Exploiting over- and under-approximations for infinite-state counterpart models*

Fabio Gadducci[1], Alberto Lluch Lafuente[2], and Andrea Vandin[2]

[1] Department of Computer Science, University of Pisa, Italy
[2] IMT Institute for Advanced Studies Lucca, Italy

Abstract. Software systems with dynamic topology are often infinite-state. Paradigmatic examples are those modeled as graph transformation systems (GTSs) with rewrite rules that allow an unbounded creation of items. For such systems, verification can become intractable, thus calling for the development of approximation techniques that may ease the verification at the cost of losing in preciseness and completeness. Both over- and under-approximations have been considered in the literature, respectively offering more and less behaviors than the original system. At the same time, properties of the system may be either preserved or reflected by a given approximation. In this paper we propose a general notion of approximation that captures some of the existing approaches for GTSs. Formulae are specified by a generic quantified modal logic, one that also generalizes many specification logics adopted in the literature for GTSs. We also propose a type system to denote part of the formulae as either reflected or preserved, together with a technique that exploits under- and over-approximations to reason about typed as well as untyped formulae.

Keywords: graph transition systems, approximated verification, abstraction, graph logics.

1 Introduction

Various approaches have been proposed to equip visual specification formalisms with suitable tools and techniques for verification. Recently, quite some attention has been devoted to those proposals that have imported and adapted traditional approaches (such as model checking) to the area of graph transformation. Among others, we mention here two research lines that have integrated the techniques they argue for into suitable verification tools, namely GROOVE [12, 9, 14, 5, 15, 16] and AUGUR [1, 3, 4, 10, 2][3].

A main ingredient in those works is the adoption of a suitable language for property specification. The language is in form of a modal logic capturing very often two essential dimensions of the state space of graph transformation systems

* Partly supported by the EU FP7-ICT IP `ASCEns` and by the MIUR PRIN `SisteR`.
[3] See groove.cs.utwente.nl and www.ti.inf.uni-due.de/research/tools/augur2.

(GTSs): the *topological* structure of states (i.e. graphs) and the *temporal* structure of transitions (i.e. graph rewrites). The topological dimension is usually handled by variants of monadic second-order (MSO) logics [6], spatial logics [7] or regular expressions [12], while the temporal dimension is typically tackled with standard modal logics from the model checking tradition like LTL, CTL or the modal μ-calculus. Our own contribution [8] to this field follows the tradition of [2] and it is based on a quantified version of the μ-calculus that mixes temporal modalities and graph expressions in the MSO-style.

Such logics are in general not decidable, mainly because the state space of the transition system associated to a GTS (usually called *graph transition system*) is very often infinite. Paradigmatic examples are GTSs with rewrite rules allowing an unbounded creation of items. Verification becomes then untractable and calls for appropriate state space reduction techniques. For example, many efforts have been devoted to the definition of approximation techniques inspired by abstract interpretation. The main idea is to consider a *finite-state* abstract system that approximates (the properties of) an *infinite-state* one, so that verification becomes feasible (at the acceptable cost of losing preciseness in the verification results). Approximated systems represent either more or less behaviours than the original one, resulting respectively in over- or under-approximations. In general, in order to consider meaningful approximations, it is necessary to relate them with the original systems via behavioural relations, like *simulation* ones. Such approximation techniques have been developed in both the above mentioned research lines: namely *neighborhood abstractions* [5] and *unfoldings* [1, 3, 4].

Contribution. Even if such techniques have been shown to be very effective, we do believe that there is still space for pushing forward their exploitation in the verification of GTSs. In this paper we propose a general formalization of similarity-based approximations, and a verification technique exploiting them. We focus on the type system of [4] proposed within the unfolding technique to classify formulae as preserved or reflected by a given approximation. We extend and generalize such type system in several directions: (i) our type system is *technique-agnostic*, meaning that it does not require the approximated systems to be obtained with a particular mechanism (e.g. the unfolding one); (ii) we consider counterpart models, a generalization of graph transition systems; (iii) our type system is parametric with respect to a given simulation relation (while the original one considers only simulations with certain properties); (iv) we use the type system to reason on all formulae (rather than just on closed ones); and (v) we propose a technique that exploits over- and under-approximations to estimate properties more precisely, handling also part of the untyped formulae.

Synopsis. §2 provides the necessary background. §3 defines simulation relations between counterpart models. §4 provides a type system to classify formulae as *preserved*, *reflected* or *strongly preserved*, exploited in §5 to define the approximated evaluation of formulae. Finally, §6 discusses related works, and concludes the paper. The soundness of our approach is proved in Appendix.

2 Background

We summarize here the basic machinery of our approach: essentially, the notion of *counterpart models* (which generalize graph transition systems) and a logic to reason about such models. A detailed presentation can be found in [8].

2.1 Counterpart models

While graph transition systems have graphs associated to states, counterpart models use many-sorted algebras to denote the structure of states (*worlds*).

Recall that a *(many-sorted) signature* Σ is a pair (S_Σ, F_Σ) composed by a set of sorts $S_\Sigma = \{\tau_1, \cdots, \tau_m\}$ and by a set of function symbols $F_\Sigma = \{f_\Sigma : \tau_1 \times \ldots \times \tau_n \to \tau \mid \tau_i, \tau \in S_\Sigma\}$ typed over S_Σ^*, and that a *(many-sorted) algebra* $\mathbf{A}$ with signature Σ (a Σ-algebra) is a pair $(A, F_\Sigma^{\mathbf{A}})$ such that: (i) the *carrier* A is a set of elements typed over S_Σ; (ii) $F_\Sigma^{\mathbf{A}} = \{f_\Sigma^{\mathbf{A}} : A_{\tau_1} \times \ldots \times A_{\tau_n} \to A_\tau \mid f_\Sigma : \tau_1 \times \ldots \times \tau_n \to \tau \in F_\Sigma\}$ is a family of functions on A typed over S_Σ^*, where $A_\tau = \{a \in A \mid a : \tau\}$, and each $f_\Sigma \in F_\Sigma$ corresponds to a function $f_\Sigma^{\mathbf{A}}$ in $F_\Sigma^{\mathbf{A}}$.

Given two Σ-algebras $\mathbf{A}$ and $\mathbf{B}$, a *(partial) morphism* ϱ is a family of partial functions $\{\varrho_\tau : A_\tau \rightharpoonup B_\tau \mid \tau \in S_\Sigma\}$ typed over S_Σ, such that, for each function symbol $f_\Sigma : \tau_1 \times \ldots \times \tau_n \to \tau \in F_\Sigma$ and list of elements $a_1, \ldots, a_n$, if each function ϱ_{τ_i} is defined for the element a_i of type τ_i, then ϱ_τ is defined for the element $f_\Sigma^{\mathbf{A}}(a_1, \ldots, a_n)$ of type τ and the elements $\varrho_\tau(f_\Sigma^{\mathbf{A}}(a_1, \ldots, a_n))$ and $f_\Sigma^{\mathbf{B}}(\varrho_{\tau_1}(a_1), \ldots, \varrho_{\tau_n}(a_n))$ coincide. A morphism is injective, surjective or bijective if all the ϱ_τ are so, meaning that they are so over its domain of definition.

Example 1. The signature for directed graphs is (S_{Gr}, F_{Gr}). The set S_{Gr} consists of the sorts of nodes τ_N and edges τ_E, while the set F_{Gr} is composed by the function symbols $s : \tau_E \to \tau_N$ and $t : \tau_E \to \tau_N$, which determine the source and the target node of an edge. For example, in Fig. 1 the graph tagged with w_1 is $(N \uplus E, \{s, t\})$, where $N = \{u, v\}$, $E = \{e_1\}$, $s = \{e_1 \mapsto u\}$ and $t = \{e_1 \mapsto v\}$.

A basic ingredient of our logic are open terms. For this purpose we consider signatures Σ_X obtained by extending a many-sorted signature Σ with an enumerable set X of variables typed over S_Σ. We let X_τ denote the τ-typed subset of variables and with x_τ or $x : \tau$ a variable with sort τ. Similarly, we let ϵ_τ or $\epsilon : \tau$ indicate a τ-sorted term. The set $T(\Sigma_X)$ of (possibly open) *terms* obtained from Σ_X is the smallest set such that $X \subseteq T(\Sigma_X)$ and $f(\epsilon_1, \ldots, \epsilon_n) : \tau \in T(\Sigma_X)$ for any $f : \tau_1 \times \ldots \times \tau_n \to \tau \in F_\Sigma$ and $\epsilon_i : \tau_i \in T(\Sigma_X)$.

For ease of presentation, we omit the sort when it is clear from the context or when it is not necessary. Moreover, we fix a generic many-sorted signature Σ.

We are finally ready to introduce counterpart models, which can be seen as a generalization of graph transition systems (see e.g. [2]).

Definition 1 (Counterpart model). *Let $\mathcal{A}$ be the set of Σ-algebras. A* counterpart model *M is a triple $(W, \rightsquigarrow, d)$ such that W is a set of worlds, $d : W \to \mathcal{A}$ is a function assigning to each world a Σ-algebra, and $\rightsquigarrow \subseteq W \times (\mathcal{A} \rightharpoonup \mathcal{A}) \times W$ is the* accessibility relation *over W, enriched with (partial) morphisms (*counterpart relations*) between the algebras of the connected worlds.*

In the following we may use $w_1 \overset{cr}{\rightsquigarrow} w_2$ for $(w_1, cr, w_2) \in \rightsquigarrow$. In particular, for each $w_1 \overset{cr}{\rightsquigarrow} w_2$ we have that $cr : d(w_1) \to d(w_2)$ defines the counterparts of (the algebra of) w_1 in (the algebra of) w_2. Counterpart relations allow hence to avoid *trans-world identity*, the implicit identification of elements of different worlds sharing the same name. Element names thus have a meaning that is local to their world. For this reason, these relations allow for the creation, deletion, and type-respecting renaming and merging of elements. Duplication is forbidden: no cr associates any element of $d(w_1)$ to more than one of $d(w_2)$.

Should Σ be a signature for graphs, a counterpart model is a two-level hierarchical graph: at the higher level the nodes are the worlds $w \in W$, and the edges are the evolution steps labeled with the associated counterpart relation; at the lower level, each world w contains a graph representing its internal structure. In standard terminology, we consider a transition system labeled with algebra morphisms, as an immediate generalization of *graph transition systems* [2].

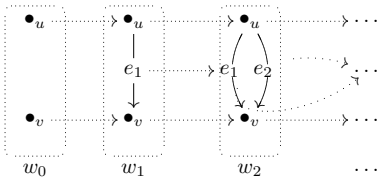

Fig. 1. A counterpart model

Example 2. The counterpart model in Fig. 1 is made of a sequence of worlds w_i, where world w_i is essentially associated to a graph $d(w_i)$ with i edges between nodes u and v. The counterpart relations (drawn with dotted lines) reflect the fact that each transition (w_i, cr_i, w_{i+1}) is such that cr_i is the identity for $d(w_i)$.

2.2 A logic to reason about counterpart models

We now present a logic for counterpart models. The main idea is that the interpretation of a formula in a model M provides sets of pairs (w, σ_w) where w is a world of M and σ_w associates first- and second-order variables to elements and to sets of elements, respectively, of $d(w)$. In what follows we fix a counterpart model M with signature Σ, and let X, $\mathcal{X}$ and $\mathcal{Z}$ denote alphabets of denumerable sets of first-order, second-order, and fix-point variables, respectively.

Definition 2 (Quantified modal formulae). *The set $\mathcal{F}_\Sigma$ of formulae ψ of our logic is generated by*

$$\psi ::= \ tt \mid \epsilon \in_\tau \chi \mid \neg\psi \mid \psi \vee \psi \mid \exists_\tau x.\psi \mid \exists_\tau \chi.\psi \mid \Diamond\psi \mid Z \mid \mu Z.\psi$$

where ϵ is a term over Σ_X, $\in_\tau$ is a family of membership predicates typed over S_Σ (stating that a term with sort τ belongs to a second-order variable with the same sort), $\exists_\tau$ quantifies over elements (sets of elements) with sort τ, $\Diamond$ is the "possibility" one-step modality, $Z \in \mathcal{Z}$, and μ denotes the least fixed point operator.

The semantics of the logic is given for *formulae-in-context* $\psi[\Gamma; \Delta]$, where $\Gamma \subset X$ and $\Delta \subset \mathcal{X}$ are the *first-* and *second-order contexts* of ψ, containing at least its free variables. However, we may omit types and contexts for the sake of presentation. As usual, we restrict to *monotonic* formulae, where fix-point variables occur under an even number of negations to ensure well-definedness.

The logic is simple, yet reasonably expressive. We can derive useful operators other than boolean connectives $\wedge$, $\rightarrow$, $\leftrightarrow$, and universal quantifiers $\forall_\tau$. For instance "$=_\tau$", the family of equivalence operators for terms in $T(S_\Sigma)$, typed over S_Σ, can be derived as $\epsilon_1 =_\tau \epsilon_2 \equiv \forall_\tau \chi.\ (\epsilon_1 \in_\tau \chi \leftrightarrow \epsilon_2 \in_\tau \chi)$. The greatest fix-point operator can be derived as $\nu Z.\psi \equiv \neg\mu Z.\neg\psi$, and the "necessarily" one-step modality as $\Box\psi \equiv \neg\Diamond\neg\psi$ *(ψ holds in all the next one-steps)*. Moreover, we can derive the standard CTL* temporal operators, as explained in detail in [8].

The semantic domain of our formulae are sets of assignments.

Definition 3 (Assignments). *An* assignment $\sigma_w = (\sigma^1_w, \sigma^2_w)$ *for a $w \in W$ is a pair of partial functions typed over S_Σ with $\sigma^1_w : X \rightharpoonup d(w)$ and $\sigma^2_w : \mathcal{X} \rightharpoonup 2^{d(w)}$. We use Ω_M (or just Ω) to denote the set of pairs (w, σ_w), for σ_w an assignment for w. A* fix-point variable assignment *is a partial function $\rho : \mathcal{Z} \rightharpoonup 2^{\Omega_M}$.*

Given a term ϵ and an assignment $\sigma = (\sigma^1, \sigma^2)$, we denote with $\sigma(\epsilon)$ or $\sigma^1(\epsilon)$ the lifting of σ^1 to $T(\Sigma_X)$. Intuitively, it evaluates ϵ under the assignment σ for its variables. If σ is undefined for any variable in ϵ, then $\sigma(\epsilon)$ is undefined as well.

Example 3. In our logic it is easy to define a predicate regarding the presence of an entity with sort τ in a world as $\mathbf{present}_\tau(x) \equiv \exists_\tau r.\ x = r$. The predicate evaluates in pairs $(w, (\{z \mapsto a\}, \lambda_2))$, with $a : \tau \in d(w)$. Consider again the model of Fig.1, then the predicate (omitting typings) $\mathbf{p}(x, y) \equiv \mathbf{present}(z) \wedge s(z) = x \wedge t(z) = y$ regards the existence of an edge z connecting a node x to a node y. Note that evaluating $\mathbf{p}(u, v)$ will provide assignments of z to edges connecting u to v.

We denote by $\Omega_M^{[\Gamma;\Delta]}$ those pairs $(w, (\sigma^1_w, \sigma^2_w))$ such that the domain of definition of σ^1_w is contained in Γ, and the one of σ^2_w is exactly Δ. As we will see, the evaluation function of our formulae is strongly based on $\Omega_M^{[\Gamma;\Delta]}$. Note the asymmetry in the definition: σ may be undefined over the elements of Γ, yet not over those of Δ. Intuitively, $\sigma(x)$ may be undefined if the element it was denoting has been deallocated, while we can always assign the empty set to $\sigma(\chi)$. We hence use partial first-order assignments to treat item deallocations.

Given models $M = (W, \rightsquigarrow, d)$, $M' = (W', \rightsquigarrow', d')$, worlds $w \in W$, $w' \in W'$, morphism $\phi : d(w) \rightarrow d'(w')$, and assignment $\sigma_w = (\sigma^1_w, \sigma^2_w)$ for w, we use $\phi \circ \sigma_w$ to denote the assignment $\sigma_{w'}$ (for w') obtained applying ϕ to the components of $\sigma_{w'}$, i.e. $\sigma^1_{w'} = \phi \circ \sigma^1_w$, and $\sigma^2_{w'} = 2^\phi \circ \sigma^2_w$, for 2^ϕ the lifting of ϕ to sets.

Assignments can be *restricted to* and *extended by* variables. Given an assignment $\sigma = (\sigma^1, \sigma^2)$ such that $(w, \sigma) \in \Omega^{[\Gamma, x; \Delta]}$, its *restriction* $\sigma \downarrow_x$ wrt. $x \notin \Gamma$ is the assignment $(\sigma^1 \downarrow_x, \sigma^2)$, such that $(w, \sigma \downarrow_x) \in \Omega^{[\Gamma;\Delta]}$, obtained by removing x from the domain of definition of σ^1. Vice versa, the *extension* $\sigma[^a/_x]$ of an assignment $\sigma = (\sigma^1, \sigma^2)$ such that $(w, \sigma) \in \Omega^{[\Gamma;\Delta]}$ wrt. mapping $x \mapsto a$ (for $x \notin \Gamma$ and $a \in d(w)$) is the assignment $(\sigma^1[^a/_x], \sigma^2)$ such that $(w, \sigma[^a/_x]) \in \Omega^{[\Gamma, x; \Delta]}$.

The notation above is analogously and implicitly given also for second-order variables, as well as for their lifting to sets $2^{\downarrow_x}$ and $2^{\uparrow_x}$. Intuitively, by extending $\Omega^{[\Gamma;\Delta]}$ with respect to a variable $x_\tau \notin \Gamma$, we replace every pair $(w, \sigma_w) \in \Omega^{[\Gamma;\Delta]}$ with the set $\{(w, \sigma_w[^a/_x]) \mid a : \tau \in d(w)\}$. Note that extensions may shrink the set of assignments, should the algebra associated to the world have no element

of the correct type. In general terms, the cardinality of $2^{\uparrow_{x_\tau}}(\{(w, \sigma_w)\})$ is the cardinality of $d(w)_\tau$, i.e. the cardinality of the set of elements of type τ in $d(w)$.

Given a transition $w \overset{cr}{\rightsquigarrow} w'$ and $(w, \sigma_w) \in \Omega^{[\Gamma;\Delta]}$, the *counterpart assignment* of σ_w relatively to cr (denoted $\sigma_w \overset{cr}{\rightsquigarrow} \sigma_{w'}$) is the assignment $\sigma_{w'} = cr \circ \sigma_w$. Thus, for $x \in \Gamma$, if $\sigma_w(x)$ is undefined, then $\sigma_{w'}(x)$ is undefined as well, meaning that if $\sigma_w(x)$ refers to an element deallocated in w, then also $\sigma_{w'}(x)$ does in w'; if $\sigma_w(x)$ is defined, but $cr(\sigma_w(x))$ is not, then the considered transition deallocates $\sigma_w(x)$. Whenever both $\sigma_w(x)$ and $cr(\sigma_w(x))$ are defined, then $\sigma_w(x)$ has to evolve in $\sigma_{w'}(x)$ accordingly to cr. As for $\chi \in \Delta$, the elements in $\sigma_w(\chi)$ preserved by cr are mapped in $\sigma_{w'}(\chi)$. If $\sigma_w(\chi)$ is defined, then $\sigma_{w'}(\chi)$ is also defined, with a cardinality equal or smaller, due to fusion or deletion of elements induced by cr.

We now introduce the evaluation of formulae in a model M, as a mapping from formulae $\psi[\Gamma;\Delta]$ into sets of pairs contained in $\Omega^{[\Gamma;\Delta]}$. Hence, the domain of the assignments in these pairs is, respectively, contained in Γ, and exactly Δ. Intuitively, a pair (w, σ_w) belongs to the semantics of $\psi[\Gamma;\Delta]$ if it holds in w under the assignment σ_w for its free variables. We assume that all the bound variables are different among themselves, and from the free ones.

Definition 4 (Semantics). *The evaluation of a formula $\psi[\Gamma;\Delta]$ in M under assignment $\rho : \mathcal{Z} \to 2^{\Omega^{[\Gamma;\Delta]}}$ is given by the function $[\![\cdot]\!]_\rho : \mathcal{F}^{[\Gamma;\Delta]} \to \Omega^{[\Gamma;\Delta]}$*

$$
\begin{aligned}
[\![tt[\Gamma;\Delta]]\!]_\rho &= \Omega^{[\Gamma;\Delta]} \\
[\![(\epsilon \in_\tau \chi)[\Gamma;\Delta]]\!]_\rho &= \{(w, \sigma_w) \in \Omega^{[\Gamma;\Delta]} \mid \sigma_w(\epsilon) \textit{ is defined and } \sigma_w(\epsilon) \in \sigma_w(\chi)\} \\
[\![\neg\psi[\Gamma;\Delta]]\!]_\rho &= \Omega^{[\Gamma;\Delta]} \setminus [\![\psi[\Gamma;\Delta]]\!]_\rho \\
[\![\psi_1 \vee \psi_2[\Gamma;\Delta]]\!]_\rho &= [\![\psi_1[\Gamma;\Delta]]\!]_\rho \cup [\![\psi_2[\Gamma;\Delta]]\!]_\rho \\
[\![\exists_\tau x.\ \psi[\Gamma;\Delta]]\!]_\rho &= 2^{\downarrow_x}(\{(w, \sigma_w) \in [\![\psi[\Gamma, x;\Delta]]\!]_{(2^{\uparrow_x} \circ \rho)} \mid \sigma_w(x) \textit{ is defined}\}) \\
[\![\exists_\tau \chi.\ \psi[\Gamma;\Delta]]\!]_\rho &= 2^{\downarrow_\chi}([\![\psi[\Gamma;\Delta,\chi]]\!]_{(2^{\uparrow_\chi} \circ \rho)}) \\
[\![\Diamond\psi[\Gamma;\Delta]]\!]_\rho &= \{(w, \sigma_w) \in \Omega^{[\Gamma;\Delta]} \mid \exists w \overset{cr}{\rightsquigarrow} w'.\ \exists (w', \sigma_{w'}) \in [\![\psi[\Gamma;\Delta]]\!]_\rho \ .\ \sigma_w \overset{cr}{\rightsquigarrow} \sigma_{w'}\} \\
[\![Z[\Gamma;\Delta]]\!]_\rho &= \rho(Z) \\
[\![\mu Z.\psi[\Gamma;\Delta]]\!]_\rho &= \mathit{lfp}(\lambda Y.[\![\psi[\Gamma;\Delta]]\!]_{\rho[Y/Z]})
\end{aligned}
$$

Notice how in order to evaluate $\exists_\tau x.\psi[\Gamma;\Delta]$, we first evaluate ψ extending Γ with x. Then, by dropping the pairs with undefined assignment for x, we obtain the ones whose worlds contain items satisfying ψ if assigned to x. The second-order case is similar, but assignments are defined for all the variables in Δ. Note that ρ is modified accordingly, thus ensuring a proper sorting for $\rho(Z)$.

Another interesting case arises evaluating formulae $\Diamond\psi[\Gamma;\Delta]$, where we search for pairs (w, σ_w) such that there exists a transition $w \overset{cr}{\rightsquigarrow} w'$ and a $\sigma_{w'}$ with $\sigma_w \overset{cr}{\rightsquigarrow} \sigma_{w'}$ and $(w', \sigma_{w'})$ belonging to the evaluation of $\psi[\Gamma;\Delta]$. In words, $\sigma_{w'}$ has to respect the relation induced by cr between the items of the two worlds.

Finally, the evaluation of a closed formula, i.e. with empty context, is a set of pairs (w, λ), for λ the empty assignment, ensuring that our proposal properly extends standard semantics for propositional modal logics.

Example 4 (Evaluation of formulae). Consider the formula of Example 3, the model of Fig. 1 and the empty assignment $\lambda = (\lambda_1, \lambda_2)$. Evaluating $[\![\mathbf{p}(u, v)]\!]^M$ results in $\{(w_1, (\{z \mapsto e_1\}, \lambda_2)), (w_2, (\{z \mapsto e_1\}, \lambda_2)), (w_2, (\{z \mapsto e_2\}, \lambda_2)), \dots\}$.

3 Behavioural equivalences for counterpart models

In this section we lift classical behavioural preorders and equivalences to counterpart models. For the sake of presentation, for the rest of the paper we fix two models $M = (W, \rightsquigarrow, d)$ and $M' = (W', \rightsquigarrow', d')$. Intuitively, we define relations from M to M' as sets of triples $(w, \phi, w') \in R$ formed by a world $w \in W$, a world $w' \in W'$ and a morphism $\phi : d(w) \to d(w')$ relating their respective topologies.

Definition 5 (Simulation). *Let $R \subseteq W \times (\mathcal{A} \rightharpoonup \mathcal{A}) \times W'$ be a set of triples (w, ϕ, w'), with $\phi : d(w) \to d'(w')$ a morphism. R is a* simulation from M to M' *if for every $(w_1, \phi_1, w_1') \in R$ we have that $w_1 \overset{cr}{\rightsquigarrow} w_2$ implies $w_1' \overset{cr'}{\rightsquigarrow} w_2'$ for some $w_2' \in W'$, with $(w_2, \phi_2, w_2') \in R$ and $\phi_2 \circ cr = cr' \circ \phi_1$. If $R^{-1} = \{(w', \phi^{-1}, w) \mid (w, \phi, w') \in R\}$ is well defined, and is also a simulation, then R (as well as R^{-1}) is called* bisimulation.

Notice how the ϕ components of bisimulations are forcibly injections. We call "iso" a bisimulation whose ϕ components are isomorphisms. We may abbreviate $(w, \phi, w') \in R$ in wRw' if ϕ is irrelevant. As usual, we define (bi)similarity as the greatest (bi)simulation, and say that M *is similar to* M' or that M' *simulates* M, written $M \sqsubseteq_R M'$ (where we may omit R), if there exists a simulation R from M to M' such that, for every $w \in W$, there exists at least a $w' \in W'$ with wRw'.

Example 5. Fig. 2 depicts three models: M (center), $\overline{M}$ (top) and $\underline{M}$ (bottom). The model M, taken from Example 2, is infinite-state and $\overline{M}$ and $\underline{M}$ can be understood as its over- and under-approximations, respectively. Indeed, we have relations $\overline{R}$ and $\underline{R}$ (denoted with double arrows) such that $\underline{M} \sqsubseteq_{\underline{R}} M \sqsubseteq_{\overline{R}} \overline{M}$.

Intuitively, $\underline{M}$ is a truncation of M considering only the first two transitions of M. Every tuple $(\underline{w}, \underline{\phi}, w)$ in $\underline{R}$ is such that $\underline{\phi} : \underline{d}(\underline{w}) \to d(w)$ is the identity.

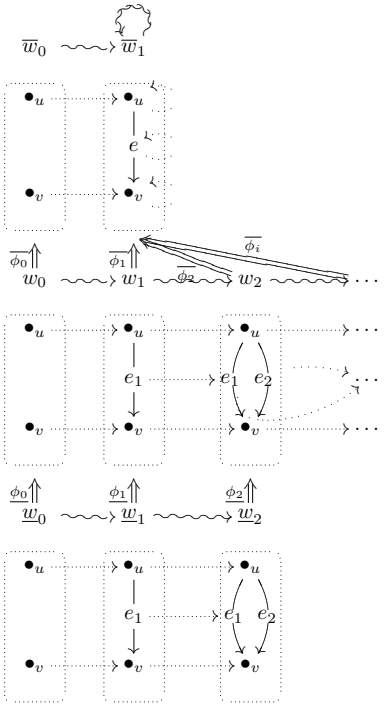

Fig. 2. Approximations

On the other hand, $\overline{M}$ can be seen as "M modulo the fusion of edges". That is, every tuple $(w, \overline{\phi}, \overline{w})$ in $\overline{R}$ is such that $\overline{\phi} : d(w) \to \overline{d}(\overline{w})$ is a bijection for nodes (in particular, the identity restricted to the nodes of $d(w)$) and a surjection on edges mapping every edge e_i into edge e.

Given a set of pairs $\omega \subseteq \Omega_M$ and a simulation R from M to M' we use $R(\omega)$ to denote the set $\{(w', \phi \circ \sigma_w) \mid (w, \sigma_w) \in \omega \wedge (w, \phi, w') \in R\}$. Note that R^{-1} is

not always well-defined since the morphisms in the triples (w, ϕ, w') may not be injective. However, we often use the pre-image $R^{-1}[\cdot]$ of R, defined for a set of pairs $\omega' \subseteq \Omega_{M'}$ as $R^{-1}[\omega'] = \{(w, \sigma_w) \in \Omega_M \mid \exists (w, \phi, w') \in R.\ (w', \phi \circ \sigma_w) \in \omega'\}$.

4 Preservation and reflection

As usual, the evaluation of formulae in a model M may be only approximated by a simulation M'. We hence introduce the usual notions of *preserved* formulae, those whose "satisfaction" in M implies their "satisfaction" in M', and *reflected* formulae, those whose "satisfaction" in M' implies their "satisfaction" in M. Of course, since the semantic domain of our logic are assignment pairs, the notion of "satisfaction" corresponds to the existence of such pairs.

Definition 6 (Preserved and reflected formulae). *Let R be a simulation from M to M' (i.e., $M \sqsubseteq_R M'$), $\psi[\Gamma; \Delta]$ a formula, and ρ an assignment. We say that ψ is* preserved *under R (written $\psi :_R \Rightarrow$) if $[\![\psi[\Gamma; \Delta]]\!]^{M'}_{R \circ \rho} \supseteq R([\![\psi[\Gamma; \Delta]]\!]^M_\rho)$;* reflected *under R (written $\psi :_R \Leftarrow$) if $R^{-1}[[\![\psi[\Gamma; \Delta]]\!]^{M'}_{R \circ \rho}] \subseteq [\![\psi[\Gamma; \Delta]]\!]^M_\rho$; and* strongly preserved *under R (written $\psi :_R \Leftrightarrow$) if $\psi :_R \Rightarrow$ and $\psi :_R \Leftarrow$.*

Note that the actual choice of ρ, Γ, and Δ is irrelevant. Note also how the definition for $\psi :_R \Leftarrow$ is stronger than the more intuitive one based on R. In particular, if $\psi :_R \Leftarrow$, then we additionally have that a pair in $\Omega_M^{[\Gamma;\Delta]} \setminus [\![\psi[\Gamma; \Delta]]\!]^M_\rho$ cannot be similar to any pair in $[\![\psi[\Gamma; \Delta]]\!]^{M'}_{R \circ \rho}$ (i.e. $R([\![\neg\psi[\Gamma; \Delta]]\!]^M_\rho) \cap [\![\psi[\Gamma; \Delta]]\!]^{M'}_{R \circ \rho} = \emptyset$).

Example 6. Consider again the predicate $\mathbf{p}(x, y)$ of Example 3 stating the existence of an edge connecting node x to node y, and the models $\underline{M} \sqsubseteq_{\underline{R}} M \sqsubseteq_{\overline{R}} \overline{M}$ of Example 5 shown in Fig. 2. It is easy to see that $\mathbf{p}(u, v)$ is strongly preserved both under $\underline{R}$ and under $\overline{R}$. Recall that in Example 4 we saw that $[\![\mathbf{p}(u, v)]\!]^M = \{(w_1, (\{z \mapsto e_1\}, \lambda_2)), (w_2, (\{z \mapsto e_1\}, \lambda_2)), (w_2, (\{z \mapsto e_2\}, \lambda_2)), \dots\}$ (for any ρ, thus neglected). Now, $[\![\mathbf{p}(u, v)]\!]^{\underline{M}} = \{(\underline{w}_1, (\{z \mapsto e_1\}, \lambda_2)), (\underline{w}_2, (\{z \mapsto e_1\}, \lambda_2)), (\underline{w}_2, (\{z \mapsto e_2\}, \lambda_2))\}$, and hence $\underline{R}([\![\mathbf{p}(u, v)]\!]^{\underline{M}})$ is $\{(w_1, (\{z \mapsto e_1\}, \lambda_2)), (w_2, (\{z \mapsto e_1\}, \lambda_2)), (w_2, (\{z \mapsto e_2\}, \lambda_2))\}$ which is clearly contained in $[\![\mathbf{p}(u, v)]\!]^M$. Moreover we also have that $\underline{R}^{-1}[[\![\mathbf{p}(u, v)[\Gamma; \Delta]]\!]^M] \subseteq [\![\mathbf{p}(u, v)[\Gamma; \Delta]]\!]^{\underline{M}}$. We hence have that $\mathbf{p}(u, v) :_{\underline{R}} \Leftrightarrow$. Similarly, we have that $[\![\mathbf{p}(u, v)]\!]^{\overline{M}} = \{(\overline{w}_1, (\{z \mapsto e\}, \lambda_2))\}$, and $\overline{R}([\![\mathbf{p}(u, v)]\!]^M) = \{(\overline{w}_1, (\{z \mapsto e\}, \lambda_2))\}$. Both conditions are again satisfied, and hence we have $\mathbf{p}(u, v) :_{\overline{R}} \Leftrightarrow$.

Of course, determining whenever a formula is preserved (or reflected) cannot be done in practice by performing the above check, since that would require to calculate the evaluation of the formula in the (possibly infinite) original model M, which is precisely what we want to avoid. Moreover, note that determining whenever a formula is preserved (and the same occurs for being reflected) is an undecidable problem, since our logic subsumes that of [4].

Nevertheless, we can apply the same approach of [4] and define a type system that approximates the preservation and reflection of formulae. In particular, our type system generalizes the one of [4] in several directions: (i) we consider

counterpart models, a generalization of graph transition systems; (ii) our type system is parametric with respect to the simulations R (while the original one is given for graph morphisms that are total and bijective for nodes, and total and surjective for edges); (iii) we use the type system to reason on all formulae (while the original proposal restricts to closed ones); and (iv) we propose a technique exploiting over- and under-approximations of a model to obtain more precise approximated formulae evaluations, and we handle part of the untyped formulae.

The type system is parametric with respect to the properties of R. In particular, we consider the properties of the morphisms in R, namely, for each sort τ, if they are τ-total (τ_t), τ-surjective (τ_s) or τ-bijective (τ_b). To easy the presentation, we say "τ_{prop} R", with $prop \in \{t, s, b\}$, whenever all $(w, \phi, w') \in R$ are such that ϕ is τ-*prop*. Moreover, we shall consider the case in which R is an iso-bisimulation.

Definition 7 (Type system). *Let R be a simulation from M to M' (i.e., $M \sqsubseteq_R M'$), ψ a formula, and $\mathcal{T} = \{\leftarrow, \rightarrow, \leftrightarrow\}$ a set of types. We say that ψ has type $d \in \mathcal{T}$ if $\psi : d$ can be inferred using the following rules*

$$\frac{}{tt :_R \leftrightarrow} \qquad \frac{d = \begin{cases} \rightarrow \text{ for } \tau_t \ R \\ \leftarrow \text{ for } \tau_b \ R \end{cases}}{\epsilon \in_\tau \chi :_R d} \qquad \frac{\psi :_R \rightarrow \quad \psi :_R \leftarrow}{\psi :_R \leftrightarrow} \qquad \frac{\psi :_R \leftrightarrow}{\psi :_R d}$$

$$\frac{\psi_i :_R d}{\psi_1 \vee \psi_2 :_R d} \qquad \frac{\psi :_R d \text{ with } d = \begin{cases} \rightarrow \text{ for } \tau_t \ R \\ \leftarrow \text{ for } \tau_s \ R \end{cases}}{\exists_\tau x.\psi :_R d \text{ and } \exists_\tau \chi.\psi :_R d} \qquad \frac{\psi :_R d}{\neg\psi :_R d^{-1}}$$

$$\frac{}{Z :_R \leftrightarrow} \qquad \frac{\psi :_R d \text{ with } d = \begin{cases} \rightarrow & \text{for any } R \\ \leftarrow & \text{for } R \text{ a iso-bisimulation} \end{cases}}{\Diamond\psi :_R d} \qquad \frac{\psi :_R d}{\mu Z.\psi :_R d}$$

where it is intended that $\rightarrow^{-1} = \leftarrow$, $\leftarrow^{-1} = \rightarrow$ and $\leftrightarrow^{-1} = \leftrightarrow$.

The type system is not complete, meaning that some formulae cannot be typed: if ψ cannot be typed, we then write $\psi :_R \bot$. However, the next proposition states its soundness: its proof can be found in the Appendix.

Proposition 1 (Type system soundness). *Let R be a simulation from M to M' (i.e., $M \sqsubseteq_R M'$) and ψ a formula. Then (i) $\psi :_R \rightarrow$ implies $\psi :_R \Rightarrow$; (ii) $\psi :_R \leftarrow$ implies $\psi :_R \Leftarrow$; and (iii) $\psi :_R \leftrightarrow$ implies $\psi :_R \Leftrightarrow$.*

Our type system can be instantiated for graph signatures, in order to obtain the one of [4] as a subsystem. In fact, the authors there consider only simulation relations R that are total on both sorts, as well as being $(\tau_N)_b$ (that is, bijective on nodes) and $(\tau_E)_s$ (surjective on edges).

Another instance is for iso-bisimulations. This is the case of the analysis of graph transition systems up to isomorphism (e.g. as implemented in [13]). In this case the type system is complete and correctly types every formula as $\psi :\leftrightarrow$.

Example 7. Consider the models $\underline{M} \sqsubseteq_{\underline{R}} M \sqsubseteq_{\overline{R}} \overline{M}$ of Fig. 2, and the formula $\mathbf{p}(u, v)$ of Example 6, where we saw that $\mathbf{p}(u, v) : \underline{R} \Leftrightarrow$ and $\mathbf{p}(u, v) : \overline{R} \Leftrightarrow$. Our

type system provides the types $\mathbf{p}(u,v) :_{\underline{R}} \leftrightarrow$ and (since $\overline{R}$ is not injective on edges) $\mathbf{p}(u,v) :_{\overline{R}} \rightarrow$. Note that the type for $\underline{R}$ is exactly inferred, while for $\overline{R}$ it is only approximated as we get *preserved* while it is actually *strongly preserved*.

5 Approximated semantics

Approximations can be used to estimate the evaluation of formulae. Consider the case of three models $\underline{M}$, M and $\overline{M}$, with $\underline{M} \sqsubseteq_{\underline{R}} M \sqsubseteq_{\overline{R}} \overline{M}$, as in Fig. 2, where $\underline{M}$ and $\overline{M}$ are, respectively, under- and over-approximations of M. Intuitively, an approximated evaluation of a formula ψ in $\underline{M}$ or $\overline{M}$ may provide us a lower- and upper-bound, defined for either $\underline{M}$ or $\overline{M}$, of the actual evaluation of ψ in M. We call under- and over-approximated evaluations the ones obtained using, respectively, under- (e.g. $\underline{M}$), and over-approximations (e.g. $\overline{M}$).

Exploiting approximated evaluations, we may address the *local* model checking problem: *"does a given assignment pair belong to the evaluation of the formula ψ in M?"*. Given that approximated semantics compute lower- and upper-bounds, we cannot define a complete procedure, i.e. one answering either *true* or *false*. A third value is required for the cases of uncertainty. For this purpose we use a standard three valued logic (namely Kleene's one) whose domain consists of the set of values $\mathbb{K} = \{T, F, ?\}$ (where ? reads "unknown"), and whose operators extend the standard Boolean ones with $T \vee ? = T$, $F \vee ? = ?$, $\neg ? = ?$ (i.e. where disjunction is the join in the complete lattice induced by the *truth* ordering relation $F < ? < T$). Moreover, we also consider a *knowledge addition* (binary, associative, commutative, partial) operation $\oplus : \mathbb{K} \times \mathbb{K} \rightharpoonup \mathbb{K}$ defined as $T \oplus T = T$, $F \oplus F = F$ and $x \oplus ? = x$ for any $x \in \mathbb{K}$. Notice how we intentionally let undefined the case of contradictory addition "$F \oplus T$".

In particular, given a formula, with our approximated semantics we are able to group the pairs of an approximating model in three distinct sets: the ones associated with T, the ones associated with F, and the ones associated with ?. For instance, the *over-approximated* semantics is defined as follows.

Definition 8 (Over-approximated semantics). *Let $\overline{R}$ be a simulation from M to $\overline{M}$ (i.e., $M \sqsubseteq_{\overline{R}} \overline{M}$) and ρ an assignment. The* over-approximated semantics *of $[\![\cdot]\!]^M_\rho$ in $\overline{M}$ via $\overline{R}$ is given by the function $\{\![\cdot]\!\}^{\overline{R}}_\rho : \mathcal{F}^{[\Gamma;\Delta]} \to (\Omega^{[\Gamma;\Delta]}_{\overline{M}} \to \mathbb{K})$, defined as $\{\![\psi[\Gamma;\Delta]]\!\}^{\overline{R}}_\rho = \{(\overline{p}, \overline{k}(\overline{p}, \psi[\Gamma;\Delta], \overline{R})) \mid \overline{p} \in \Omega^{[\Gamma;\Delta]}_{\overline{M}}\}$, where*

$$\overline{k}(\overline{p}, \psi[\Gamma;\Delta], \overline{R}) = \begin{cases} T & \text{if } \psi :_{\overline{R}} \leftarrow \text{ and } \overline{p} \in [\![\psi[\Gamma;\Delta]]\!]^{\overline{M}}_{\overline{R} \circ \rho} \\ F & \text{if } \psi :_{\overline{R}} \rightarrow \text{ and } \overline{p} \notin [\![\psi[\Gamma;\Delta]]\!]^{\overline{M}}_{\overline{R} \circ \rho} \\ ? & \text{otherwise} \end{cases}$$

Intuitively, the mapping of the pairs in $\Omega^{[\Gamma;\Delta]}_{\overline{M}}$ depends on the type of ψ. If it is typed as reflected, then all pairs in $[\![\psi[\Gamma;\Delta]]\!]^{\overline{M}}_{\overline{R} \circ \rho}$ are mapped to T, since their counterparts in M do certainly belong to the evaluation of ψ. Nothing can be said about the rest of the pairs, which are hence mapped to ?.

	$\{\!\{[\![\mathbf{p}(u,v)]\!]\}\!\}$	$\{\!\{[\![\neg\mathbf{p}(u,u)]\!]\}\!\}$	$\{\!\{[\![\mathbf{p}(u,v)\vee\neg\mathbf{p}(u,u)]\!]\}\!\}$	${}^{+}\{\!\{[\![\mathbf{p}(u,v)\vee\neg\mathbf{p}(u,u)]\!]\}\!\}$
$(\overline{w}_0,\lambda)$, $(\overline{w}_1,\lambda)$	F	T	?	T
$(\overline{w}_1,(z\mapsto e,\lambda_2))$	?	T	?	T
$(\underline{w}_0,\lambda)$,$(\underline{w}_1,\lambda)$,$(\underline{w}_2,\lambda)$,	F	T	T	T
$(\underline{w}_1,(z\mapsto e_1,\lambda_2)$	T	T	T	T
$(\underline{w}_2,(z\mapsto e_1,\lambda_2)$	T	T	T	T
$(\underline{w}_2,(z\mapsto e_2,\lambda_2)$	T	T	T	T
$(w_2,(z\mapsto e_2,\lambda_2))\models^{\overline{R}}[\![\cdot]\!]$	?	T	?	T
$(w_2,(z\mapsto e_2,\lambda_2))\models_{\underline{R}}[\![\cdot]\!]$	T	T	T	T
$(w_2,(z\mapsto e_2,\lambda_2))\models_{\underline{R}}^{\overline{R}}[\![\cdot]\!]$	T	T	T	T

Fig. 3. Approximated semantics and checks for some formulae

Dually, if ψ is typed as preserved, then all those pairs that do not belong to $[\![\psi[\Gamma;\Delta]]\!]^{\overline{M}}_{\overline{R}\circ\rho}$ are mapped to F because we know that their counterparts in M do certainly not belong to the evaluation of ψ. Again, nothing can be said about the rest of the pairs, which are hence mapped to ?.

Finally, if ψ cannot be typed, then all pairs are mapped to ?.

Notice how, in practice, we rarely have to explicitly define $\overline{R}\circ\rho$. In fact, formulae of our logic are thought to be evaluated under an empty assignment for fix-point variables, which is later manipulated during the evaluation, and clearly $\overline{R}\circ\emptyset=\emptyset$ for any $\overline{R}$. Moreover, it can be shown that the rules of the semantics preserve this equivalence (see also the case $\psi=\mu Z.\psi'$ in the proof of Proposition 1 that is given in Appendix).

We can hence use the over-approximated semantics to decide whether an assignment pair belongs to the evaluation of a formula in M as formalized below.

Definition 9 (Over-check). *Let $\overline{R}$ be a simulation from M to $\overline{M}$ (i.e., $M\sqsubseteq_{\overline{R}}\overline{M}$) and ρ an assignment. The* over-approximated model check *(shortly,* over-check*) of $[\![\cdot]\!]^M_\rho$ in $\overline{M}$ via $\overline{R}$ is given by the function $\cdot\models^{\overline{R}}[\![\cdot]\!]^M_\rho : \Omega_M^{[\Gamma;\Delta]}\times\mathcal{F}^{[\Gamma;\Delta]}\to\mathbb{K}$, defined as*

$$p\models^{\overline{R}}[\![\psi[\Gamma;\Delta]]\!]^M_\rho=\begin{cases}? & \text{if } \overline{R}(p)=\emptyset\\ \bigvee\limits_{\overline{p}\in\overline{R}(p)}\{\!\{[\![\psi[\Gamma;\Delta]]\!]\}\!\}^{\overline{R}}_\rho(\overline{p}) & \text{otherwise}\end{cases}$$

Example 8. Consider again the predicate $\mathbf{p}(x,y)$ of Example 3 stating the existence of an edge connecting node x to node y, and the models M and $\overline{M}$ with $M\sqsubseteq_{\overline{R}}\overline{M}$ of Example 5 shown in Fig. 2. In the first group of lines of Fig. 3 we exemplify the over-approximated semantics in $\overline{M}$ via $\overline{R}$ of $[\![\mathbf{p}(u,v)]\!]^M$, $[\![\neg\mathbf{p}(u,u)]\!]^M$, and $[\![\mathbf{p}(u,v)\vee\neg\mathbf{p}(u,u)]\!]^M$, considering the pairs $\Omega^{[z;\emptyset]}_{\overline{M}}=\{(\overline{w}_0,\lambda),(\overline{w}_1,\lambda),(\overline{w}_1,(z\mapsto e,\lambda_2))\}$. We recall from Example 7 that $\mathbf{p}(u,v):_{\overline{R}}\rightarrow$, and, hence, $\neg\mathbf{p}(u,u):_{\overline{R}}\leftarrow$ and $\mathbf{p}(u,v)\vee\neg\mathbf{p}(u,u):_{\overline{R}}\bot$. Moreover we know that $[\![\mathbf{p}(u,v)]\!]^{\overline{M}}=\{(w_1,(z\mapsto e,\lambda_2))\}$, and $[\![\neg\mathbf{p}(u,u)]\!]^{\overline{M}}=\Omega^{[z;\emptyset]}_{\overline{M}}$. Following Definition 8, we hence have that $(\overline{w}_0,\lambda)$ and $(\overline{w}_1,\lambda)$ are mapped to F for $\mathbf{p}(u,v)$, and to T for $\neg\mathbf{p}(u,u)$, while $(\overline{w}_1,(z\mapsto e,\lambda_2))$ is mapped to ? and to T. Different is the case of $\mathbf{p}(u,v)\vee\neg\mathbf{p}(u,u)$: it cannot be typed and its approximation hence maps the three pairs to ?.

In the third group of lines of Fig. 3 we find the over-check "$\cdot \models^{\overline{R}} [\![\cdot]\!]$" of $(w_2, (z \mapsto e_2, \lambda_2))$ in $\overline{M}$ via $\overline{R}$ for the three formulae. Note that $\overline{R}((w_2, (z \mapsto e_2, \lambda_2))) = (\overline{w}_1, (z \mapsto e, \lambda_2))$, hence the over-checks of $\mathbf{p}(u, v)$ and of $\mathbf{p}(u, v) \vee \neg\mathbf{p}(u, u)$ give ?, because no pair in $\overline{R}((w_2, (z \mapsto e_2, \lambda_2)))$ is mapped to either T or F. Instead, given that $\{[\neg\mathbf{p}(u, u)]\}((\overline{w}_1, (z \mapsto e, \lambda_2))) = T$, then we have $(w_2, (z \mapsto e_2, \lambda_2)) \models^{\overline{R}} [\![\neg\mathbf{p}(u, u)]\!] = T$.

With the next proposition we state that the above described check is sound.

Proposition 2 (Soundness of over-check). *Let $\overline{R}$ be a simulation from M to $\overline{M}$ (i.e., $M \sqsubseteq_{\overline{R}} \overline{M}$), $\psi[\Gamma; \Delta]$ a formula, and ρ an assignment. Then (i) $p \models^{\overline{R}} [\![\psi[\Gamma; \Delta]]\!]^M_\rho = T$ implies $p \in [\![\psi[\Gamma; \Delta]]\!]^M_\rho$; and (ii) $p \models^{\overline{R}} [\![\psi[\Gamma; \Delta]]\!]^M_\rho = F$ implies $p \notin [\![\psi[\Gamma; \Delta]]\!]^M_\rho$.*

Now, we can define the under-approximated semantics in a specular way.

Definition 10 (Under-approximated semantics). *Let $\underline{R}$ be a simulation from $\underline{M}$ to M (i.e., $\underline{M} \sqsubseteq_{\underline{R}} M$) and ρ an assignment. Then, the* under-approximated semantics *of $[\![\cdot]\!]^M_\rho$ in $\underline{M}$ via $\underline{R}$ is the function $\{\![\cdot]\!\}^{\underline{R}}_\rho : \mathcal{F}^{[\Gamma;\Delta]} \to (\Omega^{[\Gamma;\Delta]}_{\underline{M}} \to \mathbb{K})$, defined as $\{\![\psi[\Gamma; \Delta]]\!\}^{\underline{R}}_\rho = \{\underline{p} \mapsto \underline{k}(\underline{p}, \psi[\Gamma; \Delta], \underline{R}) \mid \underline{p} \in \Omega^{[\Gamma;\Delta]}_{\underline{M}}\}$, where*

$$\underline{k}(\underline{p}, \psi[\Gamma; \Delta], \underline{R}) = \begin{cases} T & \text{if } \psi :_{\underline{R}} \rightarrow \text{ and } \underline{p} \in [\![\psi[\Gamma; \Delta]]\!]^{\underline{M}}_{\underline{R}^{-1}[\cdot] \circ \rho} \\ F & \text{if } \psi :_{\underline{R}} \leftarrow \text{ and } \underline{p} \notin [\![\psi[\Gamma; \Delta]]\!]^{\underline{M}}_{\underline{R}^{-1}[\cdot] \circ \rho} \\ ? & \text{otherwise} \end{cases}$$

As for over-approximation, the use of $\underline{R}^{-1}[\cdot] \circ \rho$ is not a problem in practice.

We can define an under-approximated model checking procedure as follows.

Definition 11 (Under-check). *Let $\underline{R}$ be a simulation from $\underline{M}$ to M (i.e., $\underline{M} \sqsubseteq_{\underline{R}} M$) and ρ an assignment. The* under-approximated model check *(shortly,* under-check*) of $[\![\cdot]\!]^M_\rho$ in $\underline{M}$ via $\underline{R}$ is given by the function $\cdot \models_{\underline{R}} [\![\cdot]\!]^M_\rho : \Omega^{[\Gamma;\Delta]}_M \times \mathcal{F}^{[\Gamma;\Delta]} \to \mathbb{K}$, defined as*

$$p \models_{\underline{R}} [\![\psi[\Gamma; \Delta]]\!]^M_\rho = \begin{cases} ? & \text{if } \underline{R}^{-1}[p] = \emptyset \\ \bigvee\limits_{\underline{p} \in \underline{R}^{-1}[p]} \{\![\psi[\Gamma; \Delta]]\!\}^{\underline{R}}_\rho(\underline{p}) & \text{otherwise} \end{cases}$$

Next proposition states the soundness of the under-check procedure.

Proposition 3 (Soundness of under-check). *Let $\underline{R}$ be a simulation from $\underline{M}$ to M (i.e., $\underline{M} \sqsubseteq_{\underline{R}} M$), $\psi[\Gamma; \Delta]$ a formula, and ρ an assignment. Then (i) $p \models_{\underline{R}} [\![\psi[\Gamma; \Delta]]\!]^M_\rho = T$ implies $p \in [\![\psi[\Gamma; \Delta]]\!]^M_\rho$; and (ii) $p \models_{\underline{R}} [\![\psi[\Gamma; \Delta]]\!]^M_\rho = F$ implies $p \notin [\![\psi[\Gamma; \Delta]]\!]^M_\rho$.*

We finally show how to combine sets of under- and over-approximations.

Definition 12 (Approximated check). *Let $\{\underline{R}_i\}$ be a set of simulations from $\{\underline{M}_i\}$ to M and $\{\overline{R}_j\}$ a set of simulations from M to $\{\overline{M}_j\}$ (i.e., $\underline{M_i} \sqsubseteq_{\underline{R_i}} M \sqsubseteq_{\overline{R_j}} \overline{M_j}$ for any i and j) and ρ an assignment. The* approximated check *of $[\![\cdot]\!]^M_\rho$ in $\{\underline{M_i}\}$ and $\{\overline{M_j}\}$ via $\{\underline{R_i}\}$ and $\{\overline{R_j}\}$ is the function $\cdot \models^{\{\overline{R_j}\}}_{\{\underline{R_i}\}} [\![\cdot]\!]^M_\rho : \Omega^{[\Gamma;\Delta]}_M \times \mathcal{F}^{[\Gamma;\Delta]} \rightharpoonup \mathbb{K}$, defined as*

$$p \models^{\{\overline{R_j}\}}_{\{\underline{R_i}\}} [\![\psi[\Gamma;\Delta]]\!]^M_\rho = \bigoplus_j (p \models^{\overline{R}_j} [\![\psi[\Gamma;\Delta]]\!]^M_\rho) \oplus \bigoplus_i (p \models_{\underline{R}_i} [\![\psi[\Gamma;\Delta]]\!]^M_\rho) \oplus ?$$

Note that, even if $\oplus$ is partial, the approximated check is well-defined since Propositions 2 and 3 ensure that we never have to combine contradictory results (e.g. $T \oplus F$). It is also easy to see that the soundness result of Propositions 2 and 3 allows to conclude the soundness of the approximated check.

Theorem 1 (Soundness of approximated check). *Let $\{\underline{R}_i\}$ be a set of simulations from $\{\underline{M}_i\}$ to M and $\{\overline{R}_j\}$ a set of simulations from M to $\{\overline{M}_j\}$ (i.e., $\underline{M_i} \sqsubseteq_{\underline{R_i}} M \sqsubseteq_{\overline{R_j}} \overline{M_j}$ for any i and j), $\psi[\Gamma;\Delta]$ a formula, and ρ an assignment. Then (i) $p \models^{\{\overline{R_j}\}}_{\{\underline{R_i}\}} [\![\psi[\Gamma;\Delta]]\!]^M_\rho = T$ implies $p \in [\![\psi[\Gamma;\Delta]]\!]^M_\rho$; and (ii) $p \models^{\{\overline{R_j}\}}_{\{\underline{R_i}\}} [\![\psi[\Gamma;\Delta]]\!]^M_\rho = F$ implies $p \notin [\![\psi[\Gamma;\Delta]]\!]^M_\rho$.*

5.1 Dealing with untyped formulae

Approximated semantics provide us with a suitable evaluation of any formula, even though its result may not be meaningful, since we may have empty lower-bounds or unbounded upper-bounds as particular instances, namely when all the pairs are assigned to ?. Indeed, this is the case of formulae that cannot be typed with our type system. In order to obtain a more significant approximation also in those cases, we may try to enrich our approximated semantics by rules mimicking the actual semantics.

We can thus extend both under- and over-approximated semantics (Definitions 8 and 10): in the following we present the enrichment for over-approximated semantics only, with the under-approximated case treated seamlessly.

Definition 13 (Enriched over-approximated semantics). *Let $\overline{R}$ be a simulation from M to $\overline{M}$ (i.e., $M \sqsubseteq_{\overline{R}} \overline{M}$) and ρ an assignment, such that $\{\![\cdot]\!\}^{\overline{R}}_\rho$ is the over-approximated semantics of $[\![\cdot]\!]^M_\rho$ in $\overline{M}$ via $\overline{R}$. The* enriched *over-approximated semantics of $[\![\cdot]\!]^M_\rho$ in $\overline{M}$ via $\overline{R}$ is given by the function ${}^+\{\!\{\cdot\}\!\}^{\overline{R}}_\rho : \mathcal{F}^{[\Gamma;\Delta]} \to (\Omega^{[\Gamma;\Delta]}_{\overline{M}} \to \mathbb{K})$ defined as*

$${}^+\{\!\{\psi[\Gamma;\Delta]\}\!\}^{\overline{R}}_\rho = \begin{cases} {}^+\{\!\{\psi_1[\Gamma;\Delta]\}\!\}^{\overline{R}}_\rho \vee {}^+\{\!\{\psi_2[\Gamma;\Delta]\}\!\}^{\overline{R}}_\rho & \text{if } \psi :_{\overline{R}} \bot \text{ and } \psi \equiv \psi_1 \vee \psi_2 \\ \neg {}^+\{\!\{\psi_1[\Gamma;\Delta]\}\!\}^{\overline{R}}_\rho & \text{if } \psi :_{\overline{R}} \bot \text{ and } \psi \equiv \neg\psi_1 \\ \{[\psi[\Gamma;\Delta]]\}^{\overline{R}}_\rho & \text{otherwise} \end{cases}$$

We may enrich the under-approximated semantics exactly in the same way, and thus straightforwardly define an enriched version "$\cdot \,^{+}\models^{\{\overline{R_j}\}}_{\{R_i\}} [\![\cdot]\!]^M_\rho$" of the approximated checking by replacing both approximated semantics with their enriched variants. It is also easy to verify that also this new check is sound.

6 Conclusions and further works

In the present work we proposed a general framework for similarity-based approximations, and we exploited them for developing a verification technique based on a suitable type system for formulae of a second-order modal logic with fix-point operators. The logic was previously introduced for the specification of systems with dynamic topology [8, 11], and it is thus now equipped with a powerful abstraction mechanism.

Our approach can be seen as an evolution of the verification technique for graph transformation systems based on temporal graph logics and unfoldings [4, 2], which is extended on the kind of models and of simulations under analysis.

We are also confident that our proposal may provide interesting insights for other approximation techniques presented in the literature, such as *neighbourhood abstraction* [5], where states are labelled graphs, and suitable abstraction morphisms (i.e. surjective graph morphisms, similar to the morphisms of our simulations) coalesce nodes and edges of concrete states according to their neighbourhood similarity. The logic that is adopted there is less expressive than the one we use here (as well as of the one used in [2]), but it offers the advantage that all formulae are both reflected and preserved.

We foresee several directions for further research. First, we plan to enrich our prototypal model checker for finite models [11] with the techniques presented here. Second, we would like to investigate the enrichment of approximated semantics in order to deal with more untyped formulae. An interesting question in this regard is whether we can use both an under- and an over-approximation *simultaneously*, by translating assignment pairs back and forth via the composition of the corresponding abstraction and concretization functions.

References

1. Baldan, P., Corradini, A., König, B.: A static analysis technique for graph transformation systems. In: Larsen, K., Nielsen, M. (eds.) CONCUR. LNCS, vol. 2154, pp. 381–395. Springer (2001)
2. Baldan, P., Corradini, A., König, B., Lluch Lafuente, A.: A temporal graph logic for verification of graph transformation systems. In: Fiadeiro, J.L., Schobbens, P.Y. (eds.) WADT. LNCS, vol. 4409, pp. 1–20. Springer (2007)
3. Baldan, P., König, B.: Approximating the behaviour of graph transformation systems. In: Corradini, A., Ehrig, H., Kreowski, H.J., Rozenberg, G. (eds.) ICGT. LNCS, vol. 2505, pp. 14–29. Springer (2002)
4. Baldan, P., König, B., König, B.: A logic for analyzing abstractions of graph transformation systems. In: Cousot, R. (ed.) SAS. LNCS, vol. 2694, pp. 255–272. Springer (2003)

5. Bauer, J., Boneva, I., Kurbán, M.E., Rensink, A.: A modal-logic based graph abstraction. In: Ehrig, H., Heckel, R., Rozenberg, G., Taentzer, G. (eds.) ICGT. LNCS, vol. 5214, pp. 321–335. Springer (2008)
6. Courcelle, B., Engelfriet, J.: Graph structure and monadic second-order logic, a language theoretic approach. Cambridge University Press (2012)
7. Dawar, A., Gardner, P., Ghelli, G.: Expressiveness and complexity of graph logic. Information and Computation 205(3), 263–310 (2007)
8. Gadducci, F., Lluch Lafuente, A., Vandin, A.: Counterpart semantics for a second-order mu-calculus. Fundamenta Informaticae 118(1-2) (2012)
9. Ghamarian, A.H., de Mol, M., Rensink, A., Zambon, E., Zimakova, M.: Modelling and analysis using groove. STTT 14(1), 15–40 (2012)
10. König, B., Kozioura, V.: Counterexample-guided abstraction refinement for the analysis of graph transformation systems. In: Hermanns, H., Palsberg, J. (eds.) TACAS. LNCS, vol. 3920, pp. 197–211. Springer (2006)
11. Lluch Lafuente, A., Vandin, A.: Towards a maude tool for model checking temporal graph properties. In: Gadducci, F., Mariani, L. (eds.) GT-VMT. ECEASST, vol. 42. EAAST (2011)
12. Rensink, A.: Towards model checking graph grammars. In: Leuschel, M., Gruner, S., Lo Presti, S. (eds.) AVOCS. DSSE-TR, vol. 2003-2. University of Southampton (2003)
13. Rensink, A.: Isomorphism checking in groove. In: Zündorf, A., Varró, D. (eds.) GraBaTs. ECEASST, vol. 1. EAAST (2006)
14. Rensink, A., Distefano, D.: Abstract graph transformation. In: WWV. ENTCS, vol. 157(1), pp. 39–59. Elsevier (2006)
15. Rensink, A., Zambon, E.: Neighbourhood abstraction in GROOVE. In: de Lara, J., Varró, D. (eds.) GraBaTs. ECEASST, vol. 32. EAAST (2010)
16. Zambon, E., Rensink, A.: Using graph transformations and graph abstractions for software verification. In: Corradini, A. (ed.) ICGT - Doctoral Symposium. ECEASST, vol. 38. EASST (2011)

Appendix

In this appendix we prove the three Propositions 1, 2 and 3, implying consequently also Theorem 1.

We first prove that Proposition 1 holds, stating the soundness of our type system with respect to Definition 6. Namely, we show that if our type system assigns a type to a formula, then it is coherent with the preservation and/or reflection of the formula.

Proposition 1 (Type system soundness). *Let R be a simulation from M to M' (i.e., $M \sqsubseteq_R M'$) and ψ a formula. Then (i) $\psi :_R \rightarrow$ implies $\psi :_R \Rightarrow$; (ii) $\psi :_R \leftarrow$ implies $\psi :_R \Leftarrow$; and (iii) $\psi :_R \leftrightarrow$ implies $\psi :_R \Leftrightarrow$.*

Proof. We focus on the points (i) and (ii), since point (iii) follows directly. Rephrasing Definition 6, what we have to show is: for every $(w_1, \phi_1, w_1') \in R$, if (i) $\psi :_R \rightarrow$, then $(w_1, \sigma_{w_1}) \in [\![\psi[\Gamma;\Delta]]\!]^M_\rho$ implies $(w_1', \phi_1 \circ \sigma_{w_1}) \in [\![\psi[\Gamma;\Delta]]\!]^{M'}_{R\circ\rho}$; while, if (ii) $\psi :_R \leftarrow$, then $(w_1', \phi_1 \circ \sigma_{w_1}) \in [\![\psi[\Gamma;\Delta]]\!]^{M'}_{R\circ\rho}$ implies $(w_1, \sigma_{w_1}) \in [\![\psi[\Gamma;\Delta]]\!]^M_\rho$. The proof is done on structural induction on ψ.

The proposition trivially holds for the cases tt, Z and $\psi_1 \vee \psi_2$.

[$\psi \equiv \epsilon \in_\tau \chi :_R \rightarrow$] Following the semantics, we have $[\![\epsilon \in_\tau \chi[\Gamma;\Delta]]\!]^M_\rho = \{(w, \sigma) \in \Omega_M^{[\Gamma;\Delta]} \mid \sigma(\epsilon)$ is defined and $\sigma(\epsilon) \in \sigma(\chi)\}$. Hence there exists an $a : \tau \in d(w_1)$ such that $a = \sigma_{w_1}(\epsilon)$, and $a \in \sigma_{w_1}(\chi)$. Clearly, $\phi_1(a) \in \phi_1 \circ \sigma_{w_1}(\chi)$. A term ϵ is a variable or an operation applied to a term. From our type system we know that $\epsilon \in_\tau \chi$ has type $\rightarrow$ for τ_{total} R, which, together with the fact that morphisms preserve terms' operations, allows us to conclude $\phi_1 \circ \sigma_{w_1}(\epsilon) = \phi_1(a)$.

[$\psi \equiv \epsilon \in_\tau \chi :_R \leftarrow$] From $(w_1', \phi_1 \circ \sigma_{w_1}) \in [\![\epsilon \in_\tau \chi[\Gamma;\Delta]]\!]^{M'}_{R\circ\rho} \equiv \{(w', \sigma') \in \Omega_{M'}^{[\Gamma;\Delta]} \mid \sigma'(\epsilon)$ is defined and $\sigma'(\epsilon) \in \sigma'(\chi)\}$, we know that there exists an $a' : \tau \in d'(w_1')$ with $a' = \phi_1 \circ \sigma_{w_1}(\epsilon)$, and $a' \in \phi_1 \circ \sigma_{w_1}(\chi)$. From our type system we know that R is $\tau_{bijective}$, hence there exists an $a \in d(w_1)$ such that $\phi_1(a) = a'$. Clearly, $a \in \sigma_{w_1}(\chi)$, and $\sigma_{w_1}(\epsilon) = a$.

[$\psi \equiv \neg\psi' :_R \rightarrow$] We want to prove that $(w_1', \phi_1 \circ \sigma_{w_1}) \in [\![\neg\psi'[\Gamma;\Delta]]\!]^{M'}_{R\circ\rho}$, knowing that $(w_1, \sigma_{w_1}) \in [\![\neg\psi'[\Gamma;\Delta]]\!]^M_\rho \equiv \Omega_M^{[\Gamma;\Delta]} \setminus [\![\psi'[\Gamma;\Delta]]\!]^M_\rho$. In particular, $(w_1', \phi_1 \circ \sigma_{w_1})$ may belong either to $[\![\psi'[\Gamma;\Delta]]\!]^{M'}_{R\circ\rho}$ or to $\Omega_{M'}^{[\Gamma;\Delta]} \setminus [\![\psi'[\Gamma;\Delta]]\!]^{M'}_{R\circ\rho}$. By absurd consider $(w_1', \phi_1 \circ \sigma_{w_1}) \in [\![\psi'[\Gamma;\Delta]]\!]^{M'}_{R\circ\rho}$. From our type system we know that $\psi' :_R \leftarrow$, which, by induction hypothesis, implies $(w_1, \sigma_{w_1}) \in [\![\psi'[\Gamma;\Delta]]\!]^M_\rho$, obtaining a contradiction.

[$\psi \equiv \neg\psi' :_R \leftarrow$] This case is specular to the $\psi :_R \rightarrow$ one.

$[\psi \equiv \exists_\tau x.\psi' :_R \rightarrow]$ Following the semantics, $[\![\exists_\tau x.\ \psi'[\Gamma;\Delta]]\!]^M_\rho = 2^{\downarrow_x}(\{(w,\sigma) \in [\![\psi'[\Gamma,x;\Delta]]\!]^M_{2^{\uparrow_x}\circ\rho} \mid \sigma(x) \text{ is defined}\})$. From the type system we know that R is τ_{total}, hence $\phi_1 \circ \sigma_{w_1}(x)$ is defined iff $\sigma_{w_1}(x)$ is defined, allowing to reduce the problem in: $(w_1,\sigma_{w_1}) \in 2^{\downarrow_x}([\![\psi'[\Gamma,x;\Delta]]\!]^M_{2^{\uparrow_x}\circ\rho})$ implies $(w'_1,\phi_1\circ\sigma_{w_1}) \in 2^{\downarrow_x}([\![\psi'[\Gamma,x;\Delta]]\!]^{M'}_{2^{\uparrow_x}\circ\rho'})$. We also know that $\psi' :_R \rightarrow$, hence, by induction hypothesis, $(w_1,\sigma_{2w_1}) \in [\![\psi'[\Gamma,x;\Delta]]\!]^M_{2^{\uparrow_x}\circ\rho}$ implies $(w'_1,\phi_1\circ\sigma_{2w_1}) \in [\![\psi'[\Gamma,x;\Delta]]\!]^{M'}_{2^{\uparrow_x}\circ\rho'}$, with $\sigma_{2w_1} \in \Omega^{[\Gamma,x;\Delta]}_{w_1}$. Noting that $2^{\uparrow_x}$ and $2^{\downarrow_x}$ are monotone, we have $(w_1,\sigma_{w_1}) \in 2^{\downarrow_x}([\![\psi'[\Gamma,x;\Delta]]\!]^M_{2^{\uparrow_x}\circ\rho})$ implies $2^{\uparrow_x}((w_1,\sigma_{w_1})) \subseteq [\![\psi'[\Gamma,x;\Delta]]\!]^M_{2^{\uparrow_x}\circ\rho}$, and $(w_1,\sigma_{2w_1}) \in [\![\psi'[\Gamma,x;\Delta]]\!]^M_{2^{\uparrow_x}\circ\rho}$ implies $2^{\downarrow_x}((w_1,\sigma_{2w_1})) \in 2^{\downarrow_x}([\![\psi'[\Gamma,x;\Delta]]\!]^M_{2^{\uparrow_x}\circ\rho})$. It is now easy to see that for every $(w_1,\sigma_{w_1}) \in 2^{\downarrow_x}([\![\psi'[\Gamma,x;\Delta]]\!]^M_{2^{\uparrow_x}\circ\rho})$ there exists a $(w_1,\sigma_{2w_1}) \in [\![\psi'[\Gamma,x;\Delta]]\!]^M_{2^{\uparrow_x}\circ\rho}$ such that $(w_1,\sigma_{w_1}) = 2^{\downarrow_x}((w_1,\sigma_{2w_1}))$, for which, in turn, there exists a $(w'_1,\phi_1\circ\sigma_{2w_1}) \in [\![\psi'[\Gamma,x;\Delta]]\!]^{M'}_{2^{\uparrow_x}\circ\rho'}$ such that $(w'_1,\phi_1\circ\sigma_{w_1}) = 2^{\downarrow_x}((w'_1,\phi_1\circ\sigma_{2w_1}))$.

$[\psi \equiv \exists_\tau x.\psi' :_R \leftarrow]$ What we want to prove is $(w'_1,\phi_1\circ\sigma_{w_1}) \in [\![\exists_\tau x.\ \psi'[\Gamma;\Delta]]\!]^{M'}_{R\circ\rho}$ implies $(w_1,\sigma_{w_1}) \in [\![\exists_\tau x.\ \psi'[\Gamma;\Delta]]\!]^M_\rho$. From the type system we know that R is partial and surjective for the sort τ, hence we can again reduce the problem to $(w'_1,\phi_1\circ\sigma_{w_1}) \in 2^{\downarrow_x}([\![\psi'[\Gamma,x;\Delta]]\!]^{M'}_{2^{\uparrow_x}\circ\rho'})$ implies $(w_1,\sigma_{w_1}) \in 2^{\downarrow_x}([\![\psi'[\Gamma,x;\Delta]]\!]^M_{2^{\uparrow_x}\circ\rho})$. We also know that $\psi' :_R \leftarrow$, hence, by induction hypothesis, $(w'_1,\phi_1\circ\sigma_{2w_1}) \in [\![\psi'[\Gamma,x;\Delta]]\!]^{M'}_{2^{\uparrow_x}\circ\rho'}$ implies $(w_1,\sigma_{2w_1}) \in [\![\psi'[\Gamma,x;\Delta]]\!]^M_{2^{\uparrow_x}\circ\rho}$, with $\sigma_{2w_1} \in \Omega^{[\Gamma,x;\Delta]}_{w_1}$. The rest of the proof is similar to the $(\exists_\tau x.\psi' :_R \rightarrow)$ case.

$[\psi \equiv \exists_\tau \chi.\psi' :_R \leftrightarrow]$ The proofs are similar to the first-order cases.

$[\psi \equiv \Diamond\psi' :_R \rightarrow]$ From the semantics, $(w_1,\sigma_{w_1}) \in [\![\Diamond\psi'[\Gamma;\Delta]]\!]^M_\rho$ implies the existence of a $w_2 \in W$ s.t. $w_1 \overset{cr}{\rightsquigarrow} w_2$ and $(w_2,\sigma_{w_2}) \in [\![\psi'[\Gamma;\Delta]]\!]^M_\rho$, with $\sigma_{w_1} \overset{cr}{\rightsquigarrow} \sigma_{w_2}$. We apply the induction hypothesis: $(w,\phi,w') \in R$ and $(w,\sigma_w) \in [\![\psi'[\Gamma;\Delta]]\!]^M_\rho$ implies $(w',\phi\circ\sigma_w) \in [\![\psi'[\Gamma;\Delta]]\!]^{M'}_{\rho'}$. Following Def. 5, there exists a transition $w'_1 \overset{cr'}{\rightsquigarrow} w'_2$, with (at least) an $(w_2,\phi_2,w'_2) \in R$ and $\phi_2 \circ cr = cr' \circ \phi_1$. Following the induction hypothesis, $(w'_2,\phi_2\circ\sigma_{w_2}) \in [\![\psi'[\Gamma;\Delta]]\!]^{M'}_{\rho'}$. All remains to prove is $\phi_1\circ\sigma_{w_1} \overset{cr'}{\rightsquigarrow} \phi_2\circ\sigma_{w_2}$, which follows from $\sigma_{w_1} \overset{cr}{\rightsquigarrow} \sigma_{w_2}$ and $\phi_2 \circ cr = cr' \circ \phi_1$.

$[\psi \equiv \Diamond\psi' :_R \leftarrow]$ From the type system we know that R is an iso-bisimulation, hence $R^{-1} \equiv \{(w',\phi^{-1},w) \in R^{-1} \mid (w,\phi,w') \in R\}$ is defined, and is a simulation from M' to M. What we want to prove is $(w'_1,\phi_1\circ\sigma_{w_1}) \in [\![\Diamond\psi'[\Gamma;\Delta]]\!]^{M'}_{R\circ\rho}$ implies $(w_1,\sigma_{w_1}) \in [\![\Diamond\psi'[\Gamma;\Delta]]\!]^M_\rho$. From the $\Diamond\psi' :_R \rightarrow$ case we know that $(w'_1,\phi_1\circ\sigma_{w_1}) \in [\![\Diamond\psi'[\Gamma;\Delta]]\!]^{M'}_{R\circ\rho}$ implies $(w_1,\phi_1^{-1}\circ\phi_1\circ\sigma_{w_1}) \in [\![\Diamond\psi'[\Gamma;\Delta]]\!]^M_{R^{-1}\circ R\circ\rho}$. It is easy to see that for a bisimulation R, $\phi_1^{-1}\circ\phi_1\circ\sigma_{w_1} = \sigma_{w_1}$ and $R^{-1}\circ R\circ\rho = \rho$, closing the case.

$[\psi \equiv \mu Z\psi' :_R \leftrightarrow]$ Let $F = \lambda Y.[\![\psi'[\Gamma;\Delta]]\!]^M_{\rho[Y/Z]}$, and $F' = \lambda Y'.[\![\psi'[\Gamma;\Delta]]\!]^{M'}_{\rho'[Y'/Z]}$. By definition, $[\![\mu Z.\psi'[\Gamma;\Delta]]\!]^M_\rho = lfp(F)$. We are proving $(w_1,\phi,w'_1) \in R$ implies

$(w_1, \sigma_{w_1}) \in lfp(F)$ iff $(w'_1, \phi_1 \circ \sigma_{w_1}) \in lfp(F')$. We apply the induction hypothesis on ψ': for any Y,Y' with $Y' = R \circ Y$, then $(w, \phi, w') \in R$ implies $(w, \sigma_w) \in F(Y)$ iff $(w', \phi \circ \sigma_w) \in F'(Y')$, from which we have $F'(Y') = R \circ F(Y)$. From Kleene's theorem, $lfp(F) = sup(F^n(\emptyset) \mid n \in \mathbb{N})$, computable as the first Y_n such that $Y_n = Y_{n-1}$, with $Y_0 = \emptyset$, and $Y_i = F(Y_{i-1})$. Clearly $\emptyset = R \circ \emptyset$, implying $Y'_1 = F'(\emptyset) = R \circ F(\emptyset) = R \circ Y_1$. Iterating, $F'(lfp(F')) = R \circ F(lfp(F))$, closing the case. □

We now prove that Proposition 2 holds, stating the soundness of our over-approximated model checking procedure.

Proposition 2 (Soundness of over-check). *Let $\overline{R}$ be a simulation from M to $\overline{M}$ (i.e., $M \sqsubseteq_{\overline{R}} \overline{M}$), $\psi[\Gamma; \Delta]$ a formula, and ρ an assignment. Then (i) $p \models^{\overline{R}} [\![\psi[\Gamma; \Delta]]\!]^M_\rho = T$ implies $p \in [\![\psi[\Gamma; \Delta]]\!]^M_\rho$; and (ii) $p \models^{\overline{R}} [\![\psi[\Gamma; \Delta]]\!]^M_\rho = F$ implies $p \notin [\![\psi[\Gamma; \Delta]]\!]^M_\rho$.*

Proof. We first focus on case (i). From Definition 9, $p \models^{\overline{R}} [\![\psi[\Gamma; \Delta]]\!]^M_\rho = T$ iff there exists a $\overline{p} \in \overline{R}(p)$, such that $\{\![\psi[\Gamma; \Delta]]\!\}^{M \sqsubseteq_{\overline{R}} \overline{M}}_\rho(\overline{p}) = T$. This in turn implies that $\psi :_{\overline{R}} \leftarrow$ and $\overline{p} \in [\![\psi[\Gamma; \Delta]]\!]^{\overline{M}}_{\overline{R} \circ \rho}$. Finally, from Definition 6 and Proposition 1, we can conclude that all the pairs in $\overline{R}^{-1}[\overline{p}]$ (including p) belong to $[\![\psi[\Gamma; \Delta]]\!]^M_\rho$.

We now consider case (ii). From Definition 9, $p \models^{\overline{R}} [\![\psi[\Gamma; \Delta]]\!]^M_\rho = F$ iff there exists a $\overline{p} \in \overline{R}(p)$, such that $\{\![\psi[\Gamma; \Delta]]\!\}^{M \sqsubseteq_{\overline{R}} \overline{M}}_\rho(\overline{p}) = F$, and does not exist any pair $\overline{p}' \in \overline{R}(p)$, such that $\{\![\psi[\Gamma; \Delta]]\!\}^{M \sqsubseteq_{\overline{R}} \overline{M}}_\rho(\overline{p}) = T$. This in turn implies that $\psi :_{\overline{R}} \rightarrow$ and $\overline{p} \notin [\![\psi[\Gamma; \Delta]]\!]^{\overline{M}}_{\overline{R} \circ \rho}$. From Definition 6 and Proposition 1, we know that $\overline{R}([\![\psi[\Gamma; \Delta]]\!]^M_\rho) \subseteq [\![\psi[\Gamma; \Delta]]\!]^{\overline{M}}_{\overline{R} \circ \rho}$. Finally, since $\overline{p} \notin [\![\psi[\Gamma; \Delta]]\!]^{\overline{M}}_{\overline{R} \circ \rho}$, then no assignment pair in $\overline{R}^{-1}[\overline{p}]$ (including p) belongs to $[\![\psi[\Gamma; \Delta]]\!]^M_\rho$. □

Finally, we now prove that Proposition 3 holds, stating the soundness of our under-approximated model checking procedure.

Proposition 3 (Soundness of under-check). *Let $\underline{R}$ be a simulation from $\underline{M}$ to M (i.e., $\underline{M} \sqsubseteq_{\underline{R}} M$), $\psi[\Gamma; \Delta]$ a formula, and ρ an assignment. Then (i) $p \models_{\underline{R}} [\![\psi[\Gamma; \Delta]]\!]^M_\rho = T$ implies $p \in [\![\psi[\Gamma; \Delta]]\!]^M_\rho$; and (ii) $p \models_{\underline{R}} [\![\psi[\Gamma; \Delta]]\!]^M_\rho = F$ implies $p \notin [\![\psi[\Gamma; \Delta]]\!]^M_\rho$.*

Proof. We first focus on case (i). From Definition 11, $p \models_{\underline{R}} [\![\psi[\Gamma; \Delta]]\!]^M_\rho = T$ iff there exists a $\underline{p} \in \underline{R}^{-1}[p]$, such that $\{\![\psi[\Gamma; \Delta]]\!\}^{\underline{M} \sqsubseteq_{\underline{R}} M}_\rho(\underline{p}) = T$. This in turn implies that $\psi :_{\underline{R}} \rightarrow$ and $\underline{p} \in [\![\psi[\Gamma; \Delta]]\!]^{\underline{M}}_{\underline{R}^{-1}[\cdot] \circ \rho}$. Finally, from Definition 6 and Proposition 1, we can conclude that all the assignment pairs in $\underline{R}(\underline{p})$ (including p) belong to $[\![\psi[\Gamma; \Delta]]\!]^M_\rho$.

We now consider case (ii). From Definition 11, $p \models_{\underline{R}} [\![\psi[\Gamma; \Delta]]\!]^M_\rho = F$ iff there exists a $\underline{p} \in \underline{R}^{-1}[p]$, such that $\{\![\psi[\Gamma; \Delta]]\!\}^{\underline{M} \sqsubseteq_{\underline{R}} M}_\rho(\underline{p}) = F$, and does not exist any

pair $\underline{p}' \in \underline{R}^{-1}[p]$, such that $\{[\psi[\Gamma;\Delta]]\}_{\rho}^{M \sqsubseteq_{\underline{R}} M}(\underline{p}') = T$. This in turn implies that $\psi :_{\underline{R}} \leftarrow$ and $\underline{p} \notin [\![\psi[\Gamma;\Delta]]\!]^{M}_{\underline{R}^{-1}[\cdot]\circ\rho}$. From Definition 6 and Proposition 1, we know that $\underline{R}^{-1}[[\![\psi[\Gamma;\Delta]]\!]^{M}_{\rho}] \subseteq [\![\psi[\Gamma;\Delta]]\!]^{M}_{\underline{R}^{-1}[\cdot]\circ\rho}$. Finally, since $\underline{p} \notin [\![\psi[\Gamma;\Delta]]\!]^{M}_{\underline{R}^{-1}[\cdot]\circ\rho}$, then $\underline{R}(\underline{p}) \cap [\![\psi[\Gamma;\Delta]]\!]^{M}_{\rho} = \emptyset$, hence, no pair in M similar to $\underline{p}$ (including p) belongs to $[\![\psi[\Gamma;\Delta]]\!]^{M}_{\rho}$. □