Tableaux for Relation-Changing Modal Logics

Carlos Areces[1,2], Raul Fervari[1] and Guillaume Hoffmann[1]

[1] FaMAF, Universidad Nacional de Córdoba, Argentina
{areces, fervari, hoffmann}@famaf.unc.edu.ar
[2] CONICET, Argentina

Abstract. We consider dynamic modal operators that can change the relation of a model during the evaluation of a formula. In this paper, we extend the basic modal language with modalities that are able to delete, add or swap pairs of related elements of the domain; and explore tableau calculi as satisfiability procedures for these logics.

1 Relation-Changing Modal Logics

We investigate modal operators that are suitable for reasoning about *dynamic aspects* of a given situation, e.g., how relations involving a set of elements *evolve* through time or through the application of certain operations. Instead of modeling the whole space of possible evolutions of the system as a graph, we use dynamic operators whose semantics directly correspond to the model evolutions that interest us. One example of such operators is *sabotage* introduced by Johan van Benthem in [8]. In the modal logic equipped with the sabotage operator, a formula can indicate that evaluation should continue in a model identical to the current one except that some edge has been removed from its relation.

In this article we present tableau methods for various relation-changing modal logics. We consider the basic modal logic $\mathcal{ML}$ [4] extended with the following operators: the local variant of sabotage $\langle sb \rangle$ deletes an arrow while traversing it; the *bridge* modality $\langle br \rangle$ adds an arrow from the current state of evaluation to a non-accessible state and continues the evaluation there; the *swap* modality $\langle sw \rangle$ inverts the direction of an arrow while traversing it. The swap modality was introduced in [3], and the local sabotage and bridge modalities in [2].

Definition 1 (Syntax). *Let* PROP *be a countable, infinite set of propositional symbols. The set* FORM *of formulas over* PROP *is defined as:*

$$\mathsf{FORM} ::= \bot \mid p \mid \neg\varphi \mid \varphi \wedge \psi \mid \blacklozenge\varphi,$$

where $p \in \mathsf{PROP}$*,* $\blacklozenge \in \{\Diamond, \langle sb \rangle, \langle br \rangle, \langle sw \rangle\}$ *and* $\varphi, \psi \in \mathsf{FORM}$*. Other operators are defined as usual. In particular,* $\blacksquare\varphi$ *is defined as* $\neg\blacklozenge\neg\varphi$*.*

Formulas of the basic modal language $\mathcal{ML}$ *contains only* $\Diamond$ *besides the Boolean operators. We call* $\mathcal{ML}(\blacklozenge)$ *the extension of* $\mathcal{ML}$ *allowing also the* $\blacklozenge$ *operator, for* $\blacklozenge \in \{\langle sb \rangle, \langle br \rangle, \langle sw \rangle\}$*.*

Semantically, formulas are evaluated in standard relational models, and the meaning of the basic modal operators is unchanged. When we evaluate formulas containing dynamic operators, we need to keep track of the edges that have been modified. To that end, let us define precisely the models that we use.

Definition 2 (Models and Model Variants). *A model is a triple* $\mathcal{M} = \langle W, R, V\rangle$*, where* W *is a non-empty set whose elements are called states;* $R \subseteq W \times W$ *is the accessibility relation; and* $V : \mathsf{PROP} \mapsto \mathcal{P}(W)$ *is a valuation.*

Given a model $\mathcal{M} = \langle W, R, V\rangle$ *we define the following notation:*

(sabotaging) $\mathcal{M}^-_S = \langle W, R^-_S, V\rangle$*, with* $R^-_S = R \backslash S$*,* $S \subseteq R$*.*
(bridging) $\mathcal{M}^+_S = \langle W, R^+_S, V\rangle$*, with* $R^+_S = R \cup S$*,* $S \subseteq (W \times W) \backslash R$*.*
(swapping) $\mathcal{M}^*_S = \langle W, R^*_S, V\rangle$*, with* $R^*_S = (R \backslash S^{-1}) \cup S$*,* $S \subseteq W \times W$*.*

Let w *be a state in* $\mathcal{M}$*, the pair* $(\mathcal{M}, w)$ *is called a* pointed model*; we will usually drop parenthesis and call* $\mathcal{M}, w$ *a pointed model. A* model variant *of* $\mathcal{M}$ *is a model obtained from* $\mathcal{M}$ *by some of the above operations.*

In the rest of this article we will use wv as a shorthand for $\{(w, v)\}$ or (w, v).

Definition 3 (Semantics). *Given a pointed model* $\mathcal{M}, w$ *and a formula* φ *we say that* $\mathcal{M}, w$ *satisfies* φ*, and write* $\mathcal{M}, w \models \varphi$*, when*

$$
\begin{array}{lll}
\mathcal{M}, w \models p & \textit{iff} & w \in V(p) \\
\mathcal{M}, w \models \neg\varphi & \textit{iff} & \mathcal{M}, w \not\models \varphi \\
\mathcal{M}, w \models \varphi \wedge \psi & \textit{iff} & \mathcal{M}, w \models \varphi \textit{ and } \mathcal{M}, w \models \psi \\
\mathcal{M}, w \models \Diamond\varphi & \textit{iff} & \textit{for some } v \in W \textit{ s.t. } Rwv, \quad \mathcal{M}, v \models \varphi \\
\mathcal{M}, w \models \langle sb\rangle\varphi & \textit{iff} & \textit{for some } v \in W \textit{ s.t. } Rwv, \quad \mathcal{M}^-_{wv}, v \models \varphi \\
\mathcal{M}, w \models \langle br\rangle\varphi & \textit{iff} & \textit{for some } v \in W \textit{ s.t. } \neg Rwv, \mathcal{M}^+_{wv}, v \models \varphi \\
\mathcal{M}, w \models \langle sw\rangle\varphi & \textit{iff} & \textit{for some } v \in W \textit{ s.t. } Rwv, \quad \mathcal{M}^*_{vw}, v \models \varphi
\end{array}
$$

φ *is* satisfiable *if for some pointed model* $\mathcal{M}, w$ *we have* $\mathcal{M}, w \models \varphi$*.*

Adding any of the previous operators to the basic modal logic increases its expressive power. A basic result for $\mathcal{ML}$ [4] shows that it has the *tree model property*: every satisfiable formula of $\mathcal{ML}$ can be satisfied at the root of a model where the accessibility relation defines a tree. In [2] we introduced formulas using the operators above that cannot be satisfied at the root of a tree:

1. $\varphi = \Diamond\Diamond\top \;\wedge\; [sb]\Box\bot$ is true at a state w, only if w is reflexive.
   Suppose we evaluate φ at some state w of an arbitrary model. On one hand, the 'static' part of the formula $\Diamond\Diamond\top$ ensures it is possible to take two steps using the accessibility relation. On the other hand, the 'dynamic' part of the formula $[sb]\Box\bot$ tells us that after traversing any edge and eliminating it we arrive to a dead-end. This can only happen if the state w is reflexive and does not have any other outgoing links.
2. $\varphi = \Box\bot \wedge \langle br\rangle\langle br\rangle\top$ is only satisfiable in models where the root is a dead-end and there is a second, unreachable state.

3. $\varphi = p \wedge (\bigwedge_{1 \leq i \leq 3} \Box^i \neg p) \wedge \langle sw \rangle \Diamond\Diamond p$ is true at a state w, only if w has a reflexive successor.
   Suppose we evaluate φ at a state w in a model. The 'static' part of the formula $p \wedge (\bigwedge_{1 \leq i \leq 3} \Box^i \neg p)$ makes p true in w and ensures that no p state is reachable within three steps from w (also w cannot be reflexive). Because $\langle sw \rangle \Diamond\Diamond p$ is true at w, there is an R-successor v where $\Diamond\Diamond p$ holds once the accessibility relation has been updated to R^*_{vw}. That is, v has to reach a p-state in exactly two R^*_{vw}-steps. The only p-state sufficiently close is w which is reachable in one step. As w is not reflexive, v has to be reflexive so that we can linger at v for one loop and reach p in the correct number of steps.

With respect to computational complexity, satisfiability of $\mathcal{ML}(\langle sw \rangle)$ is known to be undecidable [3], and we conjecture that the same holds for the other two logics. The finite model property fails for the three logics. For this reason, and as we will not introduce control mechanism like loop checks, the tableau procedures we will define not necessarily terminate on all inputs.

In Section 2 we will introduce complete and sound tableau calculi for these logics. In Section 3 we extend the results to the global counterparts of the operators. In Section 4 we discuss a few final issues.

2 Tableau Calculi

We present basic definitions for different tableau algorithms for the relation-changing modal logics we introduced in the previous section. These algorithms will rely on the same data structures and will only differ in some of their rules.

Definition 4 (Tableau formulas). *Let* NOM *be an infinite, well ordered set of symbols we call* nominals. *A* tableau formula *is either a* prefixed formula, *an* equational formula *or a* relational formula. *A prefixed formula is of the form* $(n, X) : \varphi$, *with* $n \in \mathsf{NOM}$, $X \subseteq \mathsf{NOM}^2$, *and* φ *a formula of the considered object language. An equational formula is a Boolean combination of formulas of the form* $n \doteq m$ *or* $n \dot{\neq} m$ *for* $n, m \in \mathsf{NOM}$. *We also use the following notation:*

$$nm \doteq xy \;:=\; n \doteq x \;\wedge\; m \doteq y \qquad nm \dot{\in} X \;:=\; \bigvee_{xy \in X} nm \doteq xy$$
$$nm \dot{\neq} xy \;:=\; n \dot{\neq} x \;\vee\; m \dot{\neq} y \qquad nm \dot{\notin} X \;:=\; \bigwedge_{xy \in X} nm \dot{\neq} xy.$$

In particular $nm \dot{\in} \emptyset$ *is a notation for* $\bot$ *and* $nm \dot{\notin} \emptyset$ *is a notation for* $\top$. *A relational formula is of the form* $\dot{R}nm$ *or* $\neg\dot{R}nm$, *with* $n, m \in \mathsf{NOM}$.

The set X of a prefixed formula $(n, X) : \varphi$ is used to describe the model variant in which the formula φ is to be interpreted. According to the logic we are in this set is to be interpreted differently. This is done by fixing a function f that, out of a relation $R, S \subseteq W \times W$ yields another relation $R' = f(R, S)$.

Definition 5 (Branches and interpretations). *A* branch *is a non-empty set of tableau formulas. Let $\mathcal{M} = \langle W, R, V\rangle$ be a model, $f : W^2 \times W^2 \mapsto W^2$ a relation-changing function and $\sigma : \mathsf{NOM} \mapsto W$ a mapping from nominals to states of $\mathcal{M}$. Let $X^\sigma = \{\sigma(a)\sigma(b) \mid ab \in X\}$, for $X \subseteq \mathsf{NOM}^2$.*

Given $\mathcal{M} = \langle W, R, V\rangle$, let $\mathcal{M}^f_{X^\sigma} = \langle W, f(R, X^\sigma), V\rangle$. That is, $\mathcal{M}^f_{X^\sigma}$ is the model $\mathcal{M}$ updated by the relation-changing function f according to a set of pairs of nominals X under mapping σ.

A branch Θ is satisfiable *if there exists a model $\mathcal{M} = \langle W, R, V\rangle$ and a mapping σ such that all the formulas of Θ are satisfiable under model $\mathcal{M}$ and mapping σ. That is, they should satisfy the following conditions:*

- *if $(n, X) : \varphi \in \Theta$ then $\mathcal{M}^f_{X^\sigma}, \sigma(n) \models \varphi$,*
- *if $n \dot{=} m \in \Theta$ then $\sigma(n) = \sigma(m)$,*
- *if $n \dot{\neq} m \in \Theta$ then $\sigma(n) \neq \sigma(m)$,*
- *Boolean combinations of equational formulas are interpreted as expected,*
- *if $\dot{R}nm \in \Theta$ then $R\sigma(n)\sigma(m)$,*
- *if $\neg\dot{R}nm \in \Theta$ then $\neg R\sigma(n)\sigma(m)$.*

A branch is unsatisfiable *if it is not satisfiable.*

A *tableau calculus* is a set of rules such that each rule applies to a branch and yields one or more branches, under certain conditions. These conditions are called saturation conditions, and stipulate that no rule can be applied twice on the same premises, and that no formula can be introduced twice in a branch.

A *tableau* is a tree in which each node defines a tableau branch, and edges represent applications of tableau rules. A tableau is expanded as much as possible by the rules of the system (i.e., rules are applied whenever possible according to the saturation condition). A fully expanded branch is called *saturated*.

A tableau branch is *closed* if it contains $\bot$, otherwise it is *open*. A tableau is closed if all branches are closed, otherwise it is open.

Given a branch Θ, $\sim_\Theta$ denotes the equivalence closure of the relation $\{nm \mid n \dot{=} m \in \Theta\}$, and we write $\bar{n}$ for the smallest nominal x such that $x \sim_\Theta n$. For $X \subseteq \mathsf{NOM}^2$ we write $\bar{X} = \{\bar{n}\bar{m} \mid nm \in X\}$. Figure 1 presents the rules common to all the tableau calculus of this work. They are the Boolean rules ($\wedge$) and ($\vee$), the clashing rules ($\bot_{atom}$) and ($\bot_{\neq}$), the equational rules ($R\sim$) and (Id), and the unrestricted blocking rule (ub) [7]. We use the unrestricted blocking rule as a way to saturate branches with equational formulas. These formulas can appear as premises of tableau rules in the calculi we introduce later.

This result follows easily from the tableau rules:

Lemma 6. *Let Θ be a saturated open branch. If $nm \dot{\in} S$ is in Θ then $\bar{n}\bar{m} \in \bar{S}$. If $nm \dot{\notin} S$ is in Θ then $\bar{n}\bar{m} \notin \bar{S}$.*

When it comes to adequacy of a tableau calculus, we have to consider two properties: completeness and soundness. Given a tableau calculus $\mathcal{T}$, let us write $\mathcal{T}(\varphi)$ to refer to a tableau obtained by running $\mathcal{T}$ on the input formula $(n_0, \emptyset) : \varphi$, where n_0 is the smallest nominal in NOM. Then we define:

$$\frac{(n,X):\ \varphi\wedge\psi}{\begin{array}{c}(n,X):\varphi\\(n,X):\psi\end{array}}\,(\wedge) \qquad \frac{(n,X):\ \varphi\vee\psi}{(n,X):\varphi \mid (n,X):\psi}\,(\vee)$$

$$\frac{\begin{array}{c}(n,X_1):p\\(n,X_2):\neg p\end{array}}{\bot}\,(\bot_{atom})^1 \qquad \frac{\begin{array}{c}n\sim_\Theta m\\ n\dot{\neq}m\end{array}}{\bot}\,(\bot_{\neq})$$

$$\frac{\dot{R}nm}{\dot{R}\bar{n}\bar{m}}\,(R\sim) \qquad \frac{(n,X):\varphi}{(\bar{n},X):\varphi}\,(Id) \qquad \frac{}{n\doteq m \ \mid\ n\dot{\neq}m}\,(ub)^2$$

[1] $p \in \mathsf{PROP}$
[2] n and m are two different nominals in the branch

Fig. 1. Common tableau rules.

Definition 7 (Completeness). *A tableau calculus $\mathcal{T}$ is* complete *if for any formula φ, if $\mathcal{T}(\varphi)$ is open then φ is satisfiable.*

Definition 8 (Soundness). *A tableau calculus $\mathcal{T}$ is* sound *if for any formula φ, if φ is satisfiable then $\mathcal{T}(\varphi)$ is open.*

We define models induced from open branches.

Definition 9 (Induced Models). *Let Θ be an open branch. We define $\mathcal{M}^\Theta = \langle W^\Theta, R^\Theta, V^\Theta\rangle$, the* induced model for *Θ, as:*

$$\begin{array}{ll} W^\Theta & = \{\bar{n} \mid n \in \Theta\} \\ R^\Theta & = \{(\bar{n},\bar{m}) \mid \dot{R}nm \in \Theta\} \\ V^\Theta(p) & = \{\bar{n} \mid n : p \in \Theta\}. \end{array}$$

We want to show that a tableau system is sound and complete, i.e., that for any formula φ, $\mathcal{T}(\varphi)$ is open if, and only if, φ is satisfiable. Moreover, if $\mathcal{T}(\varphi)$ has an open branch Θ then $\mathcal{M}^\Theta$ is a model that satisfies φ. We present tableau calculi for $\mathcal{ML}(\langle sb\rangle)$, $\mathcal{ML}(\langle br\rangle)$ and $\mathcal{ML}(\langle sw\rangle)$ in the next sections.

2.1 Sabotage

Figure 2 introduces rules that, in combination with those in Figure 1, form a complete and sound tableau calculus for $\mathcal{ML}(\langle sb\rangle)$. In this calculus, a formula $(n,S):\varphi$ is understood as "φ holds at the state referred to by n in the model variant described by the set of sabotaged pairs S".

We interpret branches of this tableau calculus with the following relation-changing function: $f:(R,S)\mapsto R\setminus S$. This means that a formula $(n,S):\varphi$ in

$$\dfrac{(n,S):\Diamond\varphi}{\begin{array}{c}\dot{R}nm\\ nm\dot{\notin}S\\ (m,S):\varphi\end{array}}\ (\Diamond)^1 \qquad \dfrac{\begin{array}{c}(n,S):\Box\varphi\\ \dot{R}nm\\ nm\dot{\notin}S\end{array}}{(m,S):\varphi}\ (\Box)$$

$$\dfrac{(n,S):\langle sb\rangle\varphi}{\begin{array}{c}\dot{R}nm\\ nm\dot{\notin}S\\ (m,S\cup nm):\varphi\end{array}}\ (\langle sb\rangle)^1 \qquad \dfrac{\begin{array}{c}(n,S):[sb]\varphi\\ \dot{R}nm\\ nm\dot{\notin}S\end{array}}{(m,S\cup nm):\varphi}\ ([sb])$$

[1] m is new.

Fig. 2. Tableau rules for $\mathcal{ML}(\langle sb\rangle)$.

a branch Θ should hold in the induced model variant $\mathcal{M}^{\Theta}_{S}$ defined as $\mathcal{M}^{\Theta}_{S} = \langle W^{\Theta}, R^{\Theta}_{S}, V^{\Theta}\rangle$, where $R^{\Theta}_{S} = R^{\Theta} \setminus \bar{S}$.

The rules involve the notation $nm\dot{\notin}S$. $nm\dot{\notin}S$ specifies that the edge referred to by the pair of nominals (n,m) should not be deleted in the model variant described by S. When present as premise of a rule, this condition requires that one of the disjuncts in $nm\dot{\notin}S$ is present in the branch, which in turn means that either $n\dot{\neq}x$ or $m\dot{\neq}y$ is in the branch for all $xy \in S$.

The $(\Diamond)$ rule captures the standard meaning of the $\Diamond$ connector, but adds a new constraint that specifies that the successor has not been deleted at this point of the branch. $(\Box)$ should also take this into account. For each successor m of n in the initial model $(\dot{R}nm)$, and only if it the edge between n and m has not been sabotaged $(nm\dot{\notin}S)$, φ must hold at m in the same model variant. Rule $([sb])$ is similar to $(\Box)$, but φ must hold at m in the model variant where the edge nm is sabotaged. Rule $(\langle sb\rangle)$ corresponds similarly to $(\Diamond)$.

Figure 3 presents an example with a satisfiable formula of $\mathcal{ML}(\langle sb\rangle)$.

We will now prove completeness and soundness of the calculus for $\mathcal{ML}(\langle sb\rangle)$.

Lemma 10. *Let Θ be a saturated, open branch and φ an $\mathcal{ML}(\langle sb\rangle)$-formula. If $(n,S):\varphi \in \Theta$ then $\mathcal{M}^{\Theta}_{S}, \bar{n} \models \varphi$.*

Proof. Let $(n,S):\varphi \in \Theta$. Proceed by structural induction on φ.

- $\boldsymbol{p}$**:** By definition, $\bar{n} \in V^{\Theta}(p)$, then $\mathcal{M}^{\Theta}, \bar{n} \models p$ and $\mathcal{M}^{\Theta}_{S}, \bar{n} \models p$.
- $\boldsymbol{\neg p}$**:** By saturation of (Id), $\bar{n} : \neg p \in \Theta$. Since Θ is open, $\bar{n} : p \notin \Theta$. By definition, $\bar{n} \notin V^{\Theta}(p)$, then $\mathcal{M}^{\Theta}, \bar{n} \not\models p$ and $\mathcal{M}^{\Theta}_{S}, \bar{n} \not\models p$.
- $\boldsymbol{\psi \wedge \chi}$ **and** $\boldsymbol{\psi \vee \chi}$: Trivial by inductive hypothesis.
- $\boldsymbol{\Diamond\psi}$**:** By $(\Diamond)$, Θ contains $\dot{R}nm$, $nm\dot{\notin}S$ and $(m,S):\psi$. We want to show that $\bar{n}\bar{m} \in R^{\Theta}_{S}$. We verify the following:
  1. $\bar{n}\bar{m} \in R^{\Theta}$: this is true since $\dot{R}nm \in \Theta$.
  2. $\bar{n}\bar{m} \notin \bar{S}$: this is true since $(nm\dot{\notin}S) \in \Theta$ by Lemma 6.

  Since $\bar{n}\bar{m} \in R^{\Theta}_{S}$, and (by (Id)) $(\bar{m},S):\psi \in \Theta$, we have $\mathcal{M}^{\Theta}_{S}, \bar{n} \models \Diamond\psi$.

Example: A tableau for $\Diamond\Diamond\top \;\wedge\; [sb]\Box\bot$ follows:

(1)	$(n_0, \emptyset) : \Diamond\Diamond\top \;\wedge\; [sb]\Box\bot$		initial node
(2)	$(n_0, \emptyset) : \Diamond\Diamond\top$		$(\wedge)$ on (1)
(3)	$(n_0, \emptyset) : [sb]\Box\bot$		
(4)	$\dot{R}n_0n_1$		$(\Diamond)$ on (2)
(5)	$n_0n_1\dot{\notin}\emptyset$		
(6)	$(n_1, \emptyset) : \Diamond\top$		
(7)	$\dot{R}n_1n_2$		$(\Diamond)$ on (6)
(8)	$n_1n_2\dot{\notin}\emptyset$		
(9)	$(n_2, \emptyset) : \top$		
(10)	$(n_1, \{n_0n_1\}) : \Box\bot$		$([sb])$ on (3) and (4) with trivial condition $n_0n_1\dot{\notin}\emptyset$
(11)	$n_0\doteq n_1$	$n_0\dot{\neq}n_1$	(ub)

The right branch soon closes since $(\Box)$ applies on (10) and (7) with condition $n_1n_2\dot{\notin}\{n_0, n_1\}$ fulfilled by $n_0\dot{\neq}n_1$, and introduces $\bot$. Let us expand the left branch:

(12)	$n_1\doteq n_2$	$n_1\dot{\neq}n_2$	(ub)

Again the right branch closes by application of $(\Box)$ with condition $n_1n_2\dot{\notin}\{n_0, n_1\}$ fulfilled by $n_1\dot{\neq}n_2$. We expand the left branch:

(13)	$n_0\doteq n_2$	$n_0\dot{\neq}n_2$	(ub)

The right branch above closes by rule $(\bot_{\neq})$. Left branch is saturated and open, with the following induced model:

Fig. 3. Tableau example for $\mathcal{ML}(\langle sb\rangle)$.

- $\langle sb\rangle\psi$**:** We need to show that $\mathcal{M}^{\Theta}_{S}, \bar{n} \models \langle sb\rangle\psi$, i.e., there exists $x \in V^{\Theta}$ s.t $\mathcal{M}^{\Theta}_{S\cup pq}, x \models \psi$, where $\bar{p} = \bar{n}$ and $\bar{q} = x$. This can be checked considering $(\langle sb\rangle)$ instead of $(\Diamond)$ as for the previous case.
- $\Box\psi$**:** We only consider states $x \in W^{\Theta}$ such that $\bar{n}x \in R^{\Theta}_{S}$. That is, there exists a, b such that $\dot{R}ab \in \Theta$ and $\bar{n}x = \bar{a}\bar{b}$, and $\bar{n}x \notin \bar{S}$. The condition of rule $(\Box)$ $(nm\dot{\notin}S)$ does not prevent it from being applied on such pair of nominals. By (Id), $(\bar{n}, S) : \Box\psi \in \Theta$, i.e., $(\bar{a}, S) : \Box\psi \in \Theta$, and also by $(R\sim)$, $\dot{R}\bar{a}\bar{b} \in \Theta$. By $(\Box)$ we have $(\bar{b}, S) : \psi \in \Theta$. Now, $\bar{b} = x$, so $\mathcal{M}^{\Theta}_{S}, x \models \psi$. Hence for all $x \in V^{\Theta}$ such that $\bar{n}x \in R^{\Theta}_{S}$, $\mathcal{M}^{\Theta}_{S}, x \models \psi$, i.e., $\mathcal{M}^{\Theta}_{S}, \bar{n} \models \Box\psi$.
- $[sb]\psi$**:** We need to show that $\mathcal{M}^{\Theta}_{S}, \bar{n} \models [sb]\psi$, i.e., for all $x \in W^{\Theta}$ such that $(\bar{n}, x) \in R^{\Theta}_{S}$, $\mathcal{M}^{\Theta}_{S\cup pq}, x \models \psi$, where $\bar{p}\bar{q} = \bar{n}x$. This can be checked considering rule $([sb])$ instead of $(\Box)$ as for the previous case. $\Box$

By the previous lemma we get:

Theorem 11 (Completeness). *If $\mathcal{T}(\varphi)$ is open, then φ is satisfiable.*

We now show soundness of the calculus for $\mathcal{ML}(\langle sb \rangle)$.

Lemma 12. *Let Γ be a set of satisfiable tableau formulas, and $\varphi \in \mathcal{ML}(\langle sb \rangle)$. If there is a closed tableau $\mathcal{T}(\Gamma')$ for $\Gamma' = (\Gamma \cup \{\neg\varphi\})$, then φ is satisfiable.*

Proof. Let Θ be a satisfiable branch. Following Definition 5, Θ is satisfied by a model $\mathcal{M} = \langle W, R, V \rangle$ and a mapping $\sigma : \mathsf{NOM} \mapsto W$. We write $\sigma[m \mapsto v]$ to refer to the mapping equal to σ except, perhaps, $\sigma(m) = v$.

Assume that there is a closed tableau $\mathcal{T}(\Gamma')$ such that $\Gamma' = (\Gamma \cup \{\neg\varphi\})$. We will prove Γ' unsatisfiable, by induction on the tableau structure.

- **($\bot_{atom}$):** If this rule applies, then $n : a \in \Gamma'$ and $n : \neg a \in \Gamma'$, for some n, a. Then Γ' is trivially unsatisfiable.
- Common rules $(\bot_{\neq})$, $(\wedge)$, $(\vee)$, $(R\sim)$, (Id) and (ub) are easy to check.

It remains to verify, for each remaining rule, that their application to a satisfiable branch generates at least one satisfiable branch. In the present calculus, all remaining rules are non-branching.

- **($\Diamond$):** Suppose $(n, S) : \Diamond\varphi \in \mathcal{T}(\Gamma')$. We know that $(n, S) : \Diamond\varphi$ is satisfiable, then there is a model $\mathcal{M} = \langle W, R, V \rangle$, and a mapping $\sigma : \mathsf{NOM} \mapsto W'$ s.t. $\mathcal{M}^-_{S^\sigma}, \sigma(n) \models \Diamond\varphi$. By definition of $\models$, there exists $v \in W$ s.t. $\sigma(n)v \in R \setminus S^\sigma$ and $\mathcal{M}^-_{S^\sigma}, v \models \varphi$. The $(\Diamond)$ rule generates $\dot{R}nm$, $nm\dot{\notin}S$ and $(m, S) : \varphi$, with m new in the branch. We need to check that the branch containing these three new formulas is satisfiable. That is, there exists a model and a mapping satisfying them. Let us consider the mapping $\sigma' = \sigma[m \mapsto v]$ and check that the interpretation $\mathcal{M}, \sigma'$ satisfies the new branch:
  - $\dot{R}nm$ is satisfied since $R\sigma'(n)\sigma'(m)$, i.e., $R\sigma(n)v$, holds.
  - Consider $nm\dot{\notin}S$. It suffices to check that for all $xy \in S$, $\sigma'(n)\sigma'(m) \neq \sigma'(x)\sigma'(y)$, i.e., $\sigma(n)v \neq \sigma'(x)\sigma'(y)$. But $\sigma(n)v = \sigma'(x)\sigma'(y)$ would contradict $\sigma(n)v \in R \setminus S^\sigma$.
  - $\mathcal{M}, \sigma'$ satisfies $(m, S) : \varphi$ since $\mathcal{M}^-_{S^{\sigma'}}, \sigma'(m) \models \varphi$ holds.
- **($\langle sb \rangle$):** This case is similar to $(\Diamond)$, except that we need to check that the new tableau formula $(m, S \cup nm) : \varphi$ is satisfied. This is done considering the new mapping $\sigma' = \sigma[m \mapsto v]$ and observing that $\mathcal{M}^-_{S \cup nm^{\sigma'}}, \sigma'(m) \models \varphi$.
- **($\Box$):** Suppose $(n, S) : \Box\varphi$ and $\dot{R}nm$ are in Θ, and the condition $nm\dot{\notin}S$ holds. This implies that there exists $\mathcal{M} = \langle W, R, V \rangle$ and a mapping σ such that $\mathcal{M}^-_{S^\sigma}, \sigma(n) \models \Box\varphi$, and $R\sigma(n)\sigma(m)$, and there is no pair of nominals $xy \in S$ such that $nm = xy$. This means that for all $v \in W$ s.t. $\sigma(n)v \in (R \setminus S^\sigma)$, $\mathcal{M}^-_{S^\sigma}, v \models \varphi$ and there exists $v \in W$ s.t. $R\sigma(n)v$. We verify that $(m, S) : \varphi$ is satisfied by $\mathcal{M}, \sigma$. Since $\sigma(n)\sigma(m) \in (R \setminus S^\sigma)$, then $\mathcal{M}^-_{S^\sigma}, \sigma(m) \models \varphi$. Hence $(m, S) : \varphi$ is satisfied by $\mathcal{M}, \sigma$.
- **($[sb]$):** This is similar to the $(\Box)$ case, but we have to show that $(m, S \cup nm) : \varphi$ is satisfied by $\mathcal{M}, \sigma$. This is done by observing that if $\mathcal{M}^-_{S^\sigma}, \sigma(n) \models [sb]\varphi$ and $R\sigma(n)\sigma(m)$ then $\mathcal{M}^-_{S \cup nm^\sigma}, \sigma(m) \models \varphi$. □

From the previous lemma we get the following result:

Theorem 13 (Soundness). *If φ is satisfiable, then $\mathcal{T}(\varphi)$ is open.*

$$\dfrac{(n,B):\Diamond\varphi}{\begin{array}{c|c}\dot{R}nm & nm\dot{\in}B\\ (m,B):\varphi & (m,B):\varphi\end{array}}\ (\Diamond)^1 \qquad \dfrac{\begin{array}{c}(n,B):\Box\varphi\\ \dot{R}nm\end{array}}{(m,B):\varphi}\ (\Box) \qquad \dfrac{\begin{array}{c}(n,B):\Box\varphi\\ nm\dot{\in}B\end{array}}{(m,B):\varphi}\ (\Box_2)$$

$$\dfrac{\begin{array}{c}\dot{R}nm\\ (a,B):\varphi\\ nm\dot{\in}B\end{array}}{\bot}\ (R_\bot) \qquad \dfrac{(n,B):\langle br\rangle\varphi}{\begin{array}{c}nm\dot{\notin}B\\ (m,B\cup nm):\varphi\end{array}}\ (\langle br\rangle)^1 \qquad \dfrac{\begin{array}{c}(n,B):[br]\varphi\\ nm\dot{\notin}B\end{array}}{(m,B\cup nm):\varphi \mid \dot{R}nm}\ ([br])^2$$

[1] m is new.
[2] m is already in the branch.

Fig. 4. Tableau rules for $\mathcal{ML}(\langle br\rangle)$.

2.2 Bridge

Figure 4 presents rules for the tableau calculus corresponding to $\mathcal{ML}(\langle br\rangle)$ which should be combined with the common rules of Figure 1. The main difference with rules for sabotage is that they use as prefix a set of pairs of nominals B to keep track of edges that have been added to the relation of the original model.

The interpretation function will be $f:(R,B)\mapsto R\cup B$. This means that a formula $(n,B):\varphi$ in a branch Θ should hold in the induced model variant $\mathcal{M}_B^\Theta$ defined as $\mathcal{M}_B^\Theta=\langle W^\Theta,R_B^\Theta,V^\Theta\rangle$, where $R_B^\Theta=R^\Theta\ \cup\ \bar{B}$. The notation $nm\dot{\in}B$ means that the edge represented by the nominals n and m is one of the edges added since the initial model in the model variant described by B. When used as a premise of a rule, the condition $nm\dot{\in}B$ requires that there exists some $xy\in B$ such that $n\dot{=}x$ and $m\dot{=}y$ are present in the branch. $nm\dot{\notin}B$ means that the edge (n,m) has not been added since the initial model in the variant described by B.

Some rules are more involved in this calculus. The rule $(\Diamond)$, when applied on a formula $(n,B):\Diamond\varphi$, has to ensure that in the model variant described by B, the state referred to by the nominal n has a successor where φ holds. This model variant has a relation that is the union of the relation in the initial model and B. This is why $(\Diamond)$ is a branching rule that either chooses that the edge (n,m) belongs to the initial relation or to B.

The rule $(\Box)$ is the standard box rule for the basic modal logic. It is completed by a $(\Box_2)$ rule that ensures new edges of model variants are taken into account.

The new clash rule $(R_\bot)$ ensures that whenever some edge nm is present in a set of new edges B representing some model variant, the same edge is not present in the original model, i.e., $\dot{R}nm$ is forbidden to occur in the branch.

The rule $(\langle br\rangle)$ differs from $(\Diamond)$. This is because the $\langle br\rangle$ operator jumps to a state that should *not* be accessible from the current state, hence the introduction of $nm\dot{\notin}B$ and $(m,B\cup nm):\varphi$ to the branch. This last formula, together with rule $(R_\bot)$, ensures that the edge nm is not in the original model.

The rule $([br])$ branches when applied to a formula $(n, B) : [br]\varphi$. It decides, for every nominal m such that $nm \dot{\notin} B$, whether $(m, R \cup nm) : \varphi$ holds, or $\dot{R}nm$ holds. In the first case, together with rule $(R_\perp)$, it ensures that the edge nm is not in the original model. In the second case, it ensures the contrary, hence no bridging to m is possible and φ does not need to hold at m.

Completeness and soundness of this tableau calculus can be proved as in the previous section. Figure 5 shows an example of how the rules are used.

Example: Consider the satisfiable formula $p \wedge \Diamond\neg p \wedge [br]p$. In the following tableau we hide the branches that directly close by vacuity of quantification:

(1)	$(n_0, \emptyset) : p \ \wedge \ \Diamond\neg p \ \wedge \ [br]p$		initial node
(2)	$(n_0, \emptyset) : p$		$(\wedge)$ on (1)
(3)	$(n_0, \emptyset) : \Diamond\neg p$		
(4)	$(n_0, \emptyset) : [br]p$		
(5)	$\dot{R}n_0n_1, (n_1, \emptyset) : \neg p$		$(\Diamond)$ on (3)
(6)	$n_0 \doteq n_1$	$n_0 \dot{\neq} n_1$	(ub)

Left branch closes due to (Id) and $(\perp_{atom})$. Right branch:

(7)	$(n_1, \{n_0n_1\}) : p$	$\dot{R}n_0n_1$	$([br])$ on (3), n_1

Left branch closes by $(\perp_{atom})$ on $(n_1, \{n_0n_1\}) : p$ and $(n_1, \emptyset) : \neg p$. Right branch:

(8)	$(n_0, \{n_0n_0\}) : p$	$\dot{R}n_0n_0$	$([br])$ on (4), n_0

Both branches are open and saturated. We have the following two induced models:

Fig. 5. Tableau example for $\mathcal{ML}(\langle br \rangle)$.

2.3 Swap

Rules for the swap calculus are given in Figure 6, to be used in combination with the rules in Figure 1.

These rules have to handle the fact that swapping edges in a model can make some edges of the original model no longer usable (as when using the sabotage modality), and can make new edges usable (as with bridge). The set S that prefixes formulas of the calculus has to be understood as the pairs of states that no longer are part of the relation of the model variant. S^{-1} contains the edges that should be added to the model.

The interpretation function for this calculus is $f : (R, S) \mapsto (R \backslash S) \cup S^{-1}$. This means that a formula $(n, S) : \varphi$ in a branch Θ should hold in the induced model variant $\mathcal{M}^\Theta_S$ defined as $\mathcal{M}^\Theta_S = \langle W^\Theta, R^\Theta_S, V^\Theta \rangle$, where $R^\Theta_S = (R^\Theta \ \backslash \ \bar{S}) \ \cup \ \bar{S}^{-1}$.

$$\frac{(n,S):\Diamond\varphi}{\begin{array}{c|c}\dot{R}nm & nm\dot{\in}S^{-1}\\ nm\dot{\notin}S & (m,S):\varphi\\ (m,S):\varphi & \end{array}}\,(\Diamond)^1 \qquad \frac{\begin{array}{c}(n,S):\Box\varphi\\ \dot{R}nm\\ nm\dot{\notin}S\end{array}}{(m,S):\varphi}\,(\Box) \qquad \frac{\begin{array}{c}(n,S):\Box\varphi\\ nm\dot{\in}S^{-1}\end{array}}{(m,S):\varphi}\,(\Box_2)$$

$$\frac{\begin{array}{c}(n,S):[sw]\varphi\\ \dot{R}nn\end{array}}{(n,S):\varphi}\,([sw]) \qquad \frac{\begin{array}{c}(n,S):[sw]\varphi\\ \dot{R}nm\\ n\dot{\neq}m\\ nm\dot{\notin}(S\cup S^{-1})\end{array}}{(m,S\cup nm):\varphi}\,([sw]_2) \qquad \frac{\begin{array}{c}(n,S):[sw]\varphi\\ xy\in S\\ n\dot{=}y\end{array}}{(x,S\backslash xy\cup yx):\varphi}\,([sw]_3)$$

$$\frac{(n,S):\langle sw\rangle\varphi}{\begin{array}{c|c|l}\dot{R}nn & \dot{R}nm & \bigvee_{xy\in S}(n\dot{=}y\wedge(x,S\backslash xy\cup yx):\varphi)\\ (n,S):\varphi & n\dot{\neq}m & \\ & nm\dot{\notin}(S\cup S^{-1}) & \\ & (m,S\cup nm):\varphi & \end{array}}\,(\langle sw\rangle)^1$$

[1] m is new.

Fig. 6. Tableau rules for $\mathcal{ML}(\langle sw\rangle)$.

In this calculus, S is kept irreflexive and asymmetric. Moreover, it will not contain two different pairs of nominals that refer to the same edge in the induced model. This guarantees that the names in S can be manipulated by the calculus as expected, in particular when a swapped edge must be swapped again. $nm\dot{\in}S$ means that nm is no longer present in the model variant represented by S. $nm\dot{\in}S^{-1}$ means that nm has been added to the S model variant.

Let us examine the rules. $(\Diamond)$ is a combination of the $(\Diamond)$ rules for sabotage and bridge. It satisfies the formula $(n,S):\varphi$ in a state that is either accessible through the initial relation or through a new swapped edge (as in the bridge calculus). In the case of being accessible through the initial relation, the rule ensures that the edge used has not been deleted in the current model variant (as in the sabotage calculus). The $(\Box)$ rule, as in the sabotage calculus, works with all states accessible from n in the initial model variant, except when they have been made inaccessible in the current model variant. The $(\Box_2)$ rule, as in the bridge calculus, ensures that newly accessible states receive the formula φ.

The remaining (swapping) rules deserve more careful explanation. The three rules that handle formulas of the form $[sw]\varphi$ handle the case of swapping a reflexive edge, swapping an irreflexive edge that has never been swapped (nor its inverse), and swapping again an edge. $([sw])$ swaps reflexive edges, for which the S set does not need to be modified since swapping a reflexive edge leaves it unchanged. $([sw]_2)$ swaps irreflexive edges that have never been swapped before, i.e., usable edges (not in S) that are not in S^{-1}. This rule ensures that S is irreflexive ($n \dot{\neq} m$), asymmetric ($nm \dot{\notin} S^{-1}$) and that it does not contain two pairs of nominals that refer to the same edge in the induced model ($nm \dot{\notin} S$). Finally, ($[sw]_3$) traverses and swaps around edges of S^{-1}. If $n \doteq y$ is in the branch and $xy \in S$ then we swap again the link yx and end up at x. Hence it removes xy from S and adds yx. This preserves the three properties of the set S (irreflexivity, asymmetry and no-redundant-names).

There is only one ($\langle sw \rangle$) rule but it handles three possibilities of satisfying a swap-diamond formula similarly to the rules for swap-box formulas. The ($\langle sw \rangle$) rule can satisfy a formula $(n, S) : \langle sw \rangle \varphi$ in three possible ways. First, through a reflexive edge, having φ true at n in the same model variant. In that case S remains unchanged. Or it satisfies it by adding an irreflexive edge to the initial relation ($\dot{R}nm$, $n \dot{\neq} m$), specifying that in the model variant S it is not removed nor is a new edge added by swapping ($nm \dot{\notin} (S \cup S^{-1})$), and then satisfying φ at m in the model variant $S \cup nm$. Finally, it can satisfy the antecedent formula by swapping again a swapped edge, updating S appropriately. The meaning of the last branch of this rule is to properly maintain the set S when an edge is swapped more than once. When an edge $xy \in S$ is swapped again, we update S by removing xy and adding yx, instead of adding a new pair of nominals.

Figure 7 shows the use of the tableau rules in an example.

Now we are going to prove completeness for the $\mathcal{ML}(\langle sw \rangle)$ calculus. Soundness can be shown similarly as for sabotage.

Lemma 14. *Let Θ be a saturated, open branch and φ a $\mathcal{ML}(\langle sw \rangle)$-formula. If $(n, S) : \varphi \in \Theta$ then $\mathcal{M}^{\Theta}_{S}, \bar{n} \models \varphi$.*

Proof. Let $(n, S) : \varphi \in \Theta$, we proceed by structural induction on φ. Propositional and Boolean cases are exactly the same that for $\mathcal{ML}(\langle sb \rangle)$.

- **$\Diamond\psi$:** We have two cases:
  1. $\dot{R}nm \in \Theta$, $nm \dot{\notin} S \in \Theta$ and $(m, S) : \psi \in \Theta$. Since $\dot{R}nm \in \Theta$, we have $(\bar{n}, \bar{m}) \in R^{\Theta}$. On the other hand, since $nm \dot{\notin} S \in \Theta$ and the branch is saturated and open, by Lemma 6, $\bar{n}\bar{m} \notin \bar{S}$. Then $\bar{n}\bar{m} \in R^{\Theta}_{S}$ and (by (Id)) $(\bar{m}, S) : \psi \in \Theta$. Hence, $\mathcal{M}^{\Theta}_{S}, \bar{n} \models \Diamond\psi$.
  2. $nm \dot{\in} S^{-1} \in \Theta$ and $(m, S) : \psi \in \Theta$. From the fist sentence, by Lemma 6, we have $\bar{n}\bar{m} \in \bar{S}$, hence $\bar{n}\bar{m} \in R^{\Theta}_{S}$. With the same argument that the previous item, we have $\mathcal{M}^{\Theta}_{S}, \bar{n} \models \Diamond\psi$.
- **$\langle sw \rangle\psi$:** ($\langle sw \rangle$) rule has three branches:
  1. $\dot{R}nn \in \Theta$ and $(n, S) \in \Theta$. In this case $\bar{n}\bar{n} \in R^{\Theta}_{S}$, and by (Id) $(\bar{n}, S) : \psi \in \Theta$, so we have $\mathcal{M}^{\Theta}_{S}, \bar{n} \models \langle sw \rangle\psi$.
  2. In the second branch, the following formulas belong to Θ: *a)* $\dot{R}nm$, *b)* $n \dot{\neq} m$, *c)* $nm \dot{\notin} (S \cup S^{-1})$ and *d)* $(m, S \cup nm) : \psi$. *b)* holds since we are not in the previous case. By *a)* and *c)* (and Lemma 6), we have $\bar{n}\bar{m} \in R^{\Theta}_{S}$. By (Id) and *d)*, $(\bar{m}, S \cup nm) : \psi \in \Theta$. Hence, $\mathcal{M}^{\Theta}_{S}, \bar{n} \models \langle sw \rangle\psi$.
  3. In the third branch, there are $x, y \in W^{\Theta}$, such that $y \doteq n \in \Theta$ and $(x, S \setminus xy \cup yx) \in \Theta$.. Then $\bar{y} \doteq \bar{n} \in \Theta$ and by definition $\bar{y}\bar{x} \in R^{\Theta}_{S}$ $\otimes$. But, $(\bar{x}, S \setminus xy \cup yx) : \psi \in \Theta$, therefore $\mathcal{M}^{\Theta}_{S \setminus xy \cup yx}, \bar{x} \models \psi$. Then, since this last condition and $\otimes$, we have $\mathcal{M}^{\Theta}_{S}, \bar{n} \models \langle sw \rangle\psi$.

Example: Consider the formula $\neg p \wedge \langle sw\rangle\Diamond p$.

(1)	$(n_0,\emptyset) : \neg p \wedge \langle sw\rangle\Diamond p$		initial node
(2)	$(n_0,\emptyset) : \neg p$		$(\wedge)$ on (1)
(3)	$(n_0,\emptyset) : \langle sw\rangle\Diamond p$		
(4)	$\dot{R}n_0n_0, (n_0,\emptyset) : \Diamond p$	$\dot{R}n_0n_1, n_0\dot{\neq}n_1, (n_1,\{n_0n_1\}) : \Diamond p$	$(\langle sw\rangle)$ on (3)

Let us expand the left branch:

(5a)	$\dot{R}n_0n_1, n_0n_1\dot{\notin}\emptyset, (n_1,\emptyset) : p$		$(\Diamond)$ on (4)
(6a)	$n_0\dot{=}n_1$	$n_0\dot{\neq}n_1$	(ub)

The left branch closes by (Id) and $(\bot_{atom})$, while the right branch is fully expanded and open, with the following induced model:

Let us go back to line (4) and expand the right branch:

(5b)	$\dot{R}n_1n_2,\ n_1n_2\dot{\notin}\{n_0n_1\}$	$n_1n_2\dot{\in}\{n_1n_0\}$	$(\Diamond)$ on (4)
(6b)	$(n_2,\{n_0n_1\}) : p$	$(n_2,\{n_0n_1\}) : p$	

In the right branch, by $n_1n_2\dot{\in}\{n_1n_0\}$ we have $n_2\dot{=}n_0$. Then by (Id) and $(\bot_{atom})$, we have a clash. The left branch is open, and $n_1n_2\dot{\notin}\{n_0n_1\}$ is a notation for $n_0\dot{\neq}n_1 \vee n_1\dot{\neq}n_2$, with $n_0\dot{\neq}n_1$ already occurring in the branch (line (4), right branch).

(7b)	$n_0\dot{=}n_2$	$n_0\dot{\neq}n_2$	(ub)

Left branch closes by (Id) and $(\bot_{atom})$. Right branch:

(8b)	$n_1\dot{=}n_2$	$n_1\dot{\neq}n_2$	(ub)

Both branch are open and saturated and produce the following induced models:

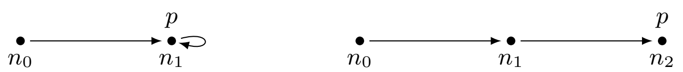

Fig. 7. Tableau example for $\mathcal{ML}(\langle sw\rangle)$.

- $\Box\psi$**:** for all $m \in W^{\Theta}$ such that $\dot{R}nm$ and $nm\dot{\notin}S \in \Theta$, we have $(m, S) : \psi \in \Theta$. Because Θ is open and saturated, by Lemma 6 it holds that $\bar{n}\bar{m} \notin \bar{S}$, which implies $\bar{n}\bar{m} \in R^{\Theta}_{S}$. Otherwise, if $nm \in S^{-1}$, then also (by definition) $\bar{n}\bar{m} \in R^{\Theta}_{S}$. In both cases, we have $(\bar{m}, S) : \psi \in \Theta$. Hence, $\mathcal{M}^{\Theta}_{S}, \bar{n} \models \Box\psi$.
- $[sw]\psi$**:** the reflexive case is the same as for $\Box$. If we have in Θ that $\dot{R}nm$, $n\dot{\neq}m$ and $nm\dot{\notin}(S \cup S^{-1})$, then $\bar{n}\bar{m} \in R^{\Theta}_{S}$. Also we have $(\bar{m}, S \cup nm) : \psi \in \Theta$. On the other hand, if $xy\dot{\in}S$ and $n\dot{=}y$ are both in Θ, (by definition) $\bar{y}\bar{x} \in R^{\Theta}_{S}$, and $(\bar{x}, S\backslash xy{\cup}yx){:}\psi \in \Theta$. With the three cases, we get $\mathcal{M}^{\Theta}_{S}, \bar{n} \models [sw]\psi$. $\Box$

By the previous lemma we get:

Theorem 15 (Completeness). *If $\mathcal{T}(\varphi)$ is open, then φ is satisfiable.*

3 Global Relation-Changing Operators

In previous sections we considered only local operators that modify the model relation from the current state of evaluation. In particular, the sabotage and swap modalities traverse an existing accessibility relation from the current state. The bridge modality is local in the sense that it creates a new link also from the current state.

We now consider the global counterparts of these three modalities. These new versions can change the accessibility relation in any part of the model, and leave the evaluation state unchanged. One motivation to consider these global operators is, again, van Benthem's original sabotage operator [8], which is actually global.

The semantics of the three global operators is formally defined as follows:

$$\begin{array}{lll}
\mathcal{M}, w \models \langle gsb \rangle \varphi & \text{iff} & \text{for some } u, v \in W, \text{ s.t. } Ruv, \ \mathcal{M}^-_{uv}, w \models \varphi \\
\mathcal{M}, w \models \langle gbr \rangle \varphi & \text{iff} & \text{for some } u, v \in W \text{ s.t. } \neg Ruv, \mathcal{M}^+_{uv}, w \models \varphi \\
\mathcal{M}, w \models \langle gsw \rangle \varphi & \text{iff} & \text{for some } u, v \in W \text{ s.t. } Ruv, \ \ \mathcal{M}^*_{vu}, w \models \varphi.
\end{array}$$

Adapting the calculi presented in Section 2, we can obtain tableau methods for the global operations. For each logic, the corresponding ($\Diamond$) and ($\Box$) rules are the same ones as for its local version. One can easily verify that the rules for $\mathcal{ML}(\langle gsb \rangle)$ and $\mathcal{ML}(\langle gbr \rangle)$ in Figure 8 are direct adaptations of the rules for $\mathcal{ML}(\langle sb \rangle)$ and $\mathcal{ML}(\langle br \rangle)$. The rules for $\mathcal{ML}(\langle gsw \rangle)$ are shown in Figure 9). Notice that ($[gsw]_3$) and (the last branch produced by) ($\langle gsw \rangle$) are simpler than ($[sw]_3$) and ($\langle sw \rangle$). This is because swapping an already swapped edge in any place is a generalization of doing it only from the evaluation state.

$$\frac{(n,S) : \langle gsb \rangle \varphi}{\begin{array}{c} \dot{R}pq \\ pq \notin S \\ (n, S \cup pq) : \varphi \end{array}} \ (\langle gsb \rangle)^1 \qquad \frac{(n,B) : \langle gbr \rangle \varphi}{\begin{array}{c} pq \notin B \\ (n, B \cup pq) : \varphi \end{array}} \ (\langle gbr \rangle)^1$$

$$\frac{\begin{array}{c} (n,S) : [gsb] \varphi \\ \dot{R}pq \\ pq \notin S \end{array}}{(n, S \cup pq) : \varphi} \ ([gsb]) \qquad \frac{\begin{array}{c} (n,B) : [gbr] \varphi \\ pq \notin B \end{array}}{(n, B \cup pq) : \varphi \ \mid \ \dot{R}pq} \ ([gbr])$$

[1] p and q are new to the branch.

Fig. 8. Tableau rules for $\mathcal{ML}(\langle gsb \rangle)$ and $\mathcal{ML}(\langle gbr \rangle)$.

The resulting calculi are sound and complete. The complexity for the satisfiability of these logics is still open but we conjecture they are undecidable (a close variant of $\mathcal{ML}(\langle gsb \rangle)$ is undecidable [6]). Applying similar arguments as for the local operators, it is possible to at least enforce infinite models.

$$\frac{(n,S):[gsw]\varphi \quad \dot{R}pp}{(n,S):\varphi}\ ([gsw]) \qquad \frac{(n,S):[gsw]\varphi \quad \dot{R}pq \quad p\dot{\neq}q \quad pq\dot{\notin}(S\cup S^{-1})}{(n,S\cup pq):\varphi}\ ([gsw]_2) \qquad \frac{(n,S):[gsw]\varphi \quad xy\in S}{(n,S\backslash xy\cup yx):\varphi}\ ([gsw]_3)$$

$$\frac{(n,S):\langle gsw\rangle\varphi}{\dot{R}pp,\ (n,S):\varphi \ \Big|\ \dot{R}pq,\ p\dot{\neq}q,\ pq\dot{\notin}(S\cup S^{-1}),\ (n,S\cup pq):\varphi \ \Big|\ \bigvee_{xy\in S}(n,S\backslash xy\cup yx):\varphi}\ (\langle gsw\rangle)^1$$

[1] p and q are new to the branch.

Fig. 9. Tableau rules for $\mathcal{ML}(\langle gsw\rangle)$.

4 Ending Remarks

In this article we considered a number of dynamic operators which can add, delete and swap edges in the accessibility relation, both locally and globally. We introduced sound and complete tableau procedures for all of them to check satisfiability.

A natural question is whether it is possible to combine these calculi into a unique calculus that would support modal logic equipped with all the dynamic operators at once. We can easily obtain local-global combinations of calculi for operators of the same kind: $\mathcal{ML}(\langle sb\rangle,\langle gsb\rangle)$, $\mathcal{ML}(\langle br\rangle,\langle gbr\rangle)$ and $\mathcal{ML}(\langle sw\rangle,\langle gsw\rangle)$, by combining the corresponding rules from Section 2 and Section 3. However, further combination seems to require deep changes since every kind of dynamic logic (sabotage, bridge, swap) requires distinct rules for the connectors $\Diamond$ and $\Box$.

As can be seen from their corresponding calculi, the logics presented here involve equality reasoning on named states. They are actually related to hybrid logics [5,1]. In particular $\mathcal{ML}(\langle sw\rangle)$ is strictly less expressive than $\mathcal{H}(:,\downarrow)$ [3]. The same can be shown about $\mathcal{ML}(\langle sb\rangle)$ and $\mathcal{H}(:,\downarrow)$. Let $S\subseteq \mathsf{NOM}^2$ and $x',y'\in \mathsf{NOM}$. Define $(\ )'_S$, a translation from formulas of $\mathcal{ML}(\langle sb\rangle)$ to formulas of $\mathcal{H}(:,\downarrow)$ as (for the non-trivial cases):

$$(\Diamond\varphi)'_S = \downarrow x'.\Diamond\downarrow y'.(\neg \bigvee_{xy\in S}(x'{:}x\wedge y'{:}y) \quad\wedge\quad (\varphi)'_S)$$
$$(\langle sb\rangle\varphi)'_S = \downarrow x'.\Diamond\downarrow y'.(\neg \bigvee_{xy\in S}(x'{:}x\wedge y'{:}y) \quad\wedge\quad (\varphi)'_{S\cup x'y'})$$

where x' and y' are nominals that do not appear in S. With this translation it holds that for any formula φ of $\mathcal{ML}(\langle sb\rangle)$ and pointed model $\mathcal{M},w$, we have $\mathcal{M},w\models\varphi$ iff $\mathcal{M},w\models(\varphi)'_S$. On the other hand, translation for the four remaining logics involve the global modality E.

All of the logics we considered can force infinite models. As a result, the tableau calculi not necessarily terminate on all inputs, given that they do not implement any kind of loop checking. Our ongoing research aims to establish the undecidability of all the presented logics using techniques from [3], showing in this way that non-termination is unavoidable.

As future work, we plan to investigate constructive interpolation results in hybrid versions of the logics we presented here.

Acknowledgments: This work was partially supported by grants ANPCyT-PICT-2008-306, ANPCyT-PICT-2010-688, the FP7-PEOPLE-2011-IRSES Project "Mobility between Europe and Argentina applying Logics to Systems" (MEALS) and the Laboratoire International Associé "INFINIS".

References

1. Areces, C., ten Cate, B.: Hybrid logics. In: Blackburn, P., Wolter, F., van Benthem, J. (eds.) Handbook of Modal Logics, pp. 821–868. Elsevier (2006)
2. Areces, C., Fervari, R., Hoffmann, G.: Moving arrows and four model checking results. In: Proceedings of WoLLIC 2012. Buenos Aires, Argentina (September 2012)
3. Areces, C., Fervari, R., Hoffmann, G.: Swap logic. To appear in the Logic Journal of IGPL (2013)
4. Blackburn, P., de Rijke, M., Venema, Y.: Modal Logic, Cambridge Tracts in Theoretical Computer Science, vol. 53. Cambridge University Press, Cambridge (2001)
5. Blackburn, P., Seligman, J.: Hybrid languages. Journal of Logic, Language and Information 4, 251–272 (1995)
6. Löding, C., Rohde, P.: Model checking and satisfiability for sabotage modal logic. In: Pandya, P., Radhakrishnan, J. (eds.) FSTTCS. Lecture Notes in Computer Science, vol. 2914, pp. 302–313. Springer (2003)
7. Schmidt, R.A., Tishkovsky, D.: Using tableau to decide expressive description logics with role negation. Lecture Notes in Computer Science, vol. 4825, pp. 438–451. Springer (2007)
8. van Benthem, J.: An essay on sabotage and obstruction. In: Mechanizing Mathematical Reasoning. pp. 268–276 (2005)