Labelled Tableaux for Distributed Temporal Logic

DAVID BASIN, *Department of Computer Science, ETH Zurich, Switzerland.*
E-mail: basin@inf.ethz.ch

CARLOS CALEIRO and JAIME RAMOS, *SQIG — Instituto de Telecomunicações, Department of Mathematics, IST, TU Lisbon, Portugal.*
E-mail: ccal@math.ist.utl.pt; jabr@math.ist.utl.pt

LUCA VIGANÒ, *Department of Computer Science, University of Verona, Italy.*
E-mail: luca.vigano@univr.it

Abstract

The distributed temporal logic DTL is a logic for reasoning about temporal properties of discrete distributed systems from the local point of view of the system's agents, which are assumed to execute sequentially and to interact by means of synchronous event sharing. We present a sound and complete labelled tableaux system for full DTL. To achieve this, we first formalize a labelled tableaux system for reasoning locally at each agent and afterwards we combine the local systems into a global one by adding rules that capture the distributed nature of DTL. We also provide examples illustrating the use of DTL and our tableaux system.

Keywords: Distributed temporal logic, discrete time, until and since, labelled tableaux system, soundness and completeness.

1 Introduction

The distributed temporal logic DTL [11] is a logic for reasoning about temporal properties of discrete distributed systems from the local point of view of the system's agents, which are assumed to execute sequentially and to interact by means of synchronous event sharing. Distribution is implicit and properties of entire systems are formulated in terms of the local properties of the system's agents and their interaction. DTL is closely related to the family of temporal logics whose semantics are based on the models of true concurrency, introduced and developed in [21, 22, 28]. In particular, the semantics of these logics are based on a conflict-free version of Winskel's event structures [36], enriched with information about sequential agents. Several different versions have been given, reflecting different perspectives on how non-local information can be accessed by each agent.

DTL was first proposed in [11] as a logic for specifying and reasoning about distributed information systems. The logic has also been used in the context of security protocol analysis to reason about the interplay between protocol models and security properties [5, 6]. However, all of the previous results have been obtained directly by semantic arguments. It would be reassuring and generally useful to have a deductive system for DTL for carrying out such proofs. There are several possibilities for deduction in temporal logics, including Hilbert calculi, resolution, sequent and tableaux systems

Vol. 19 No. 6, © The Author, 2009. Published by Oxford University Press. All rights reserved.
For Permissions, please email: journals.permissions@oxfordjournals.org
Published online 7 July 2008 doi:10.1093/logcom/exp022

and model checking [3, 7, 8, 14–20, 23, 24, 29–32, 35, 37] (P.H. Schmitt and J. Goubault-Larrecq, Unpublished data). We explore in this article two options for deduction in DTL.

First, we give a decision procedure for DTL entailment by reducing it to entailment in LTL using a polynomial-time syntactic translation. Furthermore, we show that, under this translation, DTL is well-suited for efficient model checking. In this way, existing decision procedures for LTL, as well as other automated tools for LTL, such as [3, 8], may be used for DTL. However, while decision procedures are fine for machines, they are often ill-suited for humans. In particular, our translation-based procedure does not reflect the arguments used in natural reasoning in DTL. This is also the case for Hilbert calculi, resolution and model-checking-based approaches, as well as for unlabelled tableaux procedures based on a Fischer–Ladner style construction [13, 27, 37], even if built specifically for DTL.

In contrast, an attractive possibility is a labelled tableaux system as deductions will then naturally follow semantic arguments. This is the second option we pursue, which is the main focus and contribution of this article. We present a sound and complete labelled tableaux system for DTL. To this end, we first introduce a labelled tableaux system for LTL, where reasoning is local. Afterwards, we take one such local tableaux system for each agent and combine them with rules that capture the distributed nature of DTL, via communication. The tableaux systems for local reasoning (in LTL) are, as expected, built from formulas labelled with local state information and relations between these labels (order and equality). We integrate these systems into a system for global reasoning, where we introduce an additional relation expressing synchronization. We prove the soundness and completeness of the system with respect to DTL entailment and provide examples of its use.

The tableaux system thus obtained is natural in that it closely formalizes proofs made using semantic arguments. For example, an eventuality simply leads to a future time point. This is in contrast to a Fischer–Ladner style construction, based on the fixedpoint characterizations of the temporal operators, where an eventuality becomes a condition that must be verified over the structure of a graph. We do not address the question of efficient proof search and we include an infinite closure rule that captures eventualities that are always delayed. Building a decision procedure by including loop checking directly on top of our tableaux system does not appear to be possible. Modifying our rules for the temporal operators to introduce 'control points' needed to check for loops, by following more closely the fixedpoint properties of the operators, should be possible, but would lead to an unnatural result. We choose not to go to this route as we already have decidability and our emphasis is on naturality. To our knowledge, this is the first labelled system given for full, discrete-time LTL with the until and since operators.[1]

1.1 Organization

In Section 2, we introduce DTL. In Section 3, we present our tableaux system for local reasoning and establish its soundness and completeness with respect to entailment. In Section 4, we extend the local system into a system for global reasoning by including a new synchronization relation between local labels and we also prove soundness and completeness with respect to entailment. Afterwards, in Section 5, we present examples that illustrate the use of our tableaux system. We conclude, in Section 6, by comparing with related work and discussing future work. For examples of applications of the logic, we refer the reader to [5, 6, 11, 12].

[1] In [1], we gave a labelled system for the future-only fragment of DTL.

2 DTL

2.1 The syntax and semantics of DTL

The syntax of DTL is defined over a *distributed signature* $\Sigma = \langle Id, \{Prop_i\}_{i \in Id} \rangle$ of a system, where Id is a finite set of *agents* and, for each $i \in Id$, $Prop_i$ is a set of *local state propositions*. The *global language* $\mathcal{L}_{\mathrm{DTL}}$ is defined by the grammar

$$\mathcal{L}_{\mathrm{DTL}} ::= @_{i_1}[\mathcal{L}_{i_1}] \mid \cdots \mid @_{i_n}[\mathcal{L}_{i_n}],$$

for $Id = \{i_1, \ldots, i_n\}$. The $\mathcal{L}_i$, for each $i \in Id$, are *local languages*, defined by

$$\mathcal{L}_i ::= Prop_i \mid \neg \mathcal{L}_i \mid \mathcal{L}_i \Rightarrow \mathcal{L}_i \mid \mathcal{L}_i \mathsf{U} \mathcal{L}_i \mid \mathcal{L}_i \mathsf{S} \mathcal{L}_i \mid \copyright_j[\mathcal{L}_j],$$

with $j \in Id$. A global formula $@_i[\varphi]$ means that φ holds for agent i. Local formulas, as the name indicates, hold locally for the different agents. For instance, locally for an agent i, the operators U and S are the usual (strong) *until* and *since* temporal operators, respectively, while the *communication formula* $\copyright_j[\psi]$ means that agent i has just communicated (synchronized) with agent j, for whom ψ held.[2] We will use $\mathcal{L}_i^{\not\copyright}$ to denote the set of all purely temporal formulas of $\mathcal{L}_i$, that is, excluding communication formulas.

Other logical connectives ($\bot$, $\top$, conjunction, disjunction, etc.) and temporal operators can be defined as abbreviations. For example:

$\mathsf{X}\varphi$	$\equiv \bot \mathsf{U} \varphi$	tomorrow (next)
$\mathsf{F}\varphi$	$\equiv \top \mathsf{U} \varphi$	sometime in the future
$\mathsf{F}_\circ\varphi$	$\equiv \varphi \vee \mathsf{F}\varphi$	now or sometime in the future
$\mathsf{G}\varphi$	$\equiv \neg \mathsf{F} \neg \varphi$	always in the future
$\mathsf{G}_\circ\varphi$	$\equiv \varphi \wedge \mathsf{G}\varphi$	now and always in the future
$\varphi \mathsf{W} \psi$	$\equiv (\mathsf{G}\varphi) \vee (\varphi \mathsf{U} \psi)$	weak until (unless)
$\mathsf{Y}\varphi$	$\equiv \bot \mathsf{S} \varphi$	yesterday (previous)
$\mathsf{P}\varphi$	$\equiv \top \mathsf{S} \varphi$	sometime in the past
$\mathsf{P}_\circ\varphi$	$\equiv \varphi \vee \mathsf{P}\varphi$	now or sometime in the past
$\mathsf{H}\varphi$	$\equiv \neg \mathsf{P} \neg \varphi$	always in the past
$\mathsf{H}_\circ\varphi$	$\equiv \varphi \wedge \mathsf{H}\varphi$	now and always in the past
$\varphi \mathsf{B} \psi$	$\equiv (\mathsf{H}\varphi) \vee (\varphi \mathsf{S} \psi)$	weak since (back to)
$*$	$\equiv \mathsf{H} \bot$	in the beginning
$\varphi \gg_j \psi$	$\equiv \varphi \Rightarrow \copyright_j[\psi]$	calling

Here we use the subscript $\circ$ to denote the reflexive versions of the operators. Note also that *calling* is specific to DTL as it involves communication: $@_i[\varphi \gg_j \psi]$ means that if φ holds for agent i then he calls (synchronizes with) agent j, for whom ψ must hold.

A *local life-cycle* of agent i is a countable (finite or infinite), discrete, well-founded total order $\lambda_i = \langle E_i, \leq_i \rangle$, where E_i is the set of *local events* and $\leq_i$ the *local order of causality*. We define the corresponding *local successor relation* $\rightarrow_i \subseteq E_i \times E_i$ to be the relation such that $e \rightarrow_i e'$ if $e <_i e'$ and there is no e'' such that $e <_i e'' <_i e'$. As a consequence, we have that $\leq_i = \rightarrow_i^*$, i.e. $\leq_i$ is the reflexive and transitive closure of $\rightarrow_i$.

[2] Note that the DTL syntax here differs slightly from the original presentation in [11]. Previously, the operator $\copyright_i$ was overloaded with $@_i$ and its interpretation was therefore context dependent.

A *distributed life-cycle* is a family $\lambda = \{\lambda_i\}_{i \in Id}$ of local life-cycles such that $\leq = (\bigcup_{i \in Id} \leq_i)^*$ defines a partial order of *global causality* on the set of all events $E = \bigcup_{i \in Id} E_i$. Note that communication is modelled by event sharing and thus for some event e we may have $e \in E_i \cap E_j$, for $i \neq j$. In that case, requiring $\leq$ to be a partial order amounts to requiring that the local orders are globally compatible. This excludes the existence of another $e' \in E_i \cap E_j$, where both $e <_i e'$ and $e' <_j e$.

A *local state* of agent i is a finite set $\xi \subseteq E_i$ that is downward closed for local causality, that is, if $e \leq_i e'$ and $e' \in \xi$ then also $e \in \xi$. The set Ξ_i of all local states of an agent i is totally ordered by inclusion and has $\emptyset$ as the minimal element. In general, each non-empty local state ξ of agent i is reached, by the occurrence of an event that we call $last_i(\xi)$, from the local state $\xi \setminus \{last_i(\xi)\}$. The local states of each agent are totally ordered as a consequence of the total order on local events. Since they are discrete and well-founded, we enumerate them as follows: $\emptyset$ is the 0-th state; $\{e\}$, where e is the minimum of $\langle E_i, \leq_i \rangle$, is the first state; and, in general, if ξ is the k-th state of agent i and $last_i(\xi) \rightarrow_i e'$, then $\xi \cup \{e'\}$ is the $(k+1)$-th state of agent i. We denote by ξ_i^k the k-th state of agent i. Note that $\xi_i^0 = \emptyset$ is the initial state and ξ_i^k is the state reached from the initial state after the occurrence of the first k events. In fact, ξ_i^k is the only state of agent i that contains k elements, i.e. where $|\xi_i^k| = k$. Given $e \in E_i$, observe that $(e \downarrow i) = \{e' \in E_i \mid e' \leq_i e\}$ is always a local state. Furthermore, if ξ is non-empty, then $(last_i(\xi) \downarrow i) = \xi$.

An *interpretation structure* $\mu = \langle \lambda, \sigma \rangle$ consists of a distributed life-cycle λ and a family $\sigma = \{\sigma_i\}_{i \in Id}$ of labelling functions. For each $i \in Id$, $\sigma_i : \Xi_i \rightarrow \wp(Prop_i)$ associates a set of local state propositions to each local state. We denote $\langle \lambda_i, \sigma_i \rangle$ by μ_i and define the *global satisfaction relation* by

- $\mu \Vdash_{\mathrm{DTL}} @_i[\varphi]$ iff $\mu_i \Vdash_i \varphi$ iff $\mu_i, \xi \Vdash_i \varphi$ for every $\xi \in \Xi_i$,

where the local satisfaction relations at local states are defined by

- $\mu_i, \xi \Vdash_i p$ if $p \in \sigma_i(\xi)$;
- $\mu_i, \xi \Vdash_i \neg\varphi$ if $\mu_i, \xi \nVdash_i \varphi$;
- $\mu_i, \xi \Vdash_i \varphi \Rightarrow \psi$ if $\mu_i, \xi \nVdash_i \varphi$ or $\mu_i, \xi \Vdash_i \psi$;
- $\mu_i, \xi \Vdash_i \varphi \mathsf{U} \psi$ if $|\xi| = k$ and there exists $\xi_i^n \in \Xi_i$ such that $k < n$ with $\mu_i, \xi_i^n \Vdash_i \psi$, and $\mu_i, \xi_i^m \Vdash_i \varphi$ for every $k < m < n$;
- $\mu_i, \xi \Vdash_i \varphi \mathsf{S} \psi$ if $|\xi| = k$ and there exists $\xi_i^n \in \Xi_i$ such that $n < k$ with $\mu_i, \xi_i^n \Vdash_i \psi$, and $\mu_i, \xi_i^m \Vdash_i \varphi$ for every $n < m < k$;
- $\mu_i, \xi \Vdash_i \copyright_j[\varphi]$ if $|\xi| > 0$, $last_i(\xi) \in E_j$, and $\mu_j, (last_i(\xi) \downarrow j) \Vdash_j \varphi$.

We say that μ is a *model* of $\Gamma \subseteq \mathcal{L}_{\mathrm{DTL}}$ if μ globally satisfies every formula in Γ, and given $\delta \in \mathcal{L}_{\mathrm{DTL}}$ we say that Γ *entails* δ, written $\Gamma \vDash_{\mathrm{DTL}} \delta$, if every global model of Γ is also a model of δ. Given $\Phi \cup \{\psi\} \subseteq \mathcal{L}_i$, we write $\Phi \vDash_i \psi$ to denote the fact that every local model of Φ is also a model of ψ, or equivalently, that $\{@_i[\varphi] \mid \varphi \in \Phi\} \vDash_{\mathrm{DTL}} @_i[\psi]$.[3]

Figure 1 illustrates the notion of a distributed life-cycle, where each row comprises the local life-cycle of one agent. In particular, $E_i = \{e_1, e_4, e_5, e_8, \ldots\}$ and $\rightarrow_i$ corresponds to the arrows in i's row. We can think of the occurrence of the event e_1 as leading agent i from its initial state $\emptyset$ to the state $\{e_1\}$, and then of the occurrence of the event e_4 as leading to state $\{e_1, e_4\}$, and so on; the state-transition sequence of agent i is displayed in Figure 2. Shared events at communication points are highlighted by the dotted vertical lines. Note that the numbers annotating the events are there only for convenience since no global total order on events is in general imposed.

[3]Note that we employ a floating temporal semantics, as opposed to a semantics anchored at the initial state. This is not a restriction since we can express the local initial states using $*$.

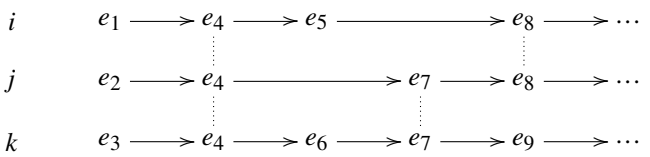

FIGURE 1. A distributed life-cycle for agents i, j and k

$$\sigma_i(\emptyset) \longrightarrow \sigma_i(\{e_1\}) \longrightarrow \sigma_i(\{e_1, e_4\}) \longrightarrow \sigma_i(\{e_1, e_4, e_5\}) \longrightarrow \cdots$$

FIGURE 2. The progress of agent i

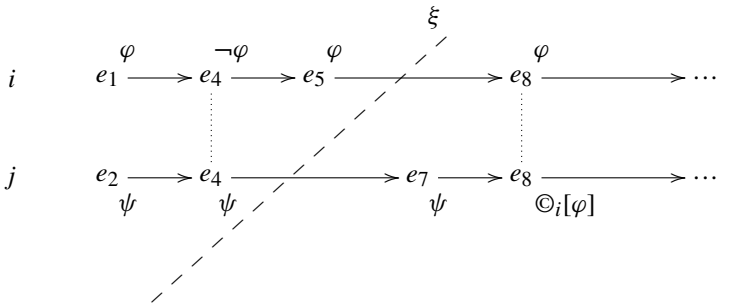

FIGURE 3. Satisfaction of formulas

Figure 3 illustrates the satisfaction relation with respect to communication formulas. Clearly, $\mu_j, \emptyset \Vdash_j \psi \mathsf{U} \copyright_i[\varphi]$, because $\mu_j, \{e_2, e_4, e_7, e_8\} \Vdash_j \copyright_i[\varphi]$ and all intermediate states of j satisfy ψ. However, $\mu_j, \{e_2, e_4\} \nVdash_j \copyright_i[\varphi]$, although $\mu_i, \{e_1, e_4, e_5\} \Vdash_i \varphi$ and $\xi = \{e_1, e_2, e_4, e_5\}$ constitutes a 'global state' compatible with the local state $\{e_1, e_4, e_5\}$ of i and $\{e_2, e_4\}$ of j. Note that global states are not necessary in this article; for more details about them see, for instance, [11].

As expected, one can extend the satisfaction relation to derived operators by using their corresponding abbreviations. In particular, the following are the satisfaction conditions for the most common temporal operators:

- $\mu_i, \xi \Vdash_i \mathsf{F}\varphi$ if $|\xi| = k$ and there exists $\xi_i^n \in \Xi_i$ such that $k < n$ with $\mu_i, \xi_i^n \Vdash_i \varphi$;
- $\mu_i, \xi \Vdash_i \mathsf{P}\varphi$ if $|\xi| = k$ and there exists $\xi_i^n \in \Xi_i$ such that $n < k$ with $\mu_i, \xi_i^n \Vdash_i \varphi$;
- $\mu_i, \xi \Vdash_i \mathsf{G}\varphi$ if $|\xi| = k$ and $\mu_i, \xi_i^n \Vdash_i \varphi$ for every $\xi_i^n \in \Xi_i$ such that $k < n$;
- $\mu_i, \xi \Vdash_i \mathsf{H}\varphi$ if $|\xi| = k$ and $\mu_i, \xi_i^n \Vdash_i \varphi$ for every $\xi_i^n \in \Xi_i$ such that $n < k$;
- $\mu_i, \xi \Vdash_i \mathsf{X}\varphi$ if $|\xi| = k$, $\xi_i^{k+1} \in \Xi_i$ exists and $\mu_i, \xi_i^{k+1} \Vdash_i \varphi$;
- $\mu_i, \xi \Vdash_i \mathsf{Y}\varphi$ if $|\xi| = k > 0$ and $\mu_i, \xi_i^{k-1} \Vdash_i \varphi$;
- $\mu_i, \xi \Vdash_i \varphi \mathsf{W} \psi$ if $|\xi| = k$ and $\mu_i, \xi_i^n \Vdash_i \varphi$ for every $\xi_i^n \in \Xi_i$ with $k < n$; or there exists $\xi_i^n \in \Xi_i$ such that $k < n$ with $\mu_i, \xi_i^n \Vdash_i \psi$, and $\mu_i, \xi_i^m \Vdash_i \varphi$ for every $k < m < n$;
- $\mu_i, \xi \Vdash_i \varphi \mathsf{B} \psi$ if $|\xi| = k$ and $\mu_i, \xi_i^n \Vdash_i \varphi$ for every $\xi_i^n \in \Xi_i$ with $n < k$; or there exists $\xi_i^n \in \Xi_i$ such that $n < k$ with $\mu_i, \xi_i^n \Vdash_i \psi$, and $\mu_i, \xi_i^m \Vdash_i \varphi$ for every $n < m < k$.

For instance, the formula $@_i[p \Rightarrow \mathsf{F}\copyright_j[\mathsf{X}q]]$ holds in a model if whenever the proposition p holds locally at a state of agent i then there must be a future state of agent i where he has just synchronized with agent j, for whom q will hold in the next state.

Note that, as is well-known, the expressive power of the set of operators $\{\mathsf{U},\mathsf{S}\}$ is identical to the set $\{\mathsf{F},\mathsf{P},\mathsf{X},\mathsf{Y},\mathsf{G},\mathsf{H},\mathsf{W},\mathsf{B}\}$ since

$$\varphi \mathsf{U} \psi \equiv (\mathsf{F}\psi) \wedge (\varphi \mathsf{W} \psi) \qquad \text{and} \qquad \varphi \mathsf{S} \psi \equiv (\mathsf{P}\psi) \wedge (\varphi \mathsf{B} \psi).$$

2.2 Decidability and trace consistency of DTL via LTL

It is not difficult to show, as suggested in [11], that DTL is decidable by a translation to LTL. An LTL signature is simply a set *Prop* of propositional symbols and the language $\mathcal{L}_{\mathrm{LTL}}$ is defined by the grammar

$$\mathcal{L}_{\mathrm{LTL}} ::= Prop \mid \neg \mathcal{L}_{\mathrm{LTL}} \mid \mathcal{L}_{\mathrm{LTL}} \Rightarrow \mathcal{L}_{\mathrm{LTL}} \mid \mathcal{L}_{\mathrm{LTL}} \mathsf{U} \mathcal{L}_{\mathrm{LTL}} \mid \mathcal{L}_{\mathrm{LTL}} \mathsf{S} \mathcal{L}_{\mathrm{LTL}}.$$

Note that, excluding communication formulas, local DTL formulas coincide with LTL formulas. That is, $\mathcal{L}_{\mathrm{LTL}} = \mathcal{L}_i^{\emptyset}$ provided that $Prop = Prop_i$. The usual interpretation structure for LTL is a map $\tau : \mathbb{N}_0 \to \wp(Prop)$, where we write $\mathbb{N}_0$ to denote the natural numbers with 0. We also use $\mathbb{N}$ to denote $\mathbb{N}_0 \setminus \{0\}$. The satisfaction of LTL formulas by τ is defined as for local DTL formulas. That is, if we define $\lambda_i = \langle E_i, \leq_i \rangle = \langle \mathbb{N}, \leq \rangle$ then we have as local states $\Xi_i = \{\xi_i^0, \xi_i^1, \xi_i^2, \xi_i^3, \ldots\} = \{\emptyset, \{1\}, \{1,2\}, \{1,2,3\}, \ldots\}$. Letting $\sigma_i(\xi_i^k) = \tau(k)$, we define $\tau, k \Vdash_{\mathrm{LTL}} \varphi$ if $\mu_i, \xi_i^k \Vdash_i \varphi$, and $\tau \Vdash_{\mathrm{LTL}} \varphi$ if $\mu_i \Vdash_i \varphi$. The entailment relation $\models_{\mathrm{LTL}}$ is defined similarly.

Given a DTL signature $\Sigma = \langle Id, Prop \rangle$, we define the corresponding LTL signature $Prop = \{@i \mid i \in Id\} \cup \biguplus_{i \in Id} Prop_i$. In the following, we assume that the element $p \in Prop_i$ is represented in *Prop* by p_i. The *translation of global formulas* is then given by the function $\alpha : \mathcal{L}_{\mathrm{DTL}} \to \mathcal{L}_{\mathrm{LTL}}$ such that

- $\alpha(@_i[\varphi]) = @i \Rightarrow \alpha_i(\varphi)$,

and for each $i \in Id$, the function $\alpha_i : \mathcal{L}_i \to \mathcal{L}_{\mathrm{LTL}}$ translates local formulas to LTL formulas as follows:

- $\alpha_i(p) = p_i$;
- $\alpha_i(\neg\varphi) = \neg\alpha_i(\varphi)$;
- $\alpha_i(\varphi \Rightarrow \psi) = \alpha_i(\varphi) \Rightarrow \alpha_i(\psi)$;
- $\alpha_i(\varphi \mathsf{U} \psi) = (@i \Rightarrow \alpha_i(\varphi)) \mathsf{U} (@i \wedge \alpha_i(\psi))$;
- $\alpha_i(\varphi \mathsf{S} \psi) = (@i \Rightarrow \alpha_i(\varphi)) \mathsf{S} (@i \wedge \alpha_i(\psi))$;
- $\alpha_i(\copyright_j[\varphi]) = @j \wedge \alpha_j(\varphi)$.

We first observe that entailment in DTL is preserved by this translation.

LEMMA 1
Let $\Gamma \cup \{\delta\} \subseteq \mathcal{L}_{\mathrm{DTL}}$. If $\Gamma \models_{\mathrm{DTL}} \delta$ then $\alpha(\Gamma) \cup \{* \Rightarrow (\bigwedge_{i \in Id} @i)\} \models_{\mathrm{LTL}} \alpha(\delta)$.

PROOF. We translate into DTL all the LTL interpretations τ that satisfy the property $(* \Rightarrow (\bigwedge_{i \in Id} @i))$, that is, τ must be such that $\{@i \mid i \in Id\} \subseteq \tau(0)$. Consider the map β from LTL interpretation structures to DTL interpretation structures such that $\beta(\tau) = \langle \lambda, \sigma \rangle$, with $\lambda_i = \langle E_i, \leq_i \rangle$, where:

- $E_i = \{n \in \mathbb{N} \mid @i \in \tau(n)\}$;
- $\leq_i$ is the restriction of the usual order on $\mathbb{N}$, with $n \to_i m$ if $n, m \in E_i$ and there is no $k \in E_i$ such that $n < k < m$;
- $\sigma_i(\emptyset) = \{p \in Prop_i \mid p_i \in \tau(0)\}$ and $\sigma_i(\{m \in E_i \mid m \leq n\}) = \{p \in Prop_i \mid p_i \in \tau(n)\}$, for each $n \in E_i$.

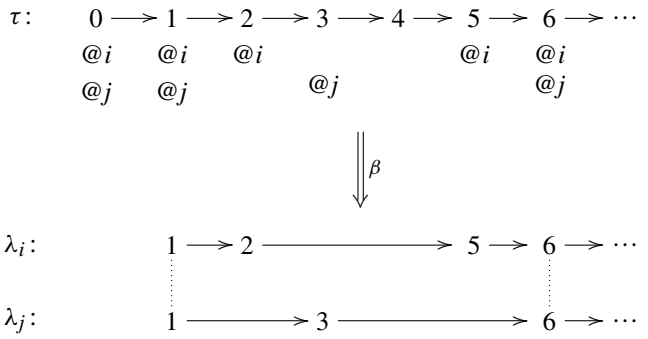

FIGURE 4. Translating τ to λ_i and λ_j

In this proof, we will assume that $last_i(\emptyset)=0$. We start by showing that for $\varphi\in\mathcal{L}_i$, $\beta(\tau)_i,\xi_i^k\Vdash_{\vdash i}\varphi$ if and only if $\tau,last_i(\xi_i^k)\Vdash_{\mathrm{LTL}}\alpha_i(\varphi)$, for every $\xi_i^k\in\Xi_i$. The proof follows by induction on φ. If φ is a propositional symbol p, then $\beta(\tau)_i,\xi_i^k\Vdash_{\vdash i}p$ iff $p\in\sigma_i(\xi_i^k)$ iff $p_i\in\tau(last_i(\xi_i^k))$ iff $\tau,last_i(\xi_i^k)\Vdash_{\mathrm{LTL}}\alpha_i(p)$. The propositional connectives are also straightforward. Assume that $\tau,last_i(\xi_i^k)\Vdash_{\mathrm{LTL}}\alpha_i(\varphi\mathsf{U}\psi)$. Then, there exists $n'>last_i(\xi_i^k)$ such that $\tau,n'\Vdash_{\mathrm{LTL}}@i\wedge\alpha_i(\psi)$. Hence $n'\in E_i$, $(n'\downarrow i)=\xi_i^n$ for some $n>k$, and so $last_i(\xi_i^n)=n'$. Therefore, by the induction hypothesis, $\beta(\tau)_i,\xi_i^n\Vdash_{\vdash i}\psi$. Moreover, $\tau,m'\Vdash_{\mathrm{LTL}}@i\Rightarrow\alpha_i(\varphi)$ for every m' such that $last_i(\xi_i^k)<m'<n'$. Given $k<m<n$, we have that $last_i(\xi_i^k)<last_i(\xi_i^m)<last_i(\xi_i^n)=n'$. Moreover, since $last_i(\xi_i^m)\in E_i$, it follows that $\tau,last_i(\xi_i^m)\Vdash_{\mathrm{LTL}}@i$. Since we have $\tau,last_i(\xi_i^m)\Vdash_{\mathrm{LTL}}\alpha_i(\varphi)$, using the induction hypothesis, we then also have that $\beta(\tau)_i,\xi_i^m\Vdash_{\vdash i}\varphi$. We can conclude that $\beta(\tau)_i,\xi_i^k\Vdash_{\vdash i}\varphi\mathsf{U}\psi$. The converse is similar, and so is the proof for S. Finally, assume that $\beta(\tau)_i,\xi_i^k\Vdash_{\vdash i}\copyright_j[\varphi]$. Then $last_i(\xi_i^k)\in E_j$ and $\beta(\tau)_i,last(\xi_i^k)\downarrow j\Vdash_{\vdash j}\varphi$. By the induction hypothesis, $\tau,last_j(last_i(\xi_i^k)\downarrow j)\Vdash_{\mathrm{LTL}}\alpha_j(\varphi)$. Furthermore, $last_j(last_i(\xi_i^k)\downarrow j)=last_i(\xi_i^k)\in E_j$ so $@j\in\tau(last_i(\xi_i^k))$, that is, $\tau,last_i(\xi_i^k)\Vdash_{\mathrm{LTL}}@j$. Hence $\tau,last_i(\xi_i^k)\Vdash_{\mathrm{LTL}}@j\wedge\alpha_j(\varphi)$, i.e, $\tau,last_i(\xi_i^k)\Vdash\alpha_i(\copyright_j[\varphi])$. Once again, the converse is similar.

Now it is straightforward to conclude that, for every $\gamma\in\mathcal{L}_{\mathrm{DTL}}$, $\beta(\tau)\Vdash_{\mathrm{DTL}}\gamma$ if and only if $\tau\Vdash_{\mathrm{LTL}}\alpha(\gamma)$. Assume that $\beta(\tau)\nVdash_{\mathrm{DTL}}@_i[\varphi]$. Then there is a ξ_i^k such that $\beta(\tau)_i,\xi_i^k\nVdash_{\vdash i}\varphi$. By the previous result, it follows that $\tau,last(\xi_i^k)\nVdash_{\mathrm{LTL}}\alpha_i(\varphi)$. We also know that $\tau,last(\xi_i^k)\Vdash_{\mathrm{LTL}}@_i$. Hence $\tau,last(\xi_i^k)\nVdash_{\mathrm{LTL}}@i\Rightarrow\alpha_i(\varphi)$, i.e. $\tau,last(\xi_i^k)\nVdash_{\mathrm{LTL}}\alpha(@_i[\varphi])$. Hence $\tau\nVdash_{\mathrm{LTL}}\alpha(@_i[\varphi])$. Conversely, assume that $\tau\nVdash_{\mathrm{LTL}}\alpha(@_i[\varphi])$. Then there is an $n\in\mathbb{N}_0$ such that $\tau,n\nVdash_{\mathrm{LTL}}@i\Rightarrow\alpha_i(\varphi)$, i.e. $\tau,n\Vdash_{\mathrm{LTL}}@i$ and $\tau,n\nVdash_{\mathrm{LTL}}\alpha_i(\varphi)$. From the first condition, it follows that either $n=0$, in which case we postulate that $(0\downarrow i)=\emptyset$, or $n\in E_i$ and so $last_i(n\downarrow i)=n$. Once again, by the previous result, it follows that $\beta(\tau)_i,n\downarrow i\nVdash_{\vdash i}\varphi$. Hence $\beta(\tau)\nVdash_{\mathrm{DTL}}@_i[\varphi]$.

The result now follows. Assume that $\Gamma\vDash_{\mathrm{DTL}}\delta$ and let τ be an LTL model satisfying $\alpha(\Gamma)\cup\{*\Rightarrow(\bigwedge_{i\in Id}@i)\}$. Then we know that $\beta(\tau)\Vdash_{\mathrm{DTL}}\Gamma$ and thus also $\beta(\tau)\Vdash_{\mathrm{DTL}}\delta$. Therefore, $\tau\Vdash_{\mathrm{LTL}}\alpha(\delta)$ and we can conclude that $\alpha(\Gamma)\cup\{*\Rightarrow(\bigwedge_{i\in Id}@i)\}\vDash_{\mathrm{LTL}}\alpha(\delta)$. ■

In Figure 4, we illustrate this translation with a simple example where the LTL interpretation τ is translated into the life-cycles λ_i and λ_j.

We now show that entailment in DTL is also reflected by the translation.

LEMMA 2
Let $\Gamma\cup\{\delta\}\subseteq\mathcal{L}_{\mathrm{DTL}}$. If $\alpha(\Gamma)\cup\{*\Rightarrow(\bigwedge_{i\in Id}@i)\}\vDash_{\mathrm{LTL}}\alpha(\delta)$ then $\Gamma\vDash_{\mathrm{DTL}}\delta$.

PROOF. We now translate interpretation structures in the opposite direction. Given a DTL interpretation structure μ it is always possible to *linearize* its underlying global order on events $\langle E, \leq \rangle$. That is, one can define an injective function $f : E \to \mathbb{N}$ that preserves the global causality relation, i.e. if $e < e'$ then $f(e) < f(e')$. We follow [2], for instance.

For each DTL interpretation structure μ and linearization f of $\langle E, \leq \rangle$, we define an associated LTL interpretation structure $\tau_{\mu,f}$ by

$$\tau_{\mu,f}(n) = \begin{cases} \{@i, p_i \mid i \in Id, p \in \sigma_i(\emptyset)\} & \text{if } n = 0, \\ \{@i, p_i \mid e \in E_i, p \in \sigma_i(e{\downarrow}i)\} & \text{if } f(e) = n, \\ \emptyset & \text{if } 0 < n \notin f(E). \end{cases}$$

Observe that, by construction, $\tau_{\mu,f}$ is a model of $(* \Rightarrow (\bigwedge_{i \in Id} @i))$.

By a simple inductive argument, similar to the one in the previous lemma, we also have that, for every $\varphi \in \mathcal{L}_i$, $\tau_{\mu,f}, f(last_i(\xi_i)) \Vdash_{\mathrm{LTL}} \alpha_i(\varphi)$ if and only if $\mu_i, \xi_i \Vdash_i \varphi$. This implies that, for every $\gamma \in \mathcal{L}_{\mathrm{DTL}}$, $\tau_{\mu,f} \Vdash_{\mathrm{LTL}} \alpha(\gamma)$ if and only if $\mu \Vdash_{\mathrm{DTL}} \gamma$.

Assume now that $\alpha(\Gamma) \cup \{* \Rightarrow (\bigwedge_{i \in Id} @i)\} \vDash_{\mathrm{LTL}} \alpha(\delta)$ and let μ be a DTL model of Γ. Then, we have that $\tau_{\mu,f} \Vdash_{\mathrm{LTL}} \alpha(\Gamma) \cup \{* \Rightarrow (\bigwedge_{i \in Id} @i)\}$ and therefore $\tau_{\mu,f} \Vdash_{\mathrm{LTL}} \alpha(\delta)$. Hence, $\mu \Vdash_{\mathrm{DTL}} \delta$ and we can conclude that $\Gamma \vDash_{\mathrm{DTL}} \delta$. ■

Putting the two previous lemmas together, we have:

COROLLARY 3
Let $\Gamma \cup \{\delta\} \subseteq \mathcal{L}_{\mathrm{DTL}}$. Then

$$\Gamma \vDash_{\mathrm{DTL}} \delta \text{ if and only if } \alpha(\Gamma) \cup \{* \Rightarrow (\bigwedge_{i \in Id} @i)\} \vDash_{\mathrm{LTL}} \alpha(\delta).$$

As a consequence, since LTL is decidable (see [3], for instance), any decision procedure for LTL can also be used for DTL. The asymptotic complexity is identical since our syntactic translation function α is polynomial. The result is actually independent of the chosen linearization function f and in general there may be many such functions. This means that DTL is *trace-consistent* in the precise sense of [34]. Namely, DTL properties can be checked by considering one arbitrary linearization of the distributed model, as opposed to checking all possible linearizations. This fact makes DTL properties particularly well suited for efficient model checking using partial-order reduction techniques [25], which has been explored in [12].

3 Tableaux for local reasoning

3.1 The local tableaux system

We first present a labelled tableaux system for reasoning locally at each agent. This essentially amounts to a labelled tableaux system for full discrete LTL with the until and since operators, which is, to our knowledge, a novelty. As defined in the previous section, we can use the set $\{\mathsf{U}, \mathsf{S}\}$ as a complete set of operators for our logic. However, for simplicity and readability of the tableaux rules of our system, we will instead take the operators $\mathsf{F}, \mathsf{P}, \mathsf{G}, \mathsf{H}, \mathsf{X}, \mathsf{Y}, \mathsf{W}$ and B as primitive. In this context, as noted above, the strong versions of until and since can be seen as derived operators.

From now on, we consider fixed a distributed signature Σ. Our tableaux for local reasoning will handle four kinds of *local judgements* for each agent $i \in Id$: labelled local formulas (excluding communication), equality between labels, inequality between labels, and a special judgement

indicating absurdity. Labels will denote the local states of agents. To define the language of labels, for the given signature Σ, we assume fixed a family $\mathcal{V}=\{\mathcal{V}_i\}_{i\in Id}$ of sets of *label variables* and also use a family $\mathcal{F}=\{\mathcal{F}_i\}_{i\in Id}$ of sets of *Skolem function symbols* defined as follows:

$$\mathcal{F}_i = \{\mathsf{f}_{\varphi \mathsf{W} \psi} \mid \varphi,\psi\in\mathcal{L}_i^{\emptyset}\}\cup\{\mathsf{f}_{\neg(\varphi \mathsf{W} \psi)} \mid \varphi,\psi\in\mathcal{L}_i^{\emptyset}\}\cup$$

$$\{\mathsf{f}_{\varphi \mathsf{B} \psi} \mid \varphi,\psi\in\mathcal{L}_i^{\emptyset}\}\cup\{\mathsf{f}_{\neg(\varphi \mathsf{B} \psi)} \mid \varphi,\psi\in\mathcal{L}_i^{\emptyset}\}.$$

The syntax of *local labels* of agent $i\in Id$ is defined by

$$\mathcal{T}_i ::= \mathbb{N}_0 \mid \mathcal{V}_i+\mathbb{Z} \mid \mathcal{F}_i(\mathcal{T}_i)+\mathbb{Z},$$

$$\mathcal{S}_i ::= (i,\mathcal{T}_i).$$

Labels involving the Skolem function symbols will be used in the tableaux to guarantee the existence of certain local states associated with the satisfaction of formulas involving the weak until and since operators. Although the use of fresh variables suffices in some cases, until and since, as well as their negations, may all require the existence of states in the model with specific properties. This fact makes the use of the Skolem functions an essential ingredient of our system. We write v to denote an arbitrary label variable, x, y and z to denote arbitrary label terms, and s_i to denote an arbitrary element of $\mathcal{S}_i$. We abbreviate $x+0$ as x. Moreover, for $c\in\mathbb{N}$, we write $x-c$ instead of $x+(-c)$, as usual.

The syntax of *local judgments* for each agent i can now be defined by

$$\mathcal{J}_i ::= \mathcal{S}_i : \mathcal{L}_i^{\emptyset} \mid \mathcal{S}_i = \mathcal{S}_i \mid \mathcal{S}_i < \mathcal{S}_i \mid \textsc{closed}.$$

When convenient, we write $s_i < s_i' < s_i''$ instead of $s_i < s_i'$ and $s_i' < s_i''$. The intended meaning of a labelled formula $(i,x):\varphi$ is that φ holds at the local state (denoted by) x of agent i. Equalities and inequalities of local labels of agent i are interpreted directly over the causality ordering. To make this formal, we extend our notion of interpretation structure with information concerning labels. We will interpret labels as natural numbers in such a way that the interpretation of a given local label identifies, by its value, the local state of the corresponding agent. An *assignment on label variables* is a family $\rho=\{\rho_i\}_{i\in Id}$ of functions $\rho_i:\mathcal{V}_i\to\mathbb{N}_0$. We also need to consider a fixed interpretation structure μ. The *denotation of labels* over μ and ρ, for each agent $i\in I$, in symbols $[\![\cdot]\!]_{\mu,\rho}:\mathcal{S}_i\to\mathbb{N}_0$, is then defined as the following partial function

- $[\![(i,k)]\!]_{\mu,\rho}=k$;
- $[\![(i,v)]\!]_{\mu,\rho}=\rho_i(v)$;
- $[\![(i,\mathsf{f}_{\varphi \mathsf{W} \psi}(x))]\!]_{\mu,\rho}=n$ provided that
  - $[\![(i,x)]\!]_{\mu,\rho}$ is defined;
  - $n>[\![(i,x)]\!]_{\mu,\rho}$ is the least number, if it exists, such that
    - $\xi_i^n\in\Xi_i$ and $\mu_i,\xi_i^n\Vdash_i\psi$;
    - $\mu_i,\xi_i^k\Vdash_i\varphi$, for every k such that $[\![(i,x)]\!]_{\mu,\rho}<k<n$;
- $[\![(i,\mathsf{f}_{\neg(\varphi \mathsf{W} \psi)}(x))]\!]_{\mu,\rho}=n$, provided that
  - $[\![(i,x)]\!]_{\mu,\rho}$ is defined;
  - $n>[\![(i,x)]\!]_{\mu,\rho}$ is the least number, if it exists, such that
    - $\xi_i^n\in\Xi_i$, $\mu_i,\xi_i^n\nVdash_i\varphi$ and $\mu_i,\xi_i^n\nVdash_i\psi$;

  * $\mu_i, \xi_i^k \not\Vdash_i \psi$, for every k such that $[\![(i,x)]\!]_{\mu,\rho} < k < n$;
- $[\![(i, \mathsf{f}_{\varphi \mathsf{B} \psi}(x))]\!]_{\mu,\rho} = n$, provided that
  – $[\![(i,x)]\!]_{\mu,\rho}$ is defined;
  – $n < [\![(i,x)]\!]_{\mu,\rho}$ is the greatest number, if it exists, such that
    * $\xi_i^n \in \Xi_i$ and $\mu_i, \xi_i^n \Vdash_i \psi$;
    * $\mu_i, \xi_i^k \Vdash_i \varphi$, for every k such that $n < k < [\![(i,x)]\!]_{\mu,\rho}$;
- $[\![(i, \mathsf{f}_{\neg(\varphi \mathsf{B} \psi)}(x))]\!]_{\mu,\rho} = n$, provided that
  – $[\![(i,x)]\!]_{\mu,\rho}$ is defined;
  – $n < [\![(i,x)]\!]_{\mu,\rho}$ is the greatest number, if it exists, such that
    * $\xi_i^n \in \Xi_i$, $\mu_i, \xi_i^n \not\Vdash_i \varphi$ and $\mu_i, \xi_i^n \not\Vdash_i \psi$;
    * $\mu_i, \xi_i^k \not\Vdash_i \psi$, for every k such that $n < k < [\![(i,x)]\!]_{\mu,\rho}$.
- $[\![(i, x+k)]\!]_{\mu,\rho} = [\![(i,x)]\!]_{\mu,\rho} + k$, provided that $[\![(i,x)]\!]_{\mu,\rho}$ is defined and $[\![(i,x)]\!]_{\mu,\rho} + k \geq 0$.

For simplicity, when $[\![(i,x)]\!]_{\mu,\rho}$ depends only on ρ_i, we write $\rho_i(x)$.

One reason why the denotation of labels is partial is that we do not consider negative values. This is unproblematic as the labels appearing in our tableaux will always denote non-negative values. A second reason for the partiality is due to the interpretation of the Skolem functions. The interpretation of the function symbols for negated until and since, that is $\mathsf{f}_{\neg(\varphi \mathsf{W} \psi)}$ and $\mathsf{f}_{\neg(\varphi \mathsf{B} \psi)}$, is defined depending on the satisfaction of the corresponding formulas $\neg(\varphi \mathsf{W} \psi)$ and $\neg(\varphi \mathsf{B} \psi)$, in which case the interpretations will have the value of the first state in the future, or respectively in the past, where φ does not hold. The interpretation of the function symbols for until and since, that is $\mathsf{f}_{\varphi \mathsf{W} \psi}$ and $\mathsf{f}_{\varphi \mathsf{B} \psi}$, do not mimic the satisfaction of the corresponding formulas so closely. Actually, it is enough for our purposes that they are only defined under the assumption that φ does not hold forever (in the future or in the past, respectively). In this case, their interpretations will take the value of the first state where ψ holds. In any case, the relevant labels of this form appearing in our tableaux will always arise in contexts where their denotation is defined.

We can now define the *satisfaction of local judgements* of agent i at μ, given an assignment ρ:

- $\mu, \rho \Vdash s_i : \varphi$ if $[\![s_i]\!]_{\mu,\rho}$ is defined, $\xi_i^{[\![s_i]\!]_{\mu,\rho}} \in \Xi_i$, and $\mu_i, \xi_i^{[\![s_i]\!]_{\mu,\rho}} \Vdash_i \varphi$;
- $\mu, \rho \Vdash s_i = s_i'$ if $[\![s_i]\!]_{\mu,\rho}$ and $[\![s_i']\!]_{\mu,\rho}$ are both defined and $[\![s_i]\!]_{\mu,\rho} = [\![s_i']\!]_{\mu,\rho}$;
- $\mu, \rho \Vdash s_i < s_i'$ if $[\![s_i]\!]_{\mu,\rho}$ and $[\![s_i']\!]_{\mu,\rho}$ are both defined and $[\![s_i]\!]_{\mu,\rho} < [\![s_i']\!]_{\mu,\rho}$;
- $\mu, \rho \not\Vdash$ CLOSED.

Recall that $\xi_i^{[\![s_i]\!]_{\mu,\rho}}$ denotes the $[\![s_i]\!]^{\text{th}}_{\mu,\rho}$ local state of agent i in μ. We can finally define our tableaux for local reasoning.

DEFINITION 4
The *local tableaux system* $\mathcal{T}_i$ for agent $i \in Id$, built over sets of local judgements in $\mathcal{J}_i$, consists of the rules shown in Figures 5–8.

We assume that the reader is familiar with standard terminology and notation for tableaux, for example from [10]. As usual, a branch of a (possibly infinite) tableau is

- *exhausted*, if no more rules are applicable,
- *closed*, if it contains CLOSED and
- *open*, if it is exhausted but not closed.

A tableau is *closed* if all of its branches are closed. Moreover, any tableau whose root is labelled by a given set of judgements Θ will be called a *tableau for* Θ. Note that we will assume that Θ contains no

$$\frac{s_i : \neg\neg\varphi}{s_i : \varphi}\ (\neg\neg) \qquad \frac{s_i : \varphi \quad s_i : \neg\varphi}{\text{CLOSED}}\ (\text{ABS}) \qquad \frac{s_i : \varphi \Rightarrow \psi}{s_i : \neg\varphi \mid s_i : \psi}\ (\Rightarrow) \qquad \frac{s_i : \neg(\varphi \Rightarrow \psi)}{s_i : \varphi\,,\, s_i : \neg\psi}\ (\neg\Rightarrow)$$

FIGURE 5. Rules for the logical connectives

$$\frac{(i,x) : \mathsf{F}\,\varphi}{(i,x) < (i,v)\,,\ (i,v) : \varphi}\ (\mathsf{F})\ [v \text{ fresh}] \qquad \frac{(i,x) : \neg\mathsf{F}\,\varphi \quad (i,x) < (i,y) \quad (i,y) : \psi}{(i,y) : \neg\varphi}\ (\neg\mathsf{F})$$

$$\frac{(i,x) : \mathsf{P}\,\varphi}{(i,v) < (i,x)\,,\ (i,v) : \varphi}\ (\mathsf{P})\ [v \text{ fresh}] \qquad \frac{(i,x) : \neg\mathsf{P}\,\varphi \qquad (i,y) < (i,x)}{(i,y) : \neg\varphi}\ (\neg\mathsf{P})$$

$$\frac{(i,x) : \mathsf{G}\,\varphi \quad (i,x) < (i,y) \quad (i,y) : \psi}{(i,y) : \varphi}\ (\mathsf{G}) \qquad \frac{(i,x) : \neg\mathsf{G}\,\varphi}{(i,x) < (i,v)\,,\ (i,v) : \neg\varphi}\ (\neg\mathsf{G})\ [v \text{ fresh}]$$

$$\frac{(i,x) : \mathsf{H}\,\varphi \quad (i,y) < (i,x)}{(i,y) : \varphi}\ (\mathsf{H}) \qquad \frac{(i,x) : \neg\mathsf{H}\,\varphi}{(i,v) < (i,x)\,,\ (i,v) : \neg\varphi}\ (\neg\mathsf{H})\ [v \text{ fresh}]$$

$$\frac{(i,x) : \mathsf{X}\,\varphi}{(i,x+1) : \varphi}\ (\mathsf{X}) \qquad \frac{(i,x) : \neg\mathsf{X}\,\varphi \quad (i,x) < (i,y) \quad (i,y) : \psi}{(i,x+1) : \neg\varphi}\ (\neg\mathsf{X})$$

$$\frac{(i,x) : \mathsf{Y}\,\varphi}{(i,x-1) : \varphi}\ (\mathsf{Y}) \qquad \frac{(i,x) : \neg\mathsf{Y}\,\varphi \quad (i,0) < (i,x)}{(i,x-1) : \neg\varphi}\ (\neg\mathsf{Y})$$

$$\frac{(i,x) : \varphi\,\mathsf{W}\,\psi}{(i,x) : \mathsf{G}\,\varphi \mid (i,x) < (i,\mathsf{f}_{\varphi\mathsf{W}\psi}(x))\,,\ (i,\mathsf{f}_{\varphi\mathsf{W}\psi}(x)) : \psi}\ (\mathsf{W}_1) \qquad \frac{(i,x) < s_i < (i,\mathsf{f}_{\varphi\mathsf{W}\psi}(x))}{s_i : \varphi\,,\ s_i : \neg\psi}\ (\mathsf{W}_2)$$

$$\frac{(i,x) : \neg(\varphi\,\mathsf{W}\,\psi)}{(i,x) < (i,\mathsf{f}_{\neg(\varphi\mathsf{W}\psi)}(x))\,,\ (i,\mathsf{f}_{\neg(\varphi\mathsf{W}\psi)}(x)) : \neg\varphi\,,\ (i,\mathsf{f}_{\neg(\varphi\mathsf{W}\psi)}(x)) : \neg\psi}\ (\neg\mathsf{W}_1)$$

$$\frac{(i,x) < s_i < (i,\mathsf{f}_{\neg(\varphi\mathsf{W}\psi)}(x))}{s_i : \neg\psi\,,\ s_i : \varphi}\ (\neg\mathsf{W}_2)$$

$$\frac{(i,x) : \varphi\,\mathsf{B}\,\psi}{(i,x) : \mathsf{H}\,\varphi \mid (i,\mathsf{f}_{\varphi\mathsf{B}\psi}(x)) < (i,x)\,,\ (i,\mathsf{f}_{\varphi\mathsf{B}\psi}(x)) : \psi}\ (\mathsf{B}_1) \qquad \frac{(i,\mathsf{f}_{\varphi\mathsf{B}\psi}(x)) < s_i < (i,x)}{s_i : \varphi\,,\ s_i : \neg\psi}\ (\mathsf{B}_2)$$

$$\frac{(i,x) : \neg(\varphi\,\mathsf{B}\,\psi)}{(i,\mathsf{f}_{\neg(\varphi\mathsf{B}\psi)}(x)) < (i,x)\,,\ (i,\mathsf{f}_{\neg(\varphi\mathsf{B}\psi)}(x)) : \neg\varphi\,,\ (i,\mathsf{f}_{\neg(\varphi\mathsf{B}\psi)}(x)) : \neg\psi}\ (\neg\mathsf{B}_1)$$

$$\frac{(i,\mathsf{f}_{\neg(\varphi\mathsf{B}\psi)}(x)) < s_i < (i,x)}{s_i : \neg\psi\,,\ s_i : \varphi}\ (\neg\mathsf{B}_2)$$

FIGURE 6. Rules for the temporal operators

Skolem function symbols, since these are meant to be used only as an internal device of the tableaux system during proof construction.

The rules for the logical connectives in Figure 5 are straightforward. Figure 6 contains, in turn, the rules for the temporal operators. Most of them are standard and simple to read. For instance, the rule (F) guarantees that in order for $\mathsf{F}\varphi$ to hold at state x, there must exist a future state v where φ holds. In contrast, the rule ($\neg\mathsf{F}$) concludes that if $\neg\mathsf{F}\varphi$ holds at state x, then φ cannot hold in any state y in the future of x. The additional premise $(i,y):\psi$ is there to control the introduction of labelled formulas. Mutatis mutandis, for the past, the same explanations apply to the rules (P) and ($\neg\mathsf{P}$). The rules (G), ($\neg\mathsf{G}$), (H) and ($\neg\mathsf{H}$) are justified similarly. The rules (X) and (Y) simply require the existence

$$\frac{\theta(i,x)}{(i,x)=(i,0) \mid (i,0)<(i,x)}\ (\textsc{Pos})$$

$$\frac{(i,x)=(i,y) \quad \theta(i,x)}{\theta(i,y)}\ (\textsc{Cong}) \qquad \frac{\theta(i,x)}{(i,x)=(i,x)}\ (\textsc{Refl})$$

$$\frac{s_i:\varphi \quad s_i'<s_i}{s_i':\top}\ (\textsc{Fill}) \qquad \frac{s_i:\varphi \quad s_i':\neg\varphi}{s_i<s_i' \mid s_i'<s_i}\ (\textsc{Dif})$$

$$\frac{(i,x)<(i,y) \quad \theta(i,y+c)}{(i,x)<(i,y+c)}\ (\textsc{Mon})\ [c>0] \qquad \frac{(i,x)<(i,y)<(i,z)}{(i,x)<(i,z-1)}\ (\textsc{DTrans})$$

$$\frac{\theta(i,x+1)}{(i,x)<(i,x+1)}\ (\textsc{Succ}) \qquad \frac{(i,0)<(i,x)}{(i,x-1)<(i,x)}\ (\textsc{Pred})$$

$$\frac{(i,x)<(i,y) \quad \theta(i,y+c)}{(i,x+c)<(i,y+c)}\ (\textsc{RShift})\ [c>0] \qquad \frac{(i,x)<(i,y) \quad \theta(i,x+c)}{(i,x+c)<(i,y+c)}\ (\textsc{LShift})\ [c<0]$$

$$\frac{(i,x)<(i,x+c)}{\text{CLOSED}}\ (\textsc{NLoop})\ [c\leq 0] \qquad \frac{(i,x+c)<(i,y) \qquad \exists^{\infty} c\geq 0}{\text{CLOSED}}\ (\textsc{Inf})$$

$$\frac{(i,x+c)=(i,x+c')}{\text{CLOSED}}\ (\textsc{Arith})\ [c\neq c']$$

FIGURE 7. Rules for the relations

of a suitable next or previous state, respectively. The rules ($\neg$X) and ($\neg$Y) follow a pattern similar to the ones above. Note, however, that the rules for the past-directed operators are not completely symmetric with respect to their future-directed counterparts. This is because our models always have an initial state, but may or may not be infinite to the future.

The rules for weak until and weak since follow closely the operators' semantics. However, some explanation is needed in order to clarify the use of the Skolem function symbols. The rule (W1) splits the satisfaction of $\varphi \mathsf{W} \psi$ at state x into two cases: either φ holds always in the future, or there is a future state $\mathsf{f}_{\varphi \mathsf{W} \psi}(x)$ where ψ holds. Of course this future state, which we have required to be the earliest possible, defines together with x an interval where φ must hold. These requirements are then imposed by the rule (W2), hence justifying the use of the Skolem function $\mathsf{f}_{\varphi \mathsf{W} \psi}$. The rules for negated until ($\neg$W1) and ($\neg$W2) are similar. The same applies, symmetrically, to the rules (B1), (B2), ($\neg$B1) and ($\neg$B2).

The rules in Figure 7 define the properties of the relations. Note that we use $\theta(i,x)$ to denote any local judgement of agent i where x occurs as a subterm. The rule (Pos) states that the values of the labels are either 0 or greater than 0. The rule (Cong) expresses the congruence of $=$, that is, if two labels (i,x) and (i,y) denote the same local state, then we may replace some occurrences of x by occurrences of y in any judgement. Similarly, the rule (Refl) asserts the reflexivity of equality. With the rule (Fill), we 'fill down' the set of states: if (i,x) denotes a state and if (i,y) is smaller than (i,x), then it should also denote a state (which we express by having truth hold there). With the rule (Dif), we force the labels of judgements containing contradictory formulas to be distinct. The rule (Mon) is a form of transitivity, given that y precedes $y+c$ when $c>0$. (DTrans) is discrete transitivity: if (i,x) is smaller than (i,y) and (i,y) is smaller than (i,z), then (i,x) is also smaller than (i,z). In fact, our rule is more specific and formalizes that (i,x) is actually smaller than $(i,z-1)$. The rules (Succ)

$$\frac{s_i : \neg \mathsf{X}\varphi \quad s_i' < s_i''}{s_i < s_i' \mid s_i = s_i' \mid s_i' < s_i} \ (\textsc{TrX})$$

$$\frac{s_i : \neg \mathsf{F}\varphi \quad s_i' < s_i''}{s_i < s_i' \mid s_i = s_i' \mid s_i' < s_i} \ (\textsc{TrF}) \qquad \frac{s_i : \neg \mathsf{P}\varphi \quad s_i'' < s_i'}{s_i < s_i' \mid s_i = s_i' \mid s_i' < s_i} \ (\textsc{TrP})$$

$$\frac{s_i : \neg \mathsf{G}\varphi \quad s_i' < s_i''}{s_i < s_i' \mid s_i = s_i' \mid s_i' < s_i} \ (\textsc{TrG}) \qquad \frac{s_i : \neg \mathsf{H}\varphi \quad s_i'' < s_i'}{s_i < s_i' \mid s_i = s_i' \mid s_i' < s_i} \ (\textsc{TrH})$$

$$\frac{(i,x) : \varphi \,\mathsf{W}\, \psi \quad s_i < s_i'}{(i,x) < s_i \mid (i,x) = s_i \mid s_i < (i,x)} \ (\textsc{TrW}_1)$$

$$\frac{(i,x) : \varphi \,\mathsf{W}\, \psi \quad s_i < s_i'}{s_i' < (i, \mathsf{f}_{\varphi \mathsf{W} \psi}(x)) \mid s_i' = (i, \mathsf{f}_{\varphi \mathsf{W} \psi}(x)) \mid (i, \mathsf{f}_{\varphi \mathsf{W} \psi}(x)) < s_i'} \ (\textsc{TrW}_2)$$

$$\frac{(i,x) : \neg(\varphi \,\mathsf{W}\, \psi) \quad s_i < s_i'}{(i,x) < s_i \mid (i,x) = s_i \mid s_i < (i,x)} \ (\textsc{Tr}\neg\textsc{W}_1)$$

$$\frac{(i,x) : \neg(\varphi \,\mathsf{W}\, \psi) \quad s_i < s_i'}{s_i' < (i, \mathsf{f}_{\neg(\varphi \mathsf{W} \psi)}(x)) \mid s_i' = (i, \mathsf{f}_{\neg(\varphi \mathsf{W} \psi)}(x)) \mid (i, \mathsf{f}_{\neg(\varphi \mathsf{W} \psi)}(x)) < s_i'} \ (\textsc{Tr}\neg\textsc{W}_2)$$

$$\frac{(i,x) : \varphi \,\mathsf{B}\, \psi \quad s_i' < s_i}{(i,x) < s_i \mid (i,x) = s_i \mid s_i < (i,x)} \ (\textsc{TrB}_1)$$

$$\frac{(i,x) : \varphi \,\mathsf{B}\, \psi \quad s_i' < s_i}{s_i' < (i, \mathsf{f}_{\varphi \mathsf{B} \psi}(x)) \mid s_i' = (i, \mathsf{f}_{\varphi \mathsf{B} \psi}(x)) \mid (i, \mathsf{f}_{\varphi \mathsf{B} \psi}(x)) < s_i'} \ (\textsc{TrB}_2)$$

$$\frac{(i,x) : \neg(\varphi \,\mathsf{B}\, \psi) \quad s_i' < s_i}{(i,x) < s_i \mid (i,x) = s_i \mid s_i < (i,x)} \ (\textsc{Tr}\neg\textsc{B}_1)$$

$$\frac{(i,x) : \neg(\varphi \,\mathsf{B}\, \psi) \quad s_i' < s_i}{s_i' < (i, \mathsf{f}_{\neg(\varphi \mathsf{B} \psi)}(x)) \mid s_i' = (i, \mathsf{f}_{\neg(\varphi \mathsf{B} \psi)}(x)) \mid (i, \mathsf{f}_{\neg(\varphi \mathsf{B} \psi)}(x)) < s_i'} \ (\textsc{Tr}\neg\textsc{B}_2)$$

FIGURE 8. Rules for trichotomy

and (PRED) order successive states, under appropriate conditions. (RSHIFT) and (LSHIFT) shift the precedence order along with addition, taking care that no new states are introduced. The closure rule (NLOOP) states that x cannot precede $x+c$ when $c \leq 0$. The rule (INF) is an infinitary closure rule: if in a branch there are infinitely many, distinct, non-negative constants that when added to (i,x) denote a value smaller than (i,y), then that branch is closed. Finally, the closure rule (ARITH) expresses the fact that distinct arithmetic constants cannot be equal.

The rules in Figure 8 introduce controlled forms of trichotomy for the local order relations. Note that they could all be replaced with one single rule expressing (full) trichotomy between any two existing labels, namely

$$\frac{\theta(i,x) \quad \theta(i,y)}{(i,x) < (i,y) \mid (i,x) = (i,y) \mid (i,y) < (i,x)} \ (\textsc{Tr}).$$

However, as this would increase branching in the tableaux, we opted for more controlled forms, where we only use trichotomy when it is strictly necessary.

Note that the rules are not independent. For instance, one may obtain the rule (ABS) using (DIF) and (NLOOP). The rule (NLOOP) can also be obtained from (INF) by infinitely many applications of (RSHIFT) and (DTRANS). The rules (SUCC) and (ARITH) are interderivable using the other rules of the system. Also the rules for G and H can be obtained from their corresponding abbreviations, using F and P, respectively.

We illustrate the use of the tableaux system with several examples.

EXAMPLE 5
We prove that the following formula is a theorem:

$$((\varphi \mathsf{W} \psi) \wedge \mathsf{X}(\neg \psi)) \Rightarrow \mathsf{X}\varphi.$$

A closed tableau for the negation of this formula is depicted in Figure 9.

EXAMPLE 6
Globally, the formula $\varphi \Rightarrow \mathsf{X}\varphi$ states that whenever φ holds in a state, then it will also hold in the next state. The usual induction schema for LTL guarantees that $\varphi \Rightarrow \mathsf{G}\varphi$ follows. This is confirmed by the closed $\mathcal{T}_i$-tableau for $\{(i,0) : \mathsf{G}_\circ(\varphi \Rightarrow \mathsf{X}\varphi), (i,v) : \neg(\varphi \Rightarrow \mathsf{G}\varphi)\}$ depicted in Figure 10. Note that we write $\wedge : \mathsf{G}_\circ$ to abbreviate the unfolding of the definition of $\mathsf{G}_\circ$ and the split of the two conjuncts. Note also the dotted line labelled with (INF) in the rightmost branch of the tableau abbreviates an infinite branch built systematically to obtain an infinite ascending chain $(i,v), (i,v+1), (i,v+2), \ldots$ below (i,v'). Finally, note that we systematically use boxes to avoid repeating sub-tableaux in the figures. For instance, the label **T1** on the left stands for the sub-tableau enclosed in the box called **T1** on the right.

3.2 Soundness

We now proceed to establish the soundness and completeness of our tableaux system $\mathcal{T}_i$. We first prove soundness where, as usual, a rule is *sound* if every structure and assignment that satisfies its premises also satisfies at least one of its conclusions, modulo a free choice for fresh variables. Of course, a closure rule, that is, a rule whose conclusion is CLOSED, is sound if no model satisfies its premises.

PROPOSITION 7
The rules of $\mathcal{T}_i$ are sound.

PROOF. Let μ be an arbitrary model and ρ an assignment. The rules for the logical connectives are straightforward. For example:

($\neg\neg$): If $\mu, \rho \Vdash s_i : \neg\neg\varphi$, then $\mu_i, \xi_i^{[\![s_i]\!]_{\mu,\rho}} \Vdash_i \neg\neg\varphi$, which implies that $\mu_i, \xi_i^{[\![s_i]\!]_{\mu,\rho}} \Vdash_i \varphi$ and so $\mu, \rho \Vdash s_i : \varphi$.

The proofs for the rules for the other connectives are similar. Let us consider now the rules for the temporal operators. Given the symmetry between past and future and the duality of some operators (like F and G), we present only the proof for some of the rules.

(F): Assume that $\mu, \rho \Vdash (i,x) : \mathsf{F}\varphi$. Then $\mu_i, \xi_i^{[\![(i,x)]\!]_{\mu,\rho}} \Vdash_i \mathsf{F}\varphi$. This implies that there exists $\xi_i^k \in \Xi_i$, with $[\![(i,x)]\!]_{\mu,\rho} < k$ such that $\mu_i, \xi_i^k \Vdash_i \varphi$. As v is fresh, we can assume that $\rho_i(v) = k$. Hence we have $[\![(i,v)]\!]_{\mu,\rho} = k$, $\mu, \rho \Vdash (i,x) < (i,v)$ and $\mu, \rho \Vdash (i,v) : \varphi$.

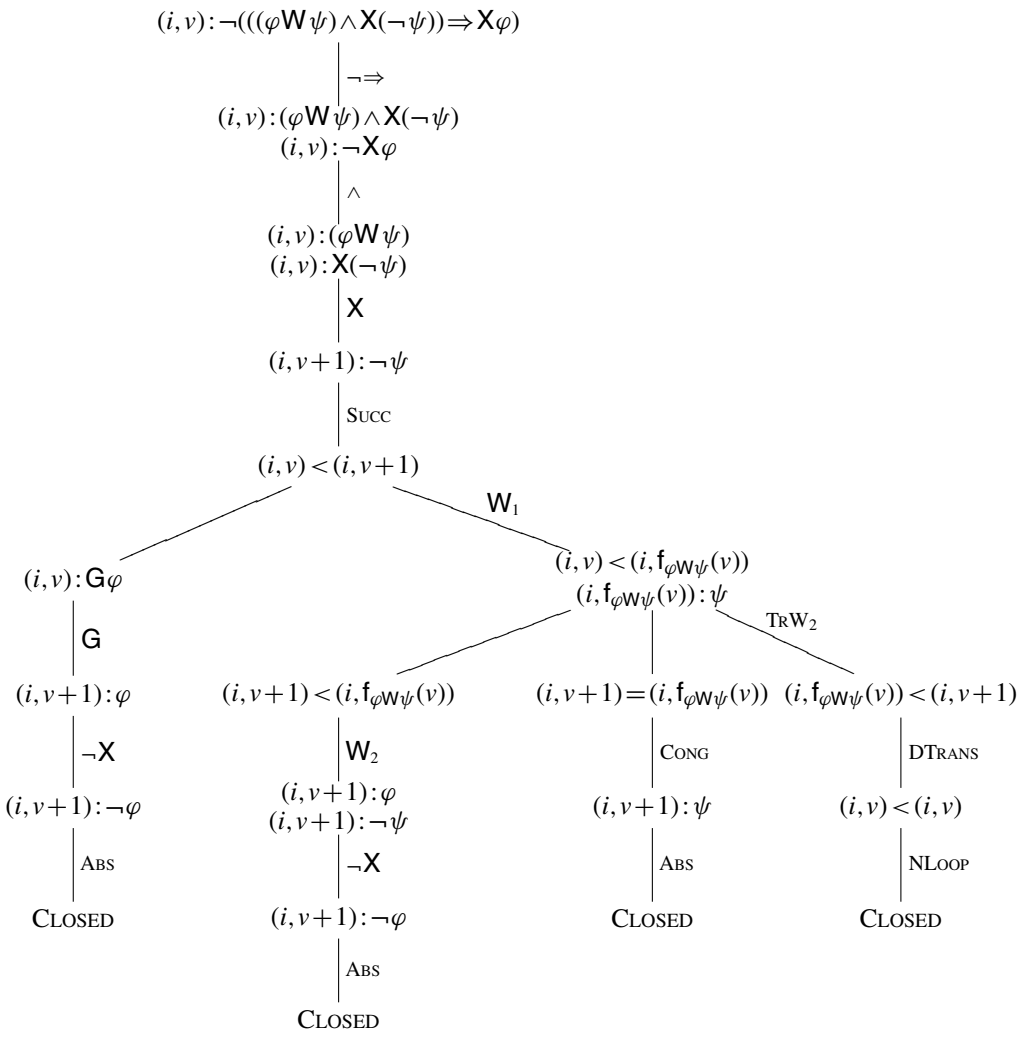

FIGURE 9. Tableau for $\neg(((\varphi \mathsf{W} \psi)\wedge \mathsf{X}(\neg\psi))\Rightarrow \mathsf{X}\varphi)$

(H): Assume that $\mu,\rho \Vdash (i,x):\mathsf{H}\varphi$ and $\mu,\rho \Vdash (i,y)<(i,x)$. Then $\mu_i,\xi_i^{[\![(i,x)]\!]_{\mu,\rho}} \Vdash_i \mathsf{H}\varphi$ and, as $[\![(i,y)]\!]_{\mu,\rho}<[\![(i,x)]\!]_{\mu,\rho}$, then $\mu_i,\xi_i^{[\![(i,y)]\!]_{\mu,\rho}} \Vdash_i \varphi$. That is, $\mu,\rho \Vdash (i,y):\varphi$.

(X): Assume that $\mu,\rho \Vdash (i,x):\mathsf{X}\varphi$. Then $\mu_i,\xi_i^{[\![(i,x)]\!]_{\mu,\rho}} \Vdash_i \mathsf{X}\varphi$, which implies that $\mu_i,\xi_i^{[\![(i,x)]\!]_{\mu,\rho}+1} \Vdash_i \varphi$. But $[\![(i,x)]\!]_{\mu,\rho}+1=[\![(i,x+1)]\!]_{\mu,\rho}$ and hence $\mu,\rho \Vdash (i,x+1):\varphi$.

($\neg\mathsf{Y}$): If $\mu,\rho \Vdash (i,0)<(i,x)$, that is $0=[\![(i,0)]\!]_{\mu,\rho}<[\![(i,x)]\!]_{\mu,\rho}$, then $[\![(i,x-1)]\!]_{\mu,\rho}=[\![(i,x)]\!]_{\mu,\rho}-1$ and the local state $\xi_i^{[\![(i,x-1)]\!]_{\mu,\rho}}$ exists. Furthermore, if $\mu,\rho \Vdash (i,x):\neg\mathsf{Y}\varphi$, then it must be the case that $\mu,\rho \Vdash (i,x-1):\neg\varphi$.

(W_1): Assume that $\mu,\rho \Vdash (i,x):\varphi \mathsf{W} \psi$. Then, $\mu_i,\xi_i^{[\![(i,x)]\!]_{\mu,\rho}} \Vdash_i \varphi \mathsf{W} \psi$ and either (1) $\mu_i,\xi_i^n \Vdash_i \varphi$ for every $\xi_i^n \in \Xi_i$ such that $[\![(i,x)]\!]_{\mu,\rho}<n$ or (2) there exists $\xi_i^n \in \Xi_i$ such that $[\![(i,x)]\!]_{\mu,\rho}<n$ with $\mu_i,\xi_i^n \Vdash_i \psi$ and $\mu_i,\xi_i^m \Vdash_i \varphi$ for every $[\![(i,x)]\!]_{\mu,\rho}<m<n$. In the first case, $\mu_i,\xi_i^{[\![(i,x)]\!]_{\mu,\rho}} \Vdash_i \mathsf{G}\varphi$

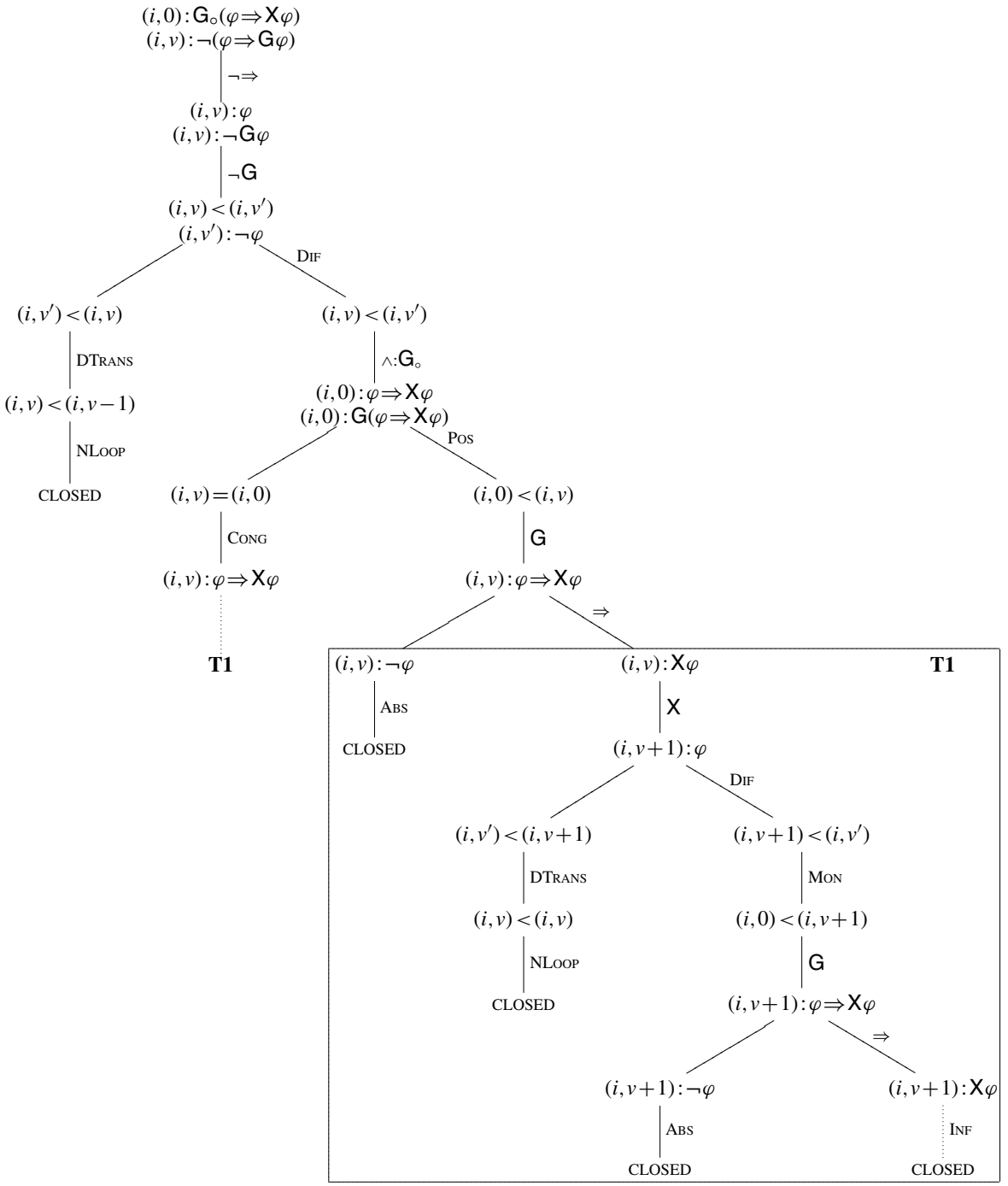

FIGURE 10. Tableau for the usual temporal induction schema

and therefore $\mu,\rho \Vdash (i,x) : \mathsf{G}\varphi$. In the second case, we know that $[\![(i,\mathsf{f}_{\varphi \mathsf{W} \psi}(x))]\!]_{\mu,\rho}$ is defined to be the least such n. In particular, $[\![(i,x)]\!]_{\mu,\rho} < [\![(i,\mathsf{f}_{\varphi \mathsf{W} \psi}(x))]\!]_{\mu,\rho}$ and $\mu_i, \xi_i^{[\![(i,\mathsf{f}_{\varphi \mathsf{W} \psi}(x))]\!]_{\mu,\rho}} \Vdash_i \psi$. That is, $\mu,\rho \Vdash (i,x) < (i,\mathsf{f}_{\varphi \mathsf{W} \psi}(x))$ and $\mu,\rho \Vdash (i,\mathsf{f}_{\varphi \mathsf{W} \psi}(x)) : \psi$.

(W_2): Assume that $\mu,\rho \Vdash (i,x) < s_i < (i,\mathsf{f}_{\varphi \mathsf{W} \psi}(x))$. Then $[\![(i,\mathsf{f}_{\varphi \mathsf{W} \psi}(x))]\!]_{\mu,\rho}$ is defined and $[\![(i,x)]\!]_{\mu,\rho} < [\![s_i]\!]_{\mu,\rho} < [\![(i,\mathsf{f}_{\varphi \mathsf{W} \psi}(x))]\!]_{\mu,\rho}$. From the definition of the interpretation of the Skolem symbols, it follows that $\mu_i, \xi^{[\![s_i]\!]_{\mu,\rho}} \Vdash_i \varphi$, i.e. $\mu,\rho \Vdash s_i : \varphi$. Furthermore, $[\![(i,\mathsf{f}_{\varphi \mathsf{W} \psi}(x))]\!]_{\mu,\rho}$ is the least value greater than $[\![(i,x)]\!]_{\mu,\rho}$ such that $\mu_i, \xi_i^{[\![(i,\mathsf{f}_{\neg(\varphi \mathsf{W} \psi)}(x))]\!]_{\mu,\rho}} \Vdash_i \psi$. Hence, $\mu_i, \xi^{[\![s_i]\!]_{\mu,\rho}} \nVdash_i \psi$, i.e. $\mu,\rho \Vdash s_i : \neg\psi$.

($\neg\mathsf{W}_1$): Assume that $\mu,\rho \Vdash (i,x) : \neg(\varphi \mathsf{W} \psi)$. Then $\mu_i, \xi_i^{[\![(i,x)]\!]_{\mu,\rho}} \nVdash_i \varphi \mathsf{W} \psi$. In particular, there exists $n > [\![(i,x)]\!]_{\mu,\rho}$ such that $\mu_i, \xi_i^n \nVdash_i \varphi$, and $\mu_i, \xi_i^m \nVdash_i \psi$ for $[\![(i,x)]\!]_{\mu,\rho} < m \leq n$. We know that $[\![(i,\mathsf{f}_{\neg(\varphi \mathsf{W} \psi)}(x))]\!]_{\mu,\rho}$ is defined to be the least such n. Hence, $[\![(i,x)]\!]_{\mu,\rho} < [\![(i,\mathsf{f}_{\neg(\varphi \mathsf{W} \psi)}(x))]\!]_{\mu,\rho}$, i.e, $\mu,\rho \Vdash (i,x) < (i,\mathsf{f}_{\neg(\varphi \mathsf{W} \psi)}(x))$. Furthermore, clearly, $\mu,\rho \Vdash (i,\mathsf{f}_{\neg(\varphi \mathsf{W} \psi)}(x)) : \neg\varphi$ and $\mu,\rho \Vdash (i,\mathsf{f}_{\neg(\varphi \mathsf{W} \psi)}(x)) : \neg\psi$.

($\neg\mathsf{W}_2$): Assume that $\mu,\rho \Vdash (i,x) < s_i < (i,\mathsf{f}_{\neg(\varphi \mathsf{W} \psi)}(x))$. Then $[\![(i,\mathsf{f}_{\neg(\varphi \mathsf{W} \psi)}(x))]\!]_{\mu,\rho}$ is defined and $[\![(i,x)]\!]_{\mu,\rho} < [\![s_i]\!]_{\mu,\rho} < [\![(i,\mathsf{f}_{\neg(\varphi \mathsf{W} \psi)}(x))]\!]_{\mu,\rho}$. From the definition of the interpretation of the Skolem symbols, it follows that $\mu_i, \xi^{[\![s_i]\!]_{\mu,\rho}} \nVdash_i \psi$, i.e. $\mu,\rho \Vdash s_i : \neg\psi$. Furthermore, $[\![(i,\mathsf{f}_{\varphi \mathsf{W} \psi}(x))]\!]_{\mu,\rho}$ is the least value greater than $[\![(i,x)]\!]_{\mu,\rho}$ such that $\mu_i, \xi_i^{[\![(i,\mathsf{f}_{\neg(\varphi \mathsf{W} \psi)}(x))]\!]_{\mu,\rho}} \nVdash_i \varphi$. Hence, $\mu_i, \xi^{[\![s_i]\!]_{\mu,\rho}} \Vdash_i \varphi$, i.e. $\mu,\rho \Vdash s_i : \varphi$.

We now turn to the rules for judgements about the relations and prove the soundness of two of them. The remaining ones are mostly trivial.

(Dif): Assume that $\mu,\rho \Vdash s_i : \varphi$ and $\mu,\rho \Vdash s_i' : \neg\varphi$. Then $\mu_i, \xi_i^{[\![s_i]\!]_{\mu,\rho}} \Vdash_i \varphi$ and $\mu_i, \xi_i^{[\![s_i']\!]_{\mu,\rho}} \nVdash_i \varphi$. Hence $\xi_i^{[\![s_i]\!]_{\mu,\rho}} \neq \xi_i^{[\![s_i']\!]_{\mu,\rho}}$ and, as Ξ_i is totally ordered, either $[\![s_i]\!]_{\mu,\rho} < [\![s_i']\!]_{\mu,\rho}$ or $[\![s_i']\!]_{\mu,\rho} < [\![s_i]\!]_{\mu,\rho}$, which implies that $\mu,\rho \Vdash s_i < s_i'$ or $\mu,\rho \Vdash s_i' < s_i$.

(Inf): $[\![(i,x)]\!]_{\mu,\rho}$ and $[\![(i,y)]\!]_{\mu,\rho}$ are natural numbers. Hence, there cannot be infinitely many distinct non-negative constants c such that $[\![(i,x)]\!]_{\mu,\rho} + c < [\![(i,y)]\!]_{\mu,\rho}$.

The soundness of the trichotomy rules is straightforward, given that the local orders are trichotomic. ■

3.3 Completeness

Before we establish completeness, we recall [26] some technical results about *integer constraints* of the form $x \leq y$, where (i,x) and (i,y) are local labels in $\mathcal{S}_i$. It is clear that any such constraint is of the form $u_1 + n \leq u_2 + m$, where u_1 and u_2 are either label variables, label terms whose head is a Skolem function, or 0. Let $A = \{A_1, A_2, \ldots\}$ be a (possibly infinite) set of such constraints. The *constraint graph* for A is a weighted, directed graph $G_A = \langle V_A, E_A \rangle$ constructed as follows:

- $V_A = \mathcal{V}(A) \cup \{0\}$, where $\mathcal{V}(A)$ is the set of variables $\mathcal{V}_i$ and of label terms headed by a Skolem function occurring in A;[4]
- $E_A = \{u_1 \xrightarrow{m-n} u_2 \mid u_1 + n \leq u_2 + m \in A\} \cup \{0 \xrightarrow{0} u \mid u \in \mathcal{V}(A)\}$.

[4] At this point, labels whose head is a Skolem function symbol are treated as if they were simply variables.

As notation, $u_1 \xrightarrow{c} u_2$ represents the directed edge (u_1, u_2) with weight c. Intuitively, this means that u_1 is at most c larger than u_2. Hence, for instance, edges of the second kind, $0 \xrightarrow{0} u$, express that $0 \leq u+0$, which is satisfied when u is non-negative, i.e. a natural number. As usual, a *path* in a graph is a finite sequence of vertices $u_1, \ldots, u_n$, where (u_i, u_{i+1}) is an edge, for all i such that $1 \leq i \leq n$. The weight of a path is the sum of the weights of its edges.

PROPOSITION 8
A (possibly infinite) set of constraints A is satisfiable if and only if for each non-zero node in G_A, there exists a minimum-weight path in G_A among all the paths from 0 to that node.

PROOF. ($\Rightarrow$) Assume that A is satisfiable and consider an arbitrary path in G_A from 0 to some u

$$0 \xrightarrow{c_0} u_1 \xrightarrow{c_1} \ldots \xrightarrow{c_{n-1}} u_n \xrightarrow{c_n} u.$$

This corresponds to the constraints

$$\begin{array}{ll} 0 & \leq u_1 + c_0 \\ & \vdots \\ u_{n-1} & \leq u_n + c_{n-1} \\ u_n & \leq u + c_n. \end{array}$$

Summing up both sides yields $0 \leq u + (c_0 + \cdots + c_{n-1} + c_n)$. This means that, given that the constraints are satisfiable, for each path from 0 to u with weight c, $0 \leq u + c$ must hold. Assume now that there is no minimum-weight path from 0 to u. This means that there is an infinite decreasing succession $\{c_i\}_{i \in \mathbb{N}}$ of integers such that there is a path from 0 to u with weight c_i, which means that $0 \leq u + c_i$, for every $i \in \mathbb{N}$, which is clearly impossible. Hence there must be a minimum-weight path.

($\Leftarrow$) Assume that, for every vertex u, there is a minimum-weight path from 0 to u, and let $\delta(u)$ denote its weight. Obviously, $\delta(0) = 0$. Let $\rho_i(u) = -\delta(u)$ for each u. We claim that ρ is a satisfying assignment for all the constraints in A. Namely, consider a constraint $u_1 + n \leq u_2 + m$. There is an edge (with weight $m - n$) from u_1 to u_2 and by the triangle inequality, $\delta(u_2) \leq \delta(u_1) + m - n$. From the definition of ρ_i, $-\rho_i(u_2) \leq -\rho_i(u_1) + m - n$ and therefore $\rho_i(u_1) + n \leq \rho_i(u_2) + m$. Thus the constraint is satisfied. ■

We now show how integer constraints can help us establish the completeness of our tableaux system. To begin with, observe that, in our tableaux, every judgement of the form $(i, x) < (i, y)$ can be equivalently stated as a constraint of the form $x \leq y - 1$. Similarly, a judgement of the form $(i, x) = (i, y)$ can be equivalently formalized as the pair of constraints $x \leq y$ and $y \leq x$. We then have the following two lemmas.

LEMMA 9
Let A be the set of integer constraints extracted from an exhausted branch of a $\mathcal{T}_i$-tableau. Then A is satisfiable if and only if the branch is open.

PROOF. ($\Rightarrow$) Assume that the branch is closed by the rule (INF). Hence, there are labels $(i, x), (i, y) \in \mathcal{S}_i$ and distinct non-negative constants $\{c_n\}_{n \in \mathbb{N}}$ such that $x + c_n < y \in A$. We can assume, without loss of generality, that the constants are ordered increasingly. Furthermore, (i, x) is either (i, k) or $(i, u + k)$. Similarly, (i, y) is either (i, k') or $(i, u' + k')$. Let us consider just the case where (i, x) is $(i, u + k)$ and (i, y) is $(i, u' + k')$. Hence, $u + (k + c_n) < u' + k' \in A$, for every $n \in \mathbb{N}$. By the rule (LSHIFT), $u < u' + (k' - (k + c_n))$. Let $c'_n = k' - (k + c_n)$. Clearly, $\{c'_n\}_{n \in \mathbb{N}}$ is a strictly decreasing succession of integers. By the rule (POS), depending on whether u is variable or a label headed by a Skolem function,

either $u=0\in A$ (and, by (Cong), $0<u'+c'_n\in A$) or $0<u\in A$ (and, by (DTrans), $0<u'+(c'_n-1)\in A$). In either case, we may conclude that there is no minimum-weight path from 0 to u', which implies that A is not satisfiable. If the branch is closed by (NLoop) then this means that there is a negative cycle in G_A and so A is not satisfiable. Note that we need not consider the case where the branch is closed by (Abs) as that rule can be obtained from (Dif) and (NLoop), as we remarked above. The same applies to closing with (Arith), although it is clear that A would then be impossible to satisfy.

($\Leftarrow$) Assume that A is not satisfiable. Then there is a strictly decreasing succession $\{c_n\}_{n\in\mathbb{N}}$ such that there is a path from 0 to some u with weight c_n, which implies that either $0<u+(c_n+1)\in A$ or $0=u+c_n\in A$. Clearly, equality can happen at most once: if $0=u+c_n\in A$ and $0=u+c_m\in A$ with $c_n\neq c_m$ then using (Cong) we would have $u+c_n=u+c_m\in A$ and we could close the branch using rule (Arith). Thus, $c_0-c_n>0$ as $\{c_n\}_{n\in\mathbb{N}}$ is strictly decreasing and we can apply (RShift) so that $c_0-c_n<u+c_0+1\in A$ for every n. By the rule (Inf), it follows that A is closed. ■

Lemma 10
Let A be the set of integer constraints extracted from an open branch of a $\mathcal{T}_i$-tableau. Let ρ_i be the assignment extracted from G_A according to Proposition 8. If $\rho_i(u)=k$ then the branch contains either $(i,u)=(i,k)$ or $(i,k-1)<(i,u)$.

Proof. Note that by Lemma 9, A is satisfiable and so we can extract an assignment ρ_i on variables and Skolem-headed labels that satisfies all the constraints in A. We consider first the case where u is a variable or a Skolem-headed label. Furthermore, assume also that $k=0$. By (Pos), either $u=0\in A$ or $0<u\in A$. But if $0<u\in A$ then, as ρ_i satisfies A, $0<\rho_i(u)$, contradicting the initial assumption. Hence, we can conclude that $u=0\in A$.

Assume now that $k>0$ and that $\rho_i(u)=k$. Then there is a path from 0 to u with weight $-k$. That is, there is a path in G_A from 0 to u

$$0\xrightarrow{c_0}u_1\xrightarrow{c_1}u_2\xrightarrow{c_2}\ldots\xrightarrow{c_n}u$$

such that $\sum_{i=0}^{n}c_i=-k$. In fact, without loss of generality, we can assume that $\sum_{i=0}^{p}c_i<0$ for all p such that $0\leq p\leq n$. Hence, the following constraints must be in A:

$$\begin{array}{ll} 0 & < u_1+c_0+1\\ & \vdots\\ u_{n-1} & < u_n+c_{n-1}+1\\ u_n & < u+c_n+1. \end{array}$$

Therefore, a simple inductive argument using (LShift) and (DTrans) allows us to conclude that also $0<u+(c_0+\cdots+c_n)+1\in A$. If $k-1>0$ then, by using (RShift), we conclude that $k-1<u\in A$. If $k-1=0$ then $-k+1=0$ and the result follows trivially.

The general result for labels follows from a direct application of the rules (RShift) and (LShift). ■

We can now prove completeness for the tableaux system $\mathcal{T}_i$. Since the Skolem function symbols are only used as an internal mechanism within our system, we will assume that the initial set of judgements contains no Skolem functions.[5]

[5]Note that this requirement could be dropped if (i) we added additional constraints to the graphs G_A imposing the required ordering between labels whose heads are Skolem functions and their subterms (e.g. stating that $(i,s_i)<(i,\mathsf{f}_{\varphi\mathsf{W}\psi}(s_i))$) and (ii) we split each of the rules (W_1), ($\neg\mathsf{W}_1$), (B_1) and ($\neg\mathsf{B}_1$) in two rules, one for introducing the Skolem symbols and one for introducing their properties.

PROPOSITION 11
$\mathcal{T}_i$ is complete, that is, a set of local judgements Θ without Skolem functions is satisfiable if and only if there is a $\mathcal{T}_i$-tableau for Θ with an open branch.

PROOF. If there is no open tableau for Θ then a simple inductive argument, using Proposition 7, establishes the unsatisfiability of Θ. We now prove the converse. Assume that there is an open tableau for Θ and let Δ be the set of judgements that appear in an open branch (which include Θ). Note that Δ is closed under the rules.

1. Let A be the set of linear constraints extracted from Δ. By Lemma 9, this set is satisfiable and so, using Proposition 8, we can extract an assignment ρ_i satisfying all linear constraints.
2. Let $\lambda_i = \langle E_i, \leq_i \rangle$ be defined as follows:
   - $E_i = \{\rho_i(x) \mid (i,x) : \varphi \in \Delta \text{ and } \rho_i(x) > 0\}$;
   - $e \leq_i e'$ is the usual order on $\mathbb{N}$.

   To continue the proof, we first establish an auxiliary lemma.

   LEMMA 12
   If $k \in E_i$ then there is some $(i,x) : \varphi \in \Delta$ such that $\rho_i(x) = k-1$.

   PROOF. Assume that $k \in E_i$. Clearly we have that $k > 0$. Then there is some $(i,y) : \varphi \in \Delta$ such that $\rho_i(y) = k$ and by Lemma 10, either $k-1 < y \in A$ or $y = k \in A$. In the first case, by (FILL), $(i,k-1) : \top \in \Delta$. In the second case, as $(i,k) > (i,0) \in \Delta$, by (PRED) and (CONG), $(i,k-1) < (i,y) \in \Delta$ and once again by (FILL), $(i,k-1) : \top \in \Delta$. Hence choose (i,x) to be $(i,k-1)$. ■

   From this lemma, we may conclude that if $k \in E_i$ and $k > 1$, then $k-1 \in E_i$ and, furthermore, $\langle E_i, \leq_i \rangle$ is a countable, discrete, well-founded total order. Local states are of the form $\{1, \ldots, e\}$. We consider any distributed interpretation structure μ such that $\mu_i = \langle \lambda_i, \sigma_i \rangle$, where $\sigma_i(\emptyset) = \{p \mid (i,x) : p \in \Delta \text{ and } \rho_i(x) = 0\}$ and $\sigma_i(\{1, \ldots, e\}) = \{p \mid (i,x) : p \in \Delta \text{ and } \rho_i(x) = e\}$. We also fix any compatible distributed assignment ρ.
3. We show that $\mu, \rho \Vdash (i,x) : \varphi$ for every $(i,x) : \varphi \in \Delta$. Simultaneously, we must also prove that $\rho_i(f(z)) = [\![(i, f(z))]\!]_{\mu,\rho}$, for every Skolem function symbol f. We will push the proof of this later fact to the application of the until and since rules, as we assume that no Skolem functions appear in Θ. The proof follows by induction on φ. If $x : p \in \Delta$, then the result follows by construction. If $(i,x) : \neg p \in \Delta$, then, by (ABS), $(i,x) : p \notin \Delta$. If $(i,y) : p \in \Delta$ then, by (DIF), either $(i,x) < (i,y) \in \Delta$ or $(i,y) < (i,x) \in \Delta$. In either case, we can conclude that $p \notin \sigma_i([\![(i,x)]\!]_{\mu,\rho})$, i.e. $\mu_i, \xi_i^{[\![(i,x)]\!]_{\mu,\rho}} \Vdash_i \neg p$ and so $\mu, \rho \Vdash (i,x) : \neg p$. The proof for implication, negated implication, and double negation follows by the induction hypothesis.
   Assume that $x : \mathsf{F}\psi \in \Delta$. Then by the rule (F), $(i,v) : \psi \in \Delta$ and $(i,x) < (i,v) \in \Delta$. By the induction hypothesis, $\mu, \rho \Vdash (i,v) : \psi$ and, since $[\![(i,x)]\!]_{\mu,\rho} < [\![(i,v)]\!]_{\mu,\rho}$, then $\mu, \rho \Vdash (i,x) : \mathsf{F}\psi$.
   Assume that $(i,x) : \neg\mathsf{F}\psi \in \Delta$. Let $k \in E_i$, such that $[\![(i,x)]\!]_{\mu,\rho} < k$. (If such a k does not exist, then the result follows immediately.) Then there is some $(i,y) : \vartheta \in \Delta$ such that $[\![(i,y)]\!]_{\mu,\rho} = k$. Hence, by Lemma 10, either $y = k \in A$ or $k-1 < y \in A$. In the first case, using (PRED) and (CONG) we also have $k-1 < y \in A$. Using (TRF), then either $(i,x) < (i,k-1) \in \Delta$ or $(i,x) = (i,k-1) \in \Delta$ or $(i,k-1) < (i,x) \in \Delta$. As A is satisfied by ρ_i, the third case is excluded. If $(i,x) < (i,k-1) \in \Delta$, then, by (DTRANS) and (MON), $(i,x) < (i,y) \in \Delta$. If $(i,x) = (i,k-1) \in \Delta$, then, by (CONG), $(i,x) < (i,y) \in \Delta$. In each case, we may apply $(\neg\mathsf{F})$ to conclude that $(i,y) : \neg\psi \in \Delta$. By the induction hypothesis, $\mu, \rho \Vdash (i,y) : \neg\psi$, i.e. $\mu_i, \xi_i^k \nVdash_i \psi$ for every $k > [\![(i,x)]\!]_{\mu,\rho}$. Hence $\mu_i, \xi_i^{[\![(i,x)]\!]_{\mu,\rho}} \nVdash_i \mathsf{F}\psi$ and therefore $\mu, \rho \Vdash (i,x) : \neg\psi$.

Assume that $(i,x):\mathsf{P}\psi\in\Delta$. Then by (P), $(i,v):\psi\in\Delta$ and $(i,v)<(i,x)\in\Delta$. By the induction hypothesis, $\mu,\rho\Vdash(i,v):\psi$ and since $[\![(i,v)]\!]_{\mu,\rho}<[\![(i,x)]\!]_{\mu,\rho}$ then $\mu,\rho\Vdash(i,x):\mathsf{P}\psi$ follows.
Assume that $(i,x):\neg\mathsf{P}\psi\in\Delta$. Furthermore, let $k\in E_i$ such that $k<[\![(i,x)]\!]_{\mu,\rho}$. (If such a k does not exist, then the result follows immediately.) Then there is some $(i,y):\vartheta\in\Delta$ such that $[\![(i,y)]\!]_{\mu,\rho}=k$. Hence, by Lemma 10, either $k-1<y\in A$ or $y=k\in A$. By an argument similar to ($\neg\mathsf{F}$), we have $k-1<y\in A$. Applying (TrF), we have either $(i,x)<(i,y)\in\Delta$ or $(i,x)=(i,y)\in\Delta$ or $(i,y)<(i,x)\in\Delta$. The first two conditions are excluded because A is satisfiable ($[\![(i,x)]\!]_{\mu,\rho}>k=[\![(i,y)]\!]_{\mu,\rho}$). Hence $(i,y)<(i,x)\in\Delta$. By ($\neg\mathsf{P}$), $(i,y):\neg\psi\in\Delta$ and by the induction hypothesis, $\mu,\rho\Vdash(i,y):\neg\psi$, i.e. $\mu_i,\xi_i^k\nVdash_i\psi$ for every $k<[\![(i,x)]\!]_{\mu,\rho}$. Hence $\mu_i,\xi_i^{[\![(i,x)]\!]_{\mu,\rho}}\nVdash_i\mathsf{P}\psi$, which implies that $\mu,\rho\Vdash(i,x):\neg\mathsf{P}\psi$.
The proofs for G and H are similar to the ones above, given their corresponding abbreviations.
Assume that $(i,x):\mathsf{X}\psi\in\Delta$. Then using the rule (X), $(i,x+1):\psi\in\Delta$. By the induction hypothesis, $\mu,\rho\Vdash(i,x+1):\psi$, which implies that $\mu,\rho\Vdash(i,x):\mathsf{X}\psi$.
Assume that $(i,x):\neg\mathsf{X}\psi\in\Delta$. If $[\![(i,x)]\!]_{\mu,\rho}$ is the maximum of E_i (if one exists) then there is no successor and so $\mu,\rho\Vdash(i,x):\neg\mathsf{X}\psi$. Otherwise, there is a $k\in E_i$ such that $[\![(i,x)]\!]_{\mu,\rho}<k$. As $k\in E_i$, there is some $(i,y):\vartheta\in\Delta$ such that $[\![(i,y)]\!]_{\mu,\rho}=k$. By Lemma 10, either $y=k\in A$ or $k-1<y\in A$. By an argument similar to the cases above, we may conclude that $k-1<y\in A$ anyway. Hence, by (TrX), either $(i,x)<(i,k-1)\in\Delta$ or $(i,x)=(i,k-1)\in\Delta$ or $(i,k-1)<(i,x)\in\Delta$. Once again, since A is satisfiable, the third possibility is excluded. If $(i,x)<(i,k-1)\in\Delta$, we may apply (Fill) and ($\neg\mathsf{X}$). If $(i,x)=(i,k-1)\in\Delta$, by (Cong), we have $(i,x)<(i,y)\in\Delta$. Once again, we may apply ($\neg\mathsf{X}$). In each case, we conclude that $(i,x+1):\psi\in\Delta$. Hence, by the induction hypothesis, $\mu,\rho\Vdash(i,x+1):\psi$, which implies that $\mu,\rho\Vdash(i,x):\neg\mathsf{X}\psi$.
Assume that $(i,x):\mathsf{Y}\psi\in\Delta$ then, by (Y), $(i,x-1):\psi\in\Delta$. By the induction hypothesis, $\mu,\rho\Vdash(i,x-1):\psi$ and thus $\mu,\rho\Vdash(i,x):\mathsf{Y}\psi$.
Assume that $(i,x):\neg\mathsf{Y}\psi\in\Delta$. By (Pos), either $(i,x)=(i,0)\in\Delta$ or $(i,0)<(i,x)\in\Delta$. If $(i,x)=(i,0)\in\Delta$, since A is satisfiable, then $[\![(i,x)]\!]_{\mu,\rho}=0$ and therefore $\mu_i,\xi_i^{[\![(i,x)]\!]_{\mu,\rho}}\nVdash_i\mathsf{Y}\psi$. If $(i,0)<(i,x)\in\Delta$, then ($\neg\mathsf{Y}$) may be applied and $(i,x-1):\neg\psi\in\Delta$. By the induction hypothesis, $\mu,\rho\Vdash(i,x-1):\neg\psi$, which implies that $\mu,\rho\Vdash(i,x):\neg\mathsf{Y}\psi$.
Assume that $(i,x):\varphi\mathsf{W}\psi\in\Delta$. Then, using the rule ($\mathsf{W}_1$), either $(i,x):\mathsf{G}\varphi\in\Delta$, or $(i,x)<(i,\mathsf{f}_{\varphi\mathsf{W}\psi}(x))\in\Delta$ and $(i,\mathsf{f}_{\varphi\mathsf{W}\psi}(x)):\psi\in\Delta$. In the first case, it follows directly from the completeness for G that $\mu,\rho\Vdash(i,x):\mathsf{G}\varphi$, and therefore $\mu,\rho\Vdash(i,x):\varphi\mathsf{W}\psi$. In the second case, we can conclude that $[\![(i,x)]\!]_{\mu,\rho}<\rho_i(\mathsf{f}_{\varphi\mathsf{W}\psi}(x))$. Furthermore, by the induction hypothesis, $\mu,\rho\Vdash(i,\mathsf{f}_{\varphi\mathsf{W}\psi}(x)):\psi$. Consider now $k\in E_i$ such that $[\![(i,x)]\!]_{\mu,\rho}<k<\rho_i(\mathsf{f}_{\varphi\mathsf{W}\psi}(x))$. Then there is $(i,y):\vartheta\in\Delta$ such that $[\![(i,y)]\!]_{\mu,\rho}=k$. As before, using Lemma 10, (Pred) and (Cong) we can conclude that $k-1<y\in A$. Hence, using (TrW_1) either $k-1<x\in A$ or $k-1=x\in A$ or $x<k-1\in A$. Since A is satisfiable, the first condition is excluded. If $k-1=x\in A$ by (Cong), we have $(i,x)<(i,y)\in\Delta$. If $x<k-1\in A$ then by (DTrans) and (Mon) we have again $(i,x)<(i,y)\in\Delta$. Using now (TrW_2), we have that either $y<\mathsf{f}_{\varphi\mathsf{W}\psi}(x)\in A$ or $y=\mathsf{f}_{\varphi\mathsf{W}\psi}(x)\in A$ or $\mathsf{f}_{\varphi\mathsf{W}\psi}(x)<y\in A$. Again, since A is satisfiable, the last two conditions are excluded. Hence $(i,y)<(i,\mathsf{f}_{\varphi\mathsf{W}\psi}(x))\in\Delta$ and we may apply (W_2) and conclude that $(i,y):\varphi\in\Delta$ and $(i,y):\neg\psi\in\Delta$. By the induction hypothesis, $\mu,\rho\Vdash(i,y):\varphi$ and $\mu,\rho\Vdash(i,y):\neg\psi$. From the conditions on k, we may finally conclude that $\rho_i(\mathsf{f}_{\varphi\mathsf{W}\psi}(x))=[\![(i,\mathsf{f}_{\varphi\mathsf{W}\psi}(x))]\!]_{\mu,\rho}$ and $\mu,\rho\Vdash(i,x):\varphi\mathsf{W}\psi$.
Assume now that $(i,x):\neg(\varphi\mathsf{W}\psi)\in\Delta$. By the rule ($\neg\mathsf{W}_1$), we have $(i,x)<(i,\mathsf{f}_{\neg(\varphi\mathsf{W}\psi)}(x)),(i,\mathsf{f}_{\neg(\varphi\mathsf{W}\psi)}(x)):\neg\varphi,(i,\mathsf{f}_{\neg(\varphi\mathsf{W}\psi)}(x)):\neg\psi\in\Delta$. Hence, we have that $[\![(i,x)]\!]_{\mu,\rho}<\rho_i(\mathsf{f}_{\neg(\varphi\mathsf{W}\psi)}(x))$. By the induction hypothesis, we also have that $\mu,\rho\Vdash(i,\mathsf{f}_{\neg(\varphi\mathsf{W}\psi)}(x)):\neg\varphi$ and $\mu,\rho\Vdash(i,\mathsf{f}_{\neg(\varphi\mathsf{W}\psi)}(x)):\neg\psi$. If $k\in E_i$ is such that $[\![(i,x)]\!]_{\mu,\rho}<k<\rho_i(\mathsf{f}_{\neg(\varphi\mathsf{W}\psi)}(x))$, then by an

argument similar to the case of W, using now the trichotomy rules for $\neg\mathsf{W}$, we can conclude that there is (i,z) such that $[\![(i,z)]\!]_{\mu,\rho}=k$ and $(i,x)<(i,z)<(i,\mathsf{f}_{\neg(\varphi\mathsf{W}\psi)}(x))\in\Delta$. Then, by the rule $(\neg\mathsf{W}_2)$, $(i,z):\varphi,(i,z):\neg\psi\in\Delta$ and hence, by the induction hypothesis, we have $\mu,\rho\Vdash(i,z):\varphi$ and $\mu,\rho\Vdash(i,z):\neg\psi$. We may finally conclude that $\rho_i(\mathsf{f}_{\neg(\varphi\mathsf{W}\psi)}(x))=[\![(i,\mathsf{f}_{\neg(\varphi\mathsf{W}\psi)}(x))]\!]_{\mu,\rho}$ and $\mu,\rho\Vdash(i,x):\neg(\varphi\mathsf{W}\psi)$.
The proofs for B are analogous to those for W. ■

As a consequence, we can reason about entailment in the logic.

COROLLARY 13
Given $\Phi\cup\{\psi\}\in\mathcal{L}_i$, $\Phi\models_i\psi$ if and only if every exhausted $\mathcal{T}_i$-tableau for $\{(i,0):\mathsf{G}_\circ\varphi\mid\varphi\in\Phi\}\cup\{(i,v):\neg\psi\}$ is closed.

Our previous examples also illustrate this. Example 5 proves that $\models_i((\varphi\mathsf{W}\psi)\wedge\mathsf{X}(\neg\psi))\Rightarrow\mathsf{X}\varphi$. Moreover, the proof in Example 6 establishes that $(\varphi\Rightarrow\mathsf{X}\varphi)\models_i(\varphi\Rightarrow\mathsf{G}\varphi)$.

4 Tableaux for global reasoning

4.1 The global tableaux system

Our aim now is to build a tableaux system $\mathcal{T}$ for full DTL by capitalizing on the local tableaux systems for each agent $i\in Id$. To do so, we now introduce an additional kind of *global judgement*: synchronization between labels. Labelled local formulas will also be unrestricted, i.e. communication formulas are allowed. Of course, the language of labels is now distributed, that is, if $Id=\{i_1,\ldots,i_n\}$ then

$$\mathcal{S}::=\mathcal{S}_{i_1}\mid\cdots\mid\mathcal{S}_{i_n}.$$

Here, the local labels of each agent $i\in Id$ are defined, as before, by

$$\mathcal{T}_i::=\mathbb{N}_0\mid\mathcal{V}_i+\mathbb{Z}\mid\mathcal{F}_i(\mathcal{T}_i)+\mathbb{Z},$$

$$\mathcal{S}_i::=(i,\mathcal{T}_i),$$

but the Skolem symbols are extended to the full language, that is,

$$\begin{aligned}\mathcal{F}_i=\ &\{\mathsf{f}_{\varphi\mathsf{W}\psi}\mid\varphi,\psi\in\mathcal{L}_i\}\cup\{\mathsf{f}_{\neg(\varphi\mathsf{W}\psi)}\mid\varphi,\psi\in\mathcal{L}_i\}\cup\\ &\{\mathsf{f}_{\varphi\mathsf{B}\psi}\mid\varphi,\psi\in\mathcal{L}_i\}\cup\{\mathsf{f}_{\neg(\varphi\mathsf{B}\psi)}\mid\varphi,\psi\in\mathcal{L}_i\}.\end{aligned}$$

The syntax of global judgements can now be defined by

$$\mathcal{J}::=\mathcal{J}_{i_1}\mid\cdots\mid\mathcal{J}_{i_n}\mid\mathcal{S}_i\bowtie\mathcal{S}_j,$$

where the local judgements are extended to also incorporate communication formulas

$$\mathcal{J}_i::=\mathcal{S}_i:\mathcal{L}_i\mid\mathcal{S}_i=\mathcal{S}_i\mid\mathcal{S}_i<\mathcal{S}_i\mid\textsc{closed}.$$

$$\frac{(i,x) : ©_j[\varphi]}{(j,v) : \varphi \ , \ (i,x) \bowtie (j,v)} \ (©) \ [v \text{ fresh}] \qquad \frac{(i,x) : \neg\, ©_j\varphi \ , \ (i,x) \bowtie (j,y)}{(j,y) : \neg\, \varphi} \ (\neg\, ©)$$

FIGURE 11. Rules for communication

$$\frac{(i,x) \bowtie (i,y)}{(i,0) < (i,x)} \ (\text{EVT}) \qquad \frac{s_i \bowtie s_j}{s_j \bowtie s_i} \ (\text{SYM}) \qquad \frac{s_i \bowtie s'_i}{s_i = s'_i} \ (\text{SELF})$$

$$\frac{s_i \bowtie s_j \quad s_j \bowtie s_k}{s_i \bowtie s_k} \ (\text{TRANS}) \qquad \frac{s_i \bowtie s_j \quad s'_i \bowtie s_k}{s_i < s'_i \mid s_i = s'_i \mid s'_i < s_i} \ (\text{TR} \bowtie)$$

$$\frac{s_{i_1} \bowtie s_{i_2} \quad s_{i_2} < s'_{i_2} \quad s'_{i_2} \bowtie s_{i_3} \quad s_{i_3} < s'_{i_3} \quad \ldots \quad s'_{i_p} \bowtie s'_{i_1}}{s_{i_1} < s'_{i_1}} \ (\text{ORDER})$$

FIGURE 12. Rules for synchronization

The intended meaning of a *synchronization judgement* $(i,x) \bowtie (j,y)$ is that the event leading to state x of agent i is synchronized with the event leading to state y of agent j. Semantically, we require a distributed *assignment on label variables* $\rho = \{\rho_i\}_{i \in Id}$. The *denotation of labels* is defined as before, given an interpretation structure μ. The satisfaction of judgements is also just extended with

- $\mu, \rho \Vdash s_i \bowtie s_j$ if $\xi_i^{[\![s_i]\!]_{\mu,\rho}} \neq \emptyset$, $\xi_j^{[\![s_j]\!]_{\mu,\rho}} \neq \emptyset$, and $last_i(\xi_i^{[\![s_i]\!]_{\mu,\rho}}) = last_j(\xi_j^{[\![s_j]\!]_{\mu,\rho}})$.

We can finally define our tableaux for global reasoning, which we then show to be sound and complete.

DEFINITION 14
The *global tableaux system* $\mathcal{T}$ for DTL, built over sets of global judgements in $\mathcal{J}$, consists of the rules of $\mathcal{T}_i$ for each agent $i \in Id$, together with the global rules in Figures 11 and 12.

The rules for communication in Figure 11 closely follow the semantics. Consider, for instance, the rule (©): if agent i, in state x, just communicated with agent j, for whom φ held, then the event leading to state x is synchronized with an event leading to some state v of agent j, where φ holds (and where v is fresh). In a similar way, for rule ($\neg$©), if agent i in state x does not communicate with agent j in a state where φ holds, but the event leading to x is synchronized with the event leading to some state y of j, then it must be the case that φ cannot hold in y for j.

Figure 12 contains the rules for synchronization. Rule (EVT) guarantees that synchronization is only possible in states following the initial state, given that the initial state of an agent is not reached by an event. Rules (SYM) and (TRANS) express the symmetry and transitivity of the synchronization relation. Rule (SELF) ensures that self-synchronization is not allowed. Finally, rule (TR $\bowtie$) applies trichotomy to any two states of agent i involved in the synchronizations and rule (ORDER) guarantees that local orders are globally compatible. If there is a chain of synchronizations linking two events of agent i, then these two events preserve the ordering imposed by the synchronization chain. For instance, assume that the events leading to states s_i and s'_i of agent i have just synchronized with the events leading to states s_j and s'_j of agent j, respectively. Furthermore, assume that s_j precedes s'_j. Then this order must be reflected in agent i and so s_i must precede s'_i. This extends to more than two agents in a straightforward way.

We illustrate the use of the tableaux system with a short example. More substantial examples are given in Section 5.

EXAMPLE 15
To show

$$\{@_i[\copyright_j[\top]\Rightarrow\copyright_j[\mathsf{X}\copyright_k[\top]]], @_j[\copyright_k[\top]\Rightarrow\copyright_k[\mathsf{X}\copyright_i[\top]]]\}\vDash_{\mathrm{DTL}} @_i[\copyright_j[\top]\Rightarrow\mathsf{F}\copyright_k[\top]]$$

it is enough (as will follow from the completeness result we give below) to build a closed $\mathcal{T}$-tableau for the corresponding judgements, as depicted in Figure 13 (where we write ABS: $\neg\top$ to abbreviate the closure of the unfolding of $\neg\top$). Figure 14 shows a possible life-cycle. The formula $@_i[\copyright_j[\top]\Rightarrow\copyright_j[\mathsf{X}\copyright_k[\top]]]$ expresses that if agent i synchronizes with agent j then, in the next state (of j), agent j will synchronize with agent k. Similarly, the formula $@_j[\copyright_k[\top]\Rightarrow\copyright_k[\mathsf{X}\copyright_i[\top]]]$ expresses that if agent j synchronizes with agent k then, in the next state (of k), agent k will synchronize with agent i. These two formulas entail that if agent i synchronizes with agent j then he will eventually also synchronize with agent k. However, observe that synchronization between agents i and k need not necessarily take place after two local state transitions of agent i. Between the synchronization with j and the synchronization with k, agent i might have changed state many times.

4.2 Soundness

We now proceed to establish the soundness and completeness of our tableaux system $\mathcal{T}_i$. We first prove the soundness of the rules.

PROPOSITION 16
The rules of $\mathcal{T}$ are sound.

PROOF. The soundness of the rules of $\mathcal{T}_i$, for each $i\in Id$, follows from Proposition 7. To show the soundness of the communication and synchronization rules, let μ be an arbitrary model and ρ an assignment.

(©): Assume that $\mu,\rho\Vdash(i,x):\copyright_j[\varphi]$. Then $\mu_i,\xi_i^{[\![(i,x)]\!]_{\mu,\rho}}\Vdash_i\copyright_j[\varphi]$. Hence, we have that $last_i(\xi_i^{[\![(i,x)]\!]_{\mu,\rho}})\in E_j$ and $\mu_j,(last_i(\xi_i^{[\![(i,x)]\!]_{\mu,\rho}})\downarrow j)\Vdash_j\varphi$. As v is fresh, let $\rho_j(v)$ be the number of the local state $(last_i(\xi_i^{[\![(i,x)]\!]_{\mu,\rho}})\downarrow j)$ of agent $j\in Id$, i.e. $\xi_j^{[\![(j,v)]\!]_{\mu,\rho}}=(last_i(\xi_i^{[\![(i,x)]\!]_{\mu,\rho}})\downarrow j)$. Thus, $\mu_j,\xi_j^{[\![(j,v)]\!]_{\mu,\rho}}\Vdash_j\varphi$, i.e. $\mu,\rho\Vdash(j,v):\varphi$. Furthermore, as $last_j(\xi_j^{[\![(j,v)]\!]_{\mu,\rho}})=last_j((last_i(\xi_i^{[\![(i,x)]\!]_{\mu,\rho}})\downarrow j))=last_i(\xi_i^{[\![(i,x)]\!]_{\mu,\rho}})$, then $\mu,\rho\Vdash(i,x)\bowtie(j,v)$.

The proof for $(\neg\copyright)$ is similar. Let us now turn to the rules for synchronization.

(EVT): Assume that $\mu,\rho\Vdash(i,x)\bowtie(j,y)$. By definition, $\xi_i^{[\![(i,x)]\!]_{\mu,\rho}}\neq\xi_i^0=\emptyset$. Clearly, then, $[\![(i,x)]\!]_{\mu,\rho}\neq 0$ and therefore $[\![(i,x)]\!]_{\mu,\rho}>[\![(i,0)]\!]_{\mu,\rho}=0$. Hence, $\mu,\rho\Vdash(i,0)<(i,x)$.

(ORDER): Assume $\mu,\rho\Vdash s_{i_1}\bowtie s_{i_2}, s_{i_2}<s'_{i_2}, s'_{i_2}\bowtie s_{i_3}, s_{i_3}<s'_{i_3},\ldots,s'_{i_p}\bowtie s_{i_2}$. It follows that $[\![s_{i_1}]\!]_{\mu,\rho},[\![s'_{i_1}]\!]_{\mu,\rho},[\![s_{i_2}]\!]_{\mu,\rho},[\![s'_{i_2}]\!]_{\mu,\rho},\ldots,[\![s_{i_p}]\!]_{\mu,\rho},[\![s'_{i_p}]\!]_{\mu,\rho}\neq\emptyset$, and also $[\![s_{i_2}]\!]_{\mu,\rho}<[\![s'_{i_2}]\!]_{\mu,\rho},\ldots,[\![s_{i_p}]\!]_{\mu,\rho}<[\![s'_{i_p}]\!]_{\mu,\rho}$. Hence, $last_{i_1}(\xi_{i_1}^{[\![s_{i_1}]\!]_{\mu,\rho}})=last_{i_2}(\xi_{i_2}^{[\![s_{i_2}]\!]_{\mu,\rho}})<last_{i_2}(\xi_{i_2}^{[\![s'_{i_2}]\!]_{\mu,\rho}})=\cdots=last_{i_p}(\xi_{i_p}^{[\![s_{i_p}]\!]_{\mu,\rho}})<last_{i_p}(\xi_{i_p}^{[\![s'_{i_p}]\!]_{\mu,\rho}})=last_{i_1}(\xi_{i_1}^{[\![s'_{i_1}]\!]_{\mu,\rho}})$. Since $<$ is a partial order of global causality, we have $last_{i_1}(\xi_{i_1}^{[\![s_{i_1}]\!]_{\mu,\rho}})<last_{i_1}(\xi_{i_1}^{[\![s'_{i_1}]\!]_{\mu,\rho}})$, which implies $[\![s_{i_1}]\!]_{\mu,\rho}<[\![s'_{i_1}]\!]_{\mu,\rho}$. Therefore, $\mu,\rho\Vdash s_{i_1}<s'_{i_1}$.

The soundness of the remaining rules is straightforward. ■

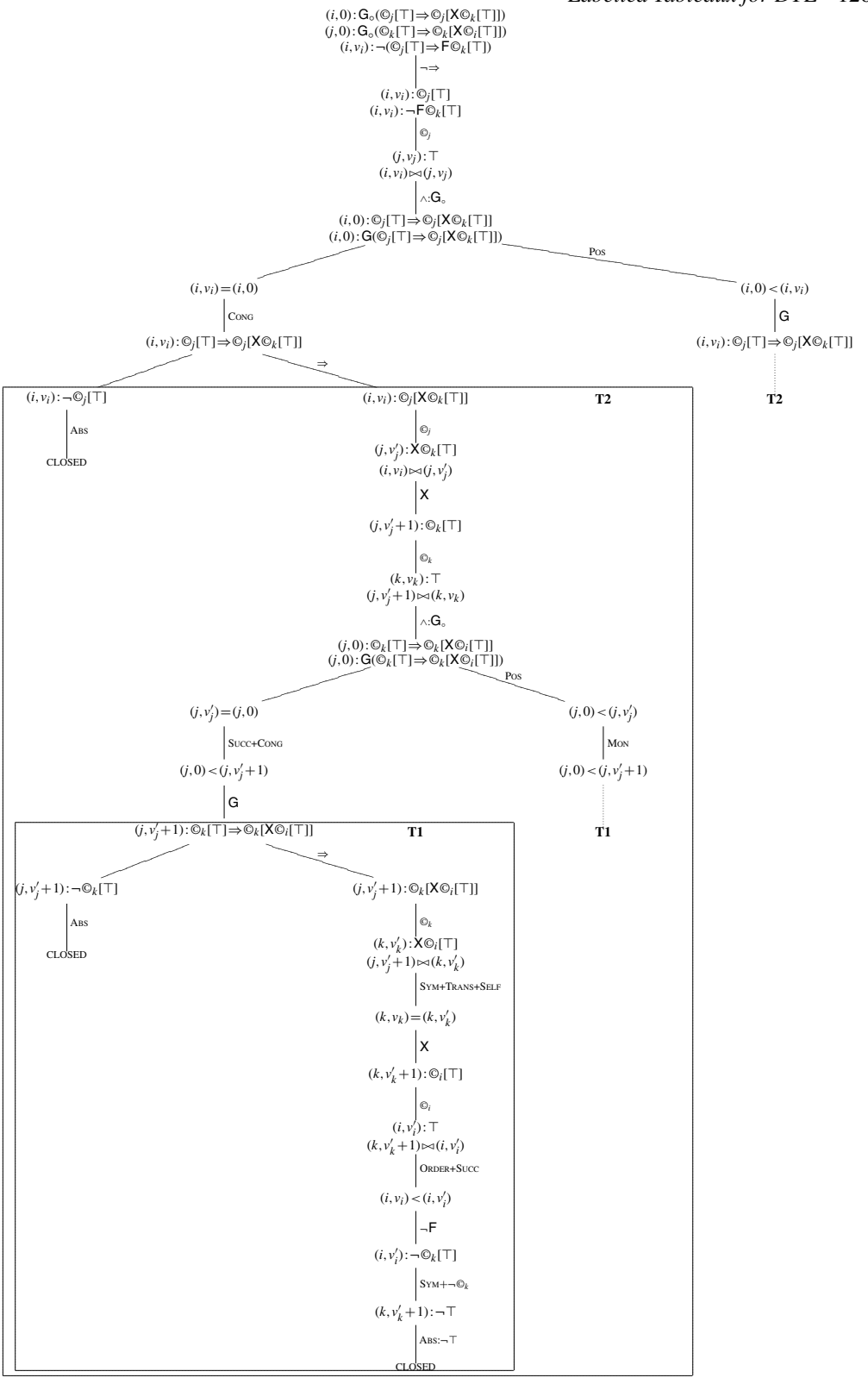

FIGURE 13. Tableau for $\{@_i[\copyright_j[\top]\Rightarrow\copyright_j[\mathsf{X}\copyright_k[\top]]], @_j[\copyright_k[\top]\Rightarrow\copyright_k[\mathsf{X}\copyright_i[\top]]]\}\vDash_{\mathrm{DTL}} @_i[\copyright_j[\top]\Rightarrow\mathsf{F}\copyright_k[\top]]$

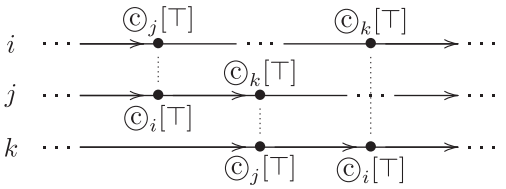

FIGURE 14. A life-cycle for Example 15

4.3 Completeness

We can now prove the completeness of the $\mathcal{T}$ system. The proof follows the lines taken above to show the completeness of $\mathcal{T}_i$.

PROPOSITION 17
$\mathcal{T}$ is complete, that is, a set of global judgements Θ without Skolem functions is satisfiable if and only if there is a $\mathcal{T}$-tableau for Θ with an open branch.

PROOF. Once again, if there is no open tableau for Θ, then by Proposition 16, Θ is not satisfiable. Hence, let us assume that we have an open tableau for Θ and let Δ be the set of judgements that appear in an open branch.

1. Let A_i be the set of linear constraints extracted from Δ involving agent i. By Proposition 8, each of these sets is satisfiable and these sets do not interact with one another. Hence, we can extract an assignment ρ on label variables satisfying all linear constraints.
2. For each $i \in Id$, let $F_i = \{(i, \rho_i(x)) | (i,x) : \varphi \in \Delta \text{ and } \rho_i(x) > 0\}$ and $F = \bigcup_{i \in Id} F_i$. Define $\approx \subseteq F \times F$ to be the reflexive closure of the relation such that $(i, \rho_i(x)) \approx (j, \rho_j(y))$ if $(i,x) \bowtie (j,y) \in \Delta$. The rules (SYM) and (TRANS) ensure that $\approx$ is an equivalence relation. Let $E = F/\approx$ and $E_i = \{e \in E \,|\, e \cap F_i \neq \emptyset\}$. For every $i \in Id$, define $\leq_i \subseteq E_i \times E_i$ to be the relation such that $e \leq_i e'$ if there are $(i,n), (i,n') \in F_i$ such that $(i,n) \in e$, $(i,n') \in e'$, and $n \leq n'$, on the natural numbers. It is not difficult to see that, with this construction, $\langle E_i, \leq_i \rangle$ is a local life-cycle (see Proposition 11 for details). Note that for every $e \in E$, using (SELF), $|e \cap F_i| \leq 1$. This means that there is at most one local event from each individual in each global event e. Therefore, rule (ORDER) guarantees that the induced global causality relation $\leq$ is indeed a partial order.
   Let $\mu = \langle \lambda, \sigma \rangle$, where each $\lambda_i = \langle E_i, \leq_i \rangle$ and σ_i are defined as in the local case.
3. Finally, we show that μ and ρ satisfy every judgement in Δ. The proof, by induction, follows exactly the same pattern as the one for the local case: showing that $\rho_i(x) = [\![(i,x)]\!]_{\mu,\rho}$ for every label x. We focus on the new judgements.
   Assume that $(i,x) : \copyright_j[\varphi] \in \Delta$. Then, by rule ($\copyright$), $(j,v) : \varphi \in \Delta$ and $(i,x) \bowtie (j,v) \in \Delta$. By the induction hypothesis, $\mu, \rho \Vdash (j,v) : \varphi$, i.e. $\mu_j, \xi_j^{[\![(j,v)]\!]_{\mu,\rho}} \Vdash_j \varphi$. Moreover, we also know that $[\![(i,x)]\!]_{\mu,\rho} = (i, \rho_i(x)) \approx (j, \rho_j(v)) = [\![(j,v)]\!]_{\mu,\rho}$. Thus, $last_i(\xi_i^{[\![(i,x)]\!]_{\mu,\rho}}) = last_j(\xi_j^{[\![(j,v)]\!]_{\mu,\rho}})$ and so it follows that $(last_i(\xi_i^{[\![(i,x)]\!]_{\mu,\rho}}) \downarrow j) = (last_j(\xi_j^{[\![(j,v)]\!]_{\mu,\rho}}) \downarrow j) = \xi_j^{[\![(j,v)]\!]_{\mu,\rho}}$. Hence $\mu_j, (last_i(\xi_i^{[\![(i,x)]\!]_{\mu,\rho}}) \downarrow j) \Vdash_j \varphi$. This allows us to conclude that $\mu, \xi_i^{[\![(i,x)]\!]_{\mu,\rho}} \Vdash_i \copyright_j[\varphi]$, that is, $\mu, \rho \Vdash (i,x) : \copyright_j[\varphi]$.
   Assume that $(i,x) : \neg\copyright_j[\varphi] \in \Delta$. If there is no (j,y) such that $(i,x) \bowtie (j,y) \in \Delta$ then $[(i, \rho_i(x))] \notin E_j$. Hence $\mu_i, \xi_i^{[\![(i,x)]\!]_{\mu,\rho}} \nVdash_i \copyright_j[\varphi]$, which implies that $\mu, \rho \Vdash (i,x) : \neg\copyright_j[\varphi]$. Assume now that $(i,x) \bowtie (j,y) \in \Delta$. Then, by rule ($\neg\copyright$), $(j,y) : \neg\varphi \in \Delta$ and, by the induction hypothesis, $\mu, \rho \Vdash (j,y) : \neg\varphi$. By an argument similar to the above, we may conclude that $\mu, \rho \Vdash (i,x) : \neg\copyright_j[\varphi]$.

Assume that $(i,x)\bowtie(j,y)\in\Delta$. Rule (Evt) guarantees that both $[\![(i,x)]\!]_{\mu,\rho}>0$ and $[\![(i,y)]\!]_{\mu,\rho}>0$, thus yielding that $(i,[\![(i,x)]\!]_{\mu,\rho})\approx(j,[\![(i,y)]\!]_{\mu,\rho})$. Hence, we have that $last_i(\xi_i^{[\![(i,x)]\!]_{\mu,\rho}})=last_j(\xi_j^{[\![(i,y)]\!]_\rho})$, since both take precisely the value of their equivalence class, and so $\mu,\rho\Vdash(i,x)\bowtie(j,y)$. ■

As a consequence, we may reason deductively about entailment in DTL.

Corollary 18
Given $\Gamma\cup\{@_i[\varphi]\}\in\mathcal{L}_{\mathrm{DTL}}$, $\Gamma\vDash_{\mathrm{DTL}}@_i[\varphi]$ if and only if every exhausted $\mathcal{T}$-tableau for $\{(j,0):\mathsf{G}_\circ\psi\mid @_j[\psi]\in\Gamma\}\cup\{(i,v):\neg\varphi\}$ is closed.

5 A detailed example

In this section, we formalize and reason about a simplified version of a *two-phase commit protocol* from [33], used to commit a transaction in a distributed system. In this protocol, one process acts as the *coordinator* and works with multiple subordinates. We designate the coordinator by C and assume that there are two *subordinates*, A and B. The behaviour of the three agents is depicted as transition diagrams in Figures 15 and 16. The commit protocol begins when the coordinator informs her subordinates that she is starting the protocol and that they should prepare to commit. She does this by executing an action (denoted by prep) that is synchronized with the actions of the subordinates (denoted by req). When a subordinate receives the commit request, he checks if he is ready to commit. When he is ready, he sends a message to the coordinator (reply) informing her of this. The corresponding $\mathsf{reply_A}$ or $\mathsf{reply_B}$ action is triggered in the coordinator. The protocol ends when the coordinator receives the replies from both subordinates.

We begin by introducing the distributed signature $\Sigma=\langle Id,\{Prop_i\}_i\in Id\rangle$ including the propositional symbols used to construct a model of the states of the processes:

- $Id=\{A,B,C\}$
- $Prop_A=Prop_B=\{\mathsf{work},\mathsf{pend}\}$
- $Prop_C=\{\mathsf{active},\mathsf{got_A},\mathsf{got_B}\}$

We can then define the transition diagram states using the following abbreviations:

- $\mathsf{idle}\equiv(\neg\mathsf{active})\wedge(\neg\mathsf{got_A})\wedge(\neg\mathsf{got_B})$
- $\mathsf{wait_{AB}}\equiv\mathsf{active}\wedge(\neg\mathsf{got_A})\wedge(\neg\mathsf{got_B})$
- $\mathsf{wait_A}\equiv\mathsf{active}\wedge(\neg\mathsf{got_A})\wedge\mathsf{got_B}$
- $\mathsf{wait_B}\equiv\mathsf{active}\wedge\mathsf{got_A}\wedge(\neg\mathsf{got_B})$
- $\mathsf{done}\equiv\mathsf{active}\wedge\mathsf{got_A}\wedge\mathsf{got_B}$

Since there are only five states, we will also employ the following constraints:

- $\mathsf{got_A}\Rightarrow\mathsf{active}$
- $\mathsf{got_B}\Rightarrow\mathsf{active}$

Similarly, we define the subordinates' states as:

- $\mathsf{free}\equiv\mathsf{work}\wedge(\neg\mathsf{pend})$
- $\mathsf{busy}\equiv\mathsf{work}\wedge\mathsf{pend}$
- $\mathsf{ready}\equiv(\neg\mathsf{work})$

and employ the following state constraint:

- $\mathsf{pend}\Rightarrow\mathsf{work}$

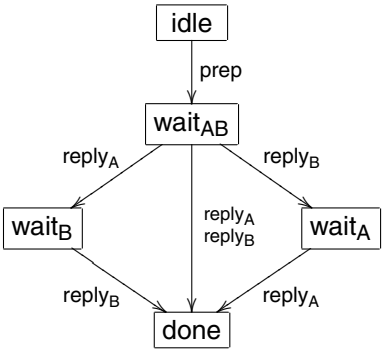

FIGURE 15. Transition diagram for the coordinator

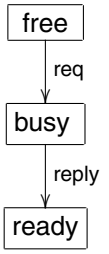

FIGURE 16. Transition diagram for the subordinates

DTL can be extended with actions (as we do, for instance, in [5]), but we do not need them here as we can model the occurrence of an action by a process changing from one state to another. To model the coordinator's actions, we define the following abbreviations:

- $\mathsf{prep} \equiv \mathsf{wait}_{\mathsf{AB}} \wedge \mathsf{Y}\ \mathsf{idle}$
- $\mathsf{reply}_{\mathsf{A}} \equiv \mathsf{got}_{\mathsf{A}} \wedge \mathsf{Y}(\neg \mathsf{got}_{\mathsf{A}})$
- $\mathsf{reply}_{\mathsf{B}} \equiv \mathsf{got}_{\mathsf{B}} \wedge \mathsf{Y}(\neg \mathsf{got}_{\mathsf{B}})$

Note that the $\mathsf{reply}_{\mathsf{A}}$ and $\mathsf{reply}_{\mathsf{B}}$ actions may occur independently or even simultaneously. With these definitions, we have not yet fully formalized the transitions in Figure 15. In particular, the states satisfying $\mathsf{got}_{\mathsf{A}}$ are $\mathsf{wait}_{\mathsf{B}}$ and done, whereas the states satisfying $\neg\mathsf{got}_{\mathsf{A}}$ are idle, $\mathsf{wait}_{\mathsf{AB}}$ and $\mathsf{wait}_{\mathsf{A}}$. Therefore, $\mathsf{reply}_{\mathsf{A}}$ specifies a possible transition from any of the states idle, $\mathsf{wait}_{\mathsf{AB}}$ and $\mathsf{wait}_{\mathsf{A}}$ to either $\mathsf{wait}_{\mathsf{B}}$ or done. Note that this allows more transitions than those in our transition diagram and hence we further restrict them as follows:

- $* \Rightarrow \mathsf{idle}$
- $\mathsf{idle} \Rightarrow (\mathsf{idle}\ \mathsf{W}\ \mathsf{prep})$
- $\mathsf{wait}_{\mathsf{AB}} \Rightarrow (\mathsf{wait}_{\mathsf{AB}}\ \mathsf{W}\ (\mathsf{reply}_{\mathsf{A}} \vee \mathsf{reply}_{\mathsf{B}}))$
- $\mathsf{wait}_{\mathsf{A}} \Rightarrow (\mathsf{wait}_{\mathsf{A}}\ \mathsf{W}\ \mathsf{reply}_{\mathsf{A}})$
- $\mathsf{wait}_{\mathsf{B}} \Rightarrow (\mathsf{wait}_{\mathsf{B}}\ \mathsf{W}\ \mathsf{reply}_{\mathsf{B}})$
- $\mathsf{got}_{\mathsf{A}} \Rightarrow (\mathsf{G}\ \mathsf{got}_{\mathsf{A}})$
- $\mathsf{got}_{\mathsf{B}} \Rightarrow (\mathsf{G}\ \mathsf{got}_{\mathsf{B}})$

Similarly, we define the subordinates' actions as follows:

- $\mathsf{req} \equiv \mathsf{busy} \wedge \mathsf{Y}\ \mathsf{free}$
- $\mathsf{reply} \equiv \mathsf{ready} \wedge \mathsf{Y}\ \mathsf{busy}$

As above, we restrict these to the transitions in Figure 16 with the propositions:

- $\mathsf{pend} \Rightarrow \mathsf{work}$
- $* \Rightarrow \mathsf{free}$
- $\mathsf{free} \Rightarrow (\mathsf{free}\ \mathsf{W}\ \mathsf{req})$
- $\mathsf{busy} \Rightarrow (\mathsf{busy}\ \mathsf{W}\ \mathsf{reply})$
- $\mathsf{ready} \Rightarrow (\mathsf{G}\ \mathsf{ready})$

Finally, the synchronization is specified as follows:

- $@_C[\mathsf{prep} \gg_A \mathsf{req}]$
- $@_C[\mathsf{prep} \gg_B \mathsf{req}]$
- $@_A[\mathsf{reply} \gg_C \mathsf{reply_A}]$
- $@_B[\mathsf{reply} \gg_C \mathsf{reply_B}]$

With respect to this specification, we prove the property:

$$\{@_A[\mathsf{req} \Rightarrow (\mathsf{F}\ \mathsf{reply})], @_B[\mathsf{req} \Rightarrow (\mathsf{F}\ \mathsf{reply})]\} \vDash @_C[\mathsf{prep} \Rightarrow (\mathsf{F}\ \mathsf{done})].$$

This property expresses that, under certain fairness assumptions on the subordinates (given by the two premises), if the coordinator begins the commit protocol, she will eventually receive a reply from both subordinates. The tableau for this proof is depicted in Figures 17–20. In Figure 17, we present the first part of the tableau, focusing on local reasoning for the coordinator C (where the leftmost branch is identical to the branch on the right and hence we omit it). This part of the tableau ends with the interaction between the coordinator C and the subordinates A and B. As before, we systematically use boxes to avoid repeating sub-tableaux in the figures. The reasoning depicted in **T1** can be repeated on the leftmost branch, by means of a straightforward application of the rule (Cong). A similar comment applies also to the sub-tableau **T2**, in Figure 18, where we depict local reasoning for the subordinate A, triggered by the interaction with the coordinator. Note that we write $\wedge : \mathsf{reply_A}$ to abbreviate the unfolding of the definition of $\mathsf{reply_A}$ and the split of the two conjuncts. This part of the tableau ends with interaction between A and C. A similar flow of reasoning applies to the subordinate B, which we refrain from showing. In any case, both must be considered in subsequent reasoning, where they are denoted by Figure 18A and B, their corresponding variables being decorated with one prime (like v'_C) or two primes (like v''_C), respectively. In Figures 19 and 20, we show the last part of the tableau, which is a mixture of local reasoning for the coordinator C and of the interaction between the subordinates and the coordinator. In these figures, we write, for instance, **T3**[B] to denote the sub-tableau **T3**[A] with A replaced by B. Moreover, we also write $\neg \wedge : \mathsf{done}$ to abbreviate the unfolding of the definition of done and the split of the three resulting disjuncts.

6 Related and future work

We have given the first sound and complete tableaux system for DTL. To do so, we first gave a system for reasoning locally (in LTL) at each agent and afterwards we combined the local systems into one for global reasoning.

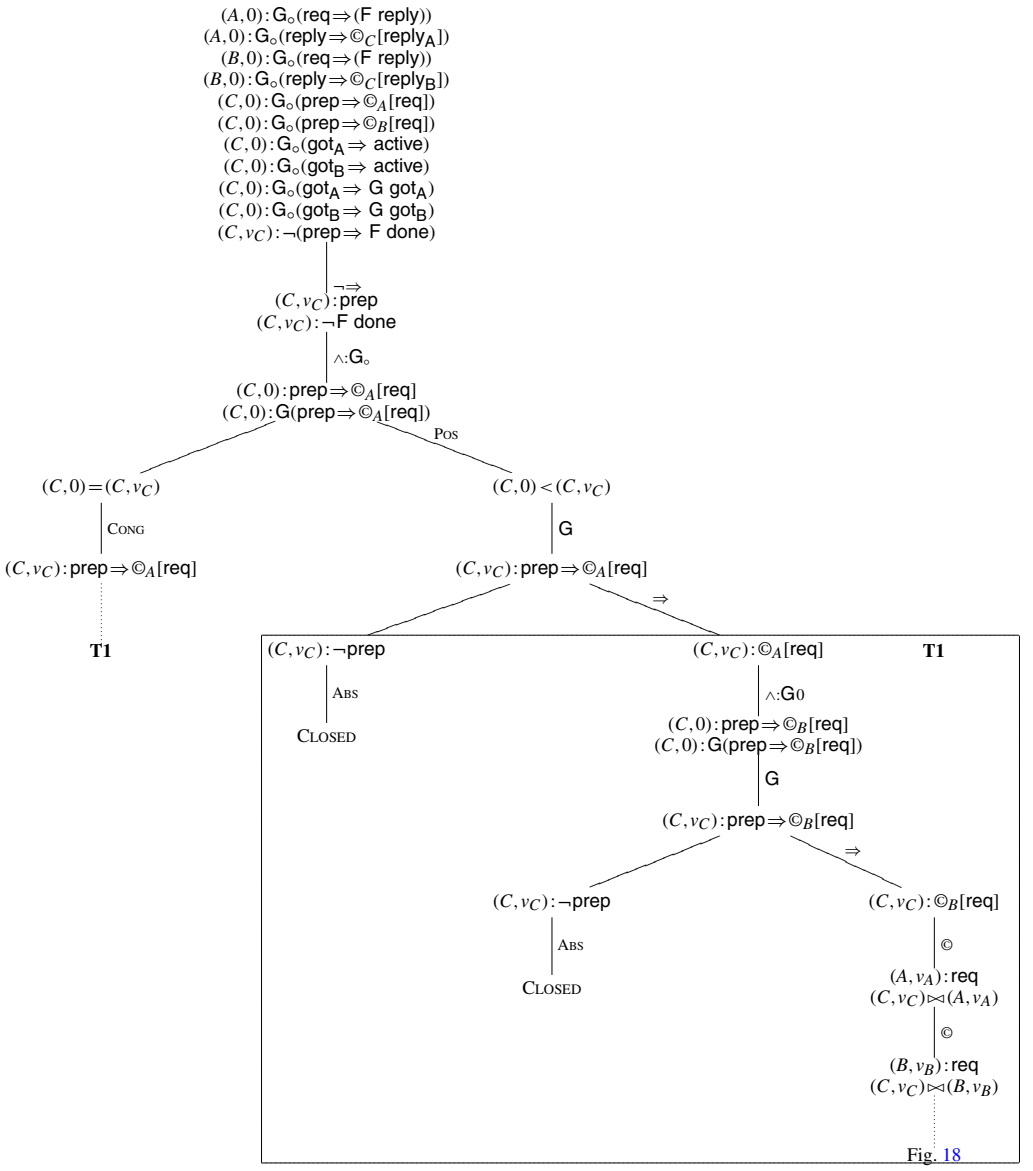

FIGURE 17. Local reasoning for agent C

A number of tableaux and other deductive systems have been given for different versions of temporal logic, e.g. [3, 14–16, 23]. For LTL, in particular, many of the proposed systems are based on the Fischer-Ladner approach [13, 27, 37] and take advantage of the fixedpoint definitions of the temporal operators to build a graph for checking the satisfaction of eventualities [17, 20, 31, 32, 37]. As noted in the introduction, this approach leads to a decision procedure based on loop checking in the graph.

Other systems, such as ours, use labels to naturally capture the logics' semantics. There are labelled systems for different temporal logics [7, 18, 19, 24, 29, 30, 35]. However, these are not for full

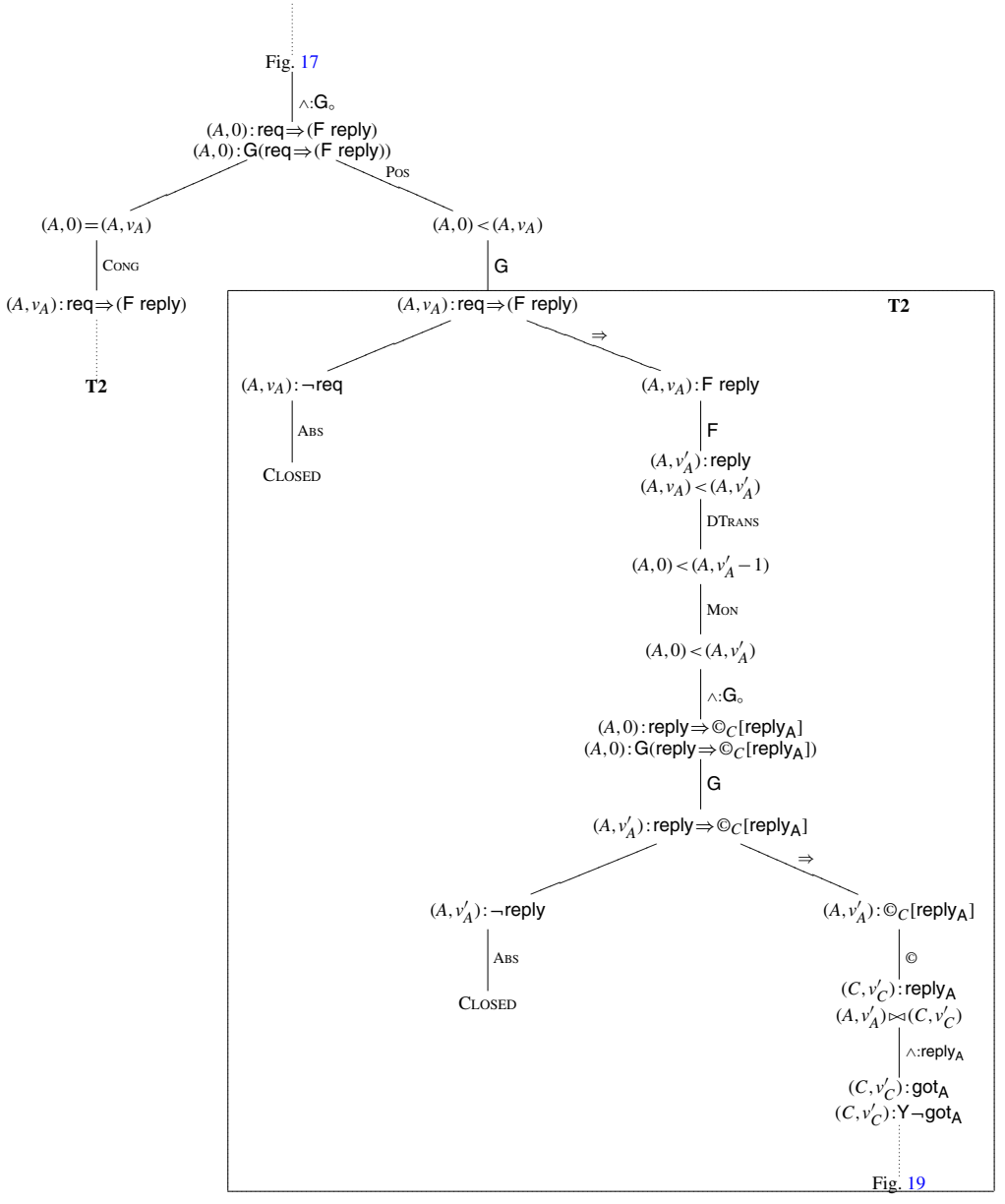

FIGURE 18. Local reasoning for agent A

discrete LTL. In this respect, the systems closest to ours are [7, 18, 30] (P.H. Schmitt and J. Goubault-Larrecq, Unpublished data). The [18] considers time points as labels for formulas, whereas [7, 30] consider time intervals. Schmitt and Goubault-Larrecq employ constraint graphs to reason about completeness of their rules where labels are time intervals, similar to what we did for our time-point labels. Most importantly, different fragments of the logic are considered in the different systems to cope with the difficulties of the full logic, e.g. the difficulties of formalizing rules for until and since. The manuscript (P.H. Schmitt and J. Goubault-Larrecq, Unpublished data) is an attempt to give a

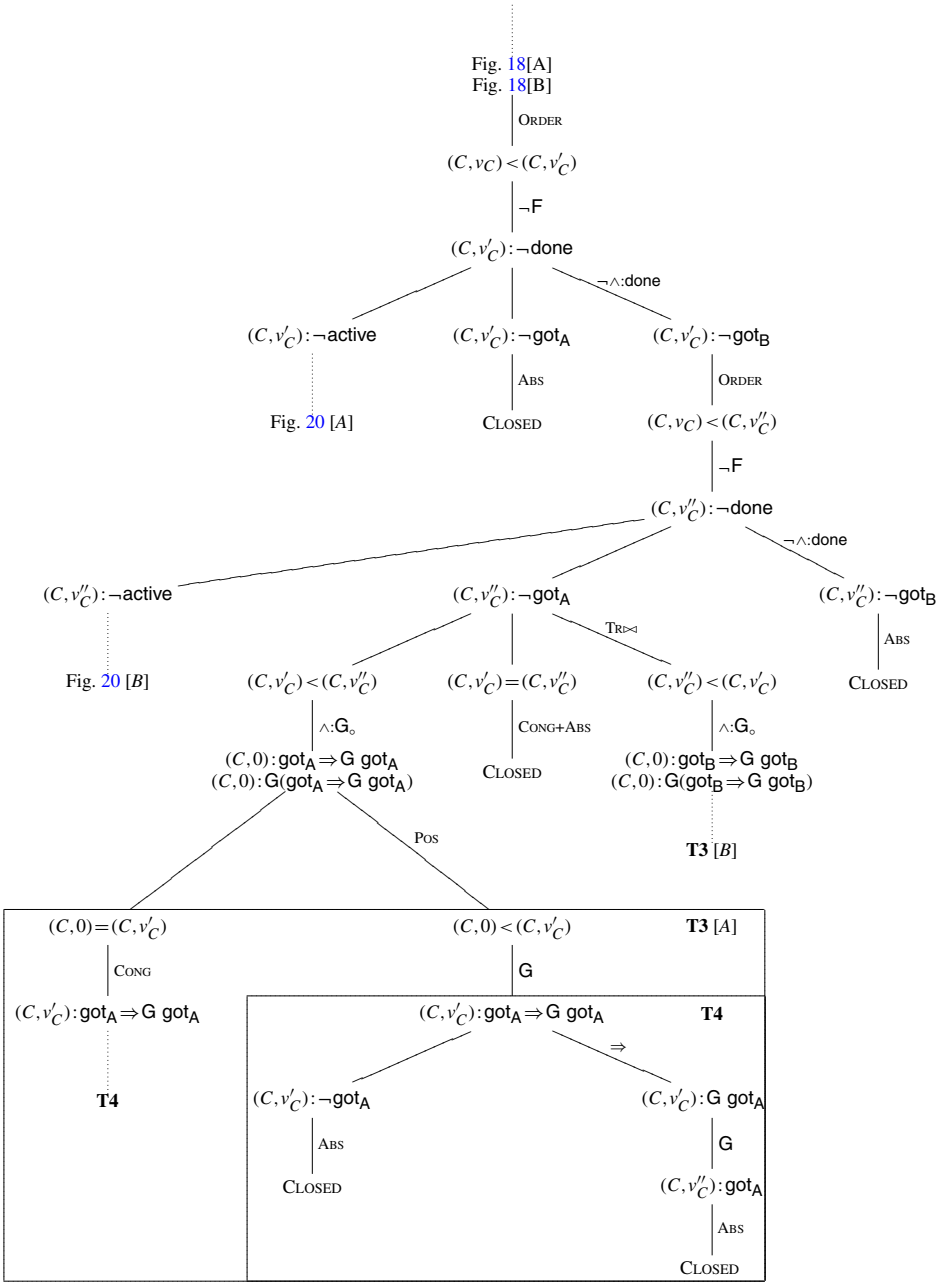

FIGURE 19. Reasoning about the last interaction

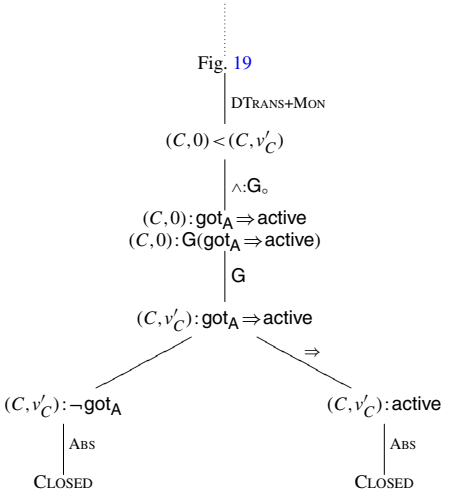

FIGURE 20. Reasoning about the state constraints of the coordinator

labelled tableaux system for the full logic, but unfortunately it has never been completed. None of these provides a decision procedure. Note also that [19, 24, 29] include tableaux-based decision procedures for versions of temporal logic without since and until, or with until but over general (not necessarily discrete) time, thus avoiding the problems of induction.

We have designed our systems with the aim of providing tableaux for full DTL, including past. However, it is interesting to note that our system for local reasoning seems to be closely related to the natural deduction system for future-time LTL of [4], which was developed in parallel with our work. We have begun investigating whether similar rules would also be suited for the extension to global reasoning in both past-time and future-time DTL and plan to report on this soon.

As we remarked above, we chose not to address decidability in the context of our tableaux system and have thus given an infinite closure rule that captures eventualities that are always delayed. If one really wants to hard-wire loop checking in our system, then exploring different rules, for instance those of [4], may be interesting. It may be possible here to capitalize on the constraint graphs we used in our tableaux system. Actually, in the finite case, our Lemma 8 is well-known to amount to checking that there are no cycles with negative weight in the graph [26], which can be done efficiently using the Bellman–Ford algorithm [9].

Another direction for future work will be to extend our system to the *distributed temporal protocol logic* DTPL that we have devised to reason about models and properties of security protocols. In [5, 6], we have applied DTPL in two different ways: first to verify (or refute) that security protocols provide claimed security properties, and second to prove metatheoretic properties of protocol models that can be used to simplify the verification of protocols or to search for attacks against them. All of these results have been obtained directly by semantic arguments. Hence, extending the tableaux system given here to DTPL will allow us to formalize, and possibly implement, (meta)reasoning about security protocols. We will report on this in a forthcoming paper.

Acknowledgements

We thank Matthias Schmalz and the anonymous referees for their useful comments on a draft of this article.

Funding

Hasler Foundation, ManCom project 2071; FCT and EU FEDER via the project KLog PTDC/MAT/68723/2006 of SQIG-IT; FP7-ICT-2007-1 Project no. 216471, 'AVANTSSAR: Automated Validation of Trust and Security of Service-oriented Architectures' (www.avantssar.eu).

References

[1] D. Basin, C. Caleiro, J. Ramos and L. Viganò. A labelled tableaux system for the Distributed Temporal Logic DTL. In *Proceedings of Temporal Representation and Reasoning (TIME 2008)*, pp. 101–109. IEEE Computer Society Press, 2008.

[2] E. Best and C. Fernández C. *Nonsequential Processes – A Petri Net View*. Springer, 1988.

[3] L. Bolc and A. Szałas, eds. *Time and Logic: A Computational Approach*. UCL Press Ltd, 1995.

[4] A. Bolotov, O. Grigoriev and V. Shangin. Automated natural deduction for propositional linear-time temporal logic. In *Proceedings of Temporal Representation and Reasoning (TIME 2007)*, pp. 47–58. IEEE Computer Society Press, 2007.

[5] C. Caleiro, L. Viganò and D. Basin. Metareasoning about security protocols using distributed temporal logic. In *Proceedings of Automated Reasoning for Security Protocol Analysis (ARSPA'04)*, *ENTCS 125(1)*, pp. 67–89, 2005.

[6] C. Caleiro, L. Viganò and D. Basin. Relating strand spaces and distributed temporal logic for security protocol analysis. *Logic Journal of the IGPL*, **13**, 637–664, 2005.

[7] S. Cerrito and M. Cialdea Mayer. Labelled tableaux for propositional linear time logic over finite frames. In *Labelled Deduction*, D. Basin, M. D'Agostino, D. M. Gabbay, S. Matthews, and L. Viganò, eds. Kluwer Academic Publishers, 2000.

[8] J. Clarke, M. Edmund, O. Grumberg and D. A. Peled. *Model Checking*. MIT Press, 1999.

[9] T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein. *Introduction to Algorithms, 2nd edn*. MIT Press, 2001.

[10] M. D'Agostino, D. M. Gabbay, R. Hähnle and J. Posegga, eds. *Handbook of Tableau Methods*. Kluwer Academic Publishers, 1999.

[11] H. -D. Ehrich and C. Caleiro. Specifying communication in distributed information systems. *Acta Informatica*, **36**, 591–616, 2000.

[12] H. -D. Ehrich, M. Kollmann and R. Pinger. Checking object system designs incrementally. *Journal of Universal Computer Science*, **9**, 106–119, 2003.

[13] M. J. Fischer and R. E. Ladner. Propositional dynamic logic of regular programs. *Journal of Computer and System Sciences*, **18**, 194–211, 1979.

[14] M. Fisher. Implementing temporal logics: tools for execution and proof. In *Proceedings of Computational Logic in Multi-Agent Systems (CLIMA IV), Lecture Notes in Artifical Intelligence 3900*, pp. 129–142. Springer, 2006.

[15] M. Fisher, D. M. Gabbay and L. Vila, eds. *Handbook of Temporal Reasoning in Artificial Intelligence I*. Elsevier, 2005.

[16] R. Gore. Tableau methods for modal and temporal logics. In *Handbook of Tableau Methods*, D'Agostino, D. M. Gabby, R. Hahnle and J. Posegga, eds. Kluwer Academic Publishers, 1999.

[17] G. D. Gough. Decision procedures for temporal logic. *Technical Report UMCS-89-10-1*. Department of Computer Science, University of Manchester, 1984.

[18] R. Hähnle and O. Ibens. Improving temporal logic tableaux using integer constraints. In *Proceedings of International Conference on Temporal Logic (ICTL'94), Lecture Notes in Artifical Intelligence 827*. Springer, 1994.
[19] A. Indrzejczak. A labelled natural deduction system for linear temporal logic. *Studia Logica*, **75**, 345–376, 2003.
[20] O. Lichtenstein and A. Pnueli. Propositional temporal logics: decidability and completeness. *Logic Journal of the IGPL*, **8**, 55–85, 2000.
[21] K. Lodaya, R. Ramanujam and P. Thiagarajan. Temporal logics for communicating sequential agents: I. *International Journal of Foundations of Computer Science*, **3**, 117–159, 1992.
[22] K. Lodaya and P. Thiagarajan. A modal logic for a subclass of event structures. In *Proceedings of International Colloquium on Automata, Languages and Programming (ICALP 14), Lecture Notes in Computer Science 267*, pp. 290–303. Springer, 1987.
[23] Z. Manna and A. Pnueli, eds. *Temporal Verification of Reactive Systems: Safety*. Springer, 1995.
[24] M. Marx, S. Mikulas and M. Reynolds. The mosaic method for temporal logics. In *Proceedings of Tableaux'00, Lecture Notes in Artifical Intelligence 1847*, pp. 324–340. Springer, 2000.
[25] D. Peled. All from one, one for all: on model checking using representatives. In *Proceedings of Computer Aided Verification (CAV'93)*, pp. 409–423. Springer, 1993.
[26] V. R. Pratt. Two easy theories whose combination is hard. *Technical report*, MIT, Cambridge, 1977.
[27] V. R. Pratt. A near-optimal method for reasoning about action. *Journal of Computer and System Sciences*, **20**, 231–254, 1980.
[28] R. Ramanujam. Locally linear time temporal logic. In *Proceedings of IEEE Symposium on Logic in Computer Science (LICS 11)*, pp. 118–127. IEEE Computer Society Press, 1996.
[29] M. Reynolds. The complexity of the temporal logic with 'until' over general linear time. *Journal of Computer and System Sciences*, **66**, 393–426, 2003.
[30] P. H. Schmitt and J. Goubault-Larrecq. A tableau system for linear-TIME temporal logic. In *Proceedings of Tools and Algorithms for the Construction and Analysis of Systems (TACAS'97), Lecture Notes in Computer Science 1217*, pp. 130–144. Springer, 1997.
[31] S. Schwendimann. A new one-pass tableau calculus for PLTL. In *Proceedings of Tableaux'98, Lecture Notes in Artifical Intelligence 1397*, pp. 277–291. Springer, 1998.
[32] R. Scott, M. Fisher and J. Keane. Parallel temporal tableaux. In *Proceedings of Euro-Par'98, Lecture Notes in Artifical Intelligence 1470*, pp. 852–861. Springer, 1998.
[33] A. S. Tanenbaum and M. van Steen. *Distributed Systems - Principles and Paradigms*, 2nd edn. Prentice Hall, 2006.
[34] P. S. Thiagarajan. A trace consistent subset of PTL. In *Proceedings of CONCUR'95, Lecture Notes in Computer Science 962*, pp. 438–452. Springer, 1995.
[35] L. Viganò and M. Volpe. Labelled natural deduction systems for a family of tense logics. In *Proceedings of Temporal Representation and Reasoning (TIME 2008)*, pp. 118–126. IEEE Computer Society Press, 2008.
[36] G. Winskel. Event structures. In *Petri Nets: Applications and Relationships to Other Models of Concurrency*, W. Brauer, W. Reisig and G. Rozenberg, eds, *Lecture Notes in Computer Science 255*, pp. 325–392. Springer, 1987.
[37] P. Wolper. The tableau method for temporal logic: an overview. *Logique et Analyse*, **110**, 119–136, 1985.

Received 3 February 2008