A tableau-based decision procedure for a branching-time interval temporal logic

Davide Bresolin   Angelo Montanari

Dipartimento di Matematica e Informatica
Università degli Studi di Udine
{bresolin, montanari}@dimi.uniud.it

M4M-4, 1-2 December 2005

Outline

Outline

1. Introduction
2. A branching-time interval temporal logic
3. A Tableau for $BTNL[R]^-$
4. Future work

Interval temporal logics

Interval temporal logics (HS, CDT, PITL) are very expressive

- simple syntax and semantics;
- can naturally express statements that refer to time intervals and continuous processes;
- the most expressive ones (HS and CDT) are strictly more expressive than every point-based temporal logic.

Interval temporal logics are (highly) undecidable

The validity problem for HS is not recursively axiomatizable.

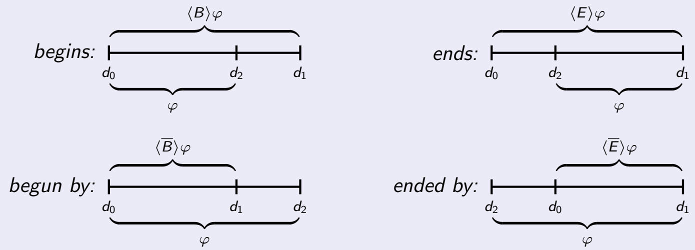
Problem

Find expressive, but decidable, fragments of interval temporal logics.

Halpern and Shoam's HS

HS features four basic unary operators:

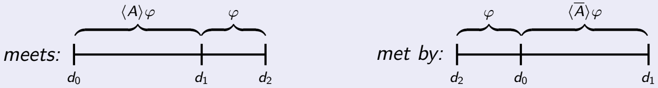
- $\langle B \rangle$ (*begins*) and $\langle E \rangle$ (*ends*), and their transposes $\langle \overline{B} \rangle$ (*begun by*) and $\langle \overline{E} \rangle$ (*ended by*).



- Given a formula φ and an interval $[d_0, d_1]$, $\langle B \rangle \varphi$ holds over $[d_0, d_1]$ if φ holds over $[d_0, d_2]$, for some $d_0 \leq d_2 < d_1$, and $\langle E \rangle \varphi$ holds over $[d_0, d_1]$ if φ holds over $[d_2, d_1]$, for some $d_0 < d_2 \leq d_1$.

Some interesting fragments of HS

- **The** $\langle B \rangle \langle E \rangle$ **fragment** (*undecidable*);
- **The** $\langle B \rangle \langle \overline{B} \rangle$ **and** $\langle E \rangle \langle \overline{E} \rangle$ **fragments** (*decidable*);
- **Goranko, Montanari, and Sciavicco's PNL:**
  - based on the derived neighborhood operators $\langle A \rangle$ (*meets*) and $\langle \overline{A} \rangle$ (*met by*);



  - decidable (by reduction to 2FO[<]), but no tableau methods.

The linear case: Right PNL (RPNL)

- future-only fragment of PNL;
- interpreted over natural numbers;
- decidable, doubly exponential tableau-based decision procedure for RPNL (TABLEAUX 2005);
- recently, we devised an optimal (NEXPTIME) tableau-based decision procedure for RPNL.

The branching case

We developed a branching-time propositional interval temporal logic.

- Such a logic combines:
  - interval quantifiers $\langle A \rangle$ and $[A]$ from RPNL;
  - path quantifiers A and E from CTL.
- We devised a tableau-based decision procedure for it, combining:
  - the tableau for RPNL (TABLEAUX 2005);
  - Emerson and Halpern's tableau for CTL (J. of Computer and System Sciences, 1985).

Outline

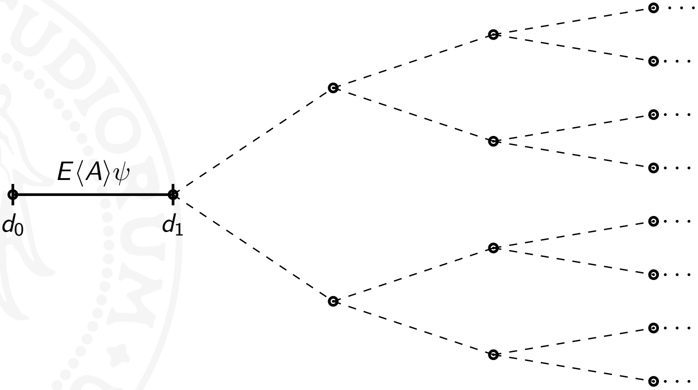
Branching Time Right-Neighborhood Logic

Syntax of BTNL[R]$^-$

$$\varphi = p \mid \neg\varphi \mid \varphi \vee \varphi \mid E\langle A\rangle\varphi \mid E[A]\varphi \mid A\langle A\rangle\varphi \mid A[A]\varphi.$$

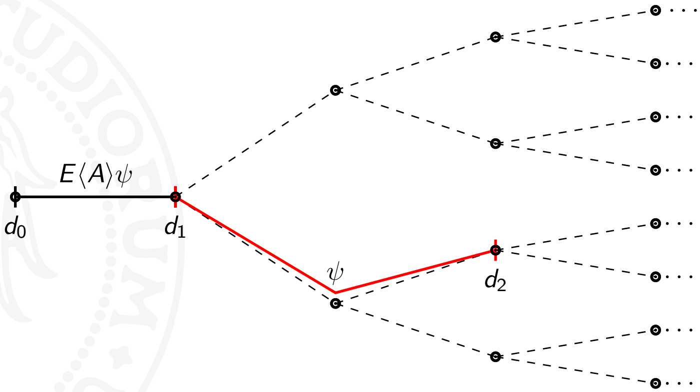
- Interpreted over **infinite trees.**
- Combines path quantifiers A (for all paths) and E (for any path) with the interval modalities $\langle A\rangle$ and $[A]$.

BTNL[R]$^-$ semantics: $E\langle A\rangle\psi$



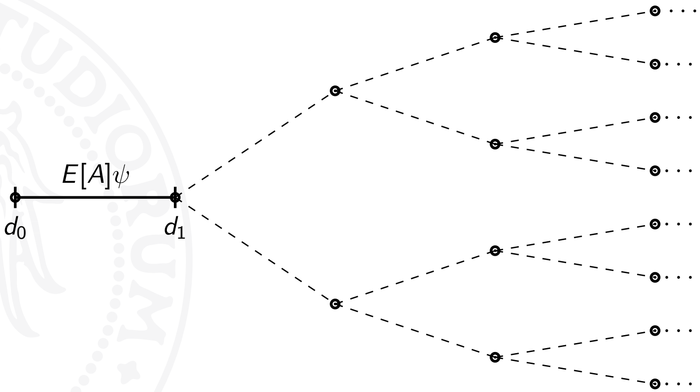
$E\langle A\rangle\psi$ holds over $[d_0, d_1]$ if ψ holds over $[d_1, d_2]$, for some $d_2 < d_1$.

BTNL[R]$^-$ semantics: $E\langle A\rangle\psi$



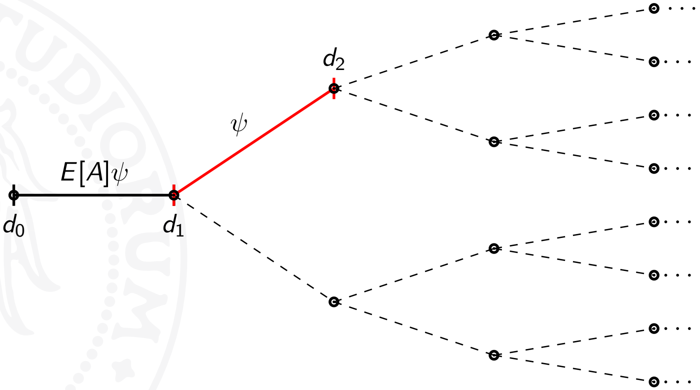
$E\langle A\rangle\psi$ holds over $[d_0, d_1]$ if ψ holds over $[d_1, d_2]$, for some $d_2 < d_1$.

BTNL[R]$^-$ semantics: $E[A]\psi$



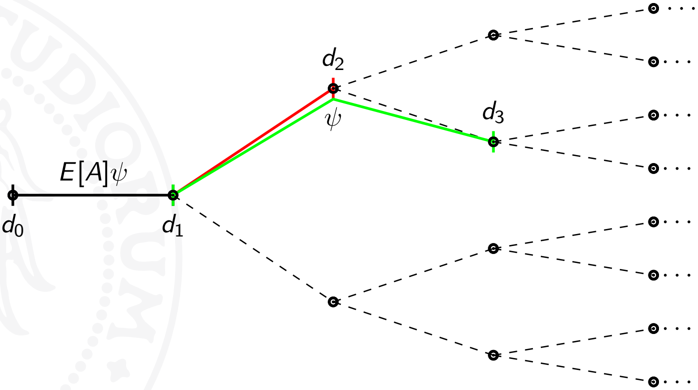
$E[A]\psi$ holds over $[d_0, d_1]$ if there exists an infinite path $d_1, d_2, \ldots$ such that ψ holds over $[d_1, d_i]$, for all $d_i > d_1$ in the path.

BTNL[R]$^-$ semantics: $E[A]\psi$



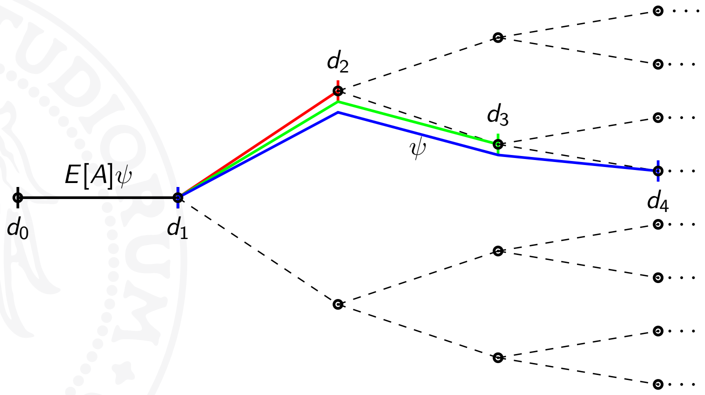
$E[A]\psi$ holds over $[d_0, d_1]$ if there exists an infinite path $d_1, d_2, \ldots$ such that ψ holds over $[d_1, d_i]$, for all $d_i > d_1$ in the path.

BTNL[R]$^-$ semantics: $E[A]\psi$

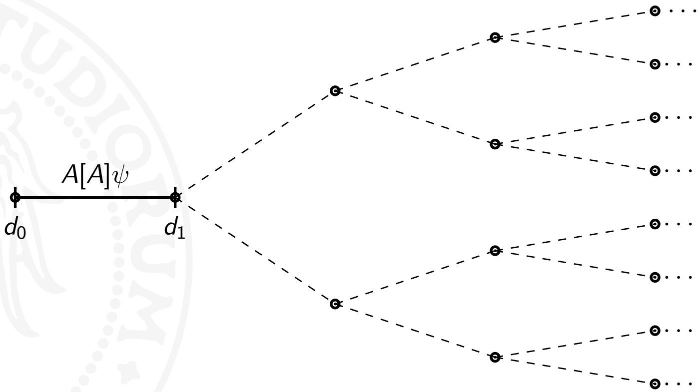


$E[A]\psi$ holds over $[d_0, d_1]$ if there exists an infinite path $d_1, d_2, \ldots$ such that ψ holds over $[d_1, d_i]$, for all $d_i > d_1$ in the path.

BTNL[R]$^-$ semantics: $E[A]\psi$

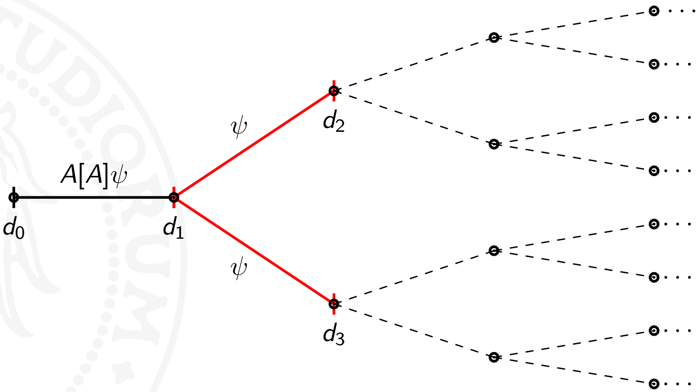


$E[A]\psi$ holds over $[d_0, d_1]$ if there exists an infinite path $d_1, d_2, \ldots$ such that ψ holds over $[d_1, d_i]$, for all $d_i > d_1$ in the path.



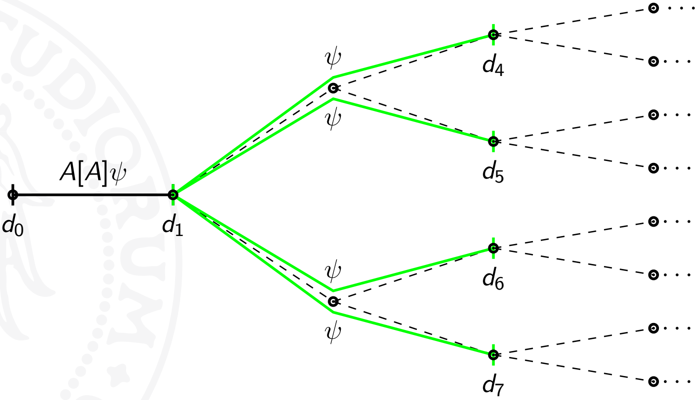
$A[A]$ is the dual of $E\langle A\rangle$: $A[A]\psi$ holds over $[d_0, d_1]$ if ψ holds over $[d_1, d_2]$, for all $d_2 < d_1$.

BTNL[R]$^-$ semantics: $A[A]\psi$



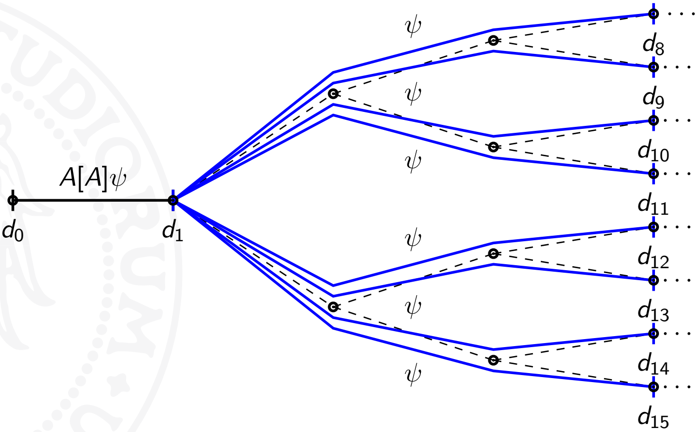
$A[A]$ is the dual of $E\langle A\rangle$: $A[A]\psi$ holds over $[d_0, d_1]$ if ψ holds over $[d_1, d_2]$, for all $d_2 < d_1$.

BTNL[R]$^-$ semantics: $A[A]\psi$



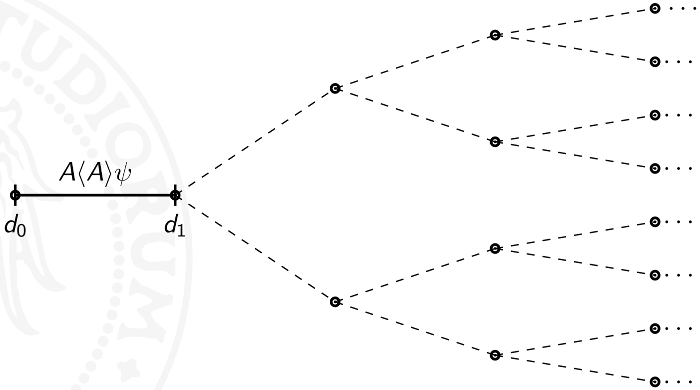
$A[A]$ is the dual of $E\langle A\rangle$: $A[A]\psi$ holds over $[d_0, d_1]$ if ψ holds over $[d_1, d_2]$, for all $d_2 < d_1$.

BTNL[R]$^-$ semantics: $A[A]\psi$



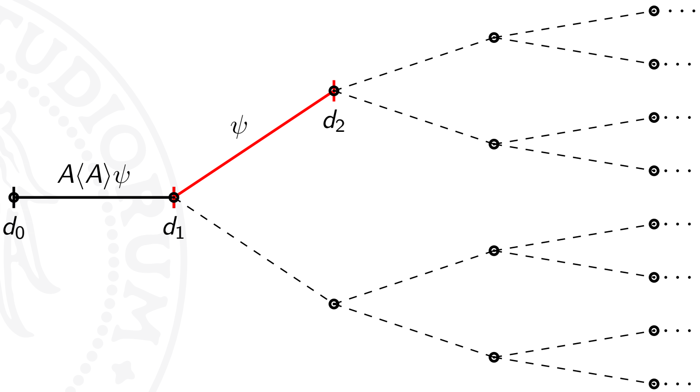
$A[A]$ is the dual of $E\langle A\rangle$: $A[A]\psi$ holds over $[d_0, d_1]$ if ψ holds over $[d_1, d_2]$, for all $d_2 < d_1$.

BTNL[R]$^-$ semantics: $A\langle A\rangle\psi$



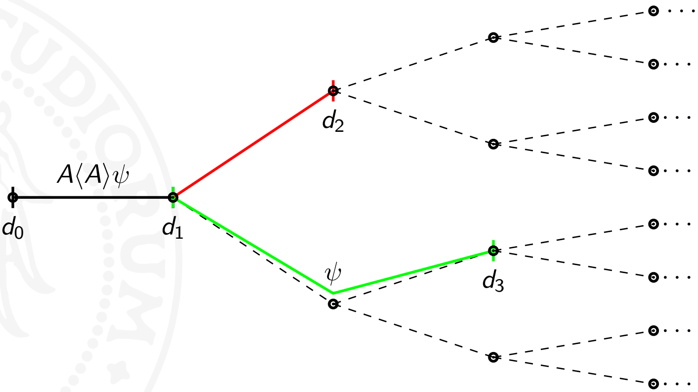
$A\langle A\rangle$ is the dual of $E[A]$: $A\langle A\rangle\psi$ holds over $[d_0, d_1]$ if, for all infinite paths $d_1, d_2, \ldots$, ψ holds over $[d_1, d_i]$, for some $d_i > d_1$ in the path.

BTNL[R]$^-$ semantics: $A\langle A\rangle\psi$



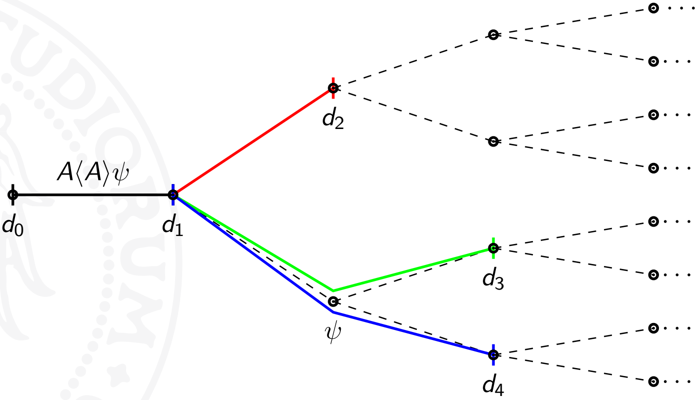
$A\langle A\rangle$ is the dual of $E[A]$: $A\langle A\rangle\psi$ holds over $[d_0, d_1]$ if, for all infinite paths $d_1, d_2, \ldots$, ψ holds over $[d_1, d_i]$, for some $d_i > d_1$ in the path.

BTNL[R]$^-$ semantics: $A\langle A\rangle\psi$



$A\langle A\rangle$ is the dual of $E[A]$: $A\langle A\rangle\psi$ holds over $[d_0, d_1]$ if, for all infinite paths $d_1, d_2, \ldots, \psi$ holds over $[d_1, d_i]$, for some $d_i > d_1$ in the path.

BTNL[R]$^-$ semantics: $A\langle A\rangle\psi$



$A\langle A\rangle$ is the dual of $E[A]$: $A\langle A\rangle\psi$ holds over $[d_0, d_1]$ if, for all infinite paths $d_1, d_2, \ldots$, ψ holds over $[d_1, d_i]$, for some $d_i > d_1$ in the path.

Outline

Basic blocks

Definition

An atom is a pair $(\mathcal{A}, \mathcal{C})$ such that:

- $\mathcal{C}$ is a maximal, locally consistent set of subformulae of φ;
- $\mathcal{A}$ is a consistent (but not necessarily complete) set of temporal formulae ($A\langle A\rangle\psi$, $A[A]\psi$, $E\langle A\rangle\psi$, and $E[A]\psi$);
- $\mathcal{A}$ and $\mathcal{C}$ must be coherent:
  - if $A[A]\psi \in \mathcal{A}$, then $\psi \in \mathcal{C}$;
  - if $E[A]\psi \in \mathcal{A}$, then $\psi \in \mathcal{C}$.

Atoms and Intervals

- Associate with every interval $[d_i, d_j]$ an atom $(\mathcal{A}, \mathcal{C})$:
  - $\mathcal{C}$ contains the formulae that (should) hold over $[d_i, d_j]$;
  - $\mathcal{A}$ contains temporal requests coming from the past.
- Connect every pair of atoms that are associated with neighbor intervals.

Tableau nodes

Definition

A node N of the tableau is a set of atoms such that, for every temporal formula ψ and every pair of atoms $(\mathcal{A},\mathcal{C}),(\mathcal{A}',\mathcal{C}') \in N$, $\psi \in \mathcal{C}$ iff $\psi \in \mathcal{C}'$.

Nodes and points

A node N represents a point d_j of the temporal domain:

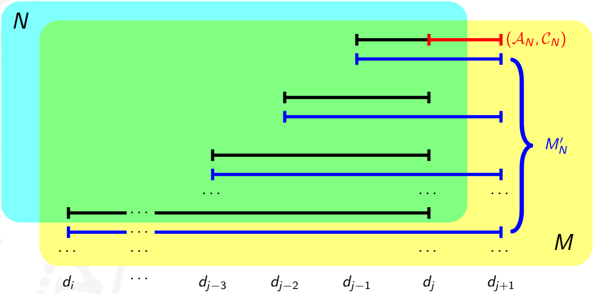
- every atom in N represents an interval $[d_i, d_j]$ ending in d_j.

A node N representing point d_1 is an initial node:

- N contains only an atom $(\emptyset,\mathcal{C})$ with $\varphi \in \mathcal{C}$.

Connecting nodes

We put an edge between two nodes if they represent successive time points:



- $(\mathcal{A}_N, \mathcal{C}_N)$ is an atom such that $\mathcal{A}_N$ contains all requests (temporal formulae) of N;
- for every $(\mathcal{A}, \mathcal{C}) \in N$ representing $[d_i, d_j]$ there is $(\mathcal{A}', \mathcal{C}') \in M'_N$ representing $[d_i, d_{j+1}]$.

The decision procedure

1. Build the (unique) initial tableau $\mathcal{T}_\varphi = \langle \mathcal{N}_\varphi, \mathcal{R}_\varphi \rangle$.
2. Delete "useless nodes" by repeatedly applying the following deletion rules, until no more nodes can be deleted:
   - delete any node which is not reachable from an initial node;
   - delete any node that contains a formula of the form $A[A]\psi$, $A\langle A\rangle\psi$, $E\langle A\rangle\psi$, or $E[A]\psi$ that is not satisfied.
3. If the final tableau is not empty, return true, otherwise return false.

Pruning the tableau: $A[A]\psi$

A[*A*]-formulas are satisfied by construction.

Given a node N and an atom $(\mathcal{A}, \mathcal{C}) \in N$:

- if $A[A]\psi \in \mathcal{C}$,
- then, for every right neighbor $(\mathcal{A}', \mathcal{C}')$, $A[A]\psi \in \mathcal{A}'$;
- hence, by definition of atom, $\psi \in \mathcal{C}'$.

Pruning the tableau: $A\langle A\rangle\psi$

$A\langle A\rangle$-formulas are checked by a marking procedure.

1. For all nodes N, mark all atoms $(\mathcal{A},\mathcal{C}) \in N$ such that $A\langle A\rangle\psi \in \mathcal{A}$ and $\psi \in \mathcal{C}$.
2. For all nodes N, mark all unmarked atoms $(\mathcal{A},\mathcal{C}) \in N$ such that there exists a successor M of N that contains a marked atom $(\mathcal{A}',\mathcal{C}')$ that is a right neighbor of $(\mathcal{A},\mathcal{C})$.
3. Repeat this last step until no more atoms can be marked.
4. Delete all nodes that either contain an unmarked atom $(\mathcal{A},\mathcal{C})$ with $A\langle A\rangle\psi \in \mathcal{A}$ or have no successors.

Pruning the tableau: $E\langle A\rangle\psi$

$E\langle A\rangle$-formulas are checked by searching for a descendant.

Given a node N and an atom $(\mathcal{A}, \mathcal{C}) \in N$, if $E\langle A\rangle\psi \in \mathcal{C}$, search for a descendant M such that:

1. M contains an atom $(\mathcal{A}', \mathcal{C}')$ that is a right-neighbor of $(\mathcal{A}, \mathcal{C})$;
2. $\psi \in \mathcal{C}'$.

Pruning the tableau: $E[A]\psi$

$E[A]$-formulas are checked by searching for a loop.

Given a node N and an atom $(\mathcal{A}, \mathcal{C}) \in N$, if $E[A]\psi \in \mathcal{C}$, search for a path leading to a loop such that:

- every node in the path and in the loop contains an atom $(\mathcal{A}', \mathcal{C}')$ such that:
  1. $(\mathcal{A}', \mathcal{C}')$ is a right-neighbor of $(\mathcal{A}, \mathcal{C})$;
  2. $\psi \in \mathcal{C}'$.

Building a model for φ

An infinite model for φ can be build by unfolding the final tableau.

1. Select an initial node N_1;
2. finite paths $N_1 N_2 \ldots N_k$ starting from the initial node becomes the points of the infinite tree;
3. define the valuation function respecting atoms (this is the key step) .

Computational Complexity

- The size of the tableau is doubly exponential in the length of the formula;
- all checkings of the algorithm can be done in time polynomial in the size of the tableau;
- after deleting at most $|\mathcal{N}_\varphi|$ nodes, the algorithm terminates.

Checking the satisfiability for a BTNL[R]$^-$ formula is doubly exponential in the length of φ.

Outline

Future work

- **Complexity issues:**
  - we do not know yet whether the satisfiability problem for BTNL[R]$^-$ is doubly EXPTIME-complete or not (we conjecture it is not!).
- **Extensions:**
  - to combine path quantifiers operators with other sets of interval logic operators, e.g., those of PNL.