Dynamic Bunched Implications Logic

J.R. Courtault - D. Galmiche

ANR DynRes Meeting - Nancy

February 2012

Introduction

Resource logics

- Formulae-as-resources
  - Linear Logic **LL** (production / consumption) (Girard 1987)
  - Logic of Bunched Implications **BI** (separation / sharing) (Pym 2002)
  - Ambients (concurrency / mobility) (Cardelli-Gordon 2000)
- Logical Kernel: $\mathbf{BI} = \begin{cases} \wedge, \vee, \rightarrow, \top, \bot \text{ (additives)} \\ *, \mathrel{-\!\!*}, I \text{ (multiplicatives)} \end{cases}$
- Extensions:
  - Pointer Logic **PL** ($x \mapsto a, b$) (O'Hearn et al. 2001)
  - **BI-Loc** (locations) (Biri-Galmiche 2007)
  - **MBI**: **SCRP** ($R, E \xrightarrow{a} R', E'$) + **BI** + modalities (Pym-Toft 2006)

▶ Dynamic Bunched Implications (**DBI**)

Introduction

BI: instantaneous resource consumption

- "I have got gasoline and a car which consumes gasoline":
  BI: $G * (C \wedge (G \mathrel{-\!\!*} \neg G))$
- Problem:
  BI: $G * (C \wedge (G \mathrel{-\!\!*} \neg G)) \models \neg G$
- New approach:
  DBI: $G * (C \wedge (G \mathrel{-\!\!*} \Diamond \neg G)) \not\models \neg G$

- **DBI**: resource consumption is not instantaneous.

Introduction

BI : Mutual exclusion

- System model:

$$\frac{}{R \longrightarrow R} \qquad \frac{}{R \bullet j \bullet e_{nc} \longrightarrow R \bullet e_c}$$

$$\frac{}{R \bullet e_c \longrightarrow R \bullet j \bullet e_{nc}} \qquad \frac{R_1 \longrightarrow R_1' \qquad R_2 \longrightarrow R_2'}{R_1 \bullet R_2 \longrightarrow R_1' \bullet R_2'}$$

- Property to prove:

$$e_{nc} \bullet e_{nc} \bullet j, s_0 \vDash \Box\neg(E_c * E_c * \top)$$

▶ Cannot be expressed in **BI**

▶ **DBI**: convergence/invariant properties on resources (in a transition system).

BI: Modelling latency

- Synchronisation of two web-services WS_a and WS_b.
  - WS_{depart}: simultaneously calls a method on WS_a et WS_b.
  - Communication channels: $C_{depart \to a}$ and $C_{depart \to b}$.

- Model:
  - $WS_{depart} \equiv go_a * go_b$
  - $C_{depart \to a} \equiv \Box(go_a \twoheadrightarrow \Diamond go_{a_{transmis}})$
  - $C_{depart \to b} \equiv \Box(go_b \twoheadrightarrow \Diamond go_{b_{transmis}})$
  - $WS_a \equiv \Box(go_{a_{transmis}} \twoheadrightarrow exec_a)$
  - $WS_b \equiv \Box(go_{b_{transmis}} \twoheadrightarrow exec_b)$

Introduction

BI: Modelling latency

- Property to prove:

$$WS_{depart} * C_{depart \to a} * C_{depart \to b} * WS_a * WS_b$$

$$\models$$

$$\Diamond(exec_a * exec_b)$$

- Not provable in **DBI** (messages do not arrive at the same time)
- Provable in **BI** (without modalities)
- **DBI** captures latencies and message loss

Introduction

BI: Place accessibility

- $r, s \vDash \phi$ (resource r in state s that satisfies ϕ).
- New signification: resource r **in place** s that satisfies ϕ.
- A building example
  - $\Box\Diamond(\mathit{emergencyExit} * \top)$
  - $\Box(\mathit{alarm} * \mathit{fireExtinguisher} * \top)$
  - $(\Diamond\Box\neg(\mathit{emergencyExit} * \top)) \rightarrow \mathit{danger}$

▶ State accessibility viewed as place accessibility

Introduction

Dynamic resource logics

- **BI** logic: static properties of resources
- **MBI** logic: dynamic properties of resources (H-M style)
- **DBI** logic: dynamic properties of resources (modalities)
  - to model **dynamic** resources (manipulated by a system)
  - to express convergence properties

▶ **DBI** logic

- Language and semantics
- Tableaux and sequent calculi
- Counter-model extraction

Plan

Plan

DBI Logic - semantics

Language - semantics

$$\phi ::= p \mid \top \mid \bot \mid I \mid \phi \wedge \phi \mid \phi \vee \phi \mid \phi \to \phi \mid \phi * \phi \mid \phi \mathrel{-\!\!*} \phi \mid \Diamond\phi \mid \Box\phi$$

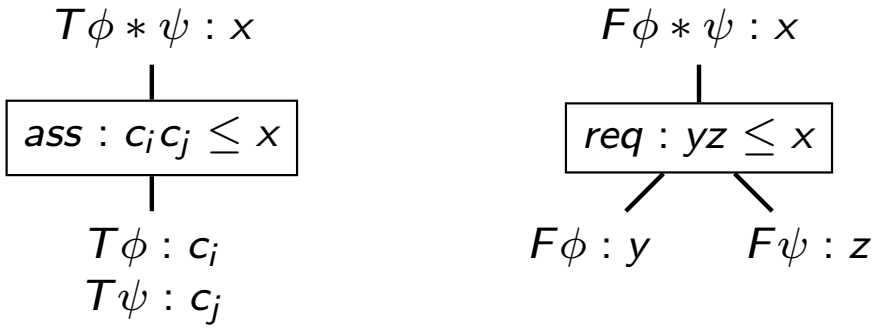
$$\neg\phi \equiv \phi \to \bot$$

- **Dynamic** resource monoid: $\mathcal{M} = (R, \bullet, e, \pi, \sqsubseteq, S, \preceq)$
  - Resource monoid $(R, \bullet, e, \pi, \sqsubseteq)$
  - S set of states
  - $\preceq \subseteq S \times S$ a preorder
- Interpretation: $[\![-]\!] : \mathit{Prop} \to \mathbb{P}(R \times S)$
- Dynamic resource model: $\mathcal{K} = (\mathcal{M}, [\![-]\!], \vDash)$
  - $r, s \vDash \Diamond\phi$ iff $\exists s' \in S \cdot s \preceq s'$ and $r, s' \vDash \phi$
  - $r, s \vDash \Box\phi$ iff $\forall s' \in S \cdot s \preceq s' \Rightarrow r, s' \vDash \phi$

DBI Logic - proof theory

A DBI tableau calculus

- An extension of **BI** calculus (Galmiche-Méry-Pym 2005)
- Resource and state labels/constraints
- Tableaux **BI** rules:



- Signed formulas: $S\phi : x, u$
- Assertions/requirements: resources and states
- Resource graphs and States graphs

DBI Logic - proof theory

Labels

- Resource labels
  - Unit label: 1
  - C_r countable set of constants ($C_r = \{1, c_1, c_2 ...\}$)
  - $\circ$ is associative, commutative and 1 is unit of $\circ$ ($x \circ y$ noted xy)
  - **Resource constraints:** $x \leq y$
- State labels
  - L_s countable set of constants
  - **State constraints:** $x \lhd y$
- Closure of resource constraints: **Resource graph**
- Closure of state constraints: **State graph**

DBI Logic - proof theory

Modal rules

- **Assertion** rules
  - Introduction of new labels and assertions
  - Modification and closure of state graph

$$\frac{F\Box\phi : x, u}{\boxed{ass_s : u \lhd l_i}} \qquad \frac{T\Diamond\phi : x, u}{\boxed{ass_s : u \lhd l_i}}$$
$$F\phi : x, l_i \qquad T\phi : x, l_i$$

- **Requirement** rules
  - Requirements must be verify in the state graph.

$$\frac{T\Box\phi : x, u}{\boxed{req_s : u \lhd v}} \qquad \frac{F\Diamond\phi : x, u}{\boxed{req_s : u \lhd v}}$$
$$T\phi : x, v \qquad F\phi : x, v$$

Plan

DBI-tableau method

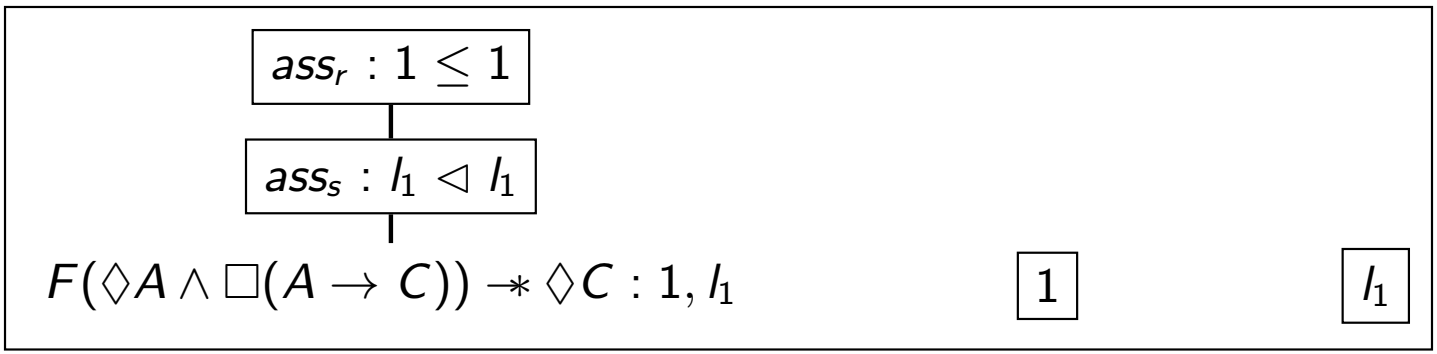
A Tableau method for ϕ

- Initialisation
- Expansion rules applying
- If all branches are closed: ϕ is valid
- If a branch is an Hintikka branch: counter-model extraction

DBI-tableau method

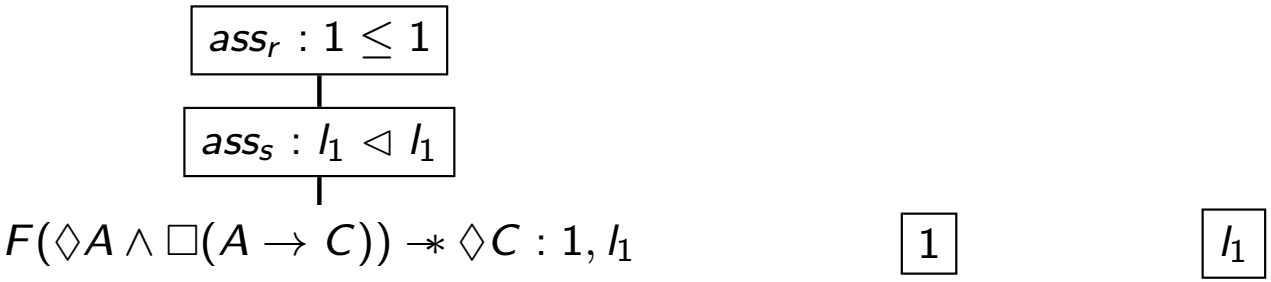
How to prove $(\Diamond A \wedge \Box(A \rightarrow C)) \twoheadrightarrow \Diamond C$?

Initialisation:

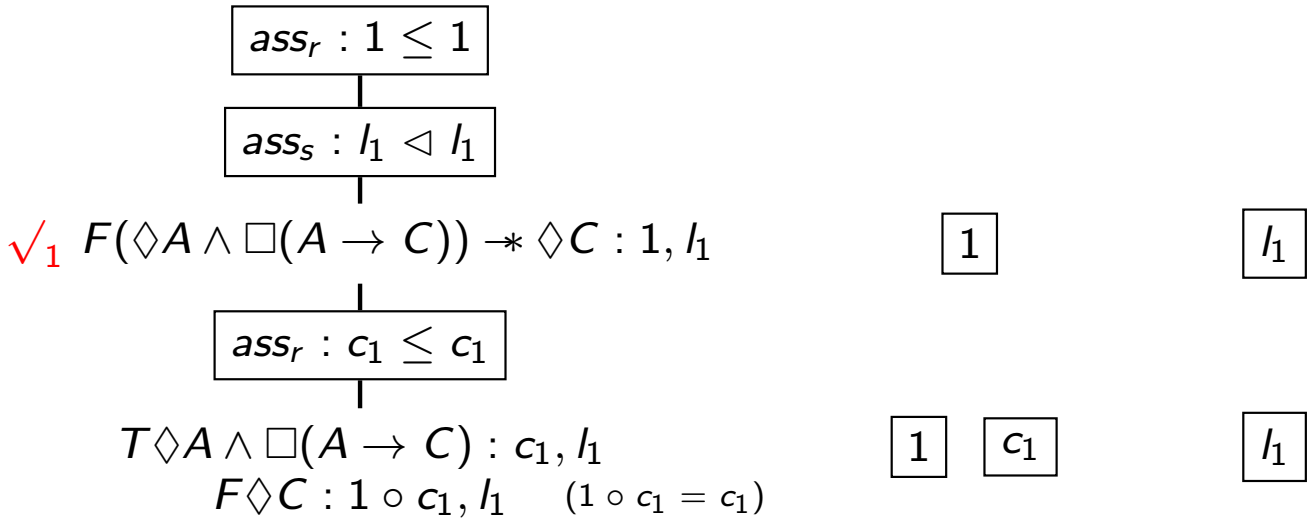


Implicit reflexive et transitive arrows

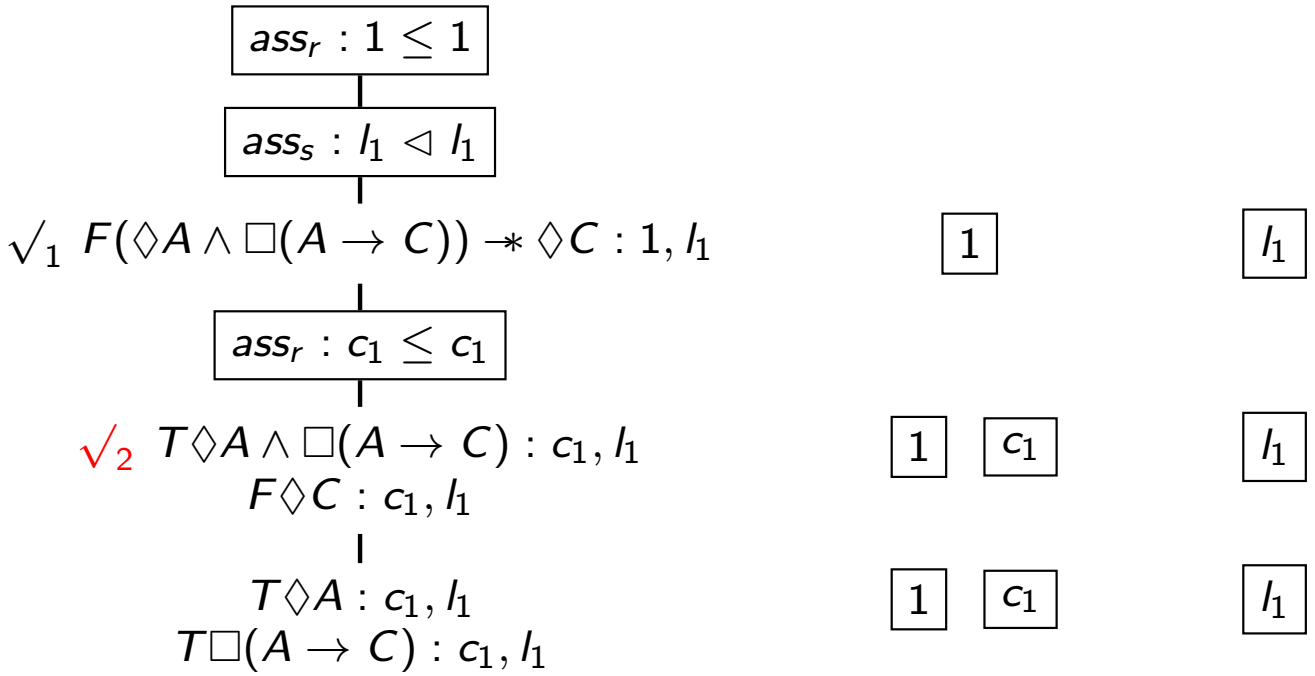
DBI-tableau method



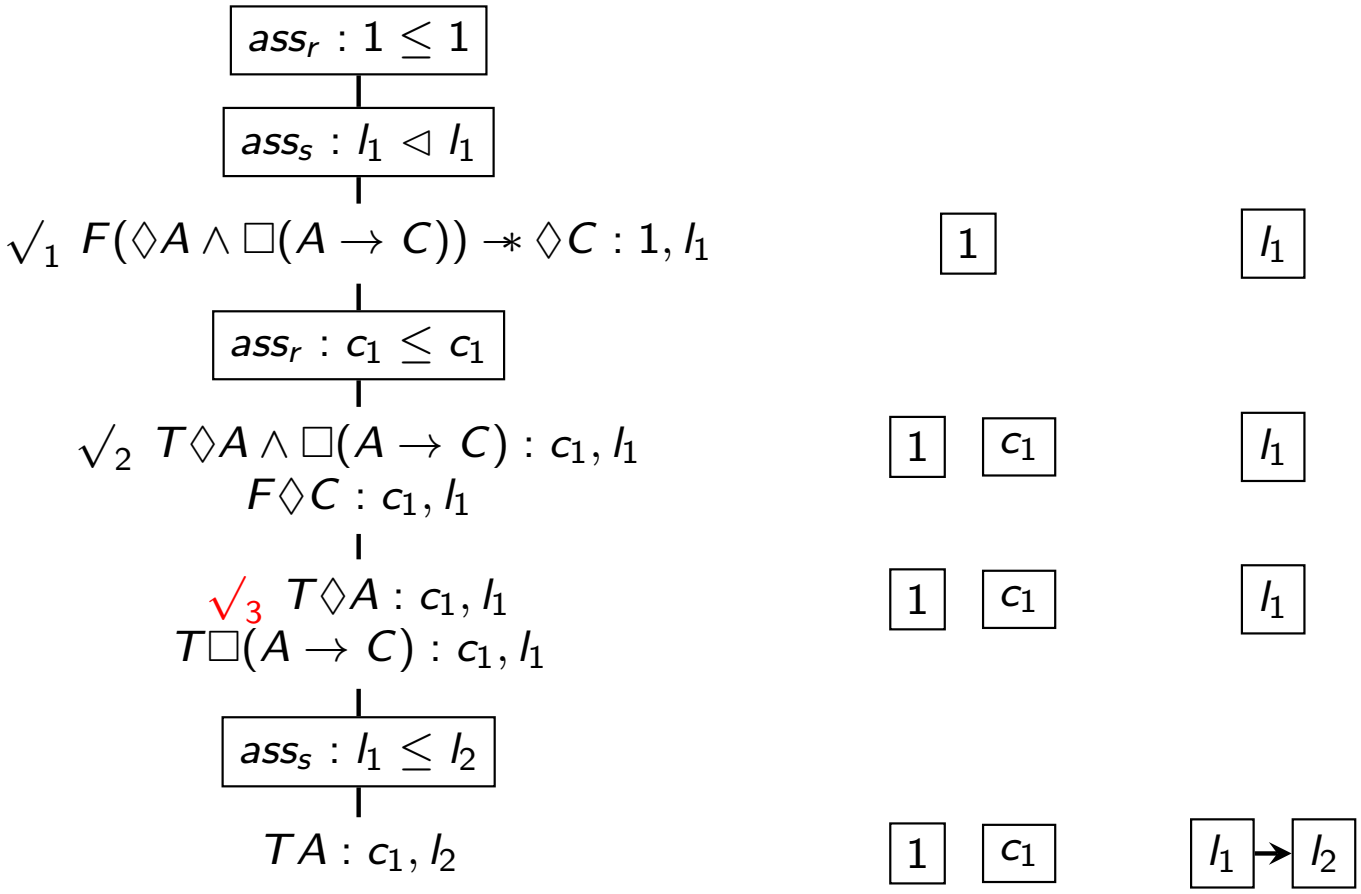
DBI-tableau method



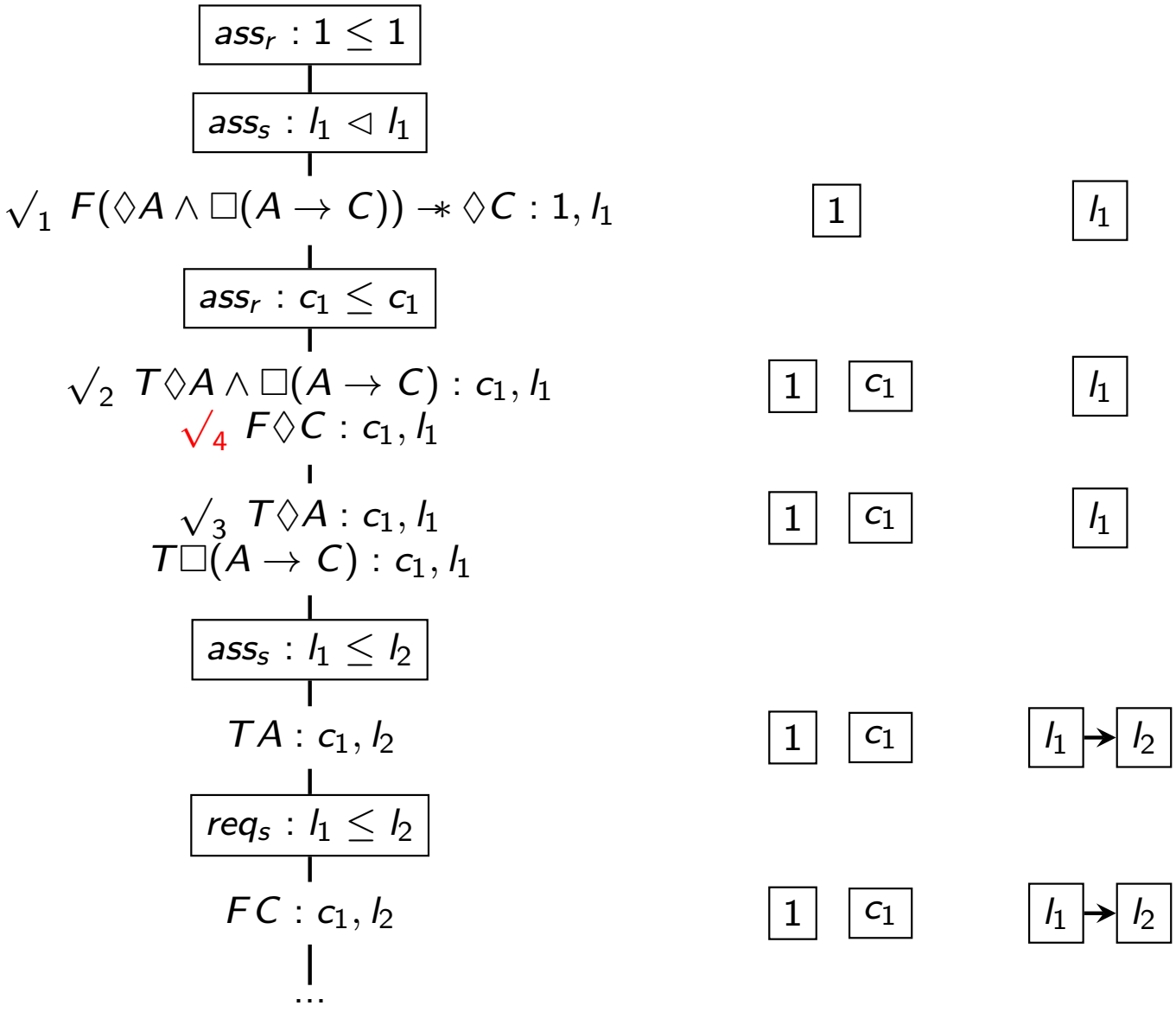
DBI-tableau method



DBI-tableau method



DBI-tableau method



DBI-tableau method

Definition: admissibility

Any requirement holds with respect to previous assertions

Definition: incoherent label

x is **incoherent** iff $yz \leq x \in \overline{Ass_r}(\mathcal{B})$ and $T\bot : y, u \in \mathcal{B}$.

Definition: closed branch

$\mathcal{B}$ is **closed** iff one of these conditions holds:

- $T\phi : x, u$ and $F\phi : y, u$ with $x \leq y \in \overline{Ass_r}(\mathcal{B})$
- $FI : x, u$ and $1 \leq x \in \overline{Ass_r}(\mathcal{B})$
- $F\top : x, u$
- $F\phi : x, u$ and x incoherent

DBI-tableau method - soundness

Soundness

Definition: DBI-tableau proof

$\mathcal{T}$ is a **DBI-tableau proof** of ϕ iff

- Applying initialisation rule with ϕ
- Applying expansion rules are **admissible**
- All branches are closed

Theorem: soundness

If $\mathcal{T}$ is a DBI-tableau proof of ϕ then ϕ is valid.

DBI-tableau method - completeness

Completeness by counter-model extraction

Definition: Hintikka branch

A Hintikka branch is a branch non close and where all information has been extracted

- $F\top : x, u \notin \mathcal{B}$
- ...
- Si $T\Diamond\phi : x, u \in \mathcal{B}$ alors $\exists v \in L_s,\ u \lhd v \in \overline{Ass_s}(\mathcal{B})$ et $T\phi : x, v \in \mathcal{B}$
- Si $F\Diamond\phi : x, u \in \mathcal{B}$ alors $\forall v \in L_s,\ u \lhd v \in \overline{Ass_s}(\mathcal{B}) \Rightarrow$ $F\phi : x, v \in \mathcal{B}$
- ...

DBI-tableau method - completeness

Completeness by counter-model extraction

Lemme: counter-model extraction

A counter-model can be extracted from a Hintikka branch.

Theorem: Completeness

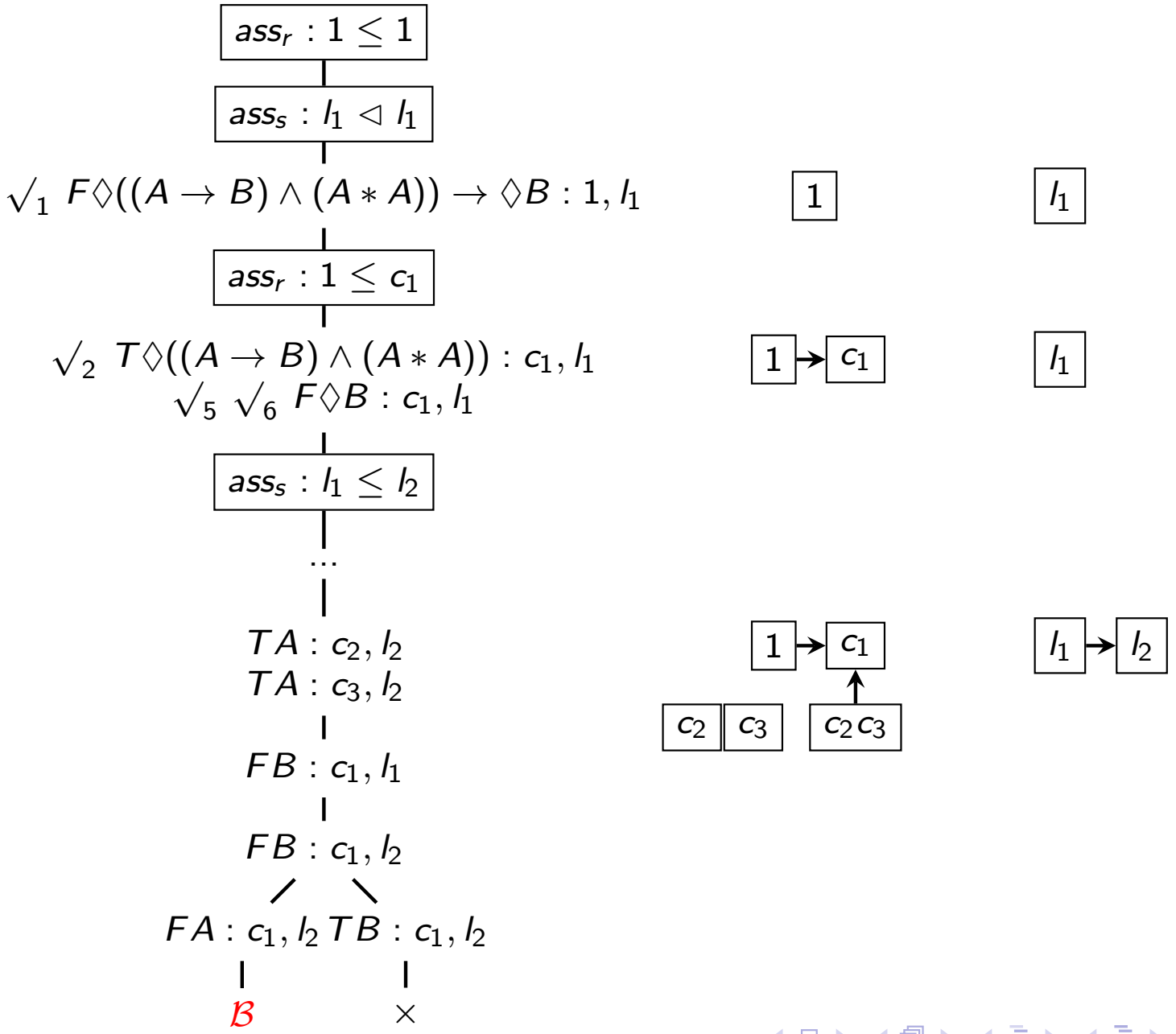
If A is a valid formula then there exists a DBI-tableau proof of A.

Principles of counter-model extraction:

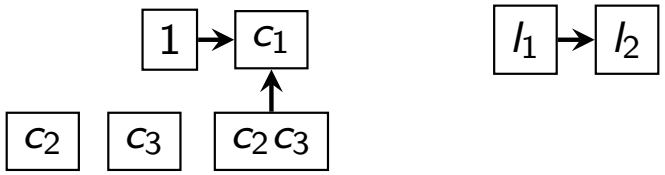
- to consider a Hintikka branch of a tableau
- to transform the two graphs into a dynamic resource monoid
- Interpretation: $(r, s) \in [\![p]\!]$ iff $(\exists r' \in R, r' \sqsubseteq r$ et $Tp : r', s \in \mathcal{B})$ or $(r = \pi)$

Plan

DBI counter-model extraction



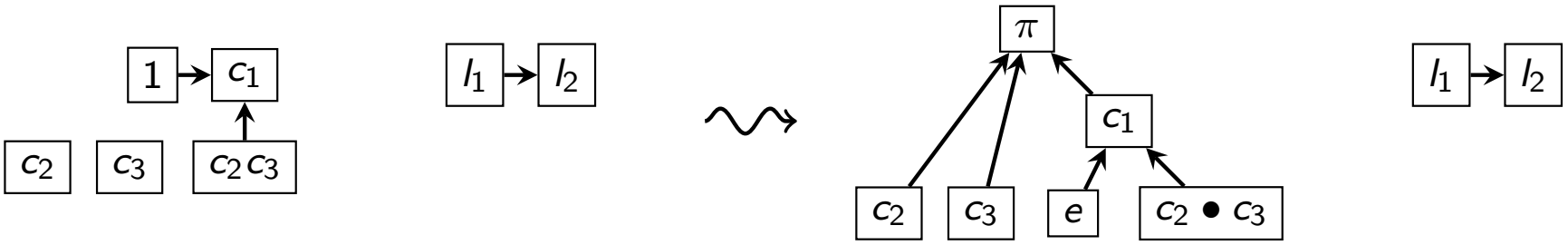
DBI counter-model extraction



- Dynamic resource monoid $\mathcal{M} = (R, \bullet, e, \pi, \sqsubseteq, S, \preceq)$:
  - R: coherent labels of resource graph $\bigcup \{\pi\}$.
    $R = \{e, c_1, c_2, c_2c_3, \pi\}$ (with $e = 1$)
  - S: labels of state graph. $S = \{l_1, l_2\}$
  - $r_1 \bullet r_2 = \begin{cases} r_1 \circ r_2 & \text{if } r_1 \circ r_2 \text{ exists and is coherent} \\ \pi & \text{else} \end{cases}$

$\bullet$	e	c_1	c_2	c_3	c_2c_3	π
e	e	c_1	c_2	c_3	c_2c_3	π
c_1	c_1	π	π	π	π	π
c_2	c_2	π	π	c_2c_3	π	π
c_3	c_3	π	c_2c_3	π	π	π
c_2c_3	c_2c_3	π	π	π	π	π
π	π	π	π	π	π	π

DBI counter-model extraction



- Interpretation:

p	$[\![p]\!]$
A	$\{(\pi, l_1), (\pi, l_2), (c_2, l_2), (c_3, l_2)\}$
B	$\{(\pi, l_1), (\pi, l_2)\}$

  En particular:
  - $TA : c_2, l_2 \in \mathcal{B} \Rightarrow (c_2, l_2) \in [\![A]\!]$
  - $\forall l_i \cdot (\pi, l_i) \in [\![p]\!]$

- Semantics: $e, l_1 \nVdash \Diamond((A \to B) \wedge (A * A)) \to \Diamond B$.
  $\Rightarrow$ counter-model extracted

Plan

DBI Sequent calculus

- DBI-sequent is of the form: $\Gamma \vdash_{R,S} \Delta$
  - Γ and Δ signed formulae sets $\phi : (c, I)$
  - R: set of resource constraints
  - S: set of state constraints

- Some rules:

$$ax_1: \quad \frac{}{\Gamma, \phi : (x, u) \vdash_{R,S} \Delta, \phi : (y, u)} \; x \leq y \in \overline{R}$$

$$\multimap\!\!*_L: \quad \frac{\Gamma \vdash_{R,S} \Delta, \phi : (y, u) \qquad \Gamma, \psi : (xy, u) \vdash_{R,S} \Delta}{\Gamma, \phi \multimap\!\!* \psi : (x, u) \vdash_{R,S} \Delta} \; xy \leq xy \in \overline{R}$$

$$\Diamond_L: \quad \frac{\Gamma, \phi : (x, I_i) \vdash_{R,S \cup \{u \lhd I_i\}} \Delta}{\Gamma, \Diamond\phi : (x, u) \vdash_{R,S} \Delta} \; I_i \notin \mathcal{D}_s(S)$$

DBI Sequent calculus

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dots}{A : (c_1, l_2), A \to C : (c_1, l_2) \vdash_{E_1, E_2} C : (c_1, l_2)} \to_L}{A : (c_1, l_2), \Box(A \to C) : (c_1, l_1) \vdash_{E_1, E_2} C : (c_1, l_2)} \Box_L}{A : (c_1, l_2), \Box(A \to C) : (c_1, l_1) \vdash_{E_1, E_2} \Diamond C : (c_1, l_1)} \Diamond_R}{\Diamond A : (c_1, l_1), \Box(A \to C) : (c_1, l_1) \vdash_{E_1, \{l_1 \lhd l_1\}} \Diamond C : (c_1, l_1)} \Diamond_L}{\Diamond A \wedge \Box(A \to C) : (c_1, l_1) \vdash_{E_1, \{l_1 \lhd l_1\}} \Diamond C : (c_1, l_1)} \wedge_L}{\vdash_{\{1 \leq 1\}, \{l_1 \lhd l_1\}} (\Diamond A \wedge \Box(A \to C)) \twoheadrightarrow \Diamond C : (1, l_1)} \twoheadrightarrow_R$$

Note: $E_1 = \{1 \leq 1, c_1 \leq c_1\}$ *et* $E_2 = \{l_1 \lhd l_1, l_2 \lhd l_2\}$

DBI Sequent calculus

DBI-sequent proof

A **DBI-sequent proof** of ϕ is a tree where:

- $\vdash_{\{1 \leq 1\},\{l_1 \lhd l_1\}} \phi : (1, l_1)$ is the root
- Nodes obtained by application of sequent calculus rules
- Leaf are axioms

Theorem: soundness

If there exists a DBI-sequent proof of Φ then Φ id valid.

Theorem: completeness

If Φ is valid then there exists a DBI-sequent proof of Φ.

Plan

Conclusions - Perspectives

- Logic **DBI** for dynamic resources management
  - Non instantaneous consumption of resources
  - Expressing convergence/invariant properties
  - Latency and message loss
  - Places
- **DBI** Tableaux and sequent calculus
  - Soundness
  - Completeness
  - Counter-model extraction
- Extension of **DBI** with statements $(R, E \xrightarrow{a} R', E')$ and Hennessy–Milner modalities
- **DBI** for Web Services composition