A Relational Dual Tableau Decision Procedure for Multimodal and Description Logics

Domenico Cantone[1], Joanna Golińska-Pilarek[2], and Marianna Nicolosi-Asmundo[1]

[1] University of Catania, Dept. of Mathematics and Computer Science, Italy
{cantone,nicolosi}@dmi.unict.it
[2] University of Warsaw, Institute of Philosophy, Poland
j.golinska@uw.edu.pl

Abstract. We present a dual tableau based decision procedure for a class of fragments of the classical relational logic of binary relations. The logics considered share a common language involving a restricted composition operator and infinitely many relational constants which may have the properties of reflexivity, transitivity, and heredity. The construction of the dual tableau is carried out by applying in a deterministic way axioms and inference rules of the system without resorting to external tools. An important feature of the dual tableau procedure is a rule to handle the relational composition operator, that permits to decompose in a single step compositional formulae and negative compositional formulae with the same left object variable.

Our relational dual tableau can be used as a decision procedure for validity verification in the multimodal logic K, the description logic $\mathcal{ALC}$, and several non-classical logics for reasoning in various AI systems.

1 Introduction

Hybrid intelligent systems are crucial and essential in solving the real-world complex problems. In the last decades, the increasing need for hybrid methods which combine and integrate different techniques of representation and computation in AI has contributed to new approaches in hybrid artificial intelligence systems (cf. [1], [5]). As stated in [9], one of the main goals of hybrid system research is to increase the efficiency, expressive power, and reasoning power of intelligent systems. Hence, the development of a powerful language of representation and of an effective system of automated deduction is one of the most important and significant challenges in hybrid systems area.

The relational representation of states is very natural and has many benefits in hybrid intelligence systems that are complex and involve many objects which interact with each other. A homogeneous relational framework, based on the logic of binary relations RL introduced in [12], has proved to be a useful logical tool for relational representation and reasoning in various AI systems. Indeed, the general methodology of relational logics provides a means for a uniform and modular representation of three basic components of a formal system:

M. Polycarpou et al. (Eds.): HAIS 2014, LNAI 8480, pp. 466–477, 2014.

its syntax, semantics, and deduction system, called *relational dual tableau*. The relational formalization of many non-classical logics that have found applications in AI has been studied systematically in the last decades [11]. In particular, relational formalisms have been constructed for various applied theories of computational logic such as temporal and spatial reasoning, fuzzy-set- and rough-set-base based reasoning, order-of-magnitude and qualitative reasoning, dynamic reasoning about programs, etc.

One of the greatest advantages of the relational methodology is that, given a theory with a relational representation, we can build its deduction system in the form of a relational dual tableau in a systematic modular way. Though the relational logic RL is undecidable, it contains several decidable fragments of great expressive power. In many cases, however, dual tableau proof systems of such decidable fragments are not their decision procedures. This is mainly due to the way decomposition and specific rules are defined and to the strategy of proof construction.

Over the years, great efforts have been spent to define dual tableau proof systems for various logics known to be decidable, but little care has been taken to construct dual tableau-based decision procedures for them. On the other hand, it is clear that when a proof system is constructed and implemented, the existence of a decision procedure for a decidable logic is important. In [6], for instance, an optimized relational dual tableau for RL, based on Binary Decision Graphs, has been implemented. However, such an implementation turns out not to be effective with respect to decidable fragments.

To the best of our knowledge, relational dual tableau-based decision procedures can be found in [11] for fragments of RL corresponding to the class of first-order formulae in prenex normal form with universal quantifiers only, in [10,8] for the relational logic corresponding to the modal logic K, in [3,4] for fragments of RL characterized by some restrictions in terms of type $(R\,;S)$, and in [7] for a class of relational logics admitting just one relational constant with the properties of reflexivity, transitivity, and heredity.

In this paper we present an extension of the results achieved [7]. We construct a dual tableau decision procedure for relational fragments allowing an unbounded number of relational constants which may enjoy the properties of reflexivity, transitivity, and heredity. We show that our procedure always terminates and that it is sound and complete. The class of relational logics presented here is more expressive than the one introduced in [7]. In fact, while in [7] only some monomodal logics can be treated, in this paper it is shown that the multimodal logic with reflexive and transitive frames and the description logic $\mathcal{ALC}$ with transitive roles are expressible within relational fragments that can be decided by our procedure. Being able to express and reason on many modalities is important for AI deduction systems where propositional dynamic logic with actions and logics for qualitative reasoning are used.

The paper is organized as follows. In Sect. 2 we define the syntax and semantics of a class $\mathcal{RDL}^m$ of fragments of the logic RL. In Sect. 3 we present our dual tableau based decision procedures for the logics belonging to $\mathcal{RDL}^m$ and state its termination, soundness, and completeness. Then, in Sect. 4 we illustrate some applications of our decision procedures to well-known logics that can be expressed by the relational fragments belonging to $\mathcal{RDL}^m$. Finally, in Sect. 5, we draw our conclusions and give some hints for future work.

2 A Class of Decidable Fragments of Relational Logic

We consider a class of fragments of the relational logic RL, called $\mathcal{RDL}^m$, and exhibit a dual-tableau based decision procedure that can be applied to each logic belonging to it.

Fragments of $\mathcal{RDL}^m$ share a common language which admits only restricted forms of terms of type $R\,;S$. Relational constants are admitted only on the left hand side of terms of type $R\,;S$, and they can enjoy properties like reflexivity, transitivity, and heredity. In addition, the left part of terms of type $R;S$, namely R, is allowed to be a relational constant only.

2.1 Syntax

Let $\mathbb{RV}$ be a countably infinite set of *relational variables* $\mathsf{p}_1, \mathsf{p}_2, \ldots$, let $\mathbb{RC}$ be a countably infinite set of *relational constants* $R_1, R_2, \ldots$. The *relational operators* admitted in terms of $\mathbb{RT}_{\mathcal{RDL}^m}$ are complement '$-$', intersection '$\cap$', and composition '$;$'. Then, the set $\mathbb{RT}_{\mathcal{RDL}^m}$ of *relational terms* of $\mathcal{RDL}^m$ is the smallest set of terms (with respect to inclusion) containing all the relational variables in $\mathbb{RV}$ and satisfying the following closure condition: if $S, T \in \mathbb{RT}_{\mathcal{RDL}^m}$, then $-S$, $S \cap T$, $(R_i\,;T) \in \mathbb{RT}_{\mathcal{RDL}^m}$, for every $R_i \in \mathbb{RC}$.

Let $\mathbb{OV}$ be a countably infinite set of *object (individual) variables* $z_0, z_1, \ldots$ $\mathcal{RDL}^m$-formulae have the form $z_n T z_0$, where $T \in \mathbb{RT}_{\mathcal{RDL}^m}$, $z_n, z_0 \in \mathbb{OV}$, $n \geq 1$.

$\mathcal{RDL}^m$-formulae of type $z_n \mathsf{p}_j z_0$ are called *atomic $\mathcal{RDL}^m$-formulae.* A *literal formula* is either an atomic formula or its complement (namely a formula of type $z_n(-\mathsf{p}_j)z_0$). If $z_n T z_0$ is a literal formula, then the term T is said to be a *literal term.* A *Boolean formula* (resp., *compositional*, *negative-compositional formula*) is a relational formula either of the form $z_n(--T)z_0$, $z_n(T \cap T')z_0$, or $z_n(-(T \cap T'))z_0$ (resp., $z_n(R_i;T)z_0$, $z_n(-(R_i;T))z_0$). If $z_n T z_0$ is a Boolean formula (resp., compositional, negative-compositional formula), then T is a *Boolean term* (resp., *compositional*, *negative-compositional term*). A formula (resp., a term) which is not compositional is called *non-compositional.*

For every i, by $(R_i^s\,;T)$ we denote the term obtained from T and from the relational constant R_i, by applying the composition operator s times, where $s \geq 0$. Formally, $(R_i^0\,;T) =_{\mathrm{Def}} T$ and $(R_i^{s+1}\,;T) =_{\mathrm{Def}} (R_i\,;(R_i^s\,;T))$.

2.2 Semantics

$\mathcal{RDL}^m$-formulae are interpreted in $\mathcal{RDL}^m$*-models.* An $\mathcal{RDL}^m$-model is a structure $\mathcal{M} = (U, h, m)$, where U is a nonempty universe, h is a function mapping each relational constant of $\mathbb{RC}$ into a binary relation on U, namely $h : \mathbb{RC} \rightarrow \wp(U \times U)$, and m is a *meaning* function satisfying the following conditions:

- $m(\mathsf{p}_j) = X \times U$, where $X \subseteq U$, for every $\mathsf{p}_j \in \mathbb{RV}$; $m(R_i) = h(R_i)$, for every $R_i \in \mathbb{RC}$;
- $m(-T) = (U \times U) \setminus m(T)$; $m(T \cap T') = m(T) \cap m(T')$;
- $m(R_i\,;T) = m(R_i)\,;m(T)$
$$= \{(a,b) \in U \times U : (a,c) \in m(R_i) \text{ and } (c,b) \in m(T), \text{ for some } c \in U\}.$$

Given an $\mathcal{RDL}^m$-model $\mathcal{M} = (U, h, m)$, a *valuation* in $\mathcal{M}$ is any function $v : \mathbb{OV} \to U$.

Next we introduce the conditions (ref_i), (tran_i), and (her_i) on $h(R_i)$, for $R_i \in \mathbb{RC}$, with the following meaning:

(ref_i) $h(R_i)$ is reflexive, namely, for all $a \in U$, $(a,a) \in h(R_i)$;
(tran_i) $h(R_i)$ is transitive, namely, for all $a,b,c \in U$, if $(a,b) \in h(R_i)$ and $(b,c) \in h(R_i)$, then $(a,c) \in h(R_i)$;
(her_i) heredity condition: For all $a,b,c \in U$, $\mathsf{p}_j \in \mathbb{RV}$, if $(a,b) \in h(R_i)$ and $(a,c) \in m(\mathsf{p}_j)$, then $(b,c) \in m(\mathsf{p}_j)$.

By $\mathcal{RDL}^m$ we denote the class of logics $\mathsf{RL_L}$ such that $\mathsf{L} \subseteq \{\mathsf{r}_i, \mathsf{t}_i, \mathsf{h}_i : i = 1, \ldots\}$ and whose set of relational terms is a subset of $\mathbb{RT}_{\mathcal{RDL}^m}$. The models of a logic $\mathsf{RL_L} \in \mathcal{RDL}^m$, referred to as $\mathsf{RL_L}$-*models*, are those $\mathcal{RDL}^m$-models that satisfy (ref_i), for $\mathsf{r}_i \in \mathsf{L}$, (tran_i), for $\mathsf{t}_i \in \mathsf{L}$, and (her_i), for $\mathsf{h}_i \in \mathsf{L}$.

An $\mathcal{RDL}^m$-formula (resp., $\mathsf{RL_L}$-formula) $z_n T z_0$ is said to be *satisfied* in an $\mathcal{RDL}^m$-model (resp., $\mathsf{RL_L}$-model) $\mathcal{M} = (U, h, m)$ by a valuation v if and only if $(v(z_n), v(z_0)) \in m(T)$. An $\mathcal{RDL}^m$-formula (resp., $\mathsf{RL_L}$-formula) is said to be *true* in an $\mathcal{RDL}^m$-model $\mathcal{M}$ (resp., $\mathsf{RL_L}$-model) if it is satisfied in $\mathcal{M}$ by all valuations. An $\mathcal{RDL}^m$-formula (resp., $\mathsf{RL_L}$-formula) is said to be $\mathcal{RDL}^m$-valid (resp., $\mathsf{RL_L}$-valid) if it is true in all $\mathcal{RDL}^m$-models (resp., $\mathsf{RL_L}$-models). A finite set of $\mathcal{RDL}^m$-formulae (resp., $\mathsf{RL_L}$-formulae) $\{\varphi_1, \ldots, \varphi_n\}$ is $\mathcal{RDL}^m$-valid (resp., $\mathsf{RL_L}$-valid) if and only if for every $\mathcal{RDL}^m$-model (resp., $\mathsf{RL_L}$-model) $\mathcal{M}$ and valuation v in $\mathcal{M}$, there exists $i \in \{1, \ldots, n\}$ such that $\mathcal{M}, v \models \varphi_i$.

3 A Dual Tableau Decision Procedure for the Relational Logics in $\mathcal{RDL}^m$

We introduce a dual tableau proof procedure, called $\mathsf{RDT}_{\mathsf{RL_L}}$, to decide $\mathsf{RL_L}$-validity of relational formulae belonging to any logic $\mathsf{RL_L}$ in $\mathcal{RDL}^m$. The proof system is constructed along the lines of the dual tableau methodology described in [11]. It consists of decomposition rules to analyze the structure of the formulae to be proved, of specific rules to deal with properties that can be enjoyed by the relational constants occurring in the formulae to be proved (namely, reflexivity, transitivity, and heredity), and of axiomatic sets which specify the closure conditions.

An $\mathsf{RDT}_{\mathsf{RL_L}}$-*axiomatic set* is any finite set of $\mathsf{RL_L}$-formulae including a subset of the form $\{z_n T z_0, z_n(-T)z_0\}$, for some $n \geq 1$ and $T \in \mathbb{RT}_{\mathcal{RDL}^m}$. Sets which are not axiomatic are referred to as *non-axiomatic*. It is evident that every $\mathsf{RDT}_{\mathsf{RL_L}}$-axiomatic set is an $\mathsf{RL_L}$-valid set.

Table 1. Boolean decomposition rules for $\mathsf{RDT}_{\mathsf{RL_L}}$

$$(--)\ \frac{X \cup \{z_n(--T)z_0\}}{X \cup \{z_n T z_0\}}$$

$$(\cap)\ \frac{X \cup \{z_n(T \cap T')z_0\}}{X \cup \{z_n T z_0\} \mid X \cup \{z_n T' z_0\}} \qquad (-\cap)\ \frac{X \cup \{z_n(-(T \cap T'))z_0\}}{X \cup \{z_n(-R)z_0, z_n(-S)z_0\}},$$

where $n \geq 1$ and $T, T' \in \mathbb{RT}_{\mathcal{RDL}^m}$.

Each relational system considered in this paper admits the following rule schemas:

$$(A)\ \frac{X \cup \Gamma}{X \cup \Delta} \qquad (B)\ \frac{X \cup \Gamma}{X \cup \Delta_1 \mid X \cup \Delta_2},$$

where $\Gamma, \Delta, \Delta_1, \Delta_2$ are finite nonempty sets of formulae, and X is a finite, possibly empty, set of formulae. X and Δ (resp., X and Δ_i, for $i = 1, 2$) are assumed to be disjoint and $X \cup \Gamma \neq X \cup \Delta$ (resp., $X \cup \Gamma \neq X \cup \Delta_i$, for some $i \in \{1, 2\}$).

Table 2. Decomposition rule for compositional formulae in $\mathsf{RDT}_{\mathsf{RL_L}}$

$$(\mathsf{glob}\ ;)\ \frac{X \cup Y \cup \bigcup_{i \in I_\Phi}\{z_n(-(R_i\,;S_{q_i}))z_0, z_n(R_i\,;T_{j_i})z_0 : q_i \in Q_i, j_i \in J_i\}}{X \cup \bigcup_{i \in I_\Phi}\{z_{n_{q_i}}(-S_{q_i})z_0, z_{n_{q_i}} T_{j_i} z_0 : q_i \in Q_i, j_i \in J_i\}}$$

where:

- $n \geq 1$ and Y is the set of literals with left variable z_n occurring in the current node;
- I_Φ is the set of indices of constants of $\mathbb{RC}$ occurring in the current node Φ;
- for all $T \in \mathbb{RT}_{\mathcal{RDL}^m}$, $z_n T z_0 \notin X$ (the only formulae in the premise that are neither compositional nor negative-compositional and have z_n as left variable are in Y);
- $Q = \bigcup_{i \in I_\Phi} Q_i$ and $J = \bigcup_{i \in I_\Phi} J_i$ are sets of indices such that $Q_i \neq \emptyset$, for some $i \in I_\Phi$ (by this condition, if in the current node there is a formula $z_n(R_i\,;T_{j_i})z_0$, for some $i \in I_\Phi$, and no formula of type $z_n(-(R_i\,;S_{q_i}))z_0$ occurs in the current node, then $z_n(R_i\,;T_{j_i})z_0$ cannot be decomposed anymore and therefore it is not repeated in the successive nodes of the dual tableau);
- $S_{q_i}, T_{j_i} \in \mathbb{RT}_{\mathcal{RDL}^m}$, for all $q_i \in Q_i$, $j_i \in J_i$, with $i \in I_\Phi$;
- the set $N = \{n_{q_i} : q_i \in Q_i, i \in I_\Phi\}$ satisfies the following conditions:
  - the elements of N are consecutive natural numbers,
  - $\min(N) = k+1$, where k is the largest number such that z_k occurs in the premise,
  - for all $n_{q_i}, n_{q'_{i'}} \in N$, we have $n_{q_i} < n_{q'_{i'}}$ if and only if $\langle R_i, S_{q_i}\rangle < \langle R_{i'}, S_{q'_{i'}}\rangle$;
- the pivot of (glob ;) is the formula $z_n(-(R_i\,;S_{q_i}))z_0$ with the minimal pair $\langle R_i\,;S_{q_i}\rangle$.

Rules of type (B) are 'conjunctive' branching rules, and in fact, in the (B) rule schema the symbol '|' is interpreted as a conjunction. On the other hand, rules of type (A) are 'disjunctive'.

A variable z_n, with $n > 1$, which appears in a conclusion of a rule while it does not appear in its premise, is called a *new variable* (introduced by the rule application). In view of the definition of the dual tableau decision procedure, it is convenient to associate with every rule a specific element of the set of formulae Γ, called the pivot of the rule, to determine the order of application of the rules. Moreover, for a set of $\mathsf{RL_L}$-formulae S, we denote by I_S the set of indices of the relational constants of $\mathbb{RC}$ occurring in S.

In order to construct a deterministic decision procedure, we define an order on the set of the relational formulae. We begin with relational terms. For $\mathsf{p}_{j_1}, \mathsf{p}_{j_2} \in \mathbb{RV}$, $R_{i_1}, R_{i_2} \in \mathbb{RC}$, and $S_0, \dots, S_6 \in \mathbb{RT}$, we put:

$$\mathsf{p}_{j_1} < (-\mathsf{p}_{j_2}) < (--S_0) < (-(S_1 \cap S_2)) < (S_3 \cap S_4) < (R_{i_1} \,;S_5) < -(R_{i_2} \,;S_6).$$

Next, we introduce an order also for the constants in $\mathbb{RC}$ by putting $R_{i_1} < R_{i_2}$ iff $i_1 < i_2$. For terms left unordered by the above, we define an order in the following way: $\mathsf{p}_{j_1} < \mathsf{p}_{j_2}$ iff $j_1 < j_2$; $(-T) < (-T')$ iff $T < T'$; $T \cap T' < S \cap S'$ iff $\langle T, T' \rangle < \langle S, S' \rangle$; $(R_{i_1};T) < (R_{i_2};T')$ iff $\langle R_{i_1}, T \rangle < \langle R_{i_2}, T' \rangle$, where $\langle a, b \rangle < \langle c, d \rangle$ if and only if either $a < c$, or both $a = c$ and $b < d$.

Finally, for formulae $z_n S z_0$ and $z_{n'} S' z_0$, we put $z_n S z_0 < z_{n'} S' z_0$ if and only if $\langle n, S \rangle < \langle n', S' \rangle$. It can easily be proved that $<$ linearly orders the set of all relational formulae.

Given a finite nonempty set of relational formulae X, *the minimal element of* X is the minimal element with respect to the order $<$ defined above. An immediate consequence of the above definition is that if $z_n S z_0$ is the minimal element of X then, for any formula $z_{n'} T' z_0$ in X, it holds that $n \leq n'$.

We also introduce a notion that is used in some restrictions on rule applications to avoid infinite loops. A finite set $\{z_n S_j z_0 : j \in J\}$ is said to be a *subcopy* of a set Y whenever there exists an $n' < n$ such that $\{z_{n'} S_j z_0 : j \in J\} \subseteq Y$.

3.1 The Decision Procedure $\mathsf{RDT_{RL_L}}$

The decomposition rules of $\mathsf{RDT_{RL_L}}$ are presented in Tables 1 and 2. Rules $(-)$, $(\cap)$, and $(-\cap)$, illustrated in Table 1, deal with Boolean operators, while rule $(\mathsf{glob}\,;)$, depicted in Table 2, deals with the composition operator. Rule $(\mathsf{glob}\,;)$ is a proper extension of rule $(R;)$, introduced in [7], because while rule $(R;)$ is applicable only to compositional and negative-compositional formulae with the same left object variable and with the same relational constant R, rule $(\mathsf{glob}\,;)$ decomposes in a single step all the compositional formulae and negative-compositional formulae with the same left object variable occurring on the current node.

Rules (ref_i), (tran_i), and (her_i) defined in Table 3 reflect the reflexivity, transitivity, and heredity of R_i, respectively, for $i \in I_\Phi$, where Φ is the set of formulae labelling the current node.[1] They are generalizations of rules (ref), (tran), and (her), introduced in [7] to deal with the reflexivity, transitivity, and heredity of a single relational constant R. Clearly, rule (ref_i) (resp., (tran_i), (her_i)) can be applied to a pivot formula only if $h(R_i)$ is reflexive (resp., transitive, enjoys the heredity property).

Definition 1 (Proof tree). *An* $\mathsf{RDT_{RL_L}}$*-proof tree of an* $\mathsf{RL_L}$*-formula* ψ *is a finitely branching tree satisfying the following conditions:*

[1] From now on, we identify nodes with the (disjunctive) sets labelling them.

Table 3. Reflexivity, transitivity, and heredity rules for $\mathsf{RDT}_{\mathsf{RL_L}}$

Reflexivity rule:

$$(\mathsf{ref}_i)\ \frac{X \cup \{z_n(R_i^s\,;T)z_0\}}{X \cup \{z_n(R_i^s\,;T)z_0\} \cup \{z_n(R_i^j\,;T)z_0 : j \in \{0,\dots,s-1\}\}},$$

where $n, s \geq 1$ and $T \in \mathbb{RT}_{\mathcal{RDL}^m}$ is either a non-compositional term or a compositional term $(R_j\,;T')$, with $j \neq i$, and $z_n(R_i^t\,;T) \notin X$, for all $t > s$.

Transitivity rule:

$$(\mathsf{tran}_i)\ \frac{X \cup \{z_n(R_i\,;T)z_0\}}{X \cup \{z_n(R_i\,;T)z_0\} \cup \{z_n(R_i^2\,;T)z_0\}},$$

where $n \geq 1$ and $T \in \mathbb{RT}_{\mathcal{RDL}^m}$ is either a non-compositional term or a compositional term $(R_j\,;T')$, with $j \neq i$.

Heredity rule:

$$(\mathsf{her}_i)\frac{X \cup \{z_n(-(R_i\,;T))z_0\} \cup \{z_n(-\mathsf{p}_j)z_0 : j \in J_\Phi\}}{X \cup \{z_n(-(R_i^s\,;T))z_0\} \cup \{z_n(-\mathsf{p}_j)z_0 : j \in J_\Phi\} \cup \{z_n(R_i\,;(-\mathsf{p}_j))z_0\ : j \in J_\Phi\}}$$

where: $n \geq 1$, $T \in \mathbb{RT}_{\mathcal{RDL}^m}$, and $z_n(-\mathsf{p}_j)z_0 \notin X$, for any $\mathsf{p}_j \in \mathbb{RV}$.
The pivot of the rule (her_i) is $z_n(-(R_i\,;T))z_0$.

1. *the root of the tree is the set* $\{\psi\}$*;*
2. *each node but the root is obtained by an application of an* $\mathsf{RDT}_{\mathsf{RL_L}}$*-rule to its direct predecessor node;*
3. *when more that one rule is applicable to a node* Φ*, the first possible schema from the following list is chosen:* $(-)$, $(-\cap)$, $(\cap)$, (ref_i)*, for every* $i \in I_\Phi$ *chosen in increasing numerical order,* (her_i)*, for every* $i \in I_\Phi$ *chosen in increasing numerical order,* (tran_i)*, for every* $i \in I_\Phi$ *chosen in increasing numerical order, and finally the rule* $(\mathsf{glob}\,;)$*;*
4. *the rule* $(\mathsf{glob}\,;)$ *applies to a node* Φ *which has the form of the premise of the rule* $(\mathsf{glob}\,;)$*, provided that* $\Phi \setminus X$ *is not a subcopy of any of its predecessor nodes;*
5. *given* φ*, the rule* (ref_i) *with pivot* φ *applies in a given branch at most once;*
6. *a node does not have successors if and only if it is* $\mathsf{RDT}_{\mathsf{RL_L}}$*-axiomatic or none of the* $\mathsf{RDT}_{\mathsf{RL_L}}$*-rules applies to it.*

Definition 2. *A branch of an* $\mathsf{RDT}_{\mathsf{RL_L}}$*-proof tree is said to be* closed *whenever it ends with an axiomatic set of formulae. An* $\mathsf{RDT}_{\mathsf{RL_L}}$*-proof tree is closed if and only if all of its branches are closed. A formula is* $\mathsf{RDT}_{\mathsf{RL_L}}$-provable *whenever it has a closed* $\mathsf{RDT}_{\mathsf{RL_L}}$*-proof tree, which is then referred to as its* $\mathsf{RDT}_{\mathsf{RL_L}}$-proof.

It can easily be checked that, given a finite set of $\mathsf{RL_L}$-formulae Φ, for each rule schema and each $\varphi \in \Phi$, there is at most one instance of that schema whose premise equals Φ and pivot equals φ. As a consequence of what observed before and of the conditions in the definition of proof tree, one can show that, for every $\mathsf{RL_L}$-formula φ, there is exactly one $\mathsf{RDT}_{\mathsf{RL_L}}$-proof tree of φ.

It can be proved that the proof system $\mathsf{RDT}_{\mathsf{RL_L}}$ always terminates by showing that each proof tree that we can construct is finite.

Theorem 1 (Termination). *For every formula* $\varphi \in \mathsf{RL_L}$ *there is exactly one finite* $\mathsf{RDT_{RL_L}}$*-proof tree of* φ*.*

In addition, by using the proof techniques from [11] and [7], it can be shown that the proof system $\mathsf{RDT_{RL_L}}$ is sound and complete.

Theorem 2 (Soundness and Completeness of $\mathsf{RDT_{RL_L}}$). *Let* φ *be a relational formula. Then,* φ *is* $\mathsf{RL_L}$*-valid if and only if* φ *is* $\mathsf{RDT_{RL_L}}$*-provable.*

Theorems 1 and 2 readily imply:

Theorem 3. *An* $\mathsf{RDT_{RL_L}}$*-dual tableau is a sound and complete deterministic decision procedure for the logic* $\mathsf{RL_L}$*.*

4 Applications to Multimodal and Description Logics

Relational logics and their dual tableaux presented in Sects. 2 and 3, respectively, can be used as decision procedures for verification of validity in some multimodal and description logics.

To begin with, we discuss multimodal logics and their relational decision procedures in dual tableaux style. The class of multimodal logics considered in this section will be denoted by ML. The common vocabulary of logics in ML consists of the following pairwise disjoint sets of symbols: $\mathbb{V} = \{p_1, p_2, p_3, \ldots\}$, a countably infinite set of propositional variables, $\{\neg, \wedge\}$, the set of the classical operations of negation and conjunction, respectively, $\{[R_i], \langle R_i \rangle : i \in I\}$, a finite set of modal propositional operations of necessity and possibility, respectively. The set of modal formulae is then the smallest set including the set of propositional variables and closed with respect to all the propositional operations.

Let $\mathsf{L} \subseteq \{\mathsf{r}_i, \mathsf{t}_i, \mathsf{h}_i : i \in I\}$. Logics in the class ML are determined by $\mathsf{ML_L}$-models which are structures of the form $\mathcal{M} = (U, h, m)$ such that U is a nonempty set (of states), m is the meaning function such that $m(p) \subseteq U$, for every propositional variable $p \in \mathbb{V}$, and h is a function such that for every $i \in I$, $h(R_i)$ is a binary relation on U (referred to as the *i-th accessibility relation*) that satisfies conditions coded by L as in the definition of $\mathsf{RL_L}$-models given in Sect. 2. Therefore, any logic in ML is determined by the properties of the accessibility relation assumed in $\mathsf{ML_L}$-models. Given $\mathsf{L} \subseteq \{\mathsf{r}_i, \mathsf{t}_i, \mathsf{h}_i : i \in I\}$, a logic in ML will be denoted by $\mathsf{ML_L}$.

Observe that among logics in the class ML are multimodal temporal logics (for instance with transitive time orderings) and multi-agent epistemic logics (for instance with reflexive and transitive knowledge relation). The notions of satisfaction relation, truth, and validity are defined as usual in modal logics.

The *satisfaction relation* is defined as usual in modal logics, that is for the modal operators we set:

$\mathcal{M}, s \models [R_i]\varphi$ iff for every $s' \in U$, if $(s, s') \in h(R_i)$, then $\mathcal{M}, s' \models \varphi$.

$\mathcal{M}, s \models \langle R_i \rangle\varphi$ iff there is $s' \in U$ such that $(s, s') \in h(R_i)$ and $\mathcal{M}, s' \models \varphi$.

A modal formula is said to be *true* in an $\mathsf{ML_L}$-model $\mathcal{M}$ whenever it is satisfied in $\mathcal{M}$ by all $s \in U$, and it is $\mathsf{ML_L}$-*valid* whenever it is true in all $\mathsf{ML_L}$-models.

The translation τ of modal formulae into relational terms is defined as follows: $\tau(p_j) = \mathsf{p}_j$, for any propositional variable $p_j \in \mathbb{V}$; $\tau(\neg\varphi) = -\tau(\varphi)$; $\tau(\varphi \wedge \psi) = \tau(\varphi) \cap \tau(\psi)$; $\tau([R_i]\varphi) = -(R_i \,;\, -\tau(\varphi))$; $\tau(\langle R_i \rangle \varphi) = (R_i \,;\, \tau(\varphi))$. Translation τ preserves validity, as stated in the following theorem:

Theorem 4. *Let* $\mathsf{L} \subseteq \{\mathsf{r}_i, \mathsf{t}_i, \mathsf{h}_i : i \in I\}$ *and let* φ *be a modal formula. Then,* φ *is* $\mathsf{ML_L}$*-valid if and only if* $z_1\tau(\varphi)z_0$ *is* $\mathsf{RL_L}$*-valid.*

The proof of the above theorem is standard in the relational formalization of non-classical logics. For details see [11]. In view of this result, and by Theorems 2 and 3, we obtain:

Theorem 5. *Let* $\mathsf{L} \subseteq \{\mathsf{r}_i, \mathsf{t}_i, \mathsf{h}_i : i \in I\}$ *and let* φ *be a modal formula. Then,* φ *is* $\mathsf{ML_L}$*-valid if and only if* $z_1\tau(\varphi)z_0$ *is* $\mathsf{RDT_{RL_L}}$*-provable. Moreover,* $\mathsf{RDT_{RL_L}}$ *is a deterministic decision procedure for a multimodal logic* $\mathsf{ML_L}$*.*

Example 1. Let $I = \{1, 2\}$, and let φ be: $\langle R_1 \rangle \langle R_1 \rangle p_1 \rightarrow \langle R_2 \rangle \langle R_1 \rangle \neg(\neg p_1 \wedge p_2)$. The formula φ is valid in all Kripke frames with two accessibility relations: transitive $h(R_1)$ and reflexive $h(R_2)$. In order to show it, by Theorem 5, it suffices to construct a closed $\mathsf{RDT_{RL_L}}$-proof tree of a formula $z_1\tau(\varphi)z_0$, where $\mathsf{L} = \{\mathsf{t}_1, \mathsf{r}_2\}$ and $\tau(\varphi)$ is the relational translation of φ of the following form: $(-((R_1 \,;\, (R_1 \,;\, \mathsf{p}_1)) \cap -(R_2 \,;\, (R_1 \,;\, -(-\mathsf{p}_1 \cap \mathsf{p}_2)))))$. Figure 1 presents an $\mathsf{RDT_{RL_L}}$-proof of $z_1\tau(\varphi)z_0$, which proves the validity of φ in Kripke frames where the first accessibility relation is transitive and the second one reflexive. Recall that a $\mathsf{RDT_{RL_L}}$-dual tableau consists of rules for Boolean operators, rules (glob ;), (tran_1), and (ref_2).

Next we show how $\mathsf{RDT_{RL_L}}$-systems can decide validity for description logics, a family of logic-based formalisms which allow one to represent knowledge about a domain of interest in terms of *concepts*, which denote sets of elements, of *roles*, which represent relations between elements, and of *individuals*, which denote domain elements [2]. Each language in this family is characterized by its set of constructors, which allow one to form complex terms. Here, we focus on the well-known description logic $\mathcal{ALC}$, considering also the case in which $\mathcal{ALC}$ is provided with transitive roles [13].

Definition 3. *Let* $N_C = \{\mathsf{p}_1, \mathsf{p}_2, \ldots\}$ *be a countably infinite set of* concept names *and let* $N_R = \{R_1, R_2, \ldots\}$ *be a countably infinite set of* role names. *The set of* $\mathcal{ALC}$*-concepts is the smallest set such that: each concept name is an* $\mathcal{ALC}$*-concept, and if* C, D *are* $\mathcal{ALC}$*-concepts and* R_i *is a role name, then* $-C$*,* $C \sqcap D$*,* $\exists R_i.C$*, and* $\forall R_i.D$ *are* $\mathcal{ALC}$*-concepts.*

An interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ consists of a nonempty set $\Delta^{\mathcal{I}}$, called the domain of $\mathcal{I}$, and of a function $\cdot^{\mathcal{I}}$ mapping each concept into a subset of $\Delta^{\mathcal{I}}$ and each role into a subset of $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$. For space reasons, we report here only the interpretations of concepts of type $\exists R_i.C$ and $\forall R_i.C$:

$(\exists R_i.C)^{\mathcal{I}} = \{d \in \Delta^{\mathcal{I}} :$ there is some $e \in \Delta^{\mathcal{I}}$ with $(d, e) \in R_i^{\mathcal{I}}$ and $e \in C^{\mathcal{I}}\}$,
$(\forall R_i.C)^{\mathcal{I}} = \{d \in \Delta^{\mathcal{I}} :$ for all $e \in \Delta^{\mathcal{I}}$, if $(d, e) \in R_i^{\mathcal{I}}$, then $e \in C^{\mathcal{I}}\}$.

$$\underline{z_1(-((R_1 ; (R_1 ; \mathsf{p}_1)) \cap -(R_2 ; (R_1 ; -(-\mathsf{p}_1 \cap \mathsf{p}_2)))))z_0}$$

$(-\cap)$ and $(--)$

$$z_1-(R_1 ; (R_1 ; \mathsf{p}_1))z_0, \underline{z_1(R_2 ; (R_1 ; -(-\mathsf{p}_1 \cap \mathsf{p}_2)))z_0}$$

(ref_2)

$$z_1-(R_1 ; (R_1 ; \mathsf{p}_1))z_0, z_1(R_2 ; (R_1 ; -(-\mathsf{p}_1 \cap \mathsf{p}_2)))z_0, \underline{z_1(R_1 ; -(-\mathsf{p}_1 \cap \mathsf{p}_2))z_0}$$

(tran_1)

$$\underline{z_1-(R_1 ; (R_1 ; \mathsf{p}_1))z_0, z_1(R_2 ; (R_1 ; -(-\mathsf{p}_1 \cap \mathsf{p}_2)))z_0,}$$
$$\underline{z_1(R_1 ; -(-\mathsf{p}_1 \cap \mathsf{p}_2))z_0, z_1(R_1 ; (R_1 ; -(-\mathsf{p}_1 \cap \mathsf{p}_2)))z_0}$$

$(\mathsf{glob}\ ;)$ with a new variable z_2

$$z_2-(R_1 ; \mathsf{p}_1)z_0, z_2-(-\mathsf{p}_1 \cap \mathsf{p}_2)z_0, \underline{z_2(R_1 ; -(-\mathsf{p}_1 \cap \mathsf{p}_2))z_0}$$

(tran_1)

$$\underline{z_2-(R_1 ; \mathsf{p}_1)z_0, z_2(R_1 ; -(-\mathsf{p}_1 \cap \mathsf{p}_2))z_0, z_2(R_1 ; (R_1 ; -(-\mathsf{p}_1 \cap \mathsf{p}_2)))z_0}, \ldots$$

$(\mathsf{glob}\ ;)$ with a new variable z_3

$$z_3-\mathsf{p}_1 z_0, \underline{z_3-(-\mathsf{p}_1 \cap \mathsf{p}_2)z_0}, z_3(R_1 ; -(-\mathsf{p}_1 \cap \mathsf{p}_2))z_0, \ldots$$

$(-\cap)$ and $(--)$

$$z_3-\mathsf{p}_1 z_0, z_3\mathsf{p}_1 z_0, z_3-\mathsf{p}_2 z_0, z_3(R_1 ; -(-\mathsf{p}_1 \cap \mathsf{p}_2))z_0, \ldots$$

closed

Fig. 1. Relational proof of $\langle R_1\rangle\langle R_1\rangle p_1 \rightarrow \langle R_2\rangle\langle R_1\rangle\neg(\neg p_1 \wedge p_2)$.

A concept C is *satisfiable* if there is an interpretation $\mathcal{I}$ such that $C^{\mathcal{I}} \neq \emptyset$, namely if there is an $e \in \Delta^{\mathcal{I}}$ such that $e \in C^{\mathcal{I}}$. A concept C is *true* in an interpretation $\mathcal{I}$ if $C^{\mathcal{I}} = \Delta^{\mathcal{I}}$, namely if for every $e \in \Delta^{\mathcal{I}}$, it holds that $e \in C^{\mathcal{I}}$. A concept C is *valid* if it is true in every interpretation $\mathcal{I}$.

$\mathcal{ALC}_{R^+}$ is an extension of $\mathcal{ALC}$ obtained by allowing the presence of transitive roles inside concepts [13]. The set of role names of $\mathcal{ALC}_{R^+}$, N_R, is the union of two disjoint sets of role names, N_P and N_+. N_P is a set of non-transitive role names and N_+ is a set of transitive role names. Any interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ for $\mathcal{ALC}_{R^+}$ has to satisfy the additional condition that if $(d, e) \in R_i^{\mathcal{I}}$ and $(e, f) \in R_i^{\mathcal{I}}$, then $(d, f) \in R_i^{\mathcal{I}}$, for each role $R_i \in N_+$.

It is well known that $\mathcal{ALC}$ is a notational variant of the multimodal logic $\mathbf{K}_n$. In fact, concepts of type $\exists R_i.C$ are syntactical variations of multimodal formulae of type $\langle R_i\rangle\varphi$ and concepts of type $\forall R_i.C$ are notational counterparts of multimodal formulae of type $[R_i]\varphi$. As a consequence, the translation of $\mathcal{ALC}$ and $\mathcal{ALC}_{R^+}$ in relational terms is analogous to the relational translation of any multimodal logic ML_L introduced above: $\mathcal{ALC}$ can be mapped into a multimodal logic ML_L such that $\mathsf{L} = \emptyset$, and $\mathcal{ALC}_{R^+}$ is a notational variant of a multimodal logic ML_L where $\mathsf{L} = \{\mathsf{t}_i : R_i \in N_+\}$.

Example 2. Consider the $\mathcal{ALC}_{R^+}$-formula:

$$\psi = \neg((\exists.R_1\neg\mathsf{p}_1 \sqcap \exists R_2.\mathsf{p}_2) \sqcap \neg((\exists R_1.\neg(\mathsf{p}_1 \sqcap \neg\mathsf{p}_2)) \sqcap (\exists R_2.\neg(\mathsf{p}_1 \sqcap \neg\mathsf{p}_2)))) ,$$

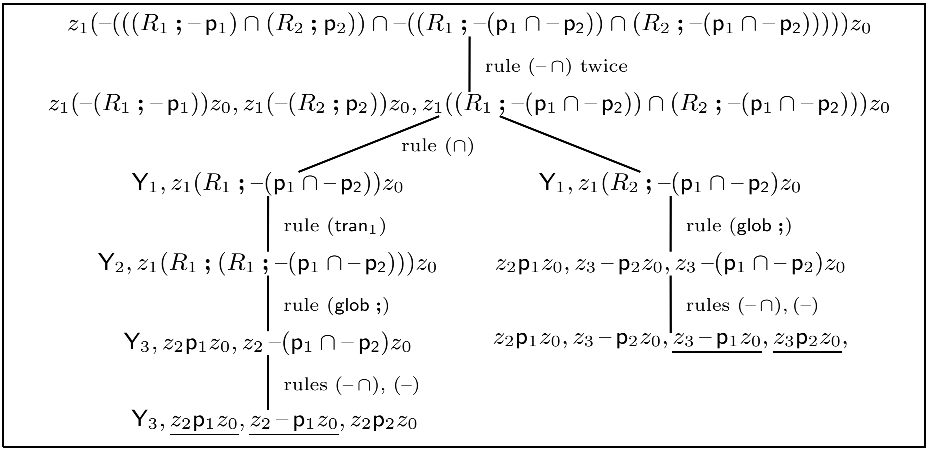

Fig. 2. $\mathsf{RDT}_{\mathsf{RL_L}}$-proof tree of the formula $z_1\tau(\psi)z_0$.

where $R_1 \in N_+$ and $R_2 \in N_P$. The formula ψ is true in all interpretations $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ for $\mathcal{ALC}_{R^+}$. Reasoning as in the previous example, in order to prove the validity of ψ in our relational setting, we just have to construct the closed $\mathsf{RDT}_{\mathsf{RL_L}}$-proof tree of the formula $z_1\tau(\psi)z_0$, with $\mathsf{L} = \{\mathsf{t}_1\}$. This is illustrated in Fig. 2. For space reasons, in the $\mathsf{RDT}_{\mathsf{RL_L}}$-proof tree of Fig. 2 we are using the following shorthands: Y_1 stands for $z_1(-(R_1 ; -\mathsf{p}_1))z_0, z_1(-(R_2 ; \mathsf{p}_2))z_0$, Y_2 stands for $z_1(-(R_1 ; -\mathsf{p}_1))z_0, z_1(-(R_2 ; \mathsf{p}_2))z_0, z_1(R_1 ; -(\mathsf{p}_1 \cap -\mathsf{p}_2))z_0$, and Y_3 stands for $z_3 - \mathsf{p}_2 z_0, z_2(R_1 ; -(\mathsf{p}_1 \cap -\mathsf{p}_2))z_0$.

5 Conclusions and Future Work

We have presented proof systems in the style of relational dual tableaux that can serve as decision procedures for some multimodal and description logics. By way of example, we showed how the systems can be used to verify validity in multimodal logic with one reflexive and one transitive accessibility relation and in a description logic corresponding to a multimodal logic with transitive accessibility relations. The results presented in the paper lead to some further questions about the possibility of extending these systems to decision procedures for other logics not captured by $\mathcal{RDL}^m$. In particular, we intend to work on dual tableau decision procedures for relational logics in which relational constants can also enjoy such properties as symmetry, seriality, Euclidean, partial functionality, functionality, weak density. In this way we would get relational decision procedures for a great variety of standard multimodal logics. Furthermore, we intend to develop decision procedures for those relational logics in which some constraints on interactions between relational constants are assumed. Such relational dual tableaux could be used as decision procedures for multimodal information logics with sufficiency modal operators, for propositional dynamic logic with actions, and for logics for qualitative reasoning. Finally, we plan to implement the decision procedures we have described, and to integrate them in the implementation of the dual tableau from [6], or in another relational theorem prover.

Acknowledgments. This work was supported by the Polish National Science Centre research project DEC-2011/02/A/HS1/00395.

References

1. Abraham, A.: Special issue: Hybrid approaches for approximate reasoning. J. Intell. Fuzzy Syst. 23(2,3), 41–42 (2012)
2. Baader, F., Calvanese, D., McGuinness, D.L., Nardi, D., Patel-Schneider, P.F. (eds.): The Description Logic Handbook: Theory, Implementation, and Applications. Cambridge University Press, New York (2003)
3. Cantone, D., Nicolosi-Asmundo, M., Orlowska, E.: Dual tableau-based decision procedures for some relational logics. In: Proceedings of the 25th Italian Conference on Computational Logic. CEUR Workshop Proceedings, vol. 598, pp. 1–16 (2010)
4. Cantone, D., Nicolosi-Asmundo, M., Orlowska, E.: Dual tableau-based decision procedures for relational logics with restricted composition operator. Journal of Applied Non-Classical Logics 21(2), 177–200 (2011)
5. Corchado, E., Wozniak, M., Abraham, A., de Carvalho, A.C.P.L.F., Snásel, V.: Recent trends in intelligent data analysis. Neurocomputing 126, 1–2 (2014)
6. Formisano, A., Nicolosi-Asmundo, M.: An efficient relational deductive system for propositional non-classical logics. Journal of Applied Non-Classical Logics 16(3,4), 367–408 (2006)
7. Golinska-Pilarek, J., Huuskonen, T., Muñoz-Velasco, E.: Relational dual tableau decision procedures and their applications to modal and intuitionistic logics. Ann. Pure Appl. Logic 165(2), 409–427 (2014)
8. Golinska-Pilarek, J., Muñoz-Velasco, E., Mora, A.: A new deduction system for deciding validity in modal logic k. Logic Journal of the IGPL 19(2), 425–434 (2011)
9. Medsker, L.R.: Hybrid intelligent systems. Kluwer (1995)
10. Mora, A., Muñoz-Velasco, E., Golinska-Pilarek, J.: Implementing a relational theorem prover for modal logic. Int. J. Comput. Math. 88(9), 1869–1884 (2011)
11. Orłowska, E., Golińska-Pilarek, J.: Dual Tableaux: Foundations, Methodology, Case Studies. Trends in Logic, vol. 36. Springer (2011)
12. Orłowska, E.: Relational interpretation of modal logics. In: Andreka, H., Monk, D., Nemeti, I. (eds.) Algebraic Logic. Colloquia Mathematica Societatis Janos Bolyai, vol. 54, pp. 443–471. North Holland (1988)
13. Sattler, U.: A concept language extended with different kinds of transitive roles. In: Görz, G., Hölldobler, S. (eds.) KI 1996. LNCS, vol. 1137, pp. 333–345. Springer, Heidelberg (1996)