd\,\tau')$. It follows that $(\tau', b, m \Rightarrow\Diamond r_\varphi) \in \mathcal{D}$, and, by definition of $\prec$, $(\tau', b, m \Rightarrow\Diamond r_\varphi) \prec (\tau, b, l_\varphi \Rightarrow\Diamond r_\varphi)$. Using the well-founded induction hypothesis, we then obtain $k_2 \le len\,\tau'$ such that $r_\varphi\,(\tau'\,k_2)$. But $k \triangleq k_1 + k_2 \le len\,\tau$, and $\tau\,k = \tau'\,k_2$, hence, $r_\varphi\,(\tau\,k)$, proving the lemma in this subcase.
- if the rule is [Stp] then the node N has one successor labelled $\mathcal{S}, \mathcal{H}' \Vdash (\mathrm{T}, \partial l_\varphi \Rightarrow\Diamond r_\varphi)$, thus, $(\mathrm{T}, \partial l_\varphi \Rightarrow\Diamond r_\varphi) \in \mathsf{Con}$, and $l_\varphi \sqcap \bullet \subseteq \bot$.
  * assume first $\tau \triangleq s$ for some $s \in S$. Hence, $\bullet\, s$ and since $l_\varphi\,(hd\,\tau)$ we get $l_\varphi\,s$ which together with $\bullet\, s$ contradict $l_\varphi \sqcap \bullet \subseteq \bot$.
  * thus, $\tau \triangleq s\,\tau'$, for some $\tau' \in fPaths$ with $s \to (hd\,\tau')$. From this and $l_\varphi\,s$ and using the definition of the ∂ endofunction, $(\partial l_\varphi)(hd\,\tau')$. Since $len\,\tau' = len\,\tau - 1$ It follows that $(\tau', \mathrm{T}, \partial l_\varphi \Rightarrow\Diamond r_\varphi) \prec (\tau, b, l_\varphi \Rightarrow\Diamond r_\varphi)$ and using the well-founded induction hypothesis, there is $k' \le len\,\tau'$ such that $r_\varphi\,(\tau'\,k')$. Setting $k = k' + 1$ we get $k \le len\,\tau$ and $\tau\,k = \tau'\,k'$, thus, $r_\varphi\,(\tau\,k)$, which proves the lemma in this case.
- if the rule is [Cut], then the node N has two successors labelled $\mathcal{S}, \mathcal{H}' \Vdash (\mathrm{F}, l'_\varphi \Rightarrow\Diamond r'_\varphi)$ and $\mathcal{S}, \mathcal{H}' \cup \{(\mathrm{F}, l'_\varphi \Rightarrow\Diamond r'_\varphi)\} \Vdash (b, l_\varphi \Rightarrow\Diamond r_\varphi)$, respectively. However, by (†), N has no successor in Θ with the conclusion $(b, l_\varphi \Rightarrow\Diamond r_\varphi)$, a contradiction. The rule is not applicable for the chosen node with its maximality property.
- if the rule is [Cof] then the node N has one successor, labelled $\mathcal{S}, \mathcal{H}' \cup \{\mathrm{F}, l_\varphi \Rightarrow\Diamond r_\varphi\} \Vdash (\mathrm{F}, l_\varphi \Rightarrow\Diamond r_\varphi)$. Assuming $b = \mathrm{F}$ we obtain as above that the rule is not applicable for the chosen node with its maximality property. Hence, $b = \mathrm{T}$. We thus have $(\mathrm{T}, l_\varphi \Rightarrow\Diamond r_\varphi) \in \mathsf{Con}$ and $(\tau, \mathrm{T}, l_\varphi \Rightarrow\Diamond r_\varphi) \in \mathcal{D}$ and $(\tau, \mathrm{F}, l_\varphi \Rightarrow\Diamond r_\varphi) \prec (\tau, \mathrm{T}, l_\varphi \Rightarrow\Diamond r_\varphi)$, and using the well-founded induction hypothesis we obtain $k \le len\,\tau$ such that $r_\varphi\,(\tau\,k)$, proving the lemma in this subcase.
- if the rule is [Clr] then $\mathcal{H}' \triangleq \mathcal{H}'' \cup \{\varphi''\}$, and the node N has one successor, labelled $\mathcal{S}, \mathcal{H}'' \Vdash (b, l_\varphi \Rightarrow\Diamond r_\varphi)$. Again we obtain as above that the rule is not applicable for the chosen node with its maximality property, which proves the lemma in this last case as well.

Hence, in all possible cases, for the freely chosen $(\tau, b, \varphi) \in \mathcal{D}$ we found $k \le len\,\tau$ such that $(rhs\,\varphi)(\tau\,k)$. □

Theorem 7 (Soundness of $\Vdash$.) *If for all $(b', \varphi') \in \mathcal{H}$, $b' = \mathrm{F}$ and $\mathcal{S} \models \varphi'$, then $\mathcal{S}, \mathcal{H} \Vdash (\mathrm{F}, \varphi)$ implies $\mathcal{S} \models \varphi$.*

Proof Let $\varphi \triangleq l \Rightarrow\Diamond r$. Lett $\mathcal{D}$ be the domain corresponding to $\mathcal{S},\mathcal{H} \Vdash (\mathrm{F}, l \Rightarrow\Diamond r)$. Consider any $\tau \in fPaths$ such that $l(hd\,\tau)$. Then, $(\tau, \mathrm{F}, l \Rightarrow\Diamond r) \in \mathcal{D}$, hence, using lemma 9, there is $k \leq len\,\tau$ such that $r(\tau\,k)$. Hence, for any $\tau \in fPaths$ such that $l(hd\,\tau)$, there is $k \leq len\,\tau$ such that $r(\tau\,k)$. Using Lemma 12 we obtain $\mathcal{S} \models l \Rightarrow\Diamond r$, which concludes the proof. □

Lemma 11. *If $\mathcal{S}',\mathcal{H} \Vdash (b,\varphi)$ and $\mathcal{S}' \lhd \mathcal{S}$ then $\mathcal{S},\mathcal{H} \Vdash (b,\varphi)$.*

Proof We first make the following observation: $\mathcal{S}',\mathcal{H} \Vdash (b,\varphi)$ implicitly means $\varphi \in \Phi_{\mathcal{S}'}$, which thanks to the state-set inclusion induced by $\mathcal{S}' \lhd \mathcal{S}$ implies $\varphi \in \Phi_{\mathcal{S}}$ as well.

The proof goes by induction on the assumed proof Θ' of $\mathcal{S}',\mathcal{H} \Vdash (b,\varphi)$. We build a proof Θ of $\mathcal{S},\mathcal{H} \Vdash (b,\varphi)$ and a partial function M from the nodes of Θ to those of Θ', such that at any point in the construction, each leaf, labelled $\mathcal{S},\mathcal{H}' \Vdash (b',\varphi')$ of a partially-constructed tree Θ is mapped by M to exactly one node labelled $\mathcal{S}',\mathcal{H}' \Vdash (b',\varphi')$ in Θ'.

The root of Θ is labelled $\mathcal{S},\mathcal{H} \Vdash (b,\varphi)$ and is mapped by M to the root of Θ', labelled $\mathcal{S}',\mathcal{H} \Vdash (b,\varphi)$.

Assume Θ and M are partially built; we show how to continue this process. Let $\mathcal{S},\mathcal{H}' \Vdash (b',\varphi')$ be the label of a current leaf L in current partially-build tree Θ. Using the induction hypothesis, L is mapped by M to exactly one node L' labelled $\mathcal{S}',\mathcal{H}' \Vdash (b',\varphi')$ of Θ'. The construction proceeds as follows:

- if L' is a leaf in Θ', then L remains a leaf in Θ.
- if L' is not a leaf in Θ and its successors $L'_1 \dots L'_k$ (for $k = 1$ or $k = 2$) are generated by any rule of the proof system except [Stp], then the same rule is applied to L and generates the same number of successor node $L_1 \dots L_k$, such that if L'_i is labelled by $\mathcal{S}',\mathcal{H}'_i \Vdash (b'_i,\varphi'_i)$ then the corresponding L_i is labelled by $\mathcal{S},\mathcal{H}'_i \Vdash (b'_i,\varphi'_i)$; and M is extended to map each new L_i of Θ to the corresponding L'_i.
- L' is not a leaf in Θ and its successor L'_1 is generated by the rule [Stp]: let $\mathcal{S}',\mathcal{H}' \Vdash (b',\varphi')$ be the label of L', with $\varphi' \triangleq l \Rightarrow\Diamond r$. Since [Stp] has been applied, $l \sqcap \bullet_{\mathcal{S}'} \sqsubseteq \bot$. Since $\to' \subseteq \to$, $\bullet_{\mathcal{S}} \subseteq \bullet_{\mathcal{S}'}$, hence, $l \sqcap \bullet_{\mathcal{S}} \subseteq l \sqcap \bullet_{\mathcal{S}'} \sqsubseteq \bot$, thus, [Stp] can also be applied to L, generating a new node $\widehat{L_1}$. Now, thanks to the rule [Stp], L'_1 is labelled $\mathcal{S}',\mathcal{H}' \Vdash (\mathrm{T}, \partial' l \Rightarrow\Diamond r)$ where $\partial' l = \lambda s.\exists s'.l\,s' \wedge s' \to' s$. Similarly, $\widehat{L_1}$ is labelled $\mathcal{S},\mathcal{H}' \Vdash (\mathrm{T}, \partial l \Rightarrow\Diamond r)$ where $\partial l = \lambda s.\exists s'.l\,s' \wedge s' \to s$. Let now $\to'' \triangleq \to \setminus \to'$. Thus, $\to = \to' \cup \to''$, and we have the inclusion $\partial l = (\lambda s.\exists s'.l\,s' \wedge s' \to s) \sqsubseteq (\partial' l) \sqcup (\lambda s.\exists s'.l\,s' \wedge s' \to'' s)$.
  Let us assume there exist $s', s \in S$ such that $l\,s' \wedge s' \to'' s$. Since $l \in \Pi_{\mathcal{S}'}$, from $l\,s'$ we obtain $s' \in S'$. It follows that $s \in S \setminus S'$ because otherwise (by the second item in the definition of $\mathcal{S}' \lhd \mathcal{S}$) one would also have $s' \to' s$, in contradiction with $\to'' = \to \setminus \to'$. Now, $s' \to'' s$ implies $s' \to s$, which together with $s' \in S'$ and with $s \in S \setminus S'$ and the third item in the definition of $\mathcal{S}' \lhd \mathcal{S}$ implies $s' \in \bullet_{\mathcal{S}'}$. From the latter and $l\,s'$ we obtain a contradiction with $l \sqcap \bullet_{\mathcal{S}'} \sqsubseteq \bot$, which arose from assuming there exist $s', s \in S$ such that $l\,s' \wedge s' \to'' s$. It follows that there does not exist $s \in S$ such that $\exists s'.l\,s' \wedge s' \to'' s$; thus, $(\lambda s.\exists s'.l\,s' \wedge s' \to'' s) \sqsubseteq \bot$, and using the above inclusions, $\partial l \sqsubseteq \partial' l$.
  We apply to $\widehat{L_1}$ the rule [Str] using $\partial l \sqsubseteq \partial' l$ and obtain a leaf L_1 in Θ, labelled $\mathcal{S},\mathcal{H}' \Vdash (\mathrm{T}, \partial' l \Rightarrow\Diamond r)$, and we extend M to map L_1 to L'_1, which does have the corresponding label $\mathcal{S}',\mathcal{H}' \Vdash (\mathrm{T}, \partial' l \Rightarrow\Diamond r)$.

The inductive construction of the proof Θ and of the map M is complete; we deduce $\mathcal{S},\mathcal{H} \Vdash (b,\varphi)$. □