Labelled Tableaux for Linear Time Bunched Implication Logic

Didier Galmiche ✉
Université de Lorraine, CNRS, LORIA, Nancy, France

Daniel Méry ✉
Université de Lorraine, CNRS, LORIA, Nancy, France

Abstract

In this paper, we define the logic of Linear Temporal Bunched Implications (LTBI), a temporal extension of the Bunched Implications logic BI that deals with resource evolution over time, by combining the BI separation connectives and the LTL temporal connectives. We first present the syntax and semantics of LTBI and illustrate its expressiveness with a significant example. Then we introduce a tableau calculus with labels and constraints, called $\mathsf{T}_{\mathsf{LTBI}}$, and prove its soundness w.r.t. the Kripke-style semantics of LTBI. Finally we discuss and analyze the issues that make the completeness of the calculus not trivial in the general case of unbounded timelines and explain how to solve the issues in the more restricted case of bounded timelines.

2012 ACM Subject Classification Theory of computation → Proof theory

Keywords and phrases Temporal Logic, Bunched Implication Logic, Labelled Deduction, Tableaux.

Digital Object Identifier 10.4230/LIPIcs.FSCD.2023.27

Funding Work supported by the ANR NARCO Project (ANR Grant 21-CE48-0011).

1 Introduction

The notion of resource is a fundamental concept in various fields, especially in computer science. For instance, resources play a central role in designing systems such as computer networks or programs that access memory and manipulate data structures using pointers [9]. It is well known that Linear Logic [8] emphasizes an aspect of resource management that is closely related with resource consumption, whereas the Logic of Bunched Implications (BI) [13, 15] focuses more on aspects related with resource sharing and separation [7]. Recent works consider modal and/or epistemic extensions of BI and Boolean BI (BBI) in order to deal with more dynamic aspects of resource management [3, 4].

In this paper, we introduce the logic of Linear Temporal Bunched Implications (LTBI), a temporal extension of BI that deals with resource evolution over time. LTBI extends BI with operators borrowed from Linear Temporal Logic (LTL) to handle temporal aspects of computer systems [16]. Both temporal and separation logics have proven themselves successful in the design and formal verification of computer systems. Temporal logics are also well-known for their ability to state and verify safety and liveness properties (e.g., using Buchi automata [11]) and have a wide range of applications including model checking, concurrent programming, and reactive systems [2]. It is therefore interesting to study a logic for which the spatial connectives of BI cohabit with the temporal modalities of LTL.

Let us remark that a temporal extension of BI, called tBI, has been introduced in [10]. This extension derives an enriched sequent calculus from LBI (the standard sequent calculus of BI) and gives various embedding of tBI into BI. In this paper, we follow another approach based on labelled tableaux, in the spirit of [3, 4]. Although tBI might at first glance seem very similar to our logic LTBI, they bear significant differences that we discuss in details in Section 5 (after the required technical notions have been introduced).

8th International Conference on Formal Structures for Computation and Deduction (FSCD 2023).
Editors: Marco Gaboardi and Femke van Raamsdonk; Article No. 27; pp. 27:1–27:17
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

The paper is organized as follows: in Section 2 we describe the syntax and semantics of our LTBI logic that mixes the separation connectives of BI [7] with the temporal connectives $\Diamond$, $\Box$, $\circ$ of LTL. We also illustrate the expressiveness of LTBI with a significant example. In Section 3, we introduce $\mathsf{T}_{\mathsf{LTBI}}$, our labelled tableau calculus for LTBI in the spirit of [7, 3]. We then illustrate how it works with some examples. In Section 4 we prove the soundness of the $\mathsf{T}_{\mathsf{LTBI}}$ calculus. Finally, Section 5 ends the paper with a discussion of the several completeness issues that arise when trying to keep the labels constraints isomorphic to the standard linear order of the natural numbers.

2 Linear Temporal Bunched Implication Logic

Separation logics like BI and its variants are well suited to state (static) spatial properties about resources [6, 7]. DBI [3], a recent extension of BI with S4 modalities $\Diamond$ and $\Box$, opens the way for more dynamic aspects of resource management, but only to some extent. In this section we introduce Linear Temporal BI (LTBI) as a combination of BI and LTL [2, 16] interpreted on a discrete timeline.

2.1 Syntax and Semantics of LTBI

LTBI is an extension of BI [7, 14] with the three main LTL unary connectives $\Box$, $\Diamond$ and $\circ$. We do not consider the binary connectives U and R ("until" and "release") in this paper and leave them for future work.

▶ **Definition 1.** *Let* $\mathbf{P}$ *be a countable set of propositional letters. The set* $\mathbf{F}$ *of* LTBI *formulas is given by the following grammar:*

$$\mathrm{A} ::= \mathbf{P} \mid \top \mid \bot \mid \mathrm{A} \wedge \mathrm{A} \mid \mathrm{A} \vee \mathrm{A} \mid \mathrm{A} \to \mathrm{A} \mid \mathrm{I} \mid \mathrm{A} * \mathrm{A} \mid \mathrm{A} \mathrel{-\!\!*} \mathrm{A} \mid \Box\mathrm{A} \mid \Diamond\mathrm{A} \mid \circ\mathrm{A}$$

Additive negation is defined as usual as $\mathrm{A} \to \bot$.

In order to define a Kripke-style semantics for LTBI, we first introduce the notions of linear resource frames (LRF), interpretation and models.

▶ **Definition 2.** *An* LTBI-frame *is a structure* $\mathcal{R} = (\mathbf{R}, \star, \epsilon, \leqslant^{\mathfrak{r}}, \pi, \mathbf{S}, \leqslant^{\mathfrak{s}}, s_0)$*, where:*

- $(\mathbf{R}, \star, \epsilon, \leqslant^{\mathfrak{r}}, \pi)$ *is a* resource monoid, *i.e., a partially ordered commutative monoid of elements, called* resources, *such that:*
  - ϵ *is the unit of* $\star$*, i.e.* $\epsilon \star r = r \star \epsilon = r$*,*
  - π *is the greatest element of* $\mathbf{R}$ *w.r.t.* $\leqslant^{\mathfrak{r}}$ *and* $\forall\, r \in \mathbf{R}.\, r \star \pi = \pi$*,*
  - $\forall\, r, r', r'' \in \mathbf{R}.\, r \leqslant^{\mathfrak{r}} r'$ *implies* $r \star r'' \leqslant^{\mathfrak{r}} r' \star r''$*.*
- $(\mathbf{S}, \leqslant^{\mathfrak{s}}, s_0)$ *is a* discrete timeline, *i.e., a subset of* $\mathbb{N}$ *totally ordered by* $\leqslant^{\mathfrak{s}}$ *taken as the restriction to* $\mathbf{S}$ *of the standard order on* $\mathbb{N}$*, and such that* s_0 *is the least element of* $\mathbf{S}$ *w.r.t.* $\leqslant^{\mathfrak{s}}$*. The elements of* $\mathbf{S}$ *are called* states.

For all $s \in \mathbf{S}$*, we define* $N(s)$ *as the set* $\{\, s' \mid s' \in \mathbf{S}$ *and* $s <^{\mathfrak{s}} s' \,\}$*. We then write* $\mathfrak{n}$ *for the function* "next" *induced on* $\mathbf{S}$ *by* $\leqslant^{\mathfrak{s}}$ *and such that for all* $s \in \mathbf{S}$*,* $\mathfrak{n}(s)$ *is the least element of* $N(s)$ *if* $N(s)$ *is not empty and undefined otherwise.*

▶ **Definition 3.** *An* LTBI-valuation *is a partial function* $[\cdot] : \mathbf{P} \to \wp(\mathbf{R} \times \mathbf{S})$ *that satisfies the following conditions:*

$(\mathcal{M}_{\mathrm{K}})$ $\forall\, \mathrm{p} \in \mathbf{P}.\, \forall\, s \in \mathbf{S}.\, \forall\, r, r' \in \mathbf{R}.$ *if* $r \leqslant^{\mathfrak{r}} r'$ *and* $(r, s) \in [\mathrm{p}]$ *then* $(r', s) \in [\mathrm{p}]$*,*

$(\mathcal{M}_{\pi})$ $\forall\, \mathrm{p} \in \mathbf{P}.\, \forall\, s \in \mathbf{S}.\, (\pi, s) \in [\mathrm{p}]$.

▶ **Definition 4.** *An* LTBI-model *is a triple* $\mathcal{M} = (\mathcal{R}, [\cdot], \Vdash)$*, where* $\mathcal{R}$ *is an* LTBI-*frame,* $[\cdot]$ *is an* LTBI-*valuation and* $\Vdash\, \subseteq \mathbf{R} \times \mathbf{S} \times \mathbf{F}$ *is the smallest* forcing relation *such that:*

- $(r, s) \Vdash \mathrm{p}$ *iff* $(r, s) \in [\mathrm{p}]$
- $(r, s) \Vdash \mathrm{I}$ *iff* $\epsilon \leqslant^{\mathfrak{r}} r$
- $(r, s) \Vdash \bot$ *iff* $\pi \leqslant^{\mathfrak{r}} r$
- $(r, s) \Vdash \top$ *always*
- $(r, s) \Vdash \mathrm{A} \vee \mathrm{B}$ *iff* $(r, s) \Vdash \mathrm{A}$ *or* $(r, s) \Vdash \mathrm{B}$
- $(r, s) \Vdash \mathrm{A} \wedge \mathrm{B}$ *iff* $(r, s) \Vdash \mathrm{A}$ *and* $(r, s) \Vdash \mathrm{B}$
- $(r, s) \Vdash \mathrm{A} \to \mathrm{B}$ *iff* $\forall r' \in \mathbf{R}.$ *if* $r \leqslant^{\mathfrak{r}} r'$ *and* $(r', s) \Vdash \mathrm{A}$ *then* $(r', s) \Vdash \mathrm{B}$
- $(r, s) \Vdash \mathrm{A} * \mathrm{B}$ *iff* $\exists r', r'' \in \mathbf{R}.\, r' \star r'' \leqslant^{\mathfrak{r}} r, (r', s) \Vdash \mathrm{A}$ *and* $(r'', s) \Vdash \mathrm{B}$
- $(r, s) \Vdash \mathrm{A} \mathrel{-\!\!*} \mathrm{B}$ *iff* $\forall r', r'' \in \mathbf{R}.$ *if* $(r', s) \Vdash \mathrm{A}$ *and* $r' \star r \leqslant^{\mathfrak{r}} r''$ *then* $(r'', s) \Vdash \mathrm{B}$
- $(r, s) \Vdash \Box \mathrm{A}$ *iff* $\forall s' \in \mathbf{S}.$ *if* $s \leqslant^{\mathfrak{s}} s'$ *then* $(r, s') \Vdash \mathrm{A}$
- $(r, s) \Vdash \Diamond \mathrm{A}$ *iff* $\exists s' \in \mathbf{S}.\, s \leqslant^{\mathfrak{s}} s'$ *and* $(r, s') \Vdash \mathrm{A}$
- $(r, s) \Vdash \circ \mathrm{A}$ *iff* $\exists s' \in \mathbf{S}.\, s' = \mathfrak{n}(s)$ *and* $(r, s') \Vdash \mathrm{A}$

▶ **Definition 5.** *A formula* A *is* satisfied in an LTBI-model $\mathcal{M}$*, written* $\mathcal{M} \vDash \mathrm{A}$*, iff* $(\epsilon, s) \Vdash \mathrm{A}$ *for all* $s \in \mathbf{S}$*. A formula* A *is* valid*, written* $\vDash \mathrm{A}$*, iff it is satisfied in all* LTBI-*models.*

It is routine to show that conditions $\mathcal{M}_{\mathrm{K}}$ and $\mathcal{M}_{\pi}$ of Definition 3 extend from propositional letters to arbitrary formulas, as stated in the following Lemma.

▶ **Lemma 6.** *For all* LTBI-*models* $\mathcal{M}$*:*
$(\mathcal{M}_{\mathrm{K}})$ $\forall \mathrm{A} \in \mathbf{F}. \forall s \in \mathbf{S}. \forall r, r' \in \mathbf{R}.$ *if* $r \leqslant^{\mathfrak{r}} r'$ *and* $(r, s) \Vdash \mathrm{A}$ *then* $(r', s) \Vdash \mathrm{A}$,
$(\mathcal{M}_{\pi})$ $\forall \mathrm{A} \in \mathbf{F}. \forall s \in \mathbf{S}. (\pi, s) \Vdash \mathrm{A}$.

Let us remark that the resource semantics we use for LTBI is based on total (and not partial) resource monoids to avoid tricky additional definedness conditions. The introduction of a greatest element π at which all formulas are satisfied is therefore required in the presence of $\bot$ (as explained in [7], for example, to enforce the validity of BI formulas such as $\mathrm{A} * (\mathrm{A} \mathrel{-\!\!*} \bot)$ where A is a theorem of intuitionistic logic).

2.2 Expressiveness of LTBI

To illustrate what kind of properties LTBI is able to express, let us consider the timeline $(\mathbf{S} = [2023 - 2025], \leqslant^{\mathfrak{s}}, 2023)$ and the resource monoid $(\mathbf{R} = \mathbb{N} \cup \{\, \infty \,\}, +, 0, \leqslant^{\mathfrak{r}}, \infty)$, where $\leqslant^{\mathfrak{r}}$ and $+$ are the extensions of the standard order and of the standard addition on natural numbers such that $r \leqslant^{\mathfrak{r}} \infty$ and $r + \infty = \infty$ for all $r \in \mathbf{R}$.

Now, let $G = \{\, g_1, g_2, g_3 \,\}$ be a set of goods the price of which (in euros) evolves over the years according to the pricing function $pr : G \times \mathbf{S} \to \mathbb{N}$ given in Table 1.

We can then define the affordability predicate on multisets of goods as follows:

$$\forall (r, s) \in \mathbf{R} \times \mathbf{S}. \quad (r, s) \Vdash Af(gs) \;\text{ iff }\; pr(gs, s) \overset{\text{def}}{=} \sum_{g \in gs} pr(g, s) \leqslant r$$

We write $x_1, \ldots, x_n$ as a shorthand for the multiset $\{\, x_1, \ldots, x_n \,\}$. Therefore, $Af(\{\, g, g' \,\})$ is more shortly written as $Af(g, g')$. It is easy to see that

$$\forall (r, s) \in \mathbf{R} \times \mathbf{S}. \forall g, g' \in G. \quad (r, s) \Vdash Af(g, g') \;\text{ iff }\; (r, s) \Vdash Af(g) * Af(g')$$

As an example, let us suppose that each year, we get an amount of money that we are required to spend buying goods on some dedicated website. LTBI allows us to state properties

	Prices (€)		
good	**2023**	**2024**	**2025**
g_1	2000	2100	2200
g_2	300	250	350
g_3	1700	1800	1500

Table 1 Prices of three goods over the years.

about our ability to buy goods depending on the year and on the amount of money available. For instance,

$$(3000, 2023) \Vdash Af(g_1) \wedge (Af(g_2) * Af(g_3))$$

intuitively means that in 2023 (the current year), with 3000 euros, we can choose to buy g_1 and we can also choose to split our money into two disjoint amounts, the first one to buy g_2 and the second one to buy g_3. Let us remark that although the two options are available to us simultaneously, it does not necessarily imply that we could afford to buy all three goods simultaneously. Indeed, with an amount of 3000 euros, we would have to make a choice since $pr(\{ g_1, g_2, g_3 \}, 2023) = 4000$. Therefore, $(3000, 2023) \nVdash Af(g_1, g_2, g_3)$.

Using the temporal modalities, we can state more complex propositions that take into account the evolution of prices over the years. For instance,

$$(3000, 2023) \Vdash \Box Af(g_2) * (\Diamond Af(g_3) \wedge (Af(g_1) * \circ Af(g_2)))$$

states that in 2023, we can split 3000 euros into two disjoint amounts of money, the first one keeping g_2 affordable every year from 2023 until 2025, the second one bringing us two choices. The first choice ensures that g_3 should become affordable at least one year during between 2023 and 2025. The second choice tells us that we could split our second amount of money once again into two new disjoint amounts, one making g_3 affordable currently (in 2023), the other making g_2 affordable only one year later (in 2024).

3 An LTBI Labelled Tableau Calculus

The labelled tableau calculus for LTBI, called $\mathsf{T}_{\mathsf{LTBI}}$, is in the spirit of the ones for BI [7] and DBI [3] and relies on the introduction of labels and constraints. $\mathsf{T}_{\mathsf{LTBI}}$ deals with two kinds of labels, namely resource labels and state labels.

We shall see that the latter require a careful and specific treatment in order to keep them isomorphic to natural numbers.

3.1 Labels and Constraints

We define a set of state labels and constraints that deals with temporality in order to capture the notion of resource evolution.

▶ **Definition 7** (Resource labels and constraints). *The set L_r of* resource labels *is built from the countable set $\gamma_r = \{ \epsilon_{\mathrm{L}}, \mathrm{c}_1, \mathrm{c}_2, \dots \}$ of* resource constants *and label composition $\circ$ according to the grammar $X ::= \gamma_r \mid X \circ X$. A* resource constraint *is an expression of the form $\mathrm{x} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{y}$, where x and y are resource labels.*

Label composition is interpreted as an associative and commutative operation on L_r that admits ϵ_{L} as its neutral element. We shall frequently write $\mathrm{x}\,\mathrm{y}$ instead of $\mathrm{x} \circ \mathrm{y}$ for convenience. We say that x is a sublabel of y iff there exists $\mathrm{z} \in L_r$ such that $\mathrm{x} \circ \mathrm{z} = \mathrm{y}$ and $E(\mathrm{x})$ denotes the set of sublabels of a label x.

▶ **Definition 8** (State labels and constraints). *The set L_s of* state labels *is built from the countable set $\gamma_s = \{\gamma_0, \gamma_1, \gamma_2, \ldots\}$ of* state constants *and the successor symbol η according to the grammar $X ::= \gamma_s \mid \eta X$. Given two state labels τ and τ', a* state constraint *is an expression of the form $\tau \leqslant^{\mathfrak{s}}_{\mathrm{L}} \tau'$, $\tau <^{s}_{\mathrm{L}} \tau'$, $\tau =^{\mathfrak{s}}_{\mathrm{L}} \tau'$ or $\tau \neq^{\mathfrak{s}}_{\mathrm{L}} \tau'$.*

▶ **Definition 9** (Domain and alphabet). *Let C_r be a set of resource constraints. The* domain *of C_r, denoted $D_r(C_r)$, is the set of all the sublabels occurring in C_r. More formally, $D_r(C_r) = \bigcup_{\mathrm{x} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{y} \in C_r} (E(\mathrm{x}) \cup E(\mathrm{y}))$. The* alphabet *(or* basis*) of C_r is the set $A_r(C_r) = \gamma_r \cap D_r(C_r)$. $D_s(C_s)$ and $A_s(C_s)$, where C_s is a set of state constraints, are defined similarly.*

▶ **Definition 10** (Closure of resource constraints). *Let C_r be a set of resource constraints, the closure $C_r^{\bullet}$ is the smallest set such that $C_r \subseteq C_r^{\bullet}$ that is closed under the following rules:*

$$\frac{\mathrm{x} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{y} \qquad \mathrm{y} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{z}}{\mathrm{x} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{z}} \quad \frac{\mathrm{x} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{y}}{\mathrm{x} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{x}} \quad \frac{\mathrm{x} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{y}}{\mathrm{y} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{y}} \quad \frac{\mathrm{x}\,\mathrm{y} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{x}\,\mathrm{y}}{\mathrm{x} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{x}} \quad \frac{\mathrm{z}\,\mathrm{y} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{z}\,\mathrm{y} \qquad \mathrm{x} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{y}}{\mathrm{z}\,\mathrm{x} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{z}\,\mathrm{y}}$$

These rules reflect the properties of transitivity and reflexivity of $\leqslant^{\mathfrak{r}}_{\mathrm{L}}$ and the compatibility of $\circ$ w.r.t. $\leqslant^{\mathfrak{r}}_{\mathrm{L}}$. Since none of these rules introduce any new resource constant, we have $A_r(C_r) = A_r(C_r^{\bullet})$.

▶ **Definition 11** (Closure of state constraints). *Let C_s be a set of state constraints, the closure $C_s^{\bullet}$ is the smallest set such that $C_s \subseteq C_s^{\bullet}$ that reflects in $\leqslant^{\mathfrak{s}}_{\mathrm{L}}, <^{s}_{\mathrm{L}}, =^{\mathfrak{s}}_{\mathrm{L}}, \neq^{\mathfrak{s}}_{\mathrm{L}}$ the properties of $\leqslant, <, =, \neq$ in $\mathbb{N}$ and such that η syntactically reflects the properties of the "next" function $\mathfrak{n}$.*

▶ **Proposition 12.** *Let C_r be a set of resource constraints:*

1. *If $\mathrm{z}\,\mathrm{x} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{y} \in C_r^{\bullet}$, then $\mathrm{x} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{x} \in C_r^{\bullet}$*
2. *If $\mathrm{x} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{z}\,\mathrm{y} \in C_r^{\bullet}$, then $\mathrm{y} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{y} \in C_r^{\bullet}$*

Proof. From $\mathrm{z}\,\mathrm{x} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{y}$ we get $\mathrm{z}\,\mathrm{x} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{z}\,\mathrm{x}$ (reflexivity), then $\mathrm{x}\,\mathrm{z} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{x}\,\mathrm{z}$ (commutativity) and then $\mathrm{x} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{x}$ (compatibility). The other case is similar. ◀

3.2 Rules of the $\mathrm{T}_{\mathrm{LTBI}}$ Tableau Calculus

▶ **Definition 13** (Labelled Formula). *A* labelled formula *is a quadruple $(\mathbb{S}, \mathrm{A}, \mathrm{x}, \tau)$, denoted $\mathbb{S}\ \mathrm{A} : (\mathrm{x}, \tau)$, where $\mathbb{S} \in \{\mathbb{T}, \mathbb{F}\}$ is a* sign*, $\mathrm{A} \in \mathbf{F}$ is a* formula*, and $(\mathrm{x}, \tau) \in L_r \times L_s$ is a* label.

▶ **Definition 14** (CTSS). *A* constrained temporal set of statements *(CTSS) is a triple noted $\langle \mathcal{F}, C_r, C_s \rangle$, where $\mathcal{F}$ is a set of labelled formulas, C_r is a set of resource constraints and C_s is a set of state constraints. A CTSS is required to satisfy the following condition:*

$$(CTSS_R) \textit{ for all } \mathbb{S}\ \mathrm{A} : (\mathrm{x}, \tau) \in \mathcal{F},\ \mathrm{x} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{x} \in C_r \textit{ and } \tau \leqslant^{\mathfrak{s}}_{\mathrm{L}} \tau \in C_s.$$

A CTSS is finite if all of its three components are finite.

▶ **Definition 15** (Inconsistent Label). *Let $\langle \mathcal{F}, C_r, C_s \rangle$ be a CTSS. The label (x, τ) is* inconsistent *if there exist two resource labels y and z such that $\mathrm{y} \circ \mathrm{z} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{x} \in C_r^{\bullet}$ and $\mathbb{T}\ \bot : (\mathrm{y}, \tau) \in \mathcal{F}$. A label is consistent if it is not inconsistent.*

▶ **Proposition 16.** *Let $\langle \mathcal{F}, C_r, C_s \rangle$ be a CTSS. The following properties hold:*

$$\frac{\mathbb{T}\ \mathrm{A} \wedge \mathrm{B} : (\mathrm{x}, \tau)}{\begin{array}{c}\mathbb{T}\ \mathrm{A} : (\mathrm{x}, \tau)\\ \mathbb{T}\ \mathrm{B} : (\mathrm{x}, \tau)\end{array}} \qquad \begin{array}{c|c}\multicolumn{2}{c}{\mathbb{F}\ \mathrm{A} \wedge \mathrm{B} : (\mathrm{x}, \tau)}\\ \mathbb{F}\ \mathrm{A} : (\mathrm{x}, \tau) & \mathbb{F}\ \mathrm{B} : (\mathrm{x}, \tau)\end{array} \qquad \frac{\mathbb{F}\ \mathrm{A} \vee \mathrm{B} : (\mathrm{x}, \tau)}{\begin{array}{c}\mathbb{F}\ \mathrm{A} : (\mathrm{x}, \tau)\\ \mathbb{F}\ \mathrm{B} : (\mathrm{x}, \tau)\end{array}}$$

$$\begin{array}{c|c}\multicolumn{2}{c}{\mathbb{T}\ \mathrm{A} \vee \mathrm{B} : (\mathrm{x}, \tau)}\\ \mathbb{T}\ \mathrm{A} : (\mathrm{x}, \tau) & \mathbb{T}\ \mathrm{B} : (\mathrm{x}, \tau)\end{array} \qquad \begin{array}{c|c}\multicolumn{2}{c}{\mathbb{T}\ \mathrm{A} \rightarrow \mathrm{B} : (\mathrm{x}, \tau)}\\ \mathbb{R}\ \mathrm{x} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{y} & \mathbb{R}\ \mathrm{x} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{y}\\ \mathbb{F}\ \mathrm{A} : (\mathrm{y}, \tau) & \mathbb{T}\ \mathrm{B} : (\mathrm{y}, \tau)\end{array} \qquad \frac{\mathbb{F}\ \mathrm{A} \rightarrow \mathrm{B} : (\mathrm{x}, \tau)}{\begin{array}{c}\mathbb{A}\ \mathrm{x} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{a}\\ \mathbb{T}\ \mathrm{A} : (\mathrm{a}, \tau)\\ \mathbb{F}\ \mathrm{B} : (\mathrm{a}, \tau)\end{array}}$$

$$\frac{\mathbb{T}\ \mathrm{I} : (\mathrm{x}, \tau)}{\mathbb{A}\ \epsilon_{\mathrm{L}} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{x}} \qquad \frac{\mathbb{T}\ \mathrm{A} * \mathrm{B} : (\mathrm{x}, \tau)}{\begin{array}{c}\mathbb{A}\ \mathrm{ab} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{x}\\ \mathbb{T}\ \mathrm{A} : (\mathrm{a}, \tau)\\ \mathbb{T}\ \mathrm{B} : (\mathrm{b}, \tau)\end{array}} \qquad \begin{array}{c|c}\multicolumn{2}{c}{\mathbb{F}\ \mathrm{A} * \mathrm{B} : (\mathrm{x}, \tau)}\\ \mathbb{R}\ \mathrm{y\,z} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{x} & \mathbb{R}\ \mathrm{y\,z} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{x}\\ \mathbb{F}\ \mathrm{A} : (\mathrm{y}, \tau) & \mathbb{F}\ \mathrm{B} : (\mathrm{z}, \tau)\end{array} \qquad \frac{\mathbb{F}\ \mathrm{A} \mathrel{-\!\!*} \mathrm{B} : (\mathrm{x}, \tau)}{\begin{array}{c}\mathbb{A}\ \mathrm{x\,a} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{b}\\ \mathbb{T}\ \mathrm{A} : (\mathrm{a}, \tau)\\ \mathbb{F}\ \mathrm{B} : (\mathrm{b}, \tau)\end{array}}$$

$$\begin{array}{c|c}\multicolumn{2}{c}{\mathbb{T}\ \mathrm{A} \mathrel{-\!\!*} \mathrm{B} : (\mathrm{x}, \tau)}\\ \mathbb{R}\ \mathrm{x\,y} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{z} & \mathbb{R}\ \mathrm{x\,y} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{z}\\ \mathbb{F}\ \mathrm{A} : (\mathrm{y}, \tau) & \mathbb{T}\ \mathrm{B} : (\mathrm{z}, \tau)\end{array} \qquad \frac{\mathbb{T}\ \circ\mathrm{A} : (\mathrm{x}, \tau)}{\begin{array}{c}\mathbb{A}\ \tau <^{s}_{\mathrm{L}} \eta\tau\\ \mathbb{T}\ \mathrm{A} : (\mathrm{x}, \eta\tau)\end{array}} \qquad \frac{\mathbb{F}\ \circ\mathrm{A} : (\mathrm{x}, \tau)}{\begin{array}{c}\mathbb{R}\ \tau <^{s}_{\mathrm{L}} \eta\tau\\ \mathbb{F}\ \mathrm{A} : (\mathrm{x}, \eta\tau)\end{array}}$$

$$\frac{\mathbb{T}\ \Box\mathrm{A} : (\mathrm{x}, \tau)}{\begin{array}{c}\mathbb{R}\ \tau \leqslant^{\mathfrak{s}}_{\mathrm{L}} \alpha\\ \mathbb{T}\ \mathrm{A} : (\mathrm{x}, \alpha)\end{array}} \qquad \frac{\mathbb{F}\ \Box\mathrm{A} : (\mathrm{x}, \tau)}{\begin{array}{c}\mathbb{A}\ \tau \leqslant^{\mathfrak{s}}_{\mathrm{L}} \upsilon\\ \mathbb{F}\ \mathrm{A} : (\mathrm{x}, \upsilon)\end{array}} \qquad \frac{\mathbb{T}\ \Diamond\mathrm{A} : (\mathrm{x}, \tau)}{\begin{array}{c}\mathbb{A}\ \tau \leqslant^{\mathfrak{s}}_{\mathrm{L}} \alpha\\ \mathbb{T}\ \mathrm{A} : (\mathrm{x}, \alpha)\end{array}} \qquad \frac{\mathbb{F}\ \Diamond\mathrm{A} : (\mathrm{x}, \tau)}{\begin{array}{c}\mathbb{R}\ \tau \leqslant^{\mathfrak{s}}_{\mathrm{L}} \upsilon\\ \mathbb{F}\ \mathrm{A} : (\mathrm{x}, \upsilon)\end{array}}$$

$$\begin{array}{c|c}\multicolumn{2}{c}{\mathrm{CD}}\\ \mathbb{R}\ \tau \leqslant^{\mathfrak{s}}_{\mathrm{L}} \upsilon & \mathbb{R}\ \tau \leqslant^{\mathfrak{s}}_{\mathrm{L}} \upsilon\\ \mathbb{A}\ \tau <^{s}_{\mathrm{L}} \upsilon & \mathbb{A}\ \tau =^{\mathfrak{s}}_{\mathrm{L}} \upsilon\end{array} \qquad \begin{array}{c|c}\multicolumn{2}{c}{\mathrm{LR}}\\ \mathbb{R}\ \tau \leqslant^{\mathfrak{s}}_{\mathrm{L}} \upsilon & \mathbb{R}\ \tau \leqslant^{\mathfrak{s}}_{\mathrm{L}} \upsilon\\ \mathbb{R}\ \tau \leqslant^{\mathfrak{s}}_{\mathrm{L}} \zeta & \mathbb{R}\ \tau \leqslant^{\mathfrak{s}}_{\mathrm{L}} \zeta\\ \mathbb{A}\ \upsilon \leqslant^{\mathfrak{s}}_{\mathrm{L}} \zeta & \mathbb{A}\ \zeta \leqslant^{\mathfrak{s}}_{\mathrm{L}} \upsilon\end{array} \qquad \frac{\mathbb{S}\ \mathrm{A} : (\mathrm{c}, \tau)}{\begin{array}{c}\mathbb{R}\ \tau =^{\mathfrak{s}}_{\mathrm{L}} \upsilon\\ \mathbb{S}\ \mathrm{A} : (\mathrm{c}, \upsilon)\end{array}}$$

Figure 1 Rules of the $\mathsf{T}_{\mathsf{LTBI}}$ calculus.

1. *If* $\mathrm{y} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{x} \in C^{\bullet}_{r}$ *and* (x, τ) *is consistent, then* (y, τ) *is a consistent label.*
2. *If* $\mathrm{x} \circ \mathrm{y} \in D_r(C^{\bullet}_{r})$ *and* $(\mathrm{x} \circ \mathrm{y}, \tau)$ *is consistent, then* (x, τ) *and* (y, τ) *are consistent.*

Proof. Assume that (y, τ) is inconsistent, then there are two resource labels z, z' and a state label τ such that $\mathrm{z} \circ \mathrm{z}' \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{y} \in C^{\bullet}_{r}$ and $\mathbb{T} \perp : (\mathrm{z}, \tau) \in \mathcal{F}$. By transitivity with $\mathrm{y} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{x} \in C^{\bullet}_{r}$ we get $\mathrm{z} \circ \mathrm{z}' \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{x} \in C^{\bullet}_{r}$, meaning that (x, τ) is inconsistent, which contradicts our assumption. The other proof is similar. ◀

The rules of $\mathsf{T}_{\mathsf{LTBI}}$ are presented in Figure 1, where a, b denote fresh resource constants and α denotes a fresh state constant. We observe that some of the rules introduce fresh constants and label constraints called *assertions*. For instance, expanding a labelled formula $\mathbb{F}\ \mathrm{A} \rightarrow \mathrm{B} : (\mathrm{x}, \tau)$ generates a (resource) assertion $\mathbb{A}\ \mathrm{x} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{a}$ where a is a fresh resource constant. Similarly, expanding a labelled formula $\mathbb{T}\ \Diamond\mathrm{A} : (\mathrm{x}, \tau)$ generates a (state) assertion $\mathbb{A}\ \tau \leqslant^{\mathfrak{s}}_{\mathrm{L}} \alpha$ where α is a fresh state constant. We also observe that some of the rules introduce label constraints on arbitrary labels called *requirements*. For instance, expanding a labelled formula $\mathbb{T}\ \mathrm{A} \rightarrow \mathrm{B} : (\mathrm{x}, \tau)$ generates a (resource) requirement $\mathbb{R}\ \mathrm{x} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{y}$. Similarly, expanding a labelled formula $\mathbb{F}\ \Diamond\mathrm{A} : (\mathrm{x}, \tau)$ generates a (state) requirement $\mathbb{R}\ \tau \leqslant^{\mathfrak{s}}_{\mathrm{L}} \upsilon$.

Before we explain how requirements work, let us note that a tableau branch $\mathcal{B}$ corresponds to a CTSS $\langle \mathcal{F}, C_r, C_s \rangle$, where $\mathcal{F}$ is the set of all labelled formulas occurring in $\mathcal{B}$ and C_r, C_s

are the sets of all resource and state assertions occurring in $\mathcal{B}$ respectively, i.e. $C_r = \{\, \mathbb{A}\ \mathrm{x} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{y} \mid \mathbb{A}\ \mathrm{x} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{y} \in \mathcal{B} \,\}$ and $C_s = \{\, \mathbb{A}\ \mathrm{x}\ \mathrm{R}^{s}_{\mathrm{L}}\ \mathrm{y} \mid \mathbb{A}\ \mathrm{x}\ \mathrm{R}^{s}_{\mathrm{L}}\ \mathrm{y} \in \mathcal{B} \,\}$ for $\mathrm{R}^{s}_{\mathrm{L}} \in \{\, \leqslant^{\mathfrak{s}}_{\mathrm{L}}, <^{s}_{\mathrm{L}}, =^{\mathfrak{s}}_{\mathrm{L}}, \neq^{\mathfrak{s}}_{\mathrm{L}} \,\}$. Now if we want to expand a labelled formula $\mathbb{T}\ \mathrm{A} \to \mathrm{B} : (\mathrm{x}, \tau)$ occurring in $\mathcal{B}$, the label constraint $\mathbb{R}\ \mathrm{x} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{y}$ requires us to find a label y such that $\mathrm{x} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{y} \in C_r^{\bullet}$, i.e., a label y for which the requirement is derivable from (the closure of) the assertions that already occur in the branch.

The last line of Figure 1 presents the structural rules of $\mathsf{T}_{\mathsf{LTBI}}$. The first one is the case distinction rule CD that disambiguates any label state constraint $\tau \leqslant^{\mathfrak{s}}_{\mathrm{L}} \upsilon$ derivable from the closure of the state assertions (hence the requirement $\mathbb{R}\ \tau \leqslant^{\mathfrak{s}}_{\mathrm{L}} \upsilon$) w.r.t. $<^{s}_{\mathrm{L}}$ and $=^{\mathfrak{s}}_{\mathrm{L}}$. The second one is the linearizing rule LR that arranges any pair of state labels υ and ζ branching from τ into a linear order $\tau \leqslant^{\mathfrak{s}}_{\mathrm{L}} \upsilon \leqslant^{\mathfrak{s}}_{\mathrm{L}} \zeta$ or $\tau \leqslant^{\mathfrak{s}}_{\mathrm{L}} \zeta \leqslant^{\mathfrak{s}}_{\mathrm{L}} \upsilon$. The last one is the equality rewriting rule which is there mostly for convenience to make the closing of a branch easier to check.

▶ **Definition 17.** *A tableau for a formula* A *is a tableau built inductively according to the rules depicted in Figure 1 the root node of which is the labelled formula* $\mathbb{F}\ \mathrm{A} : (\epsilon_{\mathrm{L}}, \gamma_0)$.

Definition 17 implies that a $\mathsf{T}_{\mathsf{LTBI}}$ tableau for a LTBI formula A begins with the initial CTSS $\langle \mathbb{F}\ \mathrm{A} : (\epsilon_{\mathrm{L}}, \gamma_0), \{\, \epsilon_{\mathrm{L}} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \epsilon_{\mathrm{L}} \,\}, \{\, \gamma_0 \leqslant^{\mathfrak{s}}_{\mathrm{L}} \gamma_0 \,\} \rangle$. Moreover, we define a rule application strategy according to the following order of precedence from highest to lowest:

1. The rules $\mathbb{T}\,\mathrm{I}$, $\mathbb{F} \to$, $\mathbb{T} *$, $\mathbb{F} \mathrel{-\!\!*}$, $\mathbb{T} \Diamond$, $\mathbb{F} \Box$, $\mathbb{T} \circ$ and $\mathbb{F} \circ$, called π_α-rules, take precedence over the other rules.
2. The structural rules CD and LR have middle precedence.
3. The rules $\mathbb{T} \to$, $\mathbb{F} *$, $\mathbb{T} \mathrel{-\!\!*}$, $\mathbb{F} \Diamond$, $\mathbb{T} \Box$, called π_β-rules, have low precedence.

▶ **Definition 18** (Closing conditions)**.** *A CTSS* $\langle \mathcal{F}, C_r, C_s \rangle$ *is* closed *if it satisfies one of the following conditions:*

1. $\mathbb{T}\ \mathrm{A} : (\mathrm{x}, \tau) \in \mathcal{F}$*,* $\mathbb{F}\ \mathrm{A} : (\mathrm{y}, \upsilon) \in \mathcal{F}$*,* $\mathrm{x} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{y} \in C_r^{\bullet}$ *and* $\tau =^{\mathfrak{s}}_{\mathrm{L}} \upsilon \in C_s^{\bullet}$.
2. $\mathbb{F}\ \mathrm{I} : (\mathrm{x}, \tau) \in \mathcal{F}$ *and* $\epsilon_{\mathrm{L}} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{x} \in C_r^{\bullet}$
3. $\mathbb{F}\ \top : (\mathrm{x}, \tau) \in \mathcal{F}$
4. $\mathbb{F}\ \mathrm{A} : (\mathrm{x}, \tau) \in \mathcal{F}$ *and* (x, τ) *is inconsistent*
5. $\tau =^{\mathfrak{s}}_{\mathrm{L}} \upsilon \in C_s^{\bullet}$ *and* $\tau \neq^{\mathfrak{s}}_{\mathrm{L}} \upsilon \in C_s^{\bullet}$.

A tableau branch is closed if its corresponding CTSS is closed. A CTSS, or a tableau branch, is open if it is not closed. A tableau is closed if all of its branches are closed.

▶ **Definition 19** ($\mathsf{T}_{\mathsf{LTBI}}$-proof)**.** *A* $\mathsf{T}_{\mathsf{LTBI}}$*-proof for a formula* A *is a closed* $\mathsf{T}_{\mathsf{LTBI}}$ *tableau for* A.

▶ **Example 20.** Let us now illustrate in Figure 2 the construction of a $\mathsf{T}_{\mathsf{LTBI}}$ tableau with an example leading to a closed tableau.

We start with $\mathbb{F}\ \Diamond\mathrm{A} \wedge \Diamond\mathrm{B} \to \Diamond(\mathrm{A} \wedge \Diamond\mathrm{B}) \vee \Diamond(\mathrm{B} \wedge \Diamond\mathrm{A}) : (\epsilon_{\mathrm{L}}, \gamma_0)$. In Step [2], expanding $\mathbb{T}\ \Diamond\mathrm{A} \wedge \Diamond\mathrm{B} : (\mathrm{c}_1, \gamma_0)$ introduces $\mathbb{T}\ \Diamond\mathrm{A} : (\mathrm{c}_1, \gamma_0)$ and $\mathbb{T}\ \Diamond\mathrm{B} : (\mathrm{c}_1, \gamma_0)$. After Steps [3, 4], we obtain two assertions $\mathbb{A}\ \gamma_0 \leqslant^{\mathfrak{s}}_{\mathrm{L}} \gamma_1$ and $\mathbb{A}\ \gamma_0 \leqslant^{\mathfrak{s}}_{\mathrm{L}} \gamma_1$. In Step [5] we expand the signed formula $\mathbb{F}\ \Diamond(\mathrm{A} \wedge \Diamond\mathrm{B}) \vee \Diamond(\mathrm{B} \wedge \Diamond\mathrm{A}) : (\mathrm{c}_1, \gamma_0)$ and then generate $\mathbb{F}\ \Diamond(\mathrm{A} \wedge \Diamond\mathrm{B}) : (\mathrm{c}_1, \gamma_0)$ and $\mathbb{F}\ \Diamond(\mathrm{B} \wedge \Diamond\mathrm{A}) : (\mathrm{c}_1, \gamma_0)$.

Before expanding them, we apply the linearizing rule LR in Step [6] and the tableau splits into two branches: the left one with the assertion $\mathbb{A}\ \gamma_1 \leqslant^{\mathfrak{s}}_{\mathrm{L}} \gamma_2$ and the right one with the assertion $\mathbb{A}\ \gamma_2 \leqslant^{\mathfrak{s}}_{\mathrm{L}} \gamma_1$. Now we consider Step [7] in the left branch (with assertion $\mathbb{A}\ \gamma_1 \leqslant^{\mathfrak{s}}_{\mathrm{L}} \gamma_2$) that corresponds to the expansion of $\mathbb{F}\ \Diamond(\mathrm{A} \wedge \Diamond\mathrm{B}) : (\mathrm{c}_1, \gamma_0)$ introducing a requirement $\mathbb{R}\ \gamma_0 \leqslant^{\mathfrak{s}}_{\mathrm{L}} \mathrm{v}_1$ and the labelled formula $\mathbb{F}\ \mathrm{A} \wedge \Diamond\mathrm{B} : (\mathrm{c}_1, \mathrm{v}_1)$ with v_1 a variable to be instantiated from the closure of the assertions in the branch. Here we choose $\mathrm{v}_1 = \gamma_1$ in order to satisfy the requirement.

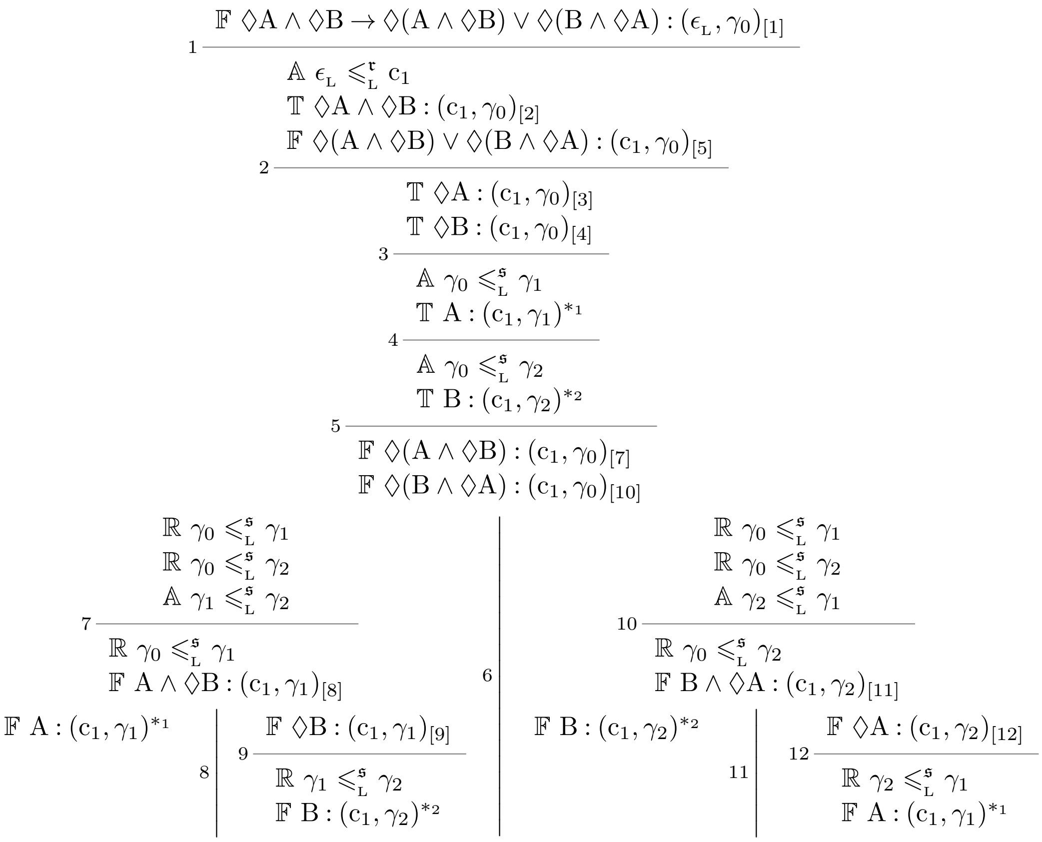

Figure 2 Closed Tableau for $\Diamond A \wedge \Diamond B \to \Diamond(A \wedge \Diamond B) \vee \Diamond(B \wedge \Diamond A)$.

Then, in Step [8] $\mathbb{F}\ A \wedge \Diamond B : (c_1, \gamma_1)$ splits the leftmost branch into two sub-branches. The first one is closed because it contains both $\mathbb{T}\ A : (c_1, \gamma_1)$, and $\mathbb{F}\ \Diamond B : (c_1, \gamma_1)$. The second one continues with Step [9] that introduces a requirement $\mathbb{R}\ \gamma_1 \leqslant^{\mathfrak{s}}_{\mathrm{L}} v_2$ and the labelled formula $\mathbb{F}\ B : (c_1, v_2)$ with v_2 a variable to be instantiated from the closure of the assertions in the branch. Here we choose $v_2 = \gamma_2$ that satisfies the requirement because $\gamma_1 \leqslant^{\mathfrak{s}}_{\mathrm{L}} \gamma_2$. Then we obtain the labelled formula $\mathbb{F}\ B : (c_1, \gamma_2)$ and the branch is closed because it also contains $\mathbb{T}\ B : (c_1, \gamma_2)$. The tableau on the right(hand side of Step [6] similarly leads to closed branches.

▶ **Example 21.** Let us now illustrate in Figure 3 the construction of a $\mathsf{T}_{\mathsf{LTBI}}$ tableau with an example leading to a non closed tableau.

We start with $\mathbb{F}\ (\Diamond A * \circ B) \to (\Diamond B * \circ A) : (\epsilon_{\mathrm{L}}, \gamma_0)$. Then, Step [2], $\mathbb{T}\ \Diamond A * \circ B : (c_1, \gamma_0)$ introduces the assertion $\mathbb{A}\ c_2\, c_3 \leqslant^{\mathfrak{r}}_{\mathrm{L}} c_1$ and to the labelled formulae $\mathbb{T}\ \Diamond A : (c_2, \gamma_0)$ and $\mathbb{T}\ \circ B : (c_3, \gamma_0)$. In Step [3] we expand the first one and generate an assertion $\mathbb{A}\ \gamma_0 \leqslant^{\mathfrak{s}}_{\mathrm{L}} \gamma_1$ and the labelled formula $\mathbb{T}\ A : (c_2, \gamma_1)$. In Step [4] we expand the second one and generate the labelled formula $\mathbb{T}\ B : (c_3, \eta\gamma_0)$. Step [5] deals with the labelled formula $\mathbb{F}\ \Diamond B * \circ A : (c_1, \gamma_0)$ and its expansion rules creates two branches: the left one with the requirement $\mathbb{R}\ y\, z \leqslant^{\mathfrak{r}}_{\mathrm{L}} c_1$ and the labelled formula $\mathbb{F}\ \Diamond B : (c_1, \gamma_0)$ and the right one with the requirement $\mathbb{R}\ y\, z \leqslant^{\mathfrak{r}}_{\mathrm{L}} c_1$ and the labelled formula $\mathbb{F}\ \circ A : (z, \gamma_0)$.

Let us consider the left branch. The requirement $\mathbb{R}\ y\, z \leqslant^{\mathfrak{r}}_{\mathrm{L}} c_1$ can only be satisfied in two cases: (1) $y = c_3$, $z = c_2$ and (2) $y = c_2$, $z = c_3$. Step [6] in the left branch corresponds to the expansion of $\mathbb{F}\ \Diamond B : (y, \gamma_0)$. It generates the requirement $\mathbb{R}\ \gamma_0 \leqslant^{\mathfrak{s}}_{\mathrm{L}} v$ and the labelled

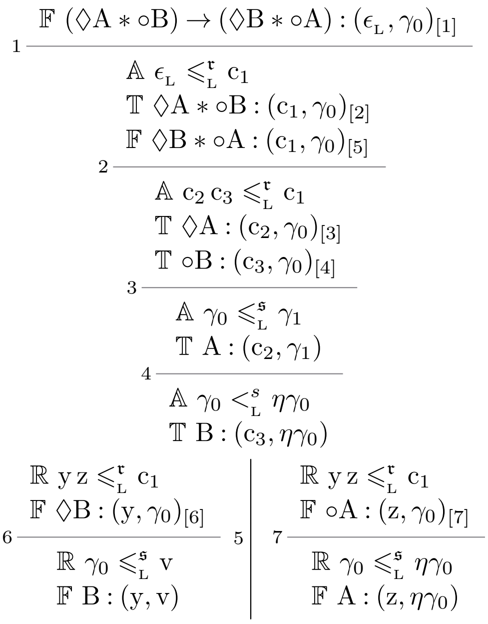

Figure 3 Non-closed Tableau for $(\Diamond \mathrm{A} * \circ \mathrm{B}) \to (\Diamond \mathrm{B} * \circ \mathrm{A})$.

formula $\mathbb{F}\ \mathrm{B} : (\mathrm{y}, \mathrm{v})$. In order to be able to close the branch with $\mathbb{T}\ \mathrm{B} : (\mathrm{c}_3, \eta\gamma_0)$ we have to set $\mathrm{y} = \mathrm{c}_3$ (with $\mathrm{z} = \mathrm{c}_2$) and to instantiate the variable v such that $\gamma_0 \leqslant^{\mathfrak{s}}_{\mathrm{L}} \mathrm{v}$. If we instantiate v with $\eta\gamma_0$ we satisfy the requirement $\mathbb{R}\ \gamma_0 \leqslant^{\mathfrak{s}}_{\mathrm{L}} \mathrm{v}$ and then the branch is closed.

Let us consider the right branch branch in which the requirement $\mathbb{R}\ \mathrm{yz} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{c}_1$ is satisfied with $\mathrm{y} = \mathrm{c}_3, \mathrm{z} = \mathrm{c}_2$. Step [7] in the left branch corresponds to the expansion of $\mathbb{F}\ \circ \mathrm{A} : (\mathrm{c}_2, \gamma_0)$ that generates the labelled formula $\mathbb{F}\ \mathrm{A} : (\mathrm{c}_2, \eta\gamma_0)$. We observe that we cannot close this branch with the latter labelled formula and $\mathbb{T}\ \mathrm{A} : (\mathrm{c}_2, \gamma_1)$ because there is no possible equality between γ_1 and $\eta\gamma_0$. Then in case (1) there is an open branch and the tableau is not closed. Developing case (2) also leads to an open branch.

4 Soundness of $\mathsf{T}_{\mathsf{LTBI}}$

In this section, we prove the soundness of $\mathsf{T}_{\mathsf{LTBI}}$ following a method based on the notion of *realizability* of a CTSS that is similar to the one used for various flavours of BI [5].

▶ **Definition 22** (Realization). *A* realization *of a CTSS* $\langle \mathcal{F}, C_r, C_s \rangle$ *is a triple* $(\mathcal{M}, [.]_r, [.]_s)$, *where* $\mathcal{M}$ *is an* LTBI*-model, and* $[.]_r$, $[.]_s$ *are order preserving homomorphisms from resource and state labels to resources and states respectively. More precisely, we have* $[.]_r : D_r(C_r^\bullet) \to \mathbf{R}$ *and* $[.]_s : D_s(C_s^\bullet) \to \mathbf{S}$, *such that:*

- $[\epsilon_{\mathrm{L}}]_r = \epsilon$, $[\mathrm{x} \circ \mathrm{y}]_r = [\mathrm{x}]_r \star [\mathrm{y}]_r$, $[\eta\tau]_s = \mathfrak{n}[\tau]_s$
- *If* $\mathbb{T}\ \mathrm{A} : (\mathrm{x}, \tau) \in \mathcal{F}$, *then* $([\mathrm{x}]_r, [\tau]_s) \Vdash \mathrm{A}$
- *If* $\mathbb{F}\ \mathrm{A} : (\mathrm{x}, \tau) \in \mathcal{F}$, *then* $([\mathrm{x}]_r, [\tau]_s) \nVdash \mathrm{A}$
- *If* $\mathrm{x} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{y} \in C_r$, *then* $[\mathrm{x}]_r \leqslant^{\mathfrak{r}} [\mathrm{y}]_r$
- *If* $\tau\, \mathrm{R}^s_{\mathrm{L}}\, \upsilon \in C_s$, *then* $[\tau]_s\, \mathrm{R}^s\, [\upsilon]_s$, *with* $\mathrm{R}^s \in \{ \leqslant^{\mathfrak{s}}, <^s, =^{\mathfrak{s}}, \neq^{\mathfrak{s}} \}$

A CTSS (or branch) is *realizable* if it has a realization. A tableau is *realizable* if it has at least one realizable branch.

▶ **Lemma 23.** *Let $(\mathcal{M}, [.]_r, [.]_s)$ be a realization of a CTSS $\langle \mathcal{F}, C_r, C_s \rangle$. For all* $\mathrm{x} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{y} \in C_r^{\bullet}$ *and for all* $\tau \, \mathrm{R}^s_{\mathrm{L}} \, \upsilon \in C_s^{\bullet}$, $[\mathrm{x}]_r \leqslant^{\mathfrak{r}} [\mathrm{y}]_r$ *and* $[\tau]_s \, \mathrm{R}^s \, [\upsilon]_s$.

Proof. Straightforward since the closure rules for C_r and C_s preserve compatibility. ◀

▶ **Lemma 24.** *If a* $\mathsf{T}_{\mathsf{LTBI}}$ *tableau is closed then it is not realizable.*

Proof. If a closed tableau is realizable then it contains at least one branch $\mathcal{B}$ that is realizable in a LTBI-model.

- If the branch is closed with complementary formulas $\mathbb{T}\ \mathrm{A} : (\mathrm{x}, \tau)$ and $\mathbb{F}\ \mathrm{A} : (\mathrm{y}, \tau)$ then by Definition 22 we have $\mathrm{x} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{y}$. By Lemma 23, we have $[\mathrm{x}]_r \leqslant^{\mathfrak{r}} [\mathrm{y}]_r$ and since the branch is realized, by Definition 22, we have $([\mathrm{x}]_r, [\tau]_s) \Vdash \mathrm{A}$ and $([\mathrm{y}]_r, [\tau]_s) \nVdash \mathrm{A}$. We thus reach a contradiction since by Lemma 6 (monotonicity) $([\mathrm{y}]_r, [\tau]_s) \Vdash \mathrm{A}$.
- if the branch is closed because of $\mathbb{F}\ \top : (\mathrm{x}, \tau)$, then $([\mathrm{x}]_r, [\tau]_s) \nVdash \top$, which is a contradiction.
- The other cases are similar.

◀

▶ **Lemma 25.** *All* $\mathsf{T}_{\mathsf{LTBI}}$ *rules preserve realizability.*

Proof. Let $\mathcal{B}$ be a tableau branch and $(\mathcal{M}, [.]_r, [.]_s)$ be a realization of its CTSS $\langle \mathcal{F}, C_r, C_s \rangle$. We proceed by case analysis on the rule that expands $\mathcal{B}$.

The cases for BI connectives are similar to the ones given in [7] for BI tableaux. We thus only consider the modal operators.

- Case $\mathbb{T} \circ$:
  Suppose that the labelled formula $\mathbb{T} \circ\mathrm{A} : (\mathrm{x}, \tau)$ has just been expanded in the branch $\mathcal{B}$. Then, $\mathcal{B}$ is extended with a new labelled formula $\mathbb{T}\ \mathrm{A}{:}(\mathrm{x}, \eta\tau)$ and a new assertion $\mathbb{A}\ \tau <^s_{\mathrm{L}} \eta\tau$. Since $\mathcal{B}$ was realizable before the expansion, we have $([\mathrm{x}]_r, [\tau]_s) \Vdash \circ\mathrm{A}$. Therefore, there exists s' such that $s' = \mathfrak{n}[\tau]_s$ and $([\mathrm{x}]_r, s') \Vdash \mathrm{A}$. Since $\mathfrak{n}[\tau]_s = [\eta\tau]_s$ and $[\tau]_s <^s [\eta\tau]_s$, both $\mathbb{T}\ \mathrm{A} : (\mathrm{x}, \eta\tau)$ and $\mathbb{A}\ \tau <^s_{\mathrm{L}} \eta\tau$ are realized.
- Case $\mathbb{F} \circ$:
  Suppose that the labelled formula $\mathbb{F} \circ\mathrm{A} : (\mathrm{x}, \tau)$ has just been expanded in the branch $\mathcal{B}$. Then, $\mathcal{B}$ is extended with a new labelled formula $\mathbb{F}\ \mathrm{A} : (\mathrm{x}, \eta\tau)$ and a new requirement $\mathbb{R}\ \tau <^s_{\mathrm{L}} \eta\tau$. A valid application of the expansion rule requires that $\tau <^s_{\mathrm{L}} \eta\tau \in C_s^{\bullet}$. Since $\mathcal{B}$ was realizable before the expansion, we have $([\mathrm{x}]_r, [\tau]_s) \nVdash \circ\mathrm{A}$ and Lemma 23 entails $[\tau]_s <^s [\eta\tau]_s$. Since $\mathfrak{n}[\tau]_s = [\eta\tau]_s$, $([\mathrm{x}]_r, [\tau]_s) \nVdash \circ\mathrm{A}$ implies $([\mathrm{x}]_r, [\eta\tau]_s) \nVdash \mathrm{A}$ by definition. Therefore, both $\mathbb{F}\ \mathrm{A} : (\mathrm{x}, \eta\tau)$ and $\mathbb{R}\ \tau <^s_{\mathrm{L}} \eta\tau$ are realized.
- The other cases are similar.

◀

▶ **Theorem 26** (Soundness). *If there exists a* $\mathsf{T}_{\mathsf{LTBI}}$ *proof for* A, *then* A *is valid.*

Proof. Let $\mathcal{T}$ be a $\mathsf{T}_{\mathsf{LTBI}}$-proof of A. Assume that A is not valid, then there exists a linear resource model $\mathcal{M}$ such that $(\epsilon, s) \nVdash \mathrm{A}$ for some state s. Since the initial CTSS $\langle \{\, \mathbb{F}\ \mathrm{A} : (\epsilon_{\mathrm{L}}, \gamma_0) \,\}, \{\, \epsilon_{\mathrm{L}} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \epsilon_{\mathrm{L}} \,\}, \{\, \gamma_0 \leqslant^{\mathfrak{s}}_{\mathrm{L}} \gamma_0 \,\} \rangle$ is trivially realizable by setting $[\gamma_0]_s = s$, Lemma 25 implies that the tableau $\mathcal{T}$ contains at least one realizable branch, which contradicts the fact that $\mathcal{T}$ is a tableau proof. Indeed, if $\mathcal{T}$ is a tableau proof for A, then all of its branches should be closed by definition, and thus not realizable by Lemma 24. Therefore, A is valid. ◀

5 Completeness

In this section we discuss the reasons why the completeness result for $\mathsf{T}_{\mathsf{LTBI}}$ is not trivial and still an open problem.

A usual way of proving the completeness of a labelled tableau calculus is by counter-model construction from an open and completed branch, as we did for BI [7], BBI [12] and various modal extensions of BI [3, 4]. This approach requires the definition of a suitable notion of what it means for a labelled formula to be completely analyzed or fulfilled. Although such a definition can be given for $\mathsf{T}_{\mathsf{LTBI}}$, the completion of an open branch raises several issues.

▶ **Definition 27.** *Let* $\langle \mathcal{F}, C_s, C_r \rangle$ *be the CTSS associated to a tableau branch* $\mathcal{B}$*. A labelled formula* $\mathbb{S}\ \mathrm{C} : (\mathrm{x}, \tau)$ *is* fulfilled *(or* completely analyzed*) in* $\mathcal{B}$*, denoted* $\mathcal{B} \vDash \mathbb{S}\ \mathrm{C} : (\mathrm{x}, \tau)$*, iff:*

- *Base cases:*
  - $\mathcal{B} \vDash \mathbb{S}\ \top : (\mathrm{x}, \tau)$ *always*
  - $\mathcal{B} \vDash \mathbb{S}\ \bot : (\mathrm{x}, \tau)$ *always*
  - $\mathcal{B} \vDash \mathbb{T}\ \mathrm{I} : (\mathrm{x}, \tau)$ *iff* $\epsilon_{\mathrm{L}} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{x} \in C_r^\bullet$
  - $\mathcal{B} \vDash \mathbb{F}\ \mathrm{I} : (\mathrm{x}, \tau)$ *always*
  - $\mathcal{B} \vDash \mathbb{T}\ \mathrm{p} : (\mathrm{x}, \tau)$ *iff* $\mathbb{T}\ \mathrm{p} : (\mathrm{y}, \tau) \in \mathcal{F}$ *for some* $\mathrm{y} \neq \mathrm{x}$ *such that* $\mathrm{y} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{x} \in C_r^\bullet$
  - $\mathcal{B} \vDash \mathbb{F}\ \mathrm{p} : (\mathrm{x}, \tau)$ *iff* $\mathbb{F}\ \mathrm{p} : (\mathrm{y}, \tau) \in \mathcal{F}$ *for some* $\mathrm{y} \neq \mathrm{x}$ *such that* $\mathrm{x} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{y} \in C_r^\bullet$
- *Induction:*
  - $\mathcal{B} \vDash \mathbb{T}\ \mathrm{A} \wedge \mathrm{B} : (\mathrm{x}, \tau)$ *iff* $\mathcal{B} \vDash \mathbb{T}\ \mathrm{A} : (\mathrm{x}, \tau)$ *and* $\mathcal{B} \vDash \mathbb{T}\ \mathrm{B} : (\mathrm{x}, \tau)$
  - $\mathcal{B} \vDash \mathbb{F}\ \mathrm{A} \wedge \mathrm{B} : (\mathrm{x}, \tau)$ *iff* $\mathcal{B} \vDash \mathbb{F}\ \mathrm{A} : (\mathrm{x}, \tau)$ *or* $\mathcal{B} \vDash \mathbb{F}\ \mathrm{B} : (\mathrm{x}, \tau)$
  - $\mathcal{B} \vDash \mathbb{T}\ \mathrm{A} \vee \mathrm{B} : (\mathrm{x}, \tau)$ *iff* $\mathcal{B} \vDash \mathbb{T}\ \mathrm{A} : (\mathrm{x}, \tau)$ *or* $\mathcal{B} \vDash \mathbb{T}\ \mathrm{B} : (\mathrm{x}, \tau)$
  - $\mathcal{B} \vDash \mathbb{F}\ \mathrm{A} \vee \mathrm{B} : (\mathrm{x}, \tau)$ *iff* $\mathcal{B} \vDash \mathbb{F}\ \mathrm{A} : (\mathrm{x}, \tau)$ *and* $\mathcal{B} \vDash \mathbb{F}\ \mathrm{B} : (\mathrm{x}, \tau)$
  - $\mathcal{B} \vDash \mathbb{T}\ \mathrm{A} * \mathrm{B} : (\mathrm{x}, \tau)$ *iff* $\mathcal{B} \vDash \mathbb{T}\ \mathrm{A} : (\mathrm{y}, \tau)$ *and* $\mathcal{B} \vDash \mathbb{T}\ \mathrm{B} : (\mathrm{z}, \tau)$ *for some* $\mathrm{y}\,\mathrm{z} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{x} \in C_r^\bullet$
  - $\mathcal{B} \vDash \mathbb{F}\ \mathrm{A} * \mathrm{B} : (\mathrm{x}, \tau)$ *iff* $\mathcal{B} \vDash \mathbb{F}\ \mathrm{A} : (\mathrm{y}, \tau)$ *and* $\mathcal{B} \vDash \mathbb{F}\ \mathrm{B} : (\mathrm{z}, \tau)$ *for all* $\mathrm{y}\,\mathrm{z} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{x} \in C_r^\bullet$
  - $\mathcal{B} \vDash \mathbb{T}\ \mathrm{A} \rightarrow \mathrm{B} : (\mathrm{x}, \tau)$ *iff* $\mathcal{B} \vDash \mathbb{F}\ \mathrm{A} : (\mathrm{y}, \tau)$ *or* $\mathcal{B} \vDash \mathbb{T}\ \mathrm{B} : (\mathrm{y}, \tau)$ *for all* $\mathrm{x} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{y} \in C_r^\bullet$
  - $\mathcal{B} \vDash \mathbb{F}\ \mathrm{A} \rightarrow \mathrm{B} : (\mathrm{x}, \tau)$ *iff* $\mathcal{B} \vDash \mathbb{T}\ \mathrm{A} : (\mathrm{y}, \tau)$ *and* $\mathcal{B} \vDash \mathbb{F}\ \mathrm{B} : (\mathrm{y}, \tau)$ *for some* $\mathrm{x} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{y} \in C_r^\bullet$
  - $\mathcal{B} \vDash \mathbb{T}\ \mathrm{A} \mathrel{-\!\!*} \mathrm{B} : (\mathrm{x}, \tau)$ *iff* $\mathcal{B} \vDash \mathbb{F}\ \mathrm{A} : (\mathrm{y}, \tau)$ *or* $\mathcal{B} \vDash \mathbb{T}\ \mathrm{B} : (\mathrm{z}, \tau)$ *for all* $\mathrm{x}\,\mathrm{y} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{z} \in C_r^\bullet$
  - $\mathcal{B} \vDash \mathbb{F}\ \mathrm{A} \mathrel{-\!\!*} \mathrm{B} : (\mathrm{x}, \tau)$ *iff* $\mathcal{B} \vDash \mathbb{T}\ \mathrm{A} : (\mathrm{y}, \tau)$ *and* $\mathcal{B} \vDash \mathbb{F}\ \mathrm{B} : (\mathrm{z}, \tau)$ *for some* $\mathrm{x}\,\mathrm{y} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{z} \in C_r^\bullet$
  - $\mathcal{B} \vDash \mathbb{S}\ \circ\mathrm{A} : (\mathrm{x}, \tau)$ *iff* $\mathcal{B} \vDash \mathbb{S}\ \mathrm{A} : (\mathrm{y}, \eta\tau)$
  - $\mathcal{B} \vDash \mathbb{T}\ \Diamond\mathrm{A} : (\mathrm{x}, \tau)$ *iff* $\mathcal{B} \vDash \mathbb{T}\ \mathrm{A} : (\mathrm{y}, \upsilon)$ *for some* $\tau \leqslant^{\mathfrak{s}}_{\mathrm{L}} \upsilon \in C_s^\bullet$
  - $\mathcal{B} \vDash \mathbb{F}\ \Diamond\mathrm{A} : (\mathrm{x}, \tau)$ *iff* $\mathcal{B} \vDash \mathbb{F}\ \mathrm{A} : (\mathrm{y}, \upsilon)$ *for all* $\tau \leqslant^{\mathfrak{s}}_{\mathrm{L}} \upsilon \in C_s^\bullet$
  - $\mathcal{B} \vDash \mathbb{T}\ \Box\mathrm{A} : (\mathrm{x}, \tau)$ *iff* $\mathcal{B} \vDash \mathbb{T}\ \mathrm{A} : (\mathrm{y}, \upsilon)$ *for all* $\tau \leqslant^{\mathfrak{s}}_{\mathrm{L}} \upsilon \in C_s^\bullet$
  - $\mathcal{B} \vDash \mathbb{F}\ \Box\mathrm{A} : (\mathrm{x}, \tau)$ *iff* $\mathcal{B} \vDash \mathbb{F}\ \mathrm{A} : (\mathrm{y}, \upsilon)$ *for some* $\tau \leqslant^{\mathfrak{s}}_{\mathrm{L}} \upsilon \in C_s^\bullet$

▶ **Definition 28.** *A branch* $\mathcal{B}$ *is* completed *(also* saturated*) if all of its labelled formulas are fulfilled and all possible expansions of the structural rules* CD *and* LR *have been applied.*

It is folklore to define a completion procedure for an open branch by defining a fair strategy for formula expansion (see [5, 4] for details). The actual problem is to turn an open and completed branch into a suitable LTBI counter-model.

5.1 Counter-Model Construction

Let us first illustrate how to construct a counter-model from an open and completed branch using the leftmost open branch of the tableau depicted in Figure 3.

Firstly, we define the set of resources as the set $D_r(C^{\bullet}_r) \cup \{\, \pi \,\}$ and the composition of resources as:

$$\begin{cases} \mathrm{x} \star \mathrm{y} = \mathrm{xy} \text{ if } \mathrm{xy} \in D_r(C^{\bullet}_r) \\ \mathrm{x} \star \epsilon_{\mathrm{L}} = \mathrm{x} \\ \mathrm{x} \star \pi = \pi \end{cases}$$

The resource ordering $\leqslant^{\mathfrak{r}}$ is induced by the closure of the resource assertions occurring in the branch, i.e.:

$$\leqslant^{\mathfrak{r}} = \; C^{\bullet}_r \cup \{\, \mathrm{x} \leqslant \pi \mid \mathrm{x} \in D_r(C^{\bullet}_r), \,\}$$

which, in our example, corresponds to the following transitive and reflevixe closure of the set of relations:

$$\{\, \epsilon_{\mathrm{L}} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{c}_1, \mathrm{c}_2\mathrm{c}_3 \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{c}_1 \,\}$$

augmented with π as the greatest element.

Secondly, the timeline is defined as the set $\{\, 0, 1, 2 \,\}$ with the state labels realized (interpreted) as follows: $[\gamma_0]_s = 0$, $[\eta\gamma_0]_s = 1$, $[\gamma_1]_s = 2$.

Thirdly, the forcing relation is induced by the following LTBI-valuation that matches the positive labelled formulas (those with a sign $\mathbb{T}$) occurring in the branch:

$$\begin{cases} [\mathrm{A}] = \{\, (\pi, 0), (\pi, 1), (\pi, 2), (\mathrm{c}_2, 2) \,\} \\ [\mathrm{B}] = \{\, (\pi, 0), (\pi, 1), (\pi, 2), (\mathrm{c}_3, 2) \,\} \end{cases}$$

Finally, the reason why we have an actual counter-model can be read directly from the labelled formulas of the completed open branch:

1. We have $(\mathrm{c}_2, 2) \Vdash \mathrm{A}$ (by definition), which implies $(\mathrm{c}_2, 0) \Vdash \Diamond\mathrm{A}$.
2. Moreover, we have $(\mathrm{c}_3, 1) \Vdash \mathrm{B}$ (by definition) and thus we get $(\mathrm{c}_3, 0) \Vdash \circ\mathrm{B}$.
3. From 1 and 2, we get $(\mathrm{c}_2\mathrm{c}_3, 0) \Vdash \Diamond\mathrm{A} * \circ\mathrm{B}$ which implies $(\mathrm{c}_1, 0) \Vdash \Diamond\mathrm{A} * \circ\mathrm{B}$ by Kripke monotonicity (as $\mathrm{c}_2\mathrm{c}_3 \leqslant \mathrm{c}_1$ by definition).
4. Besides, we have $(\mathrm{c}_0, 0) \nVdash \Diamond\mathrm{B} * \circ\mathrm{A}$ because $(\mathrm{x}, \tau) \nVdash \circ\mathrm{A}$ for all resources x and all states τ (since the timeline has no state 3 and A is only true at $(\mathrm{c}_2, 2)$).

The first and third points (construction of a total resource monoid and of a forcing relation) described above work in the general case for any open and completed branch, not just for the tableau depicted in Figure 3. The second point (construction of discrete linear timeline) is however more problematic.

5.2 The Dense Timeline Issue

A first issue in $\mathsf{T}_{\mathsf{LTBI}}$ is that the completion procedure might result in a set of state constraints that, although representing a discrete linear order, might not be isomorphic to any subset of $(\mathbb{N}, \leqslant)$ because it might be dense.

Let us for example consider the tableau depicted in Fig. 4. Its leftmost branch grows infinitely because the $\pi\beta$-formula $\mathbb{T}\ (\Diamond\mathrm{A} \to \mathrm{B}) \to \mathrm{C}$ contains a $\pi\alpha$-subformula $\mathbb{F}\ \Diamond\mathrm{A} \to \mathrm{B}$ the expansion of which repeatedly generates new resource constants $\mathrm{c}_2, \mathrm{c}'_2, \mathrm{c}''_2, \mathrm{c}^i_2$ $(i > 2)$ to be fed to the $\pi\beta$-formula for its fulfillment. For instance in Step [3], the resource assertion $\mathbb{A}\ \mathrm{c}_1 \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{c}_2$ is generated, where c_2 is fresh. Then, in Step [4], the state assertion $\mathbb{A}\ \gamma_0 \leqslant^{\mathfrak{s}}_{\mathrm{L}} \gamma_1$ is generated, where γ_1 is fresh. Since the requirement $\mathbb{R}\ \mathrm{c}_1 \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{c}_2$ is met, Step [2] must be

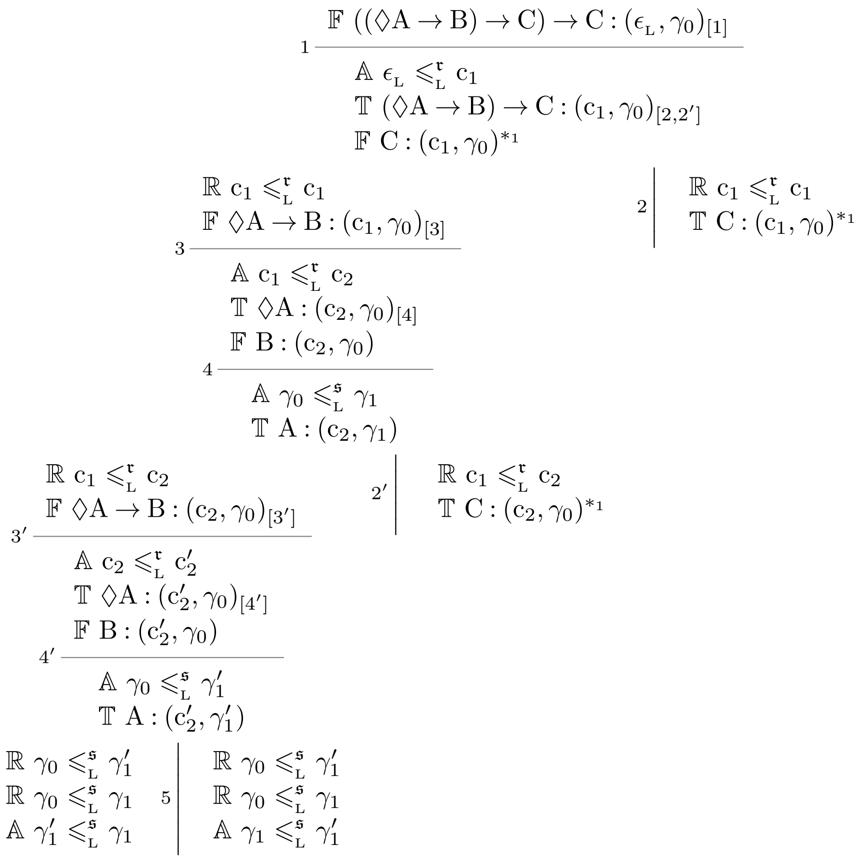

Figure 4 Liberizable Infinite Tableau.

reproduced with c_2 instead of c_1, which gives Step [2′]. After Step [2′], Steps [3′] and [4′] reproduce Steps [3] and [4] leading to new assertions $\mathbb{A}\ \mathrm{c}_2 \leqslant_{\mathrm{L}}^{\mathfrak{r}} \mathrm{c}_2'$ and $\mathbb{A}\ \gamma_0 \leqslant_{\mathrm{L}}^{\mathfrak{s}} \gamma_1'$.

After Step [4′], we get two state labels γ_1 and γ_1' that are not linearly ordered. We therefore use the linearizing rule LR in Step [5] to get (in the leftmost branch) the assertion $\mathbb{A}\ \gamma_1' \leqslant_{\mathrm{L}}^{\mathfrak{s}} \gamma_1$. Several applications of the case distinction rule CD (not represented in Fig. 4 for conciseness) allow us to get the following ordering of the state labels: $\gamma_0 <_{\mathrm{L}}^{s} \gamma_1' <_{\mathrm{L}}^{s} \gamma_1$. Repeating the previous steps infinitely many times we can generate a strictly decreasing infinite chain of state labels $(\gamma_1^i)_{i\in\mathbb{N}}$ between γ_0 and γ_1.

The situation described in Fig. 4 well illustrates the fact that our logic LTBI is not a simple and orthogonal combination of BI and LTL connectives, but induces an actual interaction between resource and state labels. Indeed, the infinite chain of state labels γ_1^i derives from the creation of an infinite chain of resource labels c_2^i.

5.3 Unsoundness of the Liberalized Rules

Tableau branches that might grow infinitely because of the creation of infinitely many fresh labels is a problem that already occurs in tableaux for BI [7]. In the case of BI, such a situation can be remedied using liberalized versions of the tableaux rules that allow the reuse of previously generated labels under specific conditions.

For example, the rule $\mathbb{F}\!\to$ would be allowed to expand $\mathbb{F}\ \mathrm{A} \to \mathrm{B} : (\mathrm{x}, \tau)$ to $\mathbb{T}\ \mathrm{A} : (\mathrm{x}, \tau), \mathbb{F}\ \mathrm{B} : (\mathrm{x}, \tau)$ without generating a fresh (resource) constant whenever the branch already contains a

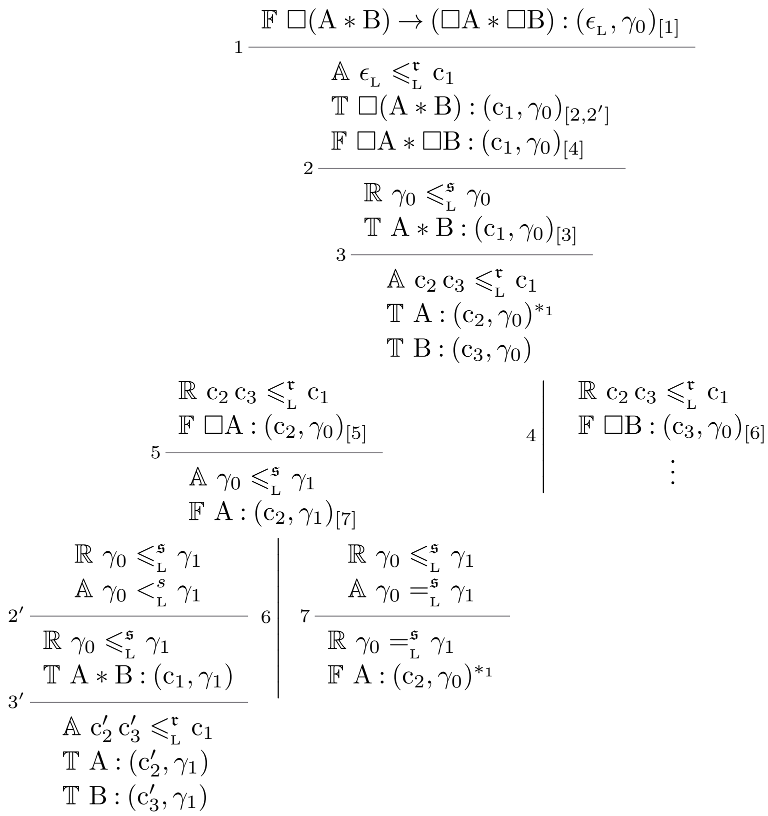

Figure 5 Unliberizable Infinite Tableau.

labelled formula $\mathbb{T}\ \mathrm{A} : (\mathrm{y}, \tau)$ for which the requirement $\mathbb{R}\ \mathrm{y} \leqslant^{\mathfrak{r}}_{\mathrm{L}} \mathrm{x}$ is met. Under the liberalized version of $\mathbb{F} \to$, the leftmost branch of the tableau depicted in Fig. 4 would be completed after Step [3′] since the introduction of $\mathbb{T}\ \Diamond\mathrm{A} : (\mathrm{c}_2, \gamma_0)$ in Step [3] would allow Step [3′] to reuse c_2 instead of generating a fresh c'_2, making Step [3′] a redundant copy of Step [3] adding no new information to the branch.

It would be tempting to think that adopting the liberalized rules given for BI in [7] would solve the problem of getting an infinite amount of state labels from the generation of an infinite number of fresh resource labels. Unfortunately, our second issue is that this approach does not work, as illustrated in Fig. 5.

The liberalized rule for $\mathbb{T} \ast$ (resp. $\mathbb{F} \mathrel{-\!\!*}$) in BI tableaux only generates fresh constants for the first instance of a labelled formula $\mathbb{T}\ \mathrm{A} \ast \mathrm{B} : \mathrm{x}$ (or $\mathbb{F}\ \mathrm{A} \mathrel{-\!\!*} \mathrm{B} : \mathrm{x}$) in a tableau branch. Every subsequent instance of the same labelled formula in the same branch is allowed to reuse the constants that have been generated by the expansion of the first instance.

After Step [4], the tableau described in Fig. 5 splits into two branches, the second one being similar to the first one (replacing occurences of A with B) and thus not fully depicted in the figure for conciseness. As easily checked, repeating Steps [2] through [6] makes the leftmost branch of the tableau grow infinitely. The repetitions Step [3^i] of Step [3] generate infinitely many decompositions $\mathrm{c}^i_2\,\mathrm{c}^i_3 (i \in \mathbb{N})$ of the resource constant c_1. In turn, this leads to the repetitions Step [5^i] of Step [5] which generate infinitely many state labels γ^i_1 and state assertions $\mathbb{A}\ \gamma_0 \leqslant^{\mathfrak{s}}_{\mathrm{L}} \gamma^i_1$.

Using the liberalized version of $\mathbb{T} \ast$ in Step [3′] as in BI tableaux would result in reusing the constants c_2 and c_3 generated during Step [3] instead of introducing the new constants

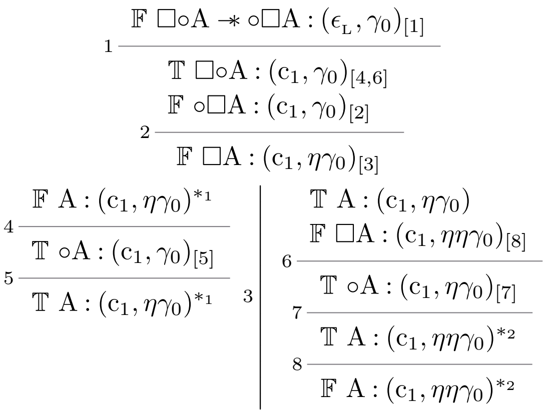

Figure 6 Tableau with Bounded Timeline of Length 3.

c_2' and c_3'. The branch would then be closed, having both $\mathbb{T}\ \mathrm{A} : (\mathrm{c}_2, \gamma_1)$ from Step [3′] and $\mathbb{F}\ \mathrm{A} : (\mathrm{c}_2, \gamma_1)$ from Step [5]. Proceeding similarly in the branch that is eluded in Fig. 5, we would finally get a closed $\mathsf{T}_{\mathsf{LTBI}}$ tableau for a formula which is not valid in LTBI. This shows that the liberalized rules for BI are not sound for LTBI.

5.4 Non-equivalence of LTBI and tBI

In BI tableaux, the soundness of the liberalized rules (as well as the decidability arguments for BI) does not rely on the widespread Kripke resource semantics of BI, but rather on its Beth resource semantics (see [7] for details). The fact that the liberalized rules are unsound for $\mathsf{T}_{\mathsf{LTBI}}$ suggests that replacing the Kripke resource monoid in Definition 2 with a Beth resource monoid would yield a non-equivalent resource semantics for LTBI.

In [10], both a logic called tBI (mixing LTL and BI) for linear bounded timelines and a corresponding purely syntactic sound and complete sequent style proof-system called GtBI are introduced. The semantics of tBI is an extension of the Grothendieck topological resource semantics of BI. The GtBI sequent system is an extension of LBI, the standard bunched sequent calculus of BI. The Grothendieck topological semantics of BI is shown in [7] to be equivalent to its Beth resource semantics w.r.t. provability in LBI, more precisely, for any BI formula A, we have $\models_{\mathsf{Beth}} \mathrm{A} \Leftrightarrow \vdash_{\mathsf{LBI}} \mathrm{A} \Leftrightarrow \models_{\mathsf{Grot}} \mathrm{A}$. Therefore, the unsoundness of the liberalized rules for $\mathsf{T}_{\mathsf{LTBI}}$ proves that even if we would extend GtBI to deal with unbounded timelines, it would be hopeless to try to show the completeness of $\mathsf{T}_{\mathsf{LTBI}}$ by translating proofs of GtBI (with liberalized rules) into closed $\mathsf{T}_{\mathsf{LTBI}}$ tableaux.

More importantly, as stated in Definition 5, the validity of a formula in $\mathsf{T}_{\mathsf{LTBI}}$ only depends on its satisfiability in all time states for the empty resource ϵ, while its validity in tBI depends on its satisfiability in all time states for all resources in the underlying Grothendieck resource monoid. Consequently, although seemingly (syntactically) similar, LTBI and tBI are semantically distinct logics and the results obtained for tBI in [10] do not apply to LTBI.

5.5 The Bounded Timeline Case

We can solve the completeness issues discussed previously by restricting the semantics of LTBI to bounded timelines. It is well known that LTL with bounded time domains can prove almost all of the typical axioms of unbounded LTL. Moreover, practical uses of LTL almost always consider bounded time domains.

Let us assume a bounded timeline $\mathbf{S} = \mathbf{S}_n = \{\, i < n \mid i \in \mathbb{N} \,\}$ of length $n \in \mathbb{N}^*$. Using the fixpoint definitions of the modal operators, we can derive a new tableau system $\mathsf{T}^{\mathsf{n}}_{\mathsf{LTBI}}$ in which the rules $\mathbb{T}\Diamond$ and $\mathbb{F}\Box$ of $\mathsf{T}_{\mathsf{LTBI}}$ are replaced by the following fixpoint rules:

- when $i < n - 1$:

$$\begin{array}{c} \mathbb{T}\ \Diamond\mathrm{A} : (\mathrm{x}, \eta^i\gamma_0) \\ \begin{array}{c|l} \mathbb{T}\ \mathrm{A} : (\mathrm{x}, \eta^i\gamma_0) & \mathbb{F}\ \mathrm{A} : (\mathrm{x}, \eta^i\gamma_0) \\ & \mathbb{T}\ \Diamond\mathrm{A} : (\mathrm{x}, \eta^{i+1}\gamma_0) \end{array} \end{array} \qquad \begin{array}{c} \mathbb{F}\ \Box\mathrm{A} : (\mathrm{x}, \eta^i\gamma_0) \\ \begin{array}{c|l} \mathbb{F}\ \mathrm{A} : (\mathrm{x}, \eta^i\gamma_0) & \mathbb{T}\ \mathrm{A} : (\mathrm{x}, \eta^i\gamma_0) \\ & \mathbb{F}\ \Box\mathrm{A} : (\mathrm{x}, \eta^{i+1}\gamma_0) \end{array} \end{array}$$

- when $i = n - 1$:

$$\frac{\mathbb{T}\ \Diamond\mathrm{A} : (\mathrm{x}, \eta^i\gamma_0)}{\mathbb{T}\ \mathrm{A} : (\mathrm{x}, \eta^i\gamma_0)} \qquad \frac{\mathbb{F}\ \Box\mathrm{A} : (\mathrm{x}, \eta^i\gamma_0)}{\mathbb{F}\ \mathrm{A} : (\mathrm{x}, \eta^i\gamma_0)}$$

Let us remark that we distinguish two cases (when $i < n-1$ and when $i = n-1$) because in our semantics (as described in Definition 4), the truth of the next modality requires the existence of a successor. A semantics in which the next modality is true whenever interpreted in a time state which is out of the bounds (as in tBI) can be obtained by using only the first pair of rules (the forking rules) in any case. Figure 6 gives an example of a closed bounded tableau of length 3 for the formula $\Box\circ\mathrm{A} \mathrel{-\!\!*} \circ\Box\mathrm{A}$.

With the fixpoint rules, we claim the following completeness result for bounded tableaux:

▷ Claim 29. $\mathsf{T}^{\mathsf{n}}_{\mathsf{LTBI}}$ is complete for bounded timelines of length n.

Proof. **(Sketch)** We first observe that in $\mathsf{T}_{\mathsf{LTBI}}$ the only rules that can introduce new state labels are the rules $\mathbb{T}\Diamond$ and $\mathbb{F}\Box$. In $\mathsf{T}^{\mathsf{n}}_{\mathsf{LTBI}}$ those rules are replaced with the fixpoint rules that no longer introduce new state labels, but create terms of the form $\eta^i\gamma_0$ from the root state label γ_0. Therefore, once γ_0 is interpreted as 0 and η is interpreted as the successor function, the generated timeline cannot be dense. Finally, since there are only finitely many terms of the form $\eta^i\gamma_0$ with $0 \leqslant i < n$, the tableau branch completion procedure necessarily terminates. Now, if the completion procedure results in an open branch, the counter-model construction procedure described in Section 5.1 yields an actual counter-model for the initial formula at the root of the tableau branch. ◁

6 Conclusion and Perspectives

In this paper we introduced a new resource logic called LTBI that mixes BI and LTL unary connectives. We proposed a labelled tableau proof system $\mathsf{T}_{\mathsf{LTBI}}$ for LTBI and proved its soundness. We discussed the various and non-trivial completeness issues that arise when trying to show the completeness of $\mathsf{T}_{\mathsf{LTBI}}$ in the general case of an unbounded timeline.

A first perspective is to give a detailed proof of the completeness result claimed previously for bounded timelines.

A second perspective is to extend the completeness result to unbounded timelines. Such an extension would necessarily require the definition of a cyclic proof system with some form of induction to decide when the fixpoint rules should stop forking. Closing conditions for sequent style cyclic proof systems have been given in the literature for unbounded LTL and the task is not at all trivial (as explained in [1]). It is presently unclear to us how to adapt such cyclic closing conditions in the context of a labelled tableau calculus and in the presence of BI multiplicative connectives.

A third perspective is to study variants of LTBI, for example variants that incorporate the binary temporal connectives U and R (until and release), or variants where the underlying resource composition is bounded (e.g. $r^n = \pi$ when $n > p$ for some $p \in \mathbb{N}^*$) or satisfies more specific axioms (e.g., $r \star r \leqslant^{\mathfrak{r}} r$).

References

1. Kai Brünnler and Martin Lange. Cut-free sequent systems for temporal logic. *The Journal of Logic and Algebraic Programming*, 76(2):216–225, 2008.
2. Edmund M Clarke and I Anca Draghicescu. Expressibility results for Linear-time and Branching-time Logics. In *Workshop/School/Symposium of the REX Project (Research and Education in Concurrent Systems), LNCS 354*, pages 428–437. Springer, 1988.
3. Jean-René Courtault and Didier Galmiche. A modal BI Logic for Dynamic Resource Properties. In *Int. Symposium on Logical Foundations of Computer Science, LFCS 2013, LNCS 7734*, pages 134–148. Springer, 2013.
4. Jean-René Courtault and Didier Galmiche. A Modal Separation Logic for Resource Dynamics. *Journal of Logic and Computation*, 28(4):733–778, 2018.
5. Didier Galmiche and Daniel Méry. Semantic Labelled Tableaux for Propositional BI without $\perp$. *Journal of Logic and Computation*, 13(5):707–753, 2003.
6. Didier Galmiche and Daniel Méry. Tableaux and Resource Graphs for Separation Logic. *Journal of Logic and Computation*, 20(1):189–231, 2010.
7. Didier Galmiche, Daniel Méry, and David Pym. The semantics of BI and Resource Tableaux. *Mathematical Structures in Computer Science*, 15(6):1033–1088, 2005.
8. Jean-Yves Girard. Linear logic. *Theoretical Computer Science*, 50(1):1−101, 1987.
9. Samin S Ishtiaq and Peter W. O'Hearn. BI as an Assertion language for Mutable Data Structures. In *Proceedings of the 28th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, pages 14−26, 2001.
10. Norohiro Kamide. Temporal BI: proof system, semantics and translations. *Theoretical Computer Science*, 492:40–69, 2013.
11. Fred Kröger and Stephan Merz. *Temporal Logic and State Systems (Texts in Theoretical Computer Science. an EATCS Series)*. Springer Publishing Company, Incorporated, 2008.
12. Dominique Larchey-Wendling and Didier Galmiche. The Undecidability of Boolean BI through phase Semantics. In *2010 25th Annual IEEE Symposium on Logic in Computer Science*, pages 140−149. IEEE, 2010.
13. Peter W. O'Hearn and David J. Pym. The Logic of Bunched Implications. *Bulletin of Symbolic Logic*, 5(2):215–244, 1999.
14. David J. Pym. *The Semantics and Proof Theory of the Logic of Bunched Implications*, volume 26. Applied Logic Series. Kluwer Academic Publishers, 2002.
15. John C. Reynolds. Separation Logic: A Logic for Shared Mutable Data Structures. *17th Annual IEEE Symposium on Logic in Computer Science (LICS'02)*, pages 55–74, 2002.
16. Kristin Y. Rozier. Linear Temporal Logic Symbolic Model Checking. *Computer Science Review*, 5(2):163–203, 2011.