Decision Procedures for Constraint Temporal Logic

Subhasree M.
subha@csa.iisc.ernet.in

Supervisor: Dr. Deepak D'Souza
Computer Science and Automation
Indian Institute of Science, Bangalore

- $\mathrm{CLTL}^{\Diamond}$
    - Expressiveness of $\mathrm{CLTL}^{\Diamond}$
    - Decidability of satisfiability problem for $\mathrm{CLTL}^{\Diamond}$

Contents

- $\mathrm{CLTL}^{\diamond}$
- Expressiveness of $\mathrm{CLTL}^{\diamond}$ and CLTL
- Decidability of CLTL over finite models
- Characterisation of $\mathrm{CLTL}^{\diamond}$ frame graphs
- Decidability of $\mathrm{CLTL}^{\diamond}$ over single variable monotonic models
- Conclusion and future work

$\mathrm{CLTL}^{\diamond}$

- $\mathrm{CLTL}^{\diamond}$: CLTL with $\diamond$ quantifier
- Constraint LTL, CLTL (DD02) : Extension of LTL
- Linear-time Temporal Logic, LTL(P77) : Tool used for Verification

LTL

- Variables : Set of Propositions, P

- Model : Finite or infinite sequence of subsets of P

Examples of LTL formula

- $P = \{p, q, r\}$
- **Example 1**

$$\begin{array}{lll} \varphi & : & \Box(p \vee r) \\ \sigma & : & \{p,q\}, \{p,r\}, \{p,q\}, \{p,r\} \\ \tau & : & \{p,q\}, \{q\}, \{p,q\}, \{q\} \end{array}$$

τ does not satisfy φ

- **Example 2**

$$\begin{array}{lll} \varphi & : & (p \,\mathcal{U}\, r) \\ \sigma & : & \{p,q\}, \{p\}, \{p,r\}, \{p,q\}, \{p\} \\ \tau & : & \{p,q\}, \{q\}, \{p,r\}, \{p,q\}, \{q\} \end{array}$$

τ does not satisfy φ

CLTL

- Extension of LTL interpreted over a sequence of valuations of $\mathbb{Z}$
- CLTL permits to refer to a variable in the next instant, using O quantifier in the logic
- Ox refers to a variable x in the next instant

  - Variables : Elements of U
  - Model : Finite or infinite sequence of $\mathbb{Z}$ valuations

Atomic constraints of CLTL

- $O^n x < O^m y$
- $O^n x = O^m y$

where $x, y \in U$, $n, m \in \mathbb{N}$

Examples of CLTL formula

- $x, y, z \in U$
- **Example 1**

  $\varphi : \Box(x < Oy)$

  $\sigma :$

$y:$	2	4	6	8	10
$x:$	1	3	5	7	9

  $\tau :$

$y:$	2	4	6	8	7
$x:$	1	3	5	7	9

  τ does not satisfy φ

Example 2

$\varphi : (x < y)\,\mathcal{U}\,(x < O^2 z)$

σ :

z :	2	4	0	2	6
y :	2	4	5	7	9
x :	1	3	5	7	9

τ :

z :	2	4	0	2	6
y :	1	4	6	7	9
x :	1	3	5	7	9

τ does not satisfy φ

CLTL$^\diamond$

- CLTL with $\diamond$ quantifier
- Permits $\diamond$ quantifier also in the logic
- $\diamond x$ refers to variable x, *some* instant in future, including the current instant
  - Variables : Elements of U
  - Model : Sequence of $\mathbb{Z}$ valuations

Atomic constraints of $\mathrm{CLTL}^{\Diamond}$

- $O^n x \sim O^m y$
- $O^n x \sim \Diamond y$
- $\Diamond x \sim O^n y$
- $\Diamond x \sim \Diamond y$

where $x, y \in U$, $n, m \in \mathbb{N}$ and $\sim \in \{<, =\}$

Examples of $\mathrm{CLTL}^{\Diamond}$ formulas

- $x, y, z \in U$
- **Example 1**

  $\varphi : (x < \Diamond y)$

  $\sigma :$

$y :$	0	1	2	3	6
$x :$	5	6	7	8	9

  $\tau :$

$y :$	0	1	2	3	4
$x :$	5	6	7	8	9

  τ does not satisfy φ

Example 2

$\varphi : (x < \Diamond y)\,\mathcal{U}\,(x < O^2 z)$

σ :

$$\begin{array}{cccccc} z: & 2 & 4 & 0 & 2 & 9 \\ y: & 0 & 0 & 2 & 0 & 0 \\ x: & 1 & 1 & 5 & 7 & 9 \end{array}$$

τ :

$$\begin{array}{cccccc} z: & 2 & 4 & 0 & 2 & 9 \\ y: & 0 & 0 & 0 & 0 & 0 \\ x: & 1 & 3 & 5 & 7 & 9 \end{array}$$

τ does not satisfy φ

Atomic constraints of $\mathrm{CLTL}^{\Diamond}$

- $O^n x \sim O^m y$
- $O^n x \sim \Diamond y$
- $\Diamond x \sim O^n y$
- $\Diamond x \sim \Diamond y$

where $x, y \in U$, $n, m \in \mathbb{N}$ and $\sim \in \{<, =\}$

- c can be written as c'

$$c \;:=\; (O^n x \sim \Diamond y)$$

$$\begin{aligned} c' \;:=\; & (O^n x \sim y) \vee (O^n x \sim O y) \vee \ldots \\ & \vee (O^n x \sim O^{n-1} y) \vee O^n (x \sim \Diamond y) \end{aligned}$$

Additional atomic constraints due to $\Diamond$ quantifier

- $(x \sim \Diamond y)$

- $(\Diamond x \sim y)$

where $x, y \in U$ and $\sim \in \{<, =\}$

Expressiveness of $\mathrm{CLTL}^{\diamond}$ and CLTL

CLTL terminology

- *Atomic Constraint*, c
  $O^n x < O^m y$ and $O^n x = O^m y$
  where $n, m \in \mathbb{N}$ and $x, y \in U$

- O-length k of an atomic constraint is the value $i + 1$ where i is the largest j for which O^j occurs in the atomic constraint
  - $c : x < Oy$
  - O-length is 2

- O-length of a formula is the largest O-length of atomic constraints in φ
  - $\varphi : (x < Oy)\mathcal{U}\,(x < O^3 z)$
  - O-length is 4

Induced k-frame

- An example of $\mathbb{Z}$ valuation sequence :

$$\begin{array}{lllll} y: & 2 & 4 & 6 & 8 \\ x: & 1 & 3 & 5 & 7 \end{array}$$

- *k-frame induced by a sequence of valuation σ*:
  - *k- frame* (σ) = $\{c|\sigma \models c\}$

y	2	4	6	8
x	1	3	5	7

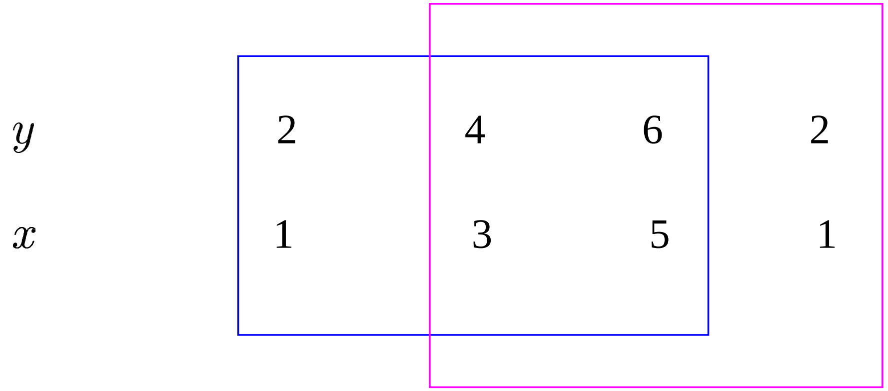
  - 2-frame (σ) : $\{(x = x), (x < y), (x < Ox), (x < Oy),$
    $(y = y), (y < Ox), (y < Oy), (Ox = Ox),$
    $(Ox < Oy), (Oy = Oy)\}$

Locally consistent k-frames

- *Locally consistent k-frames* : the frame pair (r, r') is locally consistent, for all $n, m \geq 1$
  $(O^n x < O^m y) \in r \Longrightarrow (O^{n-1} x < O^{m-1} y) \in r'$ and
  $(O^n x = O^m y) \in r \Longrightarrow (O^{n-1}$
- A $\mathbb{Z}$ -valuation sequence σ is as follows:

$$
\begin{array}{lcccc}
y: & 2 & 4 & 6 & 2 \\
x: & 1 & 3 & 5 & 1
\end{array}
$$

Example of locally consistent frames



- 3-frame (σ), r :
  $\{(x = x), (x < y), (x < Ox), (x < Oy), (x < O^2x), (x < O^2y)$
  $(y = y), (y < Ox), (y < Oy), (y < O^2x), (y < O^2y), (Ox = Ox),$
  $(Ox < Oy), (Ox < O^2x), (Ox < O^2y), (Oy = Oy), (Oy < O^2x),$
  $(Oy < O^2y), (O^2x = O^2x), (O^2x < O^2y), (O^2y = O^2y)\}$
- 3-frame (σ) r': $\{(x = x), (x < y), (x < Ox), (x < Oy), (y = y),$
  $(y < Ox), (y < Oy), (Ox = Ox), (Ox < Oy), (Oy = Oy),$
  $(O^2x = O^2x)(O^2x < x), (O^2x < Ox), (O^2x < Oy), (O^2x < O^2y),$
  $(O^2y = O^2y), (O^2y < x), (O^2y < y), (O^2y = Ox), (O^2y = Oy)\}$
- (r, r') is locally consistent

k-frame sequence induced by a valuation sequence

- *k-frame sequence* ρ : Sequences of k frames, denoted by $\rho(0)\rho(1)\ldots$
- *k-fs* (σ) : a locally consistent k-frame sequence induced by σ

Example of 3-frame sequence induced by a valuation sequence

- A $\mathbb{Z}$ -valuation sequence σ is as follows:

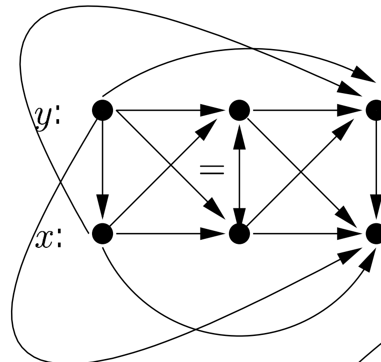
$$\begin{array}{lcccc} y: & 2 & 4 & 6 & 2 \\ x: & 1 & 3 & 5 & 1 \end{array}$$

- 3-fs (σ) :
  $\{(x = x), (x < y), (x < Ox), (x < Oy), (x < O^2x), (x < O^2y)$
  $(y = y), (y < Ox), (y < Oy), (y < O^2x), (y < O^2y), (Ox = Ox),$
  $(Ox < Oy), (Ox < O^2x), (Ox < O^2y), (Oy = Oy), (Oy < O^2x),$
  $(Oy < O^2y), (O^2x = O^2x), (O^2x < O^2y), (O^2y = O^2y)\}$
  $\{(x = x), (x < y), (x < Ox), (x < Oy), (y = y),$
  $(y < Ox), (y < Oy), (Ox = Ox), (Ox < Oy), (Oy = Oy),$
  $(O^2x = O^2x)(O^2x < x), (O^2x < Ox), (O^2x < Oy), (O^2x < O^2y),$
  $(O^2y = O^2y), (O^2y < x), (O^2y < y), (O^2y = Ox), (O^2y = Oy)\}$
- *Frame graph* G_ρ : Locally consistent frame sequence as a $\{<, =\}$-labelled, directed graph

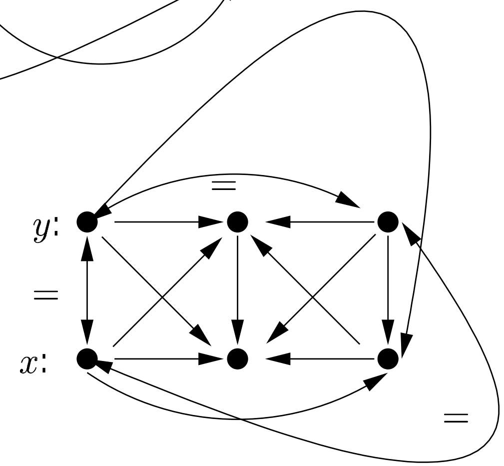
Example of frame graph with $k = 3$

y	2	4	6	4
x	3	4	7	5

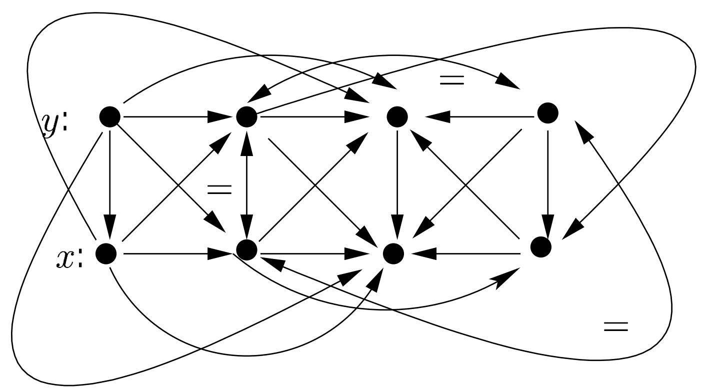
$\rho(0)$



$\rho(1)$



G_ρ



- G_ρ satisfies the following conditions :
  - There is an edge between every pair of vertices
  - If there is '='-labelled edge from x to y then there is also one from y to x
  - There are no *strict cycles*-i.e. directed cycles containing a '$<$'-labelled edge

- $L(\varphi)$ is the set of models of a CLTL formula, φ

- $k\text{-}fs(L) = \{k\text{-}fs(\sigma) \mid \sigma \in L\}$

Expressiveness of $\mathrm{CLTL}^{\diamond}$ and CLTL

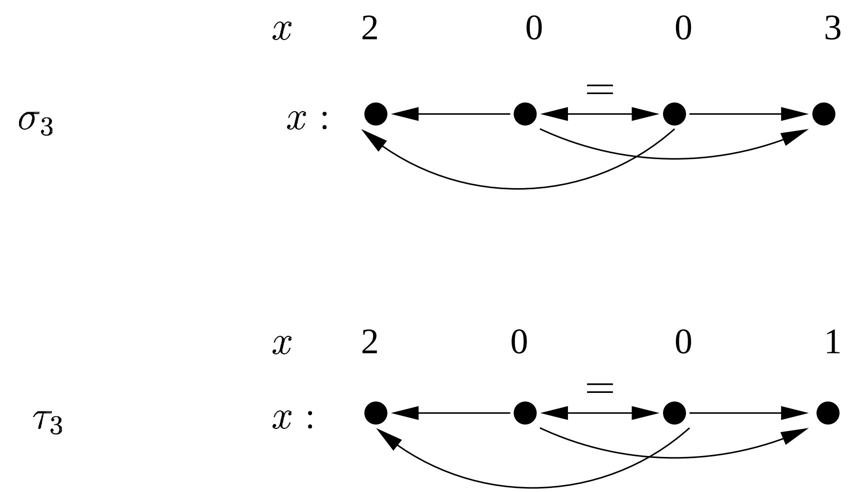
- Two logics $\mathcal{L}_1$ and $\mathcal{L}_2$ are said to be equivalent if
  - $\{L(\varphi_1) : \varphi_1 \in \mathcal{L}_1\} = \{L(\varphi_2) : \varphi_2 \in \mathcal{L}_2\}$

- **Theorem 1** *:* $\mathrm{CLTL}^{\diamond}$ *is strictly more expressive than* CLTL*.*

Outline of the proof of theorem 1

- $\mathrm{CLTL}^{\Diamond}$ formula $(x < \Diamond x)$ has no equivalent CLTL formula.

$$\begin{array}{cccccc} \sigma & X: & 2 & 0 & 0 & 3 \\ \tau & X: & 2 & 0 & 0 & 1 \end{array}$$



- No CLTL formula can distinguish between σ and τ because the k-frame sequences induced by both of them are same
- $\mathrm{CLTL}^{\Diamond}$ formula distinguishes between σ and τ

Proof

- The proof is by contradiction
- Suppose there exists a CLTL formula which distinguishes σ_i and τ_i
- Consider two families of models σ_i and τ_i for $i \geq 1$
- Length of σ_i and τ_i is $(i+1)$.
- σ_i and τ_i are as follows:

$$\begin{array}{llllllllllllll}
\sigma_1 & 2 & 0 & 3 & & & & \tau_1 & 2 & 0 & 1 & & & \\
\sigma_2 & 2 & 0 & 0 & 3 & & & \tau_2 & 2 & 0 & 0 & 1 & & \\
\sigma_3 & 2 & 0 & 0 & 0 & 3 & & \tau_3 & 2 & 0 & 0 & 0 & 1 & \\
\sigma_4 & 2 & 0 & 0 & 0 & 0 & 3 & \tau_4 & 2 & 0 & 0 & 0 & 0 & 1 \\
\vdots & & & & & & & \vdots & & & & & &
\end{array}$$

- Either both σ_i and τ_i satisfy the CLTL formula or both do not satisfy the formula

Decidability of the satisfiability problem

Satisfiability problem

Satisfiability problem for a logic is : given a formula φ of the logic, does there exist a $\mathbb{Z}$-valuation sequence which satisfies φ? In other words, is $L(\varphi) \neq \phi$?

Decidability of the satisfiability problem for CLTL over *finite models*

Link between CLTL and LTL

- CLTL formula φ of O-length k can be viewed as a LTL formula by replacing the constraints with propositions
- **Example**

  $\varphi : \Box(x < Oy)$

  $\sigma :$

y	2	4	6	8
x	1	3	5	7

- 2-frame (σ) : $\{(x = x), (x < y), (x < Ox), (x < Oy), (y = y), (y < Ox), (y < Oy), (Ox = Ox), (Ox < Oy), (Oy = Oy)\}$

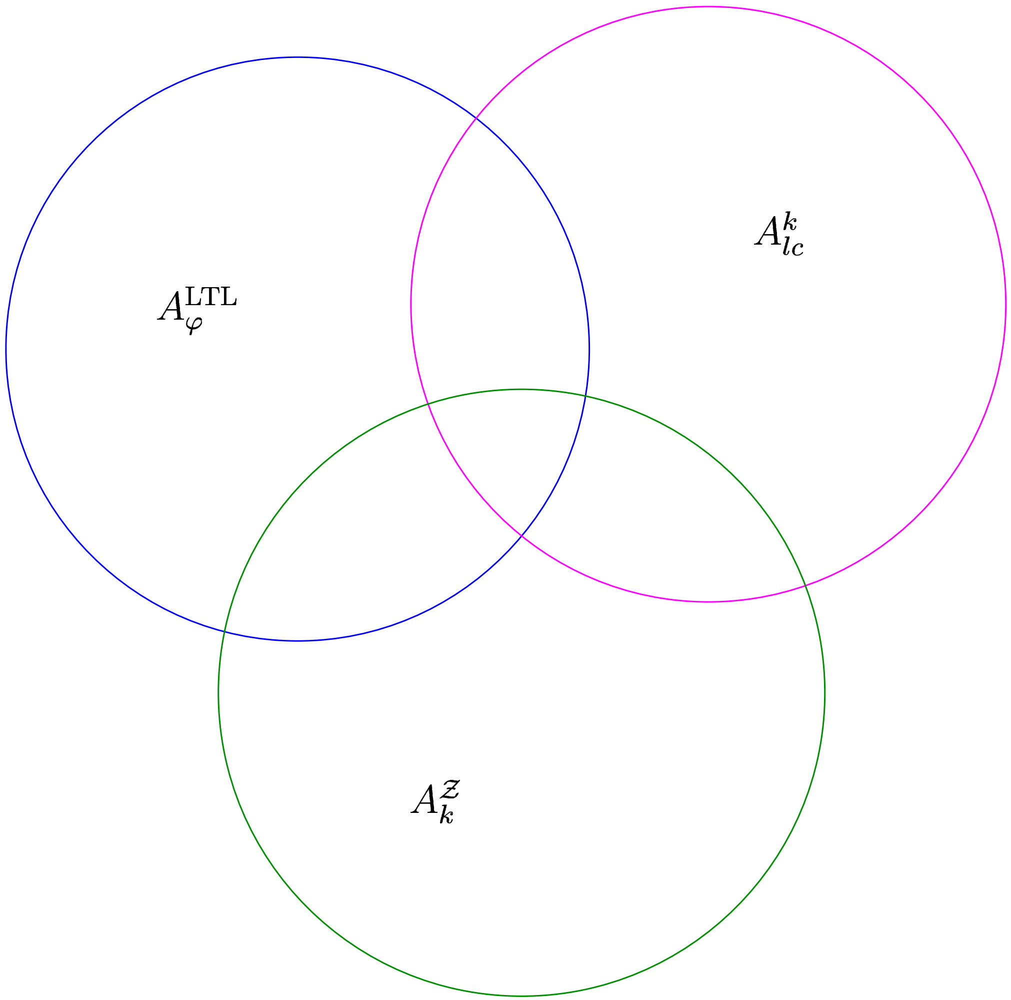
Link between CLTL and LTL

- From Lemma 3.1(DD02)
  - $\sigma \models \varphi$ iff $\rho \models_{\mathrm{LTL}} \varphi$
    - φ : CLTL formula of O-length k
    - σ : $\mathbb{Z}$ - valuation sequence
    - ρ : Induced k frame sequence

From Lemma 3.1(DD02) we get a corollary which describes :

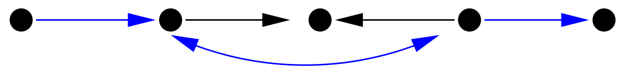
- A CLTL formula φ of O-length k is satisfiable
  - Iff there exist a k-frame sequence :
    - Locally consistent
    - Satisfies φ as a classical LTL formula
    - Admits a $\mathbb{Z}$-valuation sequence

Automata theoretic approach for decidability of CLTL



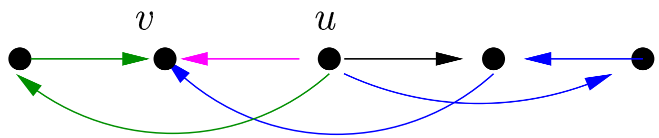
Strict length of a path

- *Slen(p): strict length of a path p in G_ρ* i.e- number of '$<$'-labelled edges in p



  - Slen$(p) = 2$

- *Slen*(u, v) is the maximum of slen(p) where p is the directed path from u to v



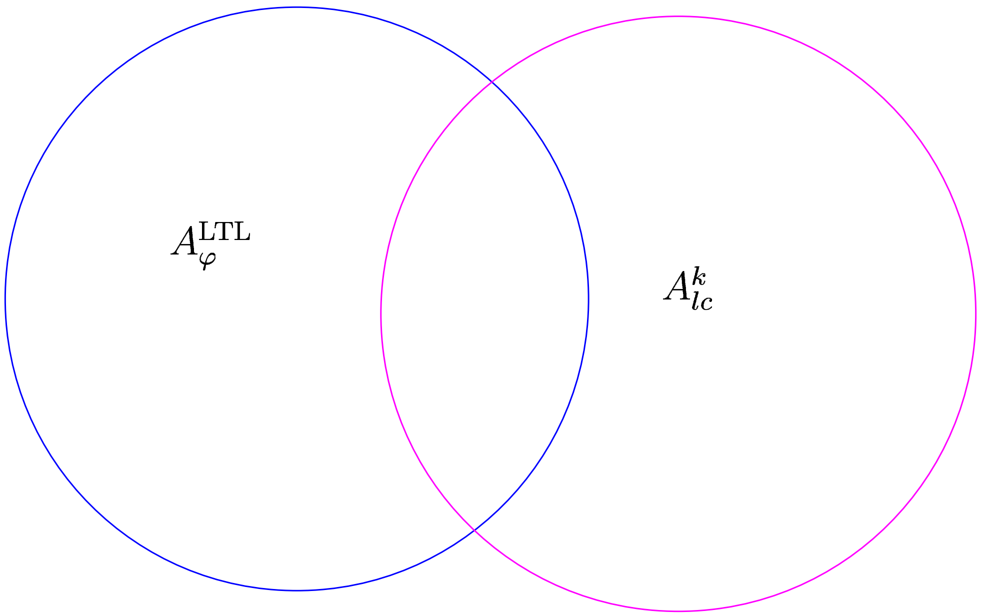
  - Slen$(u, v) = 3$

- In Lemma 6.1 (DD02), ρ admits a $\mathbb{Z}$ -valuation sequence iff for all $u, v \in G_\rho$, slen$(u, v) < \omega$

- In finite models slen(u, v) is bounded

- Lemma 6.1(DD02) applies for finite models also

Automata theoretic approach for decidability of CLTL over finite models



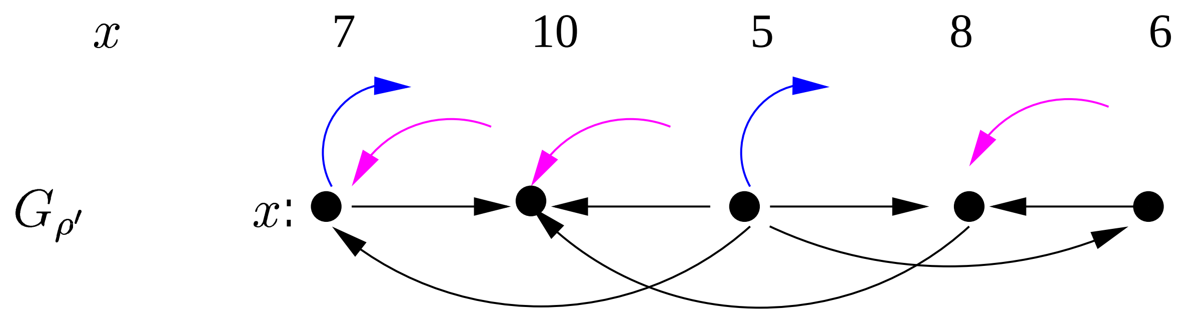
Characterisation of $\mathrm{CLTL}^{\diamond}$ frame graphs

Characterisation of $\mathrm{CLTL}^{\Diamond}$ frame graphs for *single variable models*

Annotated frame graph $G_{\rho'}$

- $(x < \Diamond x)$
  - represented as open forward arc on x in G_ρ
- $(\Diamond x < x)$
  - represented as open backward arc on x in G_ρ
- $(x = \Diamond x)$ and $(\Diamond x = x)$ are always true

Example of an annotated 3-frame graph

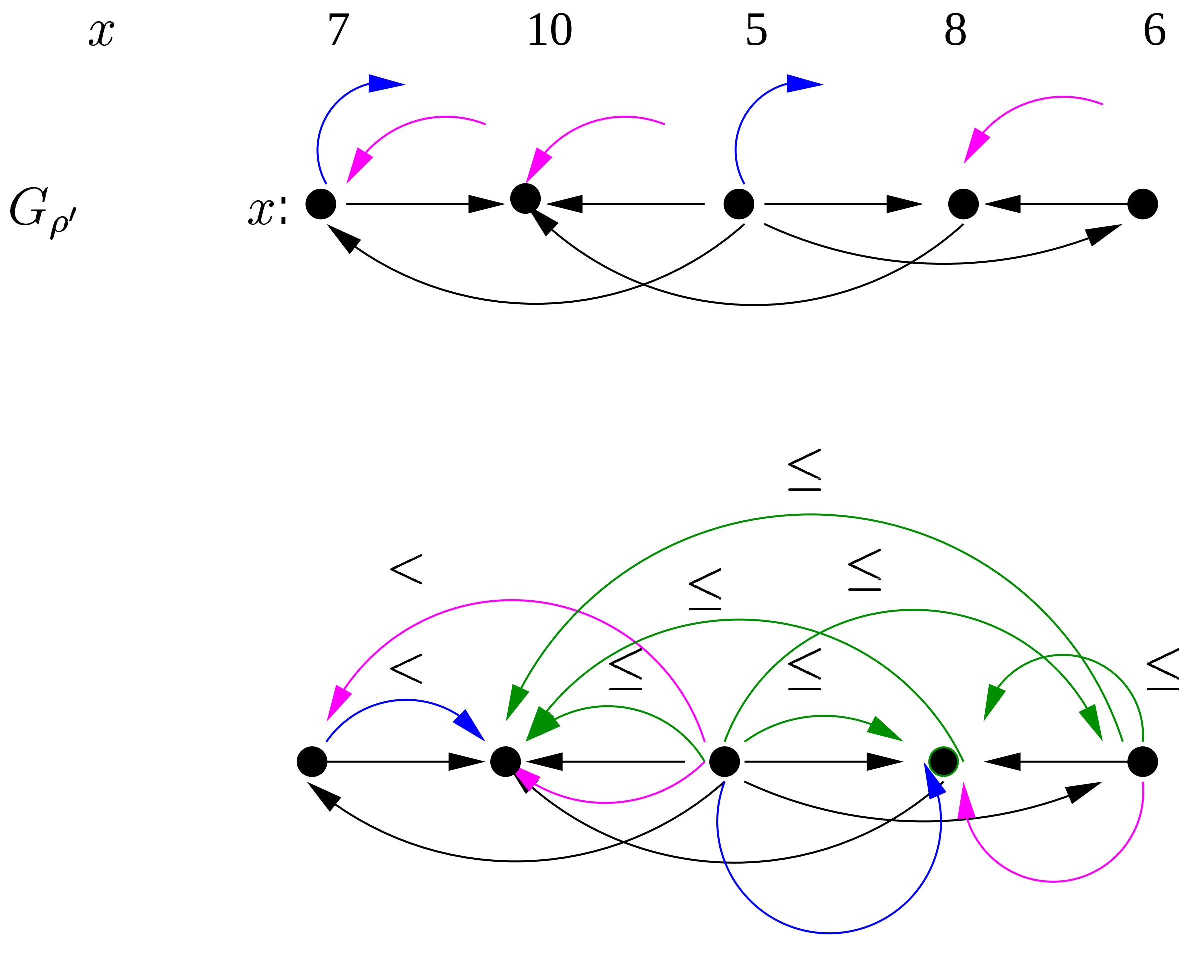


Completion of an annotated frame graph

Annotated graph with matched and implied edges

- *Matched edges*
  - if open forward arc at (x, i), put a '$<$'-labelled edge from (x, i) to some (x, q), $i < q$
  - if open backward arc at (x, i), put a '$<$'-labelled edge from some (x, q) to (x, i), $i < q$
- *Implied edges*
  - if no open forward arc at (x, i), put a '$\leq$'-labelled edge from (x, i) to (x, q), $\forall q, i < q$
  - if no open backward arc at (x, i), put a '$\leq$'-labelled edge from (x, q) to (x, i), $\forall q, i < q$

Example of completion of an annotated frame graph



- *Edge respecting* labelling : If the labels on the vertices connected by an edge (including the implied and matched edges) satisfy the edge relation

- An annotated locally consistent k-frame sequence ρ' admits a $\mathbb{Z}$-valuation sequence σ iff σ is an edge-respecting labelling. From Lemma 5.2 (DD02)

Characterisation of frame graphs which admit a $\mathbb{Z}$ valuation sequence

Lemma 1 *Let ρ' be an annotated locally consistent finite k-frame sequence. Then ρ' admits a $\mathbb{Z}$-valuation sequence iff there is no strict cycle in completion of $G_{\rho'}$.*

Proof

- ρ' admits a $\mathbb{Z}$ -valuation sequence $\Longrightarrow$ edge respecting labelling l for $G_{\rho'}$
- Cycle in completion of $G_{\rho'} \Longrightarrow l(x,i) < l(x,i)$ leads to contradiction

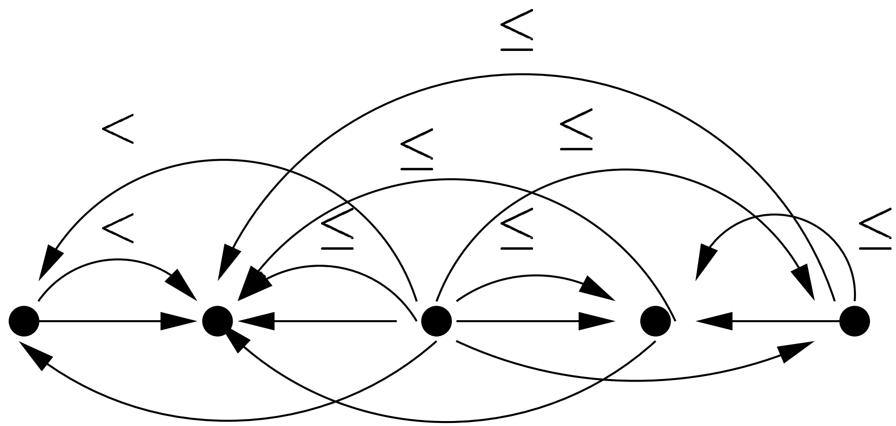
- If there is no cycle in completion of $G_{\rho'}$, the procedure for labelling gives an edge respecting labelling
- Edge respecting labelling $\Longrightarrow$ ρ' admits a $\mathbb{Z}$ -valuation

Procedure for labelling $G_{\rho'}$

1. Label the vertices in order. Begin by labelling the first vertex $(x, 0)$, by 0 .
2. In general if X is the part of the graph which is already labelled, and u is the next vertex to be labelled:
   (a) if there is a directed path from u to a vertex in X, set $l(u) = min\,\{l(v) - sle^{n}(u, v) \mid v \in X$ and there exists a path from u to $v\}$, else,
   (b)set $l(u) = max\,\{l(v) + sle^{n}(v, u) \mid v \in X$ and there exists a path from v to $u\}$.

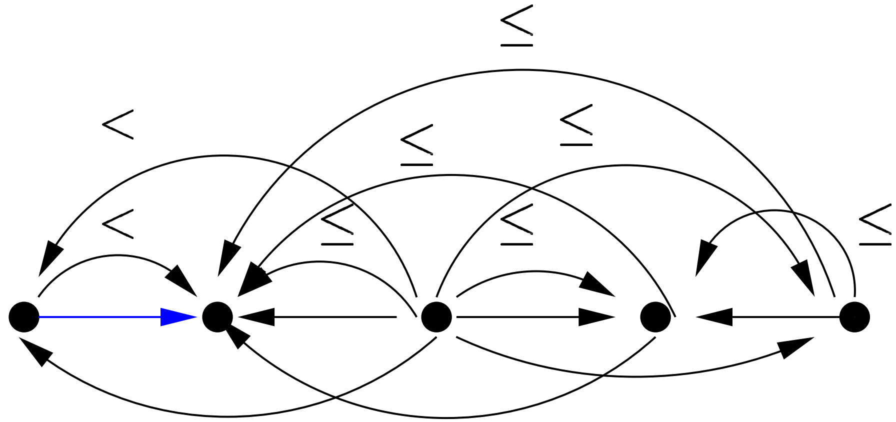
How does the labelling procedure work

$x \qquad 0$



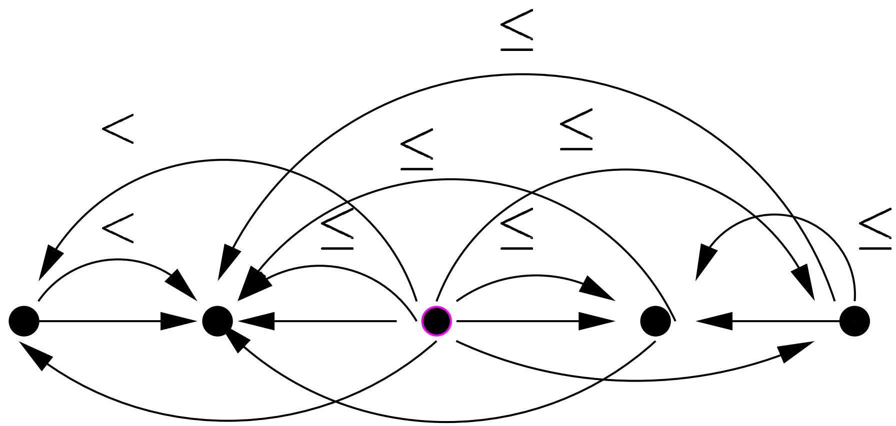
How does the labelling procedure work

x $\quad 0$



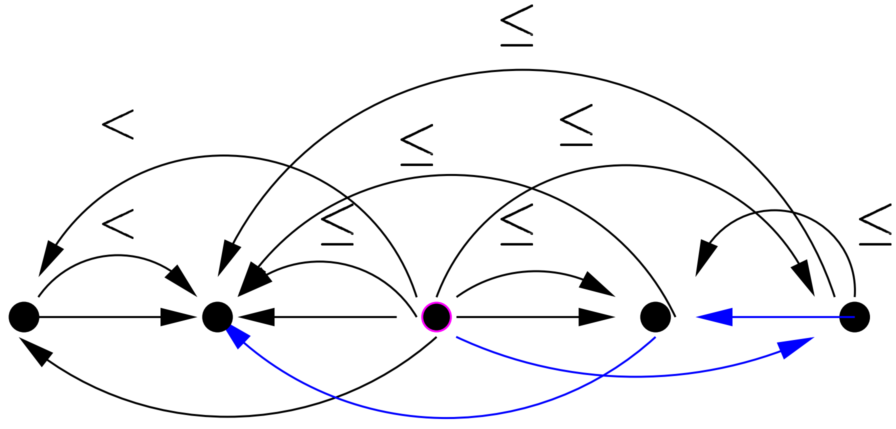
How does the labelling procedure work

x $\quad 0 \quad 1$

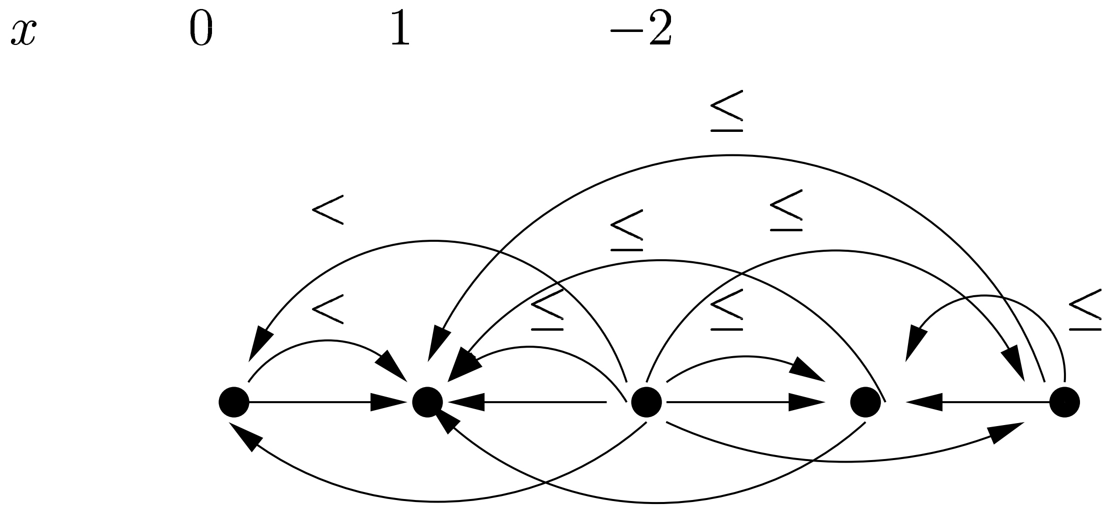


How does the labelling procedure work

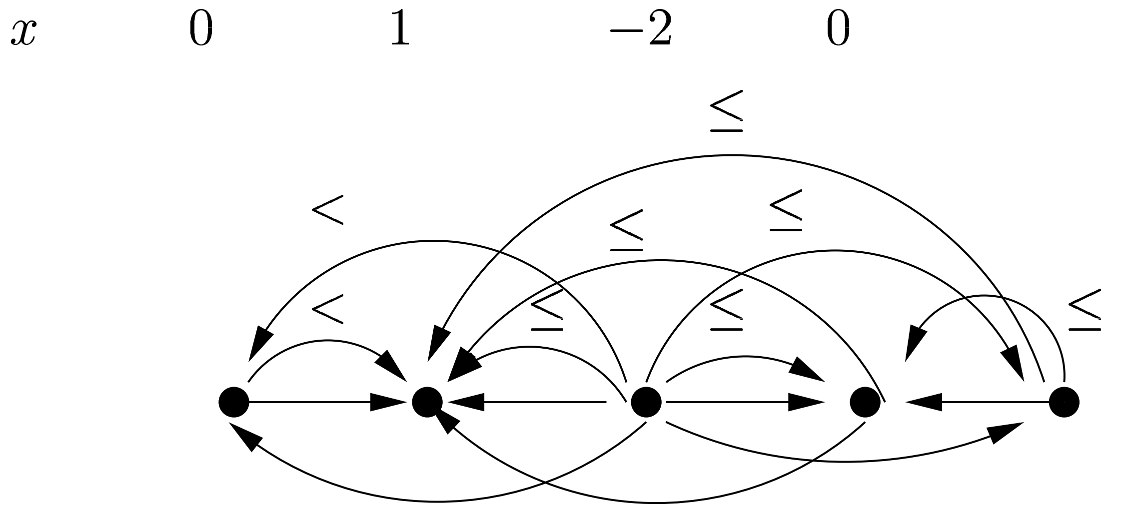
x 0 1



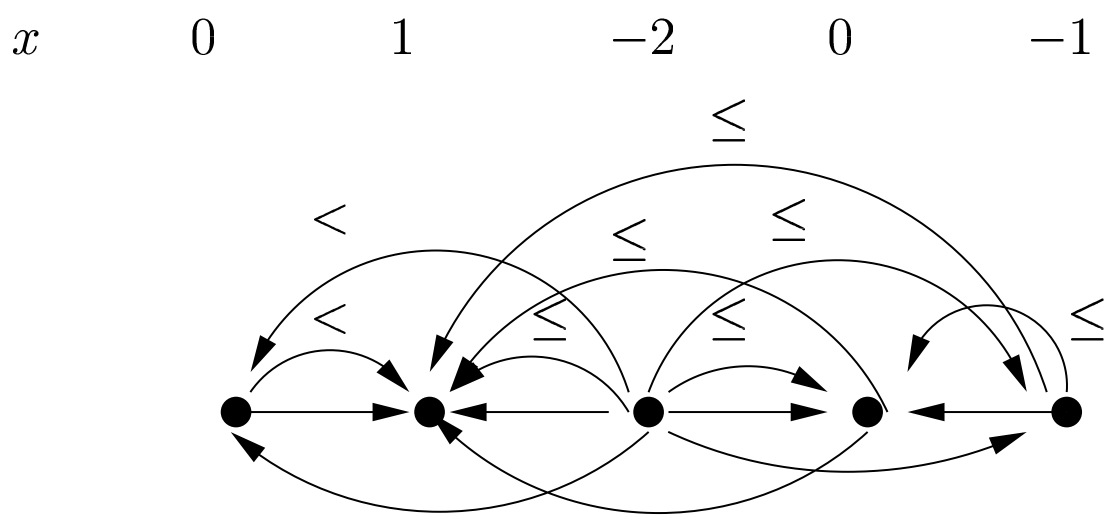
How does the labelling procedure work



How does the labelling procedure work



How does the labelling procedure work



Lemma 2 *Let ρ' be an annotated locally consistent infinite k-frame sequence. Then ρ' admits a $\mathbb{Z}$ -valuation sequence iff $G_{\rho'}$ satisfies the following conditions:*

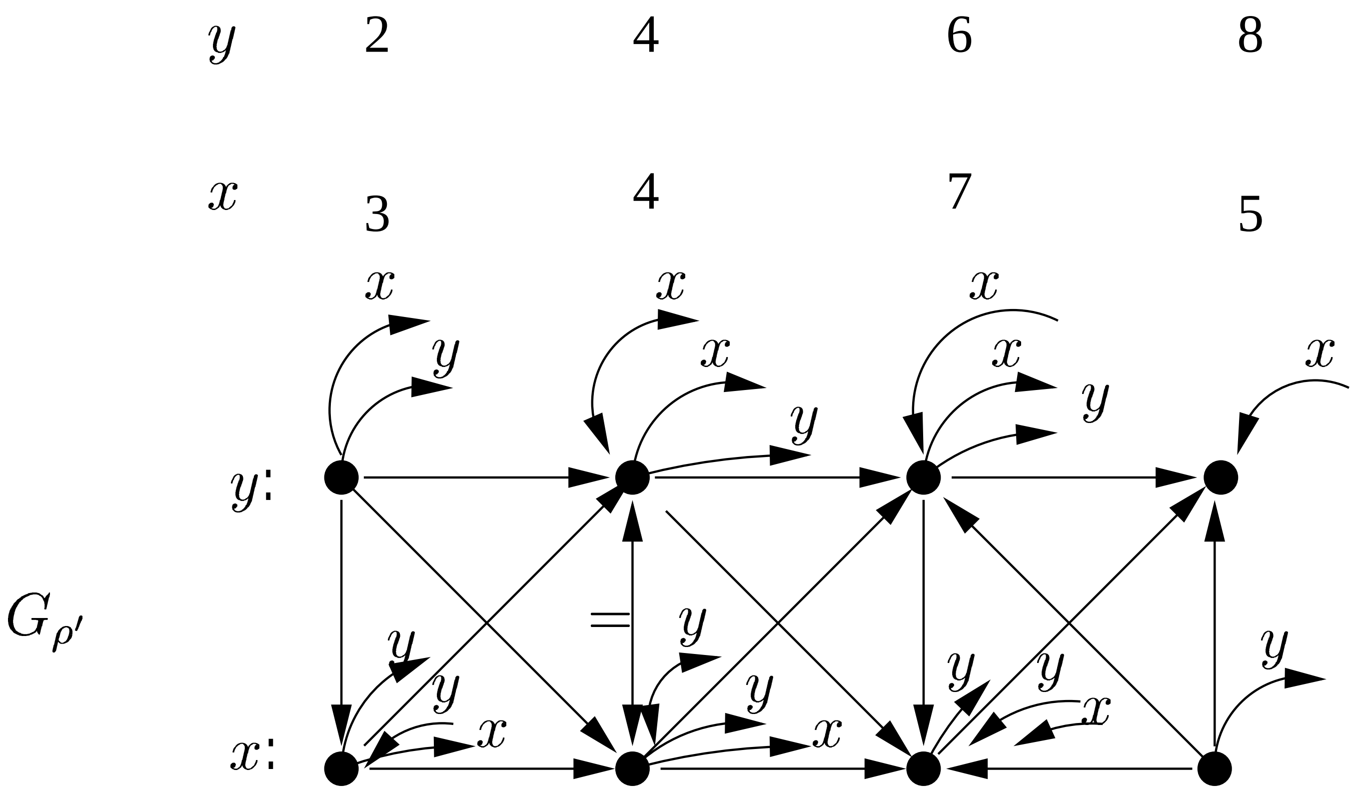
- *There is no strict cycle in the completion of $G_{\rho'}$.*
- *For all the vertices u, v in the completion of $G_{\rho'}$, slen$(u, v) < \omega$.*

Characterisation of $\mathrm{CLTL}^{\diamond}$ frame graphs *for multiple variable models*

Annotated frame graph

- $(x < \Diamond y)$ represented as open forward arc on x labelled y
- $(\Diamond x < y)$ represented as open backward arc on x labelled y
- $(x = \Diamond y)$ represented as open equal to arc labelled y

Example of annotated 2-frame graph $G_{\rho'}$



Completion of annotated frame graphs

Annotated frame graph with matched and implied edges

- Matched edges
  - If open `=' arc at (x, i) labelled y, put a ='-labelled edge from (x, i) to some (y, q), $i \leq q$
- Implied edges
  - If no open `=' arc at $(x, i)$ labelled $y$, put a `¡'-labelled edge from either (x, i) to (y, q) or (y, q) to $(x, i), \forall q, i \leq q$

Characterisation of frame graphs which admit a $\mathbb{Z}$ valuation sequence

Lemma 3 *Let ρ' be an annotated locally consistent finite* k*-frame sequence. Then ρ' admits a $\mathbb{Z}$-valuation sequence iff there is no strict cycle in completion of $G_{\rho'}$.*

Decidability of $\mathrm{CLTL}^{\Diamond}$ over single variable monotonic models

Reduction of satisfiability problem for $\mathrm{CLTL}^{\Diamond}$ to that for CLTL

- c : atomic constraint of $\mathrm{CLTL}^{\Diamond}$
- ϕ' : CLTL formula equivalent to c defined as follows:

Case 1 :

$$
\begin{aligned}
c &:= (x < \Diamond x) \\
\phi' &:= \neg\Box(x = Ox)
\end{aligned}
$$

Case 2 :

$$
\begin{aligned}
c &:= (\Diamond x < x) \\
\phi' &:= \bot
\end{aligned}
$$

In CLTL the constant $\bot$ represents "false".

- $\sigma \models c$ iff $\sigma \models \phi'$ where σ is a monotonic model
- $\sigma \models \phi$ iff $\sigma \models \phi'$ for any $\mathrm{CLTL}^{\Diamond}$ formula ϕ
- CLTL formula is interpreted over non-monotonic models
- The monotonicity condition can be specified as a CLTL formula φ,
  $\varphi := \Box(\neg(O\top) \vee (x < Ox) \vee (x = Ox))$
- Append φ with each ϕ'
- The resultant CLTL formula $(\phi' \wedge \varphi)$ is interpreted over non-monotonic models

Lemma 4 *The satisfiability problem for* $\mathrm{CLTL}^{\Diamond}$ *over single variable monotonic models is decidable.*

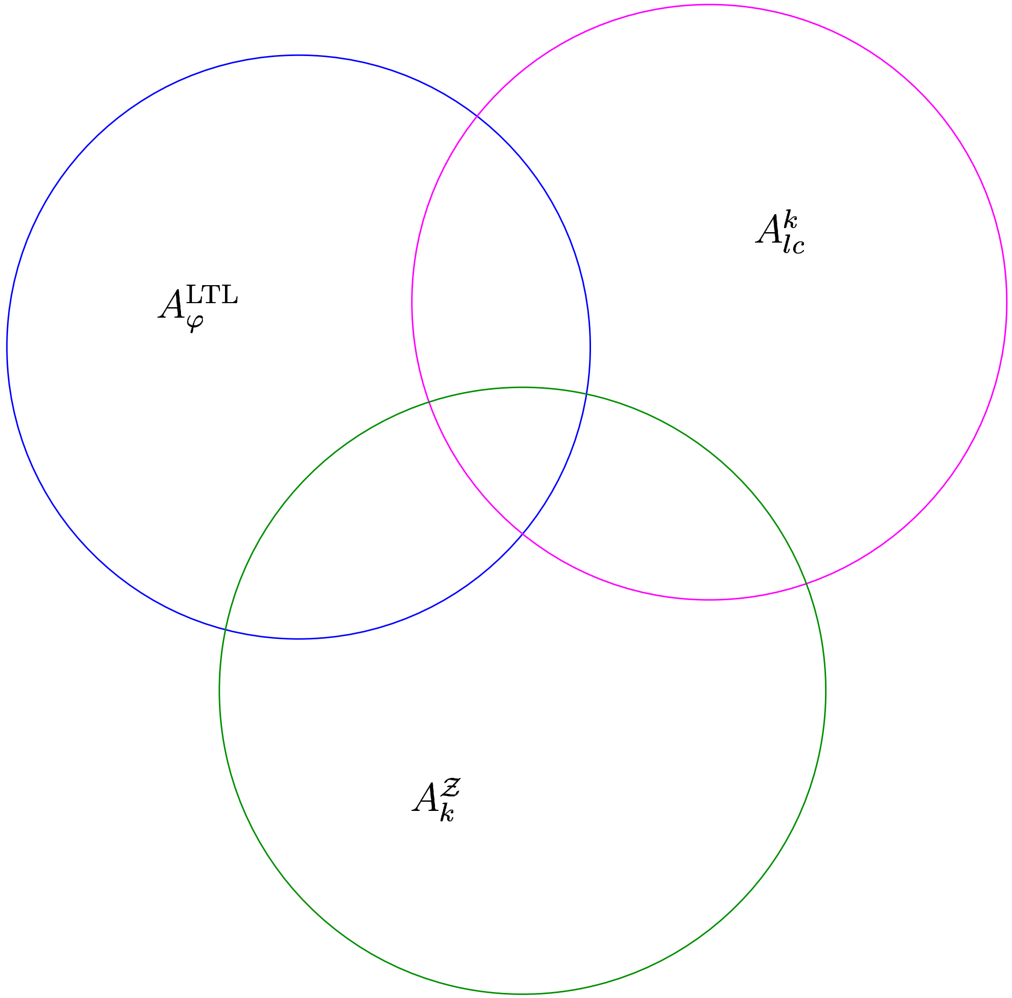
Automata theoretic approach for single variable monotonic model

Annotated frame graph for single variable monotonic model

- Only open forward arcs in $G_{\rho'}$

- Conditions to be checked :
  - Local consistency of frames
  - Completion of $G_{\rho'}$ has no strict cycle
  - For all vertices u, v in the completion of $G_{\rho'}$, slen$(u, v) < \omega$

Automata theoretic approach for $\mathrm{CLTL}^{\diamond}$ monotonic finite models



Construction of $A_k^{\mathcal{Z}}$

- $A_k^{\mathcal{Z}}$ is a nondeterministic finite automaton
  - Guesses a vertex which has an open forward arc
  - When a strict forward edge is found, all the open forward arcs up to that vertex are matched
  - At the end of the model if there is no unmatched open forward arc, then the frame sequence is accepted

Overview of related work

- In TPTL (RT89) employs a novel quantifier construct to reference time: the *freeze quantifier* binds a variable to the time of the local temporal context

- $\mathrm{CLTL}^{\downarrow}(\mathcal{D})$(DRN05) denotes a logic where Constraint LTL is augmented with freeze operator

Conclusion

- $\mathrm{CLTL}^{\Diamond}$ is strictly more expressive than CLTL
- Satisfiability problem for CLTL over finite models is decidable
- Characterisation of $\mathrm{CLTL}^{\Diamond}$ frame graphs
- Satisfiability problem for $\mathrm{CLTL}^{\Diamond}$ over single variable monotonic models is decidable

Future work

- Decidability of the logic or its sublogics
  - By push down automata ?
- Undecidability of the logic
  - Reducing Post correspondence problem to satisfiability problem for the logic ?

References

RT89 Rajeev Alur and Thomas A. Henzinger. A Really Temporal Logic. In *Proc. 30th IEEE Symposium on Foundations of Computer Science(FOCS 1989)*, pp.164-169, and in *Journal of the ACM* 41, pp.181-204, 1994.

DD02 Stephane Demri and Deepak D'Souza. An automata Theoretic Approach to Constraint LTL. Technical Report LSV-03-11, LSV, 2003.40 pages, *Proceedings of FST & TCS'02,Kanpur,* volume 2256 of *Lecture notes in Computer Science*, pages 121-132 Springer, Berlin, 2002.

DRN05 Stephane Demri, Ranko Lazic and David Nowak. On the freeze quantifier in Constraint LTL: decidability and complexity. In *Proc. 12th International Symposium on Temporal Representation and Reasoning'05*, Technical report LSV-05-03, LSV, 2005.

P77 Amir Pnueli. The temporal logic of programs. In *Proc. 18th IEEE Symposium on Foundation of Computer Science*, pages 46-57, 1977.

VW86 M. Vardi and P. Wolper. An automata theoretic approach to automatic program verification. In *Logic in Computer Science*, pages 332-334. IEEE, 1986.

Thank You

LTL as verification tool

- Example of a protocol which implements mutual exclusion
  - Process 1

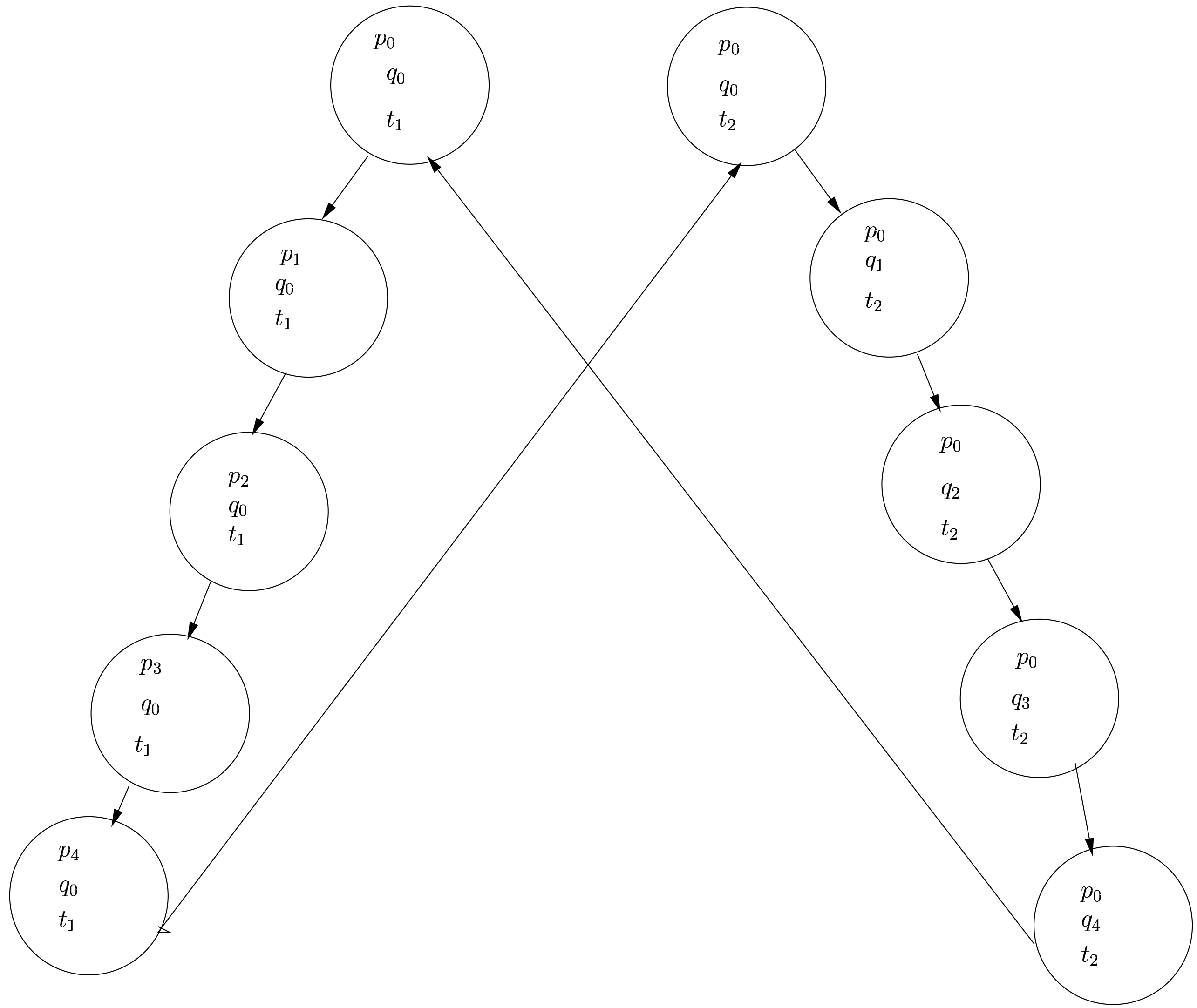
    repeat forever{
    l_0:/* do other jobs*/
    l_1:while (turn!=1){/*do nothing*/ }
    l_2: enter CS;
    l_3:exit CS;
    l_4:turn=2;
    }
  - Process 2

    repeat forever{
    l_0:/* do other jobs*/
    l_1:while (turn!=2){/*do nothing*/ }
    l_2: enter CS;
    l_3:exit CS;
    l_4:turn=1;
    }

- Execution of a program as a state labelled system
  - process 1 is at l_0: p_0
  - process 1 is at l_1: p_1
  - process 1 is at l_2: p_2
  - process 1 is at l_3: p_3
  - process 1 is at l_4: p_4
  - turn = 1 :t_1

  - process 2 is at l_0: q_0
  - process 2 is at l_1: q_1
  - process 2 is at l_2: q_2
  - process 2 is at l_3: q_2
  - process 2 is at l_4: q_4
  - turn =2 :t_2
- Property of a program as a LTL formula
  - Safety condition : $\Box \neg (p_3 \wedge q_3)$
  - Starvation condition : $\Box (t_1 \implies p_3$

State labelled transition system



Syntax and semantics of LTL formula

- Syntax

$$\varphi \quad :: \quad p \mid \neg\varphi \mid \varphi_1 \vee \varphi_2 \mid O\varphi \mid \varphi_1\, \mathcal{U}\, \varphi_2$$

- Semantics of the logic is inductively defined as follows:

$$\begin{array}{lll}
\sigma, i \models p & \text{iff} & p \in \sigma(i) \\
\sigma, i \models \neg\varphi & \text{iff} & \sigma, i \not\models \varphi. \\
\sigma, i \models \varphi_1 \vee \varphi_2 & \text{iff} & \sigma, i \models \varphi_1 \text{ or } \sigma, i \models \varphi_2. \\
\sigma, i \models O\varphi & \text{iff} & \sigma, i+1 \models \varphi. \\
\sigma, i \models \varphi_1\, \mathcal{U}\, \varphi_2 & \text{iff} & \exists k \geq i \text{ such that } \sigma, k \models \varphi_2 \\
 & & \text{and } \forall j : i \leq j < k,\ \ \sigma, j \models \varphi_1.
\end{array}$$

Semantics of $\mathrm{CLTL}^{\Diamond}$ atomic constraints

- $\sigma[i,j] \models O^n x \sim O^m y$ iff $(i+n), (i+m) \leq j$ and $\sigma(i+n)(x) \sim \sigma(i+m)(y)$.
- $\sigma[i,j] \models O^n x \sim \Diamond y$ iff there exists m such that $(i+n), (i+m) \leq j$ and $\sigma(i+n)(x) \sim \sigma(i+m)(y)$.
- $\sigma[i,j] \models \Diamond x \sim O^m y$ iff there exists n such that $(i+n), (i+m) \leq j$ and $\sigma(i+n)(x) \sim \sigma(i+m)(y)$.
- $\sigma[i,j] \models \Diamond x \sim \Diamond y$ iff there exists n and m such that $(i+n)$ and $(i+m) \leq j$ and $\sigma(i+n)(x) \sim \sigma(i+m)(y)$.

Syntax of $\mathrm{CLTL}^{\Diamond}$ formula

- $\varphi ::= c \mid \neg\varphi \mid (\varphi \vee \varphi) \mid O\varphi \mid (\varphi \cup \varphi)$, where c is an atomic constraint.

Semantics of $\mathrm{CLTL}^{\Diamond}$ formula

$\sigma[i,j] \models c$	iff	$\sigma[i,j] \models c.$
$\sigma[i,j] \models \neg\varphi$	iff	$\sigma[i,j] \not\models \varphi.$
$\sigma[i,j] \models \varphi_1 \vee \varphi_2$	iff	$\sigma[i,j] \models \varphi_1$ or $\sigma[i,j] \models \varphi_2.$
$\sigma[i,j] \models O\varphi$	iff	$\sigma[i+1,j] \models \varphi$
$\sigma[i,j] \models \varphi_1 \cup \varphi_2$	iff	$\exists k : i \leq k \leq j$ such that $\sigma[k,j] \models \varphi_2$ and $\forall i' : i \leq i' < k,$ $\sigma[i',j] \models \varphi_1.$

Additional atomic constraints due to $\Diamond$ quantifier

Case 1 :

$$
\begin{aligned}
c &:= (O^n x \sim \Diamond y) \\
c' &:= (O^n x \sim y) \vee (O^n x \sim O y) \vee \ldots \\
&\quad \vee (O^n x \sim O^{n-1} y) \vee O^n (x \sim \Diamond y)
\end{aligned}
$$

Case 2 :

$$
\begin{aligned}
c &:= (\Diamond x \sim O^n y) \\
c' &:= (x \sim O^n y) \vee (O x \sim O^n y) \vee \ldots \\
&\quad \vee (O^{n-1} x \sim O^n y) \vee O^n (\Diamond x \sim y)
\end{aligned}
$$

Case 3 :

$$
\begin{aligned}
c &:= (\Diamond x \sim \Diamond y) \\
c' &:= \Diamond((x \sim \Diamond y) \vee (\Diamond x \sim y))
\end{aligned}
$$

where c is the $\mathrm{CLTL}^{\Diamond}$ atomic constraint and c' is the atomic constraint equivalent to c parsed using $(x \sim \Diamond y)$ and $(\Diamond x \sim y)$

atc(k)

- *atc* (k): The set of all atomic constraints over U of O-length at most k

  σ :

  $$\begin{array}{llllll} y: & 2 & 4 & 6 & 8 & 10 \\ x: & 1 & 3 & 5 & 7 & 9 \end{array}$$

- *atc* (2) is as follows:

  $\{$ $(x{=}x)$, $(x{=}y)$, $(x{=}Ox)$, $(x{=}Oy)$, $(x{<}x)$, $(x{<}y)$, $(x{<}Ox)$, $(x{<}Oy)$, $(y{=}x)$, $(y{=}y)$, $(y{=}Ox)$,$(y{=}Oy)$, $(y{<}x)$,$(y{<}y)$, $(y{<}Ox)$,$(y{<}Oy)$,$(Ox{=}x)$,$(Ox{=}y)$, $(Ox{=}Ox)$,$(Ox{=}Oy)$,$(Ox{<}x)$, $(Ox{<}y)$, $(Ox{<}Ox)$, $(Ox{<}Oy)$,$(Oy{=}x)$,$(Oy{=}y)$, $(Oy{=}Ox)$,$(Oy{=}Oy)$, $(Oy{<}x)$,$(Oy{<}y)$, $(Oy{<}Ox)$,$(Oy{<}Oy)$$\}$

Link between CLTL and LTL

- CLTL formula φ of *O*-length *k* can be viewed as a LTL formula over *atc* (*k*)
- **Example**

  $\varphi : \Box(x < Oy)$

  σ :

  $$\begin{array}{llllll} y: & 2 & 4 & 6 & 8 & 10, \ldots \\ x: & 1 & 3 & 5 & 7 & 9, \ldots \end{array}$$

- Let *O*-length $= 2$

  $2\text{-}frame(\sigma)$: $\{$ $(x{=}x)$, $(x{<}y)$, $(x{<}Ox)$, $(x{<}Oy)$, $(y{=}y)$, $(y{<}Ox)$, $(y{<}Oy)$, $(Ox{=}Ox)$, $(Ox{<}Oy)$, $(Oy{=}Oy)\}$

- $atc(2)$ is as follows:
  { (x=x), (x=y), (x=Ox), (x=Oy), (x<x), (x<y), (x<Ox), (x<Oy), (y=x), (y =y), (y=Ox), (y=Oy), (y<x), (y<y), (y<Ox), (y<Oy), (Ox=x), (Ox=y), (Ox=Ox), (Ox=Oy), (Ox<x), (Ox<y), (Ox<Ox), (Ox<Oy), (Oy=x), (Oy =y), (Oy=Ox), (Oy=Oy), (Oy<x), (Oy<y), (Oy<Ox), (Oy<Oy)}

- Each constraint can be replaced by propositions, $\{p_1, p_2, \ldots, p_{3^2}\}$

- CLTL formula can be written as LTL formula
  - φ_{LTL} : $\Box\, p_8$
  - ρ : $\{p_1, p_6, p_7, p_8, p_{10}, p_{15}, p_{16}, p_{19}, p_{24}, p_{28}\}$, $\{p_1, p_6, p_7, p_8, p_{10}, p_{15}, p_{16}, p_{19}, p_{24}, p_{28}\}, \ldots$
  - $\rho \models_{LTL} \varphi$
- **Lemma 5** (from (DD02)): *Let φ be a CLTL formula of O-length* k. *Let σ be a $\mathbb{Z}$ valuation sequence and let ρ be the induced* k *frame sequence. Then $\sigma \models \varphi$ iff $\rho \models_{\mathrm{LTL}} \varphi$.*

- **Corollary 1** *Let σ and τ be $\mathbb{Z}$-valuation sequences of same length. If frame sequence induced by σ is identical to the frame sequence induced by τ, then for any CLTL formula φ of O-length k, $\tau \models \varphi$ iff $\sigma \models \varphi$.*
- **Proof** :

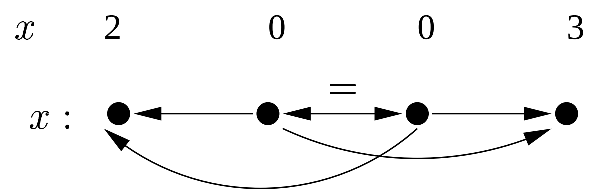
$$
\begin{aligned}
\sigma \in L(\varphi) &\iff k\text{-}fs(\sigma) \in L(\varphi_{\mathrm{LTL}}) && \{\ \because \text{Lemma 1}\,\} \\
&\iff k\text{-}fs(\tau) \in L(\varphi_{\mathrm{LTL}}) && \{\ \because k\text{-}fs(\tau) = k\text{-}fs(\sigma)\,\} \\
&\iff \tau \in L(\varphi) && \{\ \because \text{Lemma 1}\}
\end{aligned}
$$

Outline of the proof of theorem 1

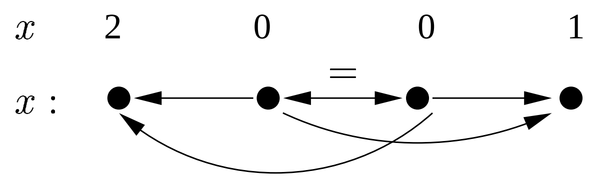
- **Claim** :CLTL$^{\Diamond}$ formula $(x < \Diamond x)$ has no equivalent CLTL formula.

σ	x :	2	0	0	3
τ	x :	2	0	0	1

σ_3



τ_3



- No CLTL formula can distinguish between σ and τ because the *k*-frame sequences induced by both of them are same
- CLTL$^{\Diamond}$ formula distinguishes between σ and τ

Proof

- The proof is by contradiction
- Consider two families of models σ_i and τ_i for i $\geq$ 1
- Length of σ_i and τ_i is $i+1$.
- σ_i and τ_i are as follows:

σ_1	2	0	3				τ_1	2	0	1			
σ_2	2	0	0	3			τ_2	2	0	0	1		
σ_3	2	0	0	0	3		τ_3	2	0	0	0	1	
σ_4	2	0	0	0	0	3	τ_4	2	0	0	0	0	1
$\vdots$							$\vdots$						

$$k\text{-}fs(\sigma) = k\text{-}fs(\tau) \qquad \{Observation\ 1\}$$

For any $k \leq i$,

$$\sigma \models (x < \Diamond x) \text{ and } \tau \not\models (x < \Diamond x)$$

$$\Longrightarrow \sigma \in L(x < \Diamond x) \text{ and } \tau \notin L(x < \Diamond x) \qquad \{Observation\ 2\}$$

Suppose there exists a formula, φ in CLTL such that $L(\varphi) = L(x < \Diamond x)$, $k = O$-length of φ

$\sigma_k \in L(\varphi)$	$\{ \because$ *Definition*$\}$
$k\text{-}fs(\sigma) \in L(\varphi_{\mathrm{LTL}})$	$\{ \because$ *Lemma 1*$\}$
$k\text{-}fs(\tau) \in L(\varphi_{\mathrm{LTL}})$	$\{ \because$ *Observation 1* and *Corollary 2*$\}$
$\tau_k \in L(\varphi)$	$\{ \because$ *Lemma 1*$\}$
But $\tau_k \notin L(x < \Diamond x)$	$\{ \because$ *Observation 2*$\}$
Contradiction	

Validity of the labelling procedure

- suppose the labelling is not valid
- there is a first time when the procedure labels the vertex with a value which contradicts the strict length
- let this vertex be *u*, let the vertex at the other end of the offending path be *v*, $v \in X$ and the vertices labelled up to this point be X

- three cases:
  - p from u to v and $slen(p) > l(v) - l(u)$.
    - Step 2 (a) of the procedure is applicable, and $l(u) \leq l(v) - slen(p)$.
  - p from v to u and $slen(p) > l(u) - l(v)$
    - Two possibilities:
    - u was labelled by step 2 (a) of the procedure.
    - w in X with a path q from u to w, s.t $l(u) = l(w) - slen(q)$.
    - v and w are labelled consistently
    - $l(w) \geq l(v) + slen(p) + slen(q)$.
    - $l(u) \geq l(v) + slen(p)$, leads to contradiction.
    - u was labelled by step 2(b) of the procedure and $l(u) \geq l(v) + slen(p)$.
  - p is a strict path from u to u, leads to contradiction because there is no strict cycle in the graph.

- *infinite backward path* in completion of $G_{\rho'}$: a sequence $d : \mathbb{N} \to U \times \mathbb{N}$, U is the nonempty set of variables in the *k*-frame sequence, satisfying:
  - for all $i \in \mathbb{N}$, there is an edge from $d(i+1)$ to $d(i)$.
  - for all $i \in \mathbb{N}$, if $d(i)$ is in level j, then $d(i+1)$ is in a level greater than or equal to $j+1$. ("level" of a vertex (x, i) is i.
- a path d is strict if there exist infinitely many i for which there is a $<'$-labelled edge from $d(i+1)$ to $d(i)$.

Lemma 6 *Let ρ' be an annotated locally consistent infinite* k-*frame sequence. Then ρ' admits an $\mathbb{N}$-valuation sequence iff $G_{\rho'}$ satisfies the following conditions:*

- *There is no strict cycle in the completion of $G_{\rho'}$.*
- *For all vertices* u,v *in the completion of* $G_{\rho'}$, slen(u,v) $< \omega$.
- *There is no strict infinite backward path in the completion of $G_{\rho'}$.*

Automata theoretic approach for $\mathrm{CLTL}^{\Diamond}$ monotonic infinite models

- Build an automaton ($A_{\varphi}^{\mathcal{Z}}$) which is an intersection of :
  - $A_{\varphi}^{\mathrm{LTL}}$: Vardi Wolper Automaton construction for infinite models
  - $A_{lc}^{k} = (Q, q_0, \longrightarrow, F)$
  - $A_{k}^{\mathcal{Z}}$ is nondeterministic Buchi automaton
- Check the resulting automaton for emptiness to decide the satisfiability of φ

Construction of automata

- A_φ^{LTL} : Vardi Wolper Automaton construction for finite models
  - A state is final iff there is no next state formula in that state

- $A_{lc}^{k} = (Q, q_0, \longrightarrow, F)$
  - Q is the set of k-frames along with a separate start state q_0
  - $\longrightarrow$ is given by $q_0 \longrightarrow r$ on r and $r \longrightarrow r'$ on r' iff (r, r') is locally consistent
  - $F = Q$

Consistent relations from a vertex

- There can be both open forward and open backward arcs
- There can be either open forward or open backward arc
- Both arcs are absent

For finite models there is no open arc from the last vertex

- Build the formula automaton $A_{\varphi}^{\mathrm{LTL}}$: LTL version of the given formula φ.If automata could be build that
  - filter out k-frame sequences that are not locally consistent and
  - filter out k-frame sequences that don't admit $\mathbb{Z}$-valuation sequences
- Intersect and check the resulting automaton for emptiness to decide the satisfiability of φ