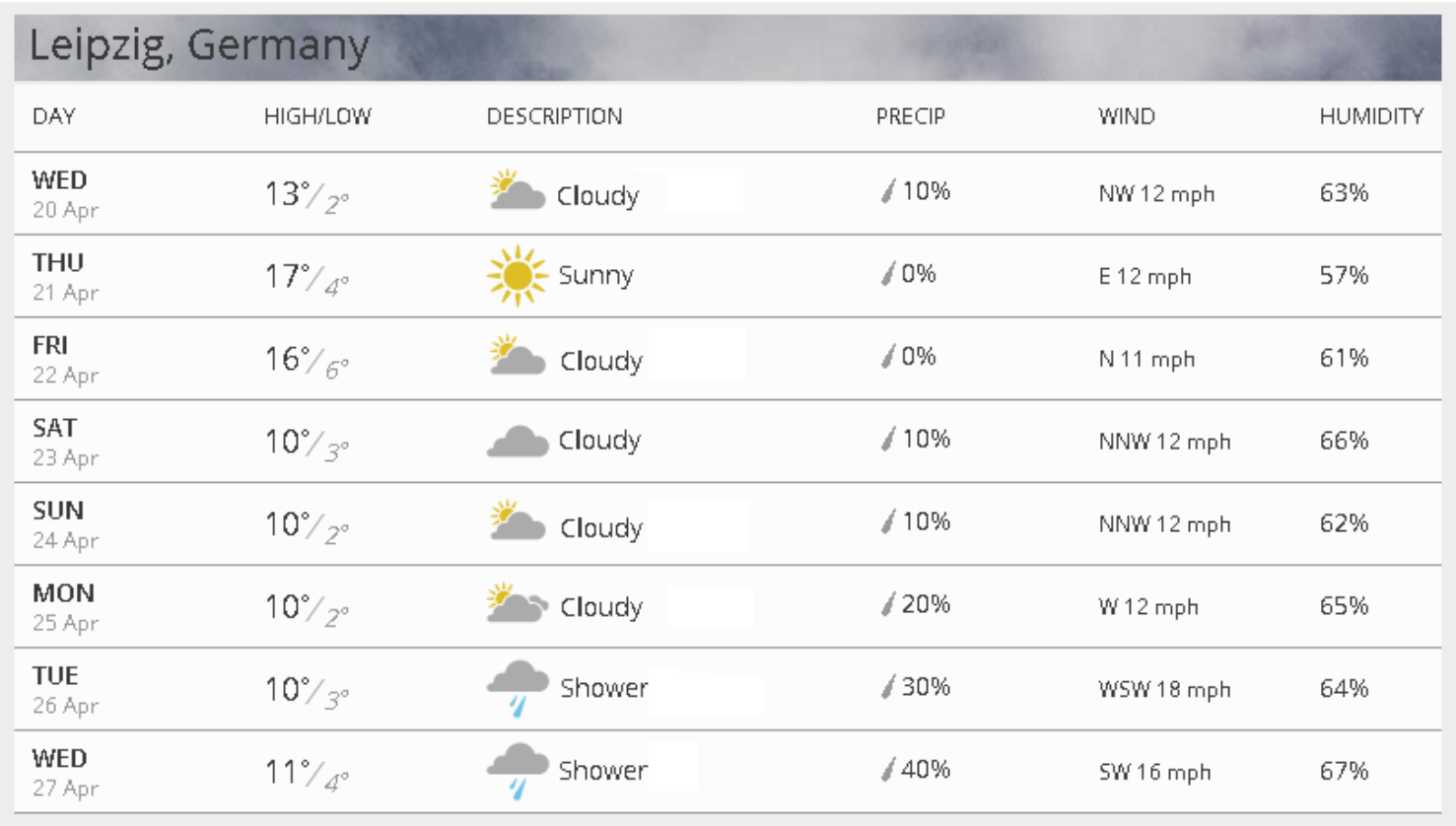
# The expressive power, satisfiability and path checking problems of MTL and TPTL over data words

*Supervisors*
Prof. Markus Lohrey
Prof. Manfred Droste

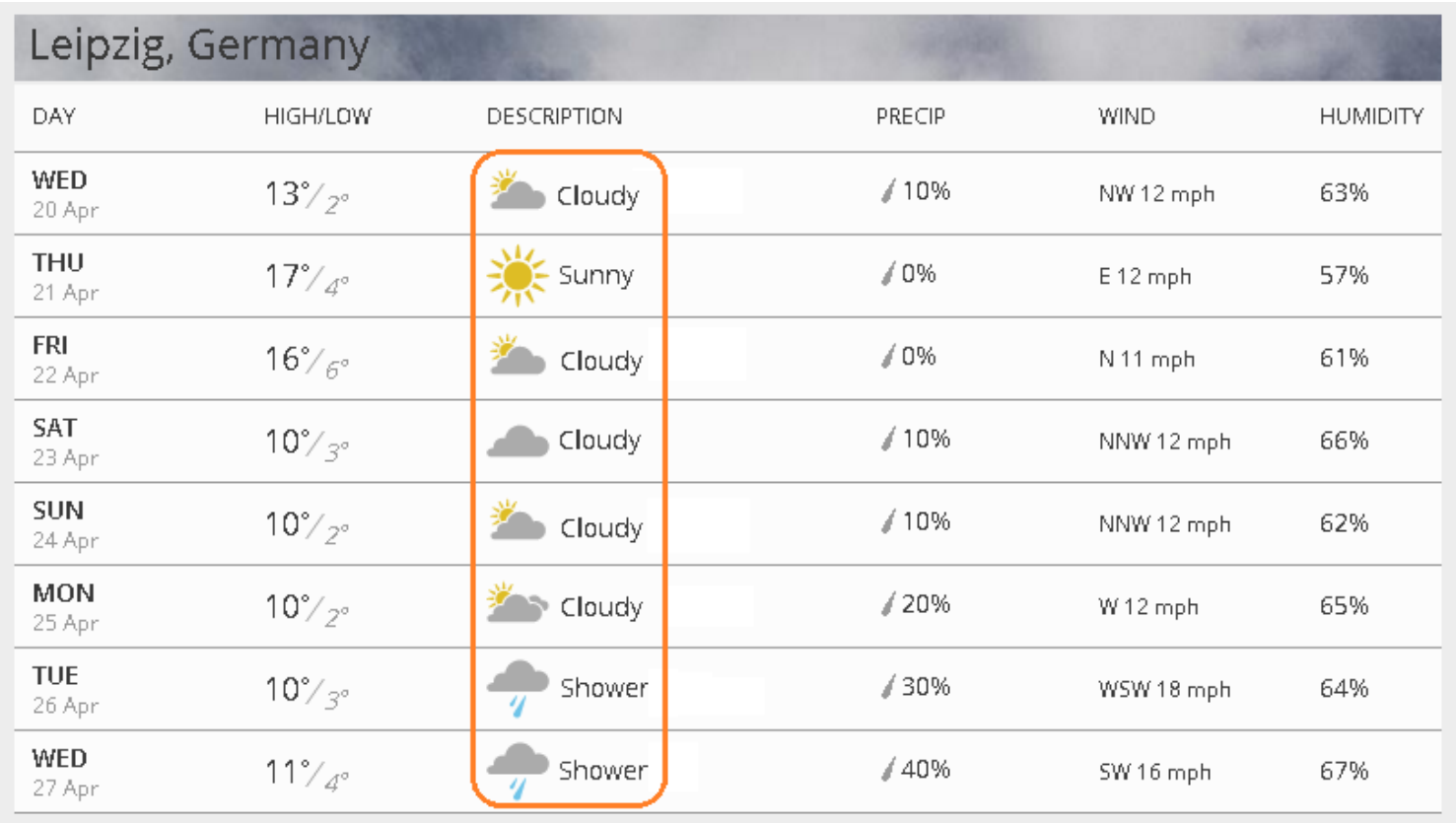
*Candidate*
Shiguang Feng
*Universität Leipzig*

19.04.2016



## Leipzig, Germany

| DAY | HIGH/LOW | DESCRIPTION | PRECIP | WIND | HUMIDITY |
|---|---|---|---|---|---|
| **WED** 20 Apr | 13°/2° | Cloudy | 10% | NW 12 mph | 63% |
| **THU** 21 Apr | 17°/4° | Sunny | 0% | E 12 mph | 57% |
| **FRI** 22 Apr | 16°/6° | Cloudy | 0% | N 11 mph | 61% |
| **SAT** 23 Apr | 10°/3° | Cloudy | 10% | NNW 12 mph | 66% |
| **SUN** 24 Apr | 10°/2° | Cloudy | 10% | NNW 12 mph | 62% |
| **MON** 25 Apr | 10°/2° | Cloudy | 20% | W 12 mph | 65% |
| **TUE** 26 Apr | 10°/3° | Shower | 30% | WSW 18 mph | 64% |
| **WED** 27 Apr | 11°/4° | Shower | 40% | SW 16 mph | 67% |

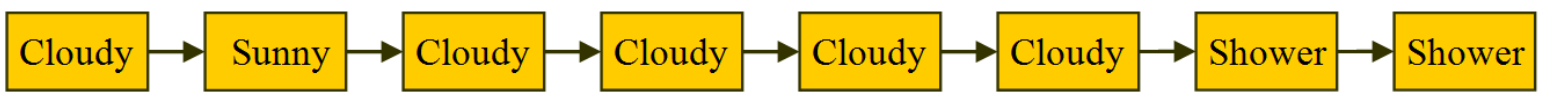


## Leipzig, Germany

| DAY | HIGH/LOW | DESCRIPTION | PRECIP | WIND | HUMIDITY |
|---|---|---|---|---|---|
| **WED** 20 Apr | 13°/2° | Cloudy | 10% | NW 12 mph | 63% |
| **THU** 21 Apr | 17°/4° | Sunny | 0% | E 12 mph | 57% |
| **FRI** 22 Apr | 16°/6° | Cloudy | 0% | N 11 mph | 61% |
| **SAT** 23 Apr | 10°/3° | Cloudy | 10% | NNW 12 mph | 66% |
| **SUN** 24 Apr | 10°/2° | Cloudy | 10% | NNW 12 mph | 62% |
| **MON** 25 Apr | 10°/2° | Cloudy | 20% | W 12 mph | 65% |
| **TUE** 26 Apr | 10°/3° | Shower | 30% | WSW 18 mph | 64% |
| **WED** 27 Apr | 11°/4° | Shower | 40% | SW 16 mph | 67% |

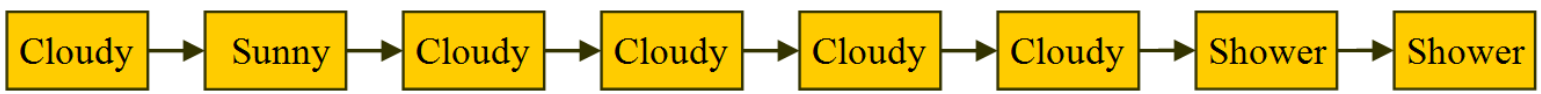
## Leipzig, Germany

| DAY | HIGH/LOW | DESCRIPTION | PRECIP | WIND | HUMIDITY |
|---|---|---|---|---|---|
| WED 20 Apr | 13°/2° | Cloudy | 10% | NW 12 mph | 63% |
| THU 21 Apr | 17°/4° | Sunny | 0% | E 12 mph | 57% |
| FRI 22 Apr | 16°/6° | Cloudy | 0% | N 11 mph | 61% |
| SAT 23 Apr | 10°/3° | Cloudy | 10% | NNW 12 mph | 66% |
| SUN 24 Apr | 10°/2° | Cloudy | 10% | NNW 12 mph | 62% |
| MON 25 Apr | 10°/2° | Cloudy | 20% | W 12 mph | 65% |
| TUE 26 Apr | 10°/3° | Shower | 30% | WSW 18 mph | 64% |
| WED 27 Apr | 11°/4° | Shower | 40% | SW 16 mph | 67% |



## Leipzig, Germany

| DAY | HIGH/LOW | DESCRIPTION | PRECIP | WIND | HUMIDITY |
|---|---|---|---|---|---|
| WED 20 Apr | 13°/2° | Cloudy | 10% | NW 12 mph | 63% |
| THU 21 Apr | 17°/4° | Sunny | 0% | E 12 mph | 57% |
| FRI 22 Apr | 16°/6° | Cloudy | 0% | N 11 mph | 61% |
| SAT 23 Apr | 10°/3° | Cloudy | 10% | NNW 12 mph | 66% |
| SUN 24 Apr | 10°/2° | Cloudy | 10% | NNW 12 mph | 62% |
| MON 25 Apr | 10°/2° | Cloudy | 20% | W 12 mph | 65% |
| TUE 26 Apr | 10°/3° | Shower | 30% | WSW 18 mph | 64% |
| WED 27 Apr | 11°/4° | Shower | 40% | SW 16 mph | 67% |




is a ***word***.

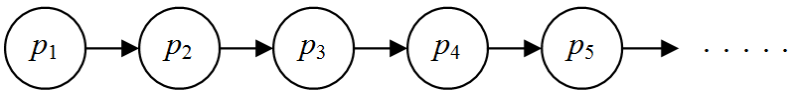
# What is a WORD?

# What is a WORD?

P: a finite set of propositions.

# What is a WORD?

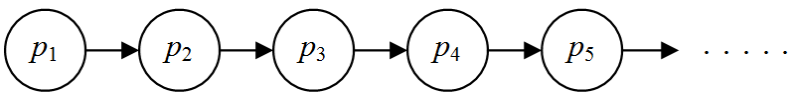
P: a finite set of propositions.



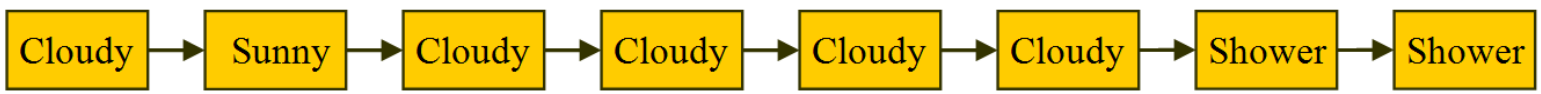

where $p_1, p_2, p_3, \cdots \in \mathrm{P}$.

# What is a WORD?

P: a finite set of propositions.



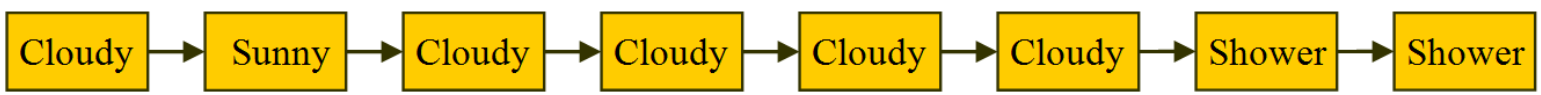

where $p_1, p_2, p_3, \cdots \in \mathrm{P}$.




is a word over the set $\{\mathrm{Sunny}, \mathrm{Cloudy}, \mathrm{Shower}\}$.

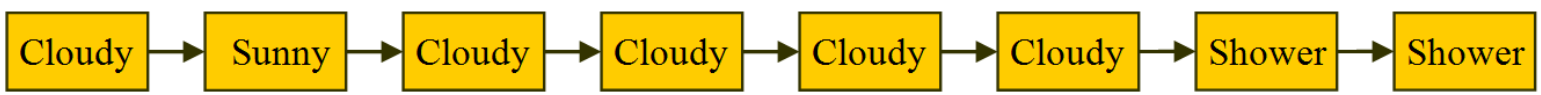
## Leipzig, Germany

| DAY | HIGH/LOW | DESCRIPTION | PRECIP | WIND | HUMIDITY |
|---|---|---|---|---|---|
| **WED** 20 Apr | 13°/2° | Cloudy | 10% | NW 12 mph | 63% |
| **THU** 21 Apr | 17°/4° | Sunny | 0% | E 12 mph | 57% |
| **FRI** 22 Apr | 16°/6° | Cloudy | 0% | N 11 mph | 61% |
| **SAT** 23 Apr | 10°/3° | Cloudy | 10% | NNW 12 mph | 66% |
| **SUN** 24 Apr | 10°/2° | Cloudy | 10% | NNW 12 mph | 62% |
| **MON** 25 Apr | 10°/2° | Cloudy | 20% | W 12 mph | 65% |
| **TUE** 26 Apr | 10°/3° | Shower | 30% | WSW 18 mph | 64% |
| **WED** 27 Apr | 11°/4° | Shower | 40% | SW 16 mph | 67% |



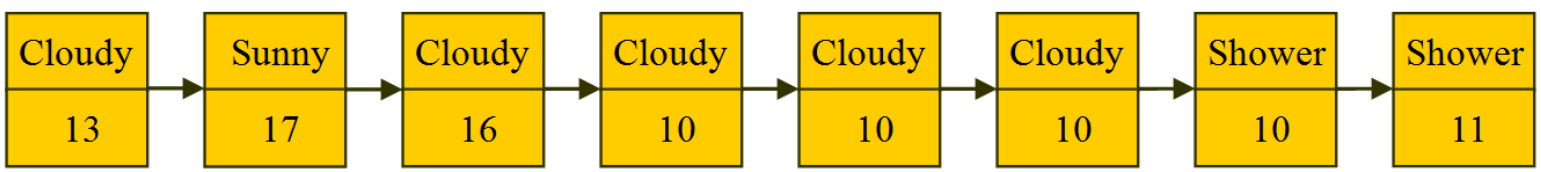
## Leipzig, Germany

| DAY | HIGH/LOW | DESCRIPTION | PRECIP | WIND | HUMIDITY |
|---|---|---|---|---|---|
| WED 20 Apr | 13°/2° | Cloudy | 10% | NW 12 mph | 63% |
| THU 21 Apr | 17°/4° | Sunny | 0% | E 12 mph | 57% |
| FRI 22 Apr | 16°/6° | Cloudy | 0% | N 11 mph | 61% |
| SAT 23 Apr | 10°/3° | Cloudy | 10% | NNW 12 mph | 66% |
| SUN 24 Apr | 10°/2° | Cloudy | 10% | NNW 12 mph | 62% |
| MON 25 Apr | 10°/2° | Cloudy | 20% | W 12 mph | 65% |
| TUE 26 Apr | 10°/3° | Shower | 30% | WSW 18 mph | 64% |
| WED 27 Apr | 11°/4° | Shower | 40% | SW 16 mph | 67% |



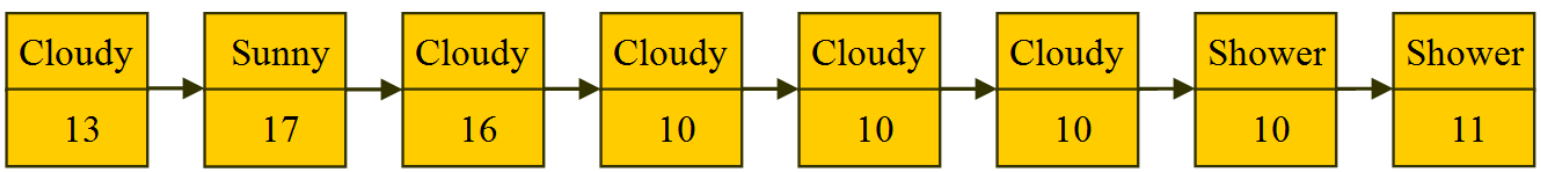
## Leipzig, Germany

| DAY | HIGH/LOW | DESCRIPTION | PRECIP | WIND | HUMIDITY |
|---|---|---|---|---|---|
| WED 20 Apr | 13°/2° | Cloudy | 10% | NW 12 mph | 63% |
| THU 21 Apr | 17°/4° | Sunny | 0% | E 12 mph | 57% |
| FRI 22 Apr | 16°/6° | Cloudy | 0% | N 11 mph | 61% |
| SAT 23 Apr | 10°/3° | Cloudy | 10% | NNW 12 mph | 66% |
| SUN 24 Apr | 10°/2° | Cloudy | 10% | NNW 12 mph | 62% |
| MON 25 Apr | 10°/2° | Cloudy | 20% | W 12 mph | 65% |
| TUE 26 Apr | 10°/3° | Shower | 30% | WSW 18 mph | 64% |
| WED 27 Apr | 11°/4° | Shower | 40% | SW 16 mph | 67% |



## Leipzig, Germany

| DAY | HIGH/LOW | DESCRIPTION | PRECIP | WIND | HUMIDITY |
|---|---|---|---|---|---|
| WED 20 Apr | 13°/2° | Cloudy | 10% | NW 12 mph | 63% |
| THU 21 Apr | 17°/4° | Sunny | 0% | E 12 mph | 57% |
| FRI 22 Apr | 16°/6° | Cloudy | 0% | N 11 mph | 61% |
| SAT 23 Apr | 10°/3° | Cloudy | 10% | NNW 12 mph | 66% |
| SUN 24 Apr | 10°/2° | Cloudy | 10% | NNW 12 mph | 62% |
| MON 25 Apr | 10°/2° | Cloudy | 20% | W 12 mph | 65% |
| TUE 26 Apr | 10°/3° | Shower | 30% | WSW 18 mph | 64% |
| WED 27 Apr | 11°/4° | Shower | 40% | SW 16 mph | 67% |




is a ***data word***.

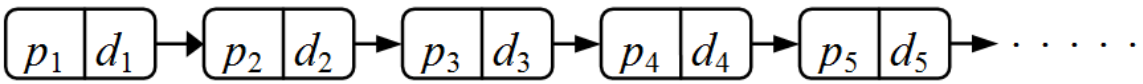
# What is a DATA WORD?

# What is a DATA WORD?

P: a finite set of propositions.

# What is a DATA WORD?

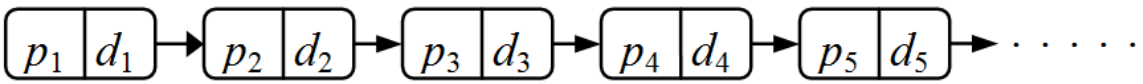
P: a finite set of propositions.



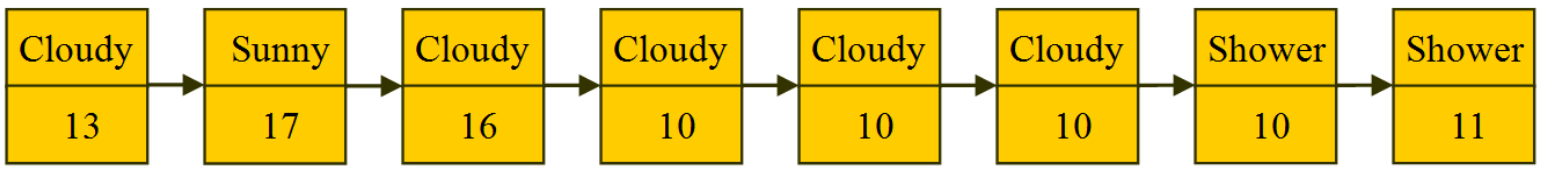

where $p_1, p_2, p_3, \cdots \in \mathrm{P}$, and $d_1, d_2, d_3, \cdots \in \mathbb{N}$.

# What is a DATA WORD?

P: a finite set of propositions.

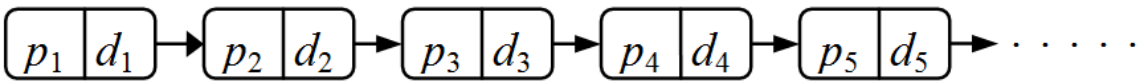



where $p_1, p_2, p_3, \cdots \in \mathrm{P}$, and $d_1, d_2, d_3, \cdots \in \mathbb{N}$.

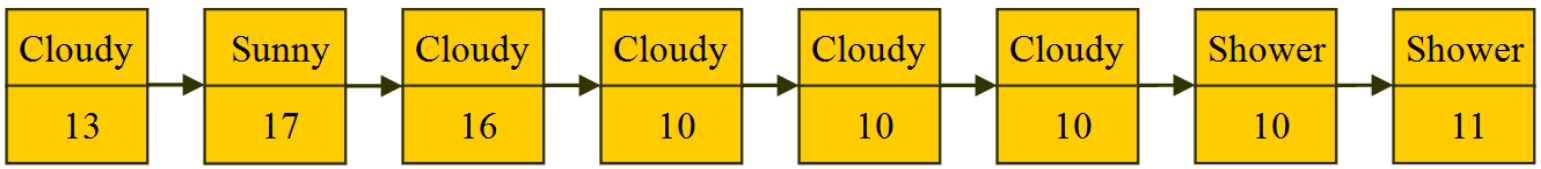


# What is a DATA WORD?

P: a finite set of propositions.



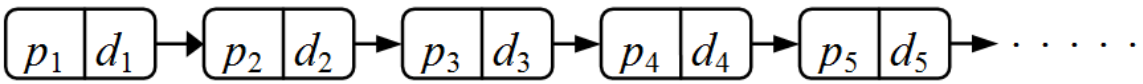

where $p_1, p_2, p_3, \cdots \in \mathrm{P}$, and $d_1, d_2, d_3, \cdots \in \mathbb{N}$.



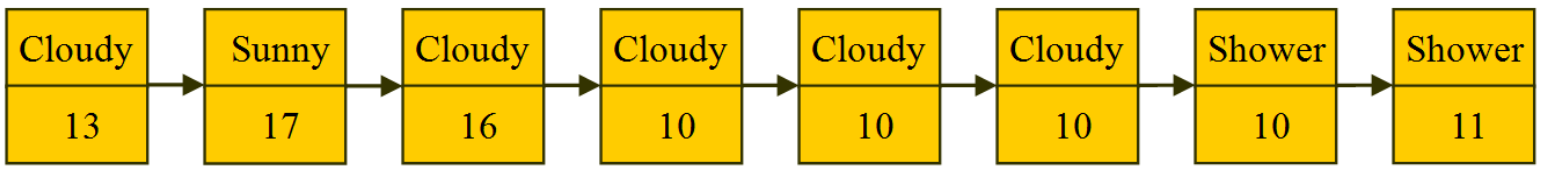

**Monotonic**: if $d_i \leq d_{i+1}$ $(i = 1, 2, \ldots)$.

# What is a DATA WORD?

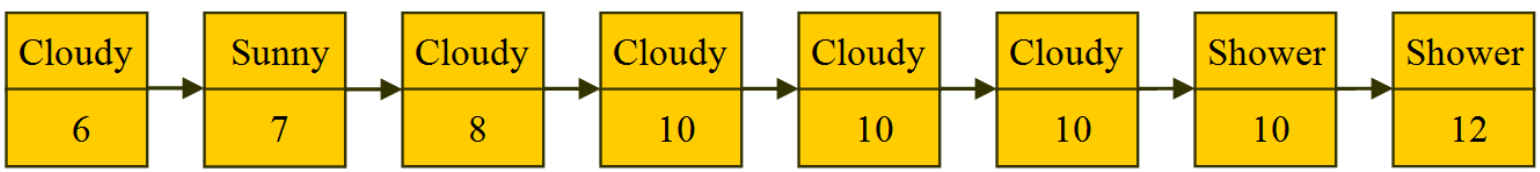
P: a finite set of propositions.




where $p_1, p_2, p_3, \cdots \in \mathrm{P}$, and $d_1, d_2, d_3, \cdots \in \mathbb{N}$.



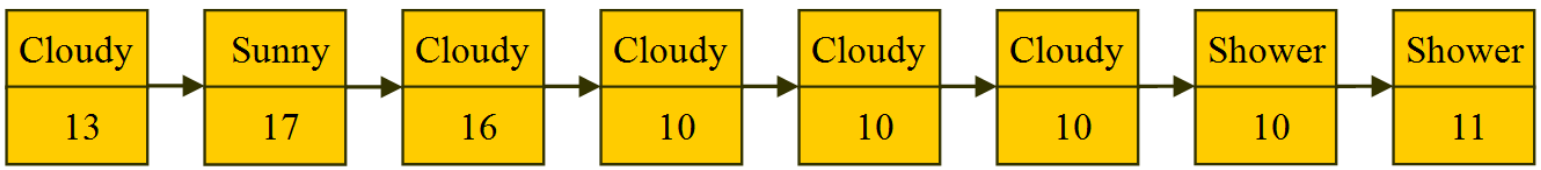

**Monotonic**: if $d_i \leq d_{i+1}$ $(i = 1, 2, \ldots)$.




is monotonic.

How to express the following statements in a logic?

- Tomorrow is sunny, and the temperature is 4 degrees higher than that of today.
- It is cloudy until a day with showers.



# Linear Temporal Logic (LTL)

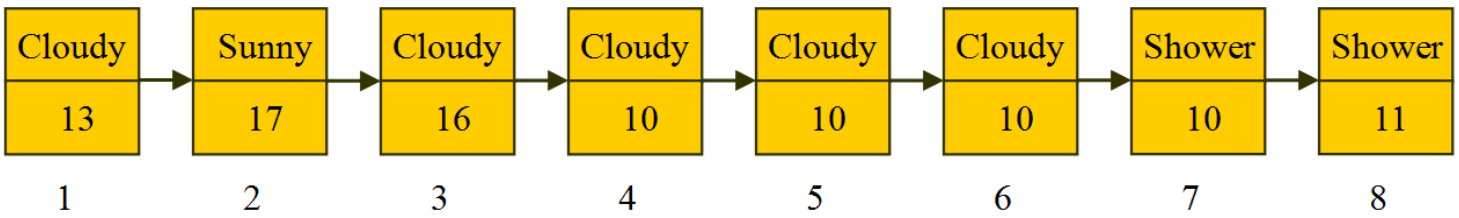
Syntax of LTL:

$$p \mid \neg\varphi \mid \varphi \wedge \psi \mid \varphi \mathsf{U} \psi \mid \mathsf{X}\varphi \quad (p \in \mathsf{P})$$

# Linear Temporal Logic (LTL)

## Syntax of LTL:

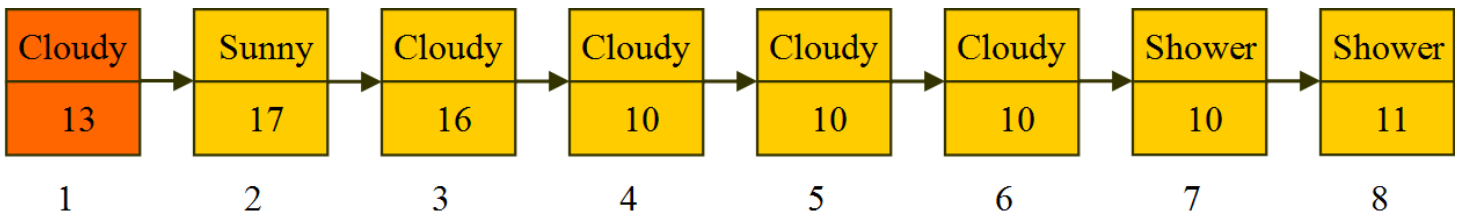
$$p \mid \neg\varphi \mid \varphi \wedge \psi \mid \varphi \mathsf{U} \psi \mid \mathsf{X}\varphi \quad (p \in \mathsf{P})$$



# Linear Temporal Logic (LTL)

Syntax of LTL:

$$p \mid \neg\varphi \mid \varphi \wedge \psi \mid \varphi \mathsf{U} \psi \mid \mathsf{X}\varphi \quad (p \in \mathsf{P})$$

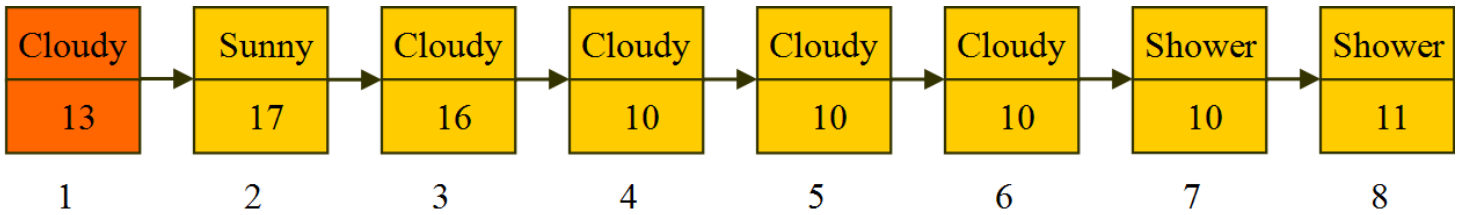



$(w, 1) \models \text{cloudy} \wedge \neg\text{sunny}.$

# Linear Temporal Logic (LTL)

Syntax of LTL:

$$p \mid \neg\varphi \mid \varphi \wedge \psi \mid \varphi \mathsf{U} \psi \mid \mathsf{X}\varphi \quad (p \in \mathsf{P})$$



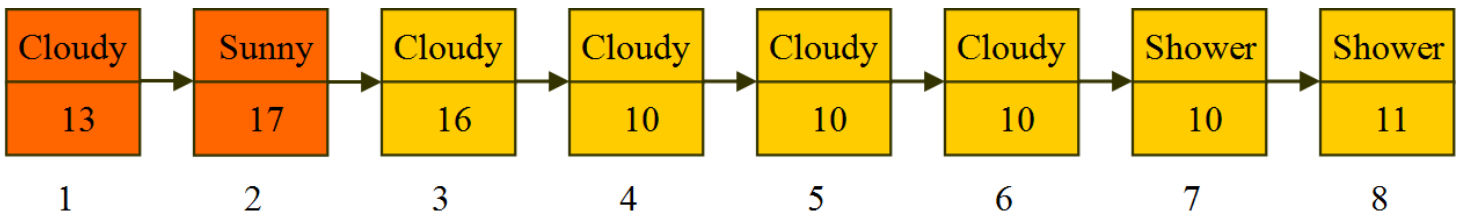

$(w, 1) \models \text{cloudy} \wedge \neg\text{sunny}.$

$(w, 1) \models \mathsf{X}\,\text{sunny}$ : the next day is sunny (i.e. $(w, 2) \models \text{sunny}$).

# Linear Temporal Logic (LTL)

Syntax of LTL:

$$p \mid \neg\varphi \mid \varphi \wedge \psi \mid \varphi \mathsf{U} \psi \mid \mathsf{X}\varphi \quad (p \in \mathsf{P})$$



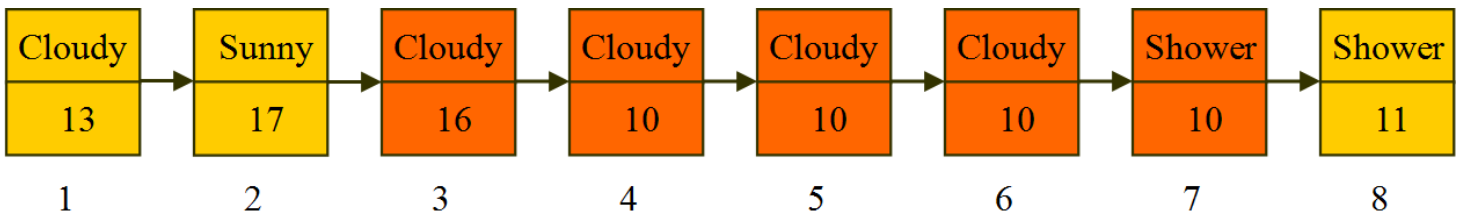

$(w, 1) \models \text{cloudy} \wedge \neg\text{sunny}.$

$(w, 1) \models \mathsf{X}\,\text{sunny}$ : the next day is sunny (i.e. $(w, 2) \models \text{sunny}$).

# Linear Temporal Logic (LTL)

Syntax of LTL:

$$p \mid \neg\varphi \mid \varphi \wedge \psi \mid \varphi \mathsf{U} \psi \mid \mathsf{X}\varphi \quad (p \in \mathsf{P})$$




$(w,1) \models \text{cloudy} \wedge \neg\text{sunny}$.

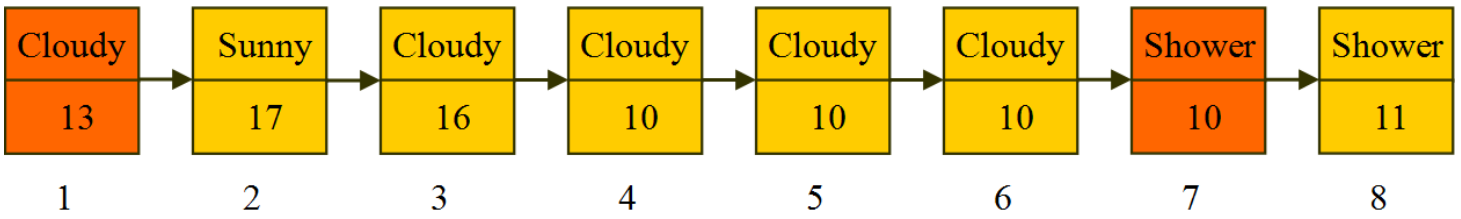
$(w,1) \models \mathsf{X}\,\text{sunny}$ : the next day is sunny (i.e. $(w,2) \models \text{sunny}$).

$(w,3) \models \text{cloudy}\,\mathsf{U}\,\text{shower}$ : it is cloudy until a day with showers.

# Linear Temporal Logic (LTL)

Syntax of LTL:

$$p \mid \neg\varphi \mid \varphi \wedge \psi \mid \varphi \mathsf{U} \psi \mid \mathsf{X}\varphi \quad (p \in \mathsf{P})$$




$(w,1) \models \text{cloudy} \wedge \neg\text{sunny}$.

$(w,1) \models \mathsf{X}\,\text{sunny}$ : the next day is sunny (i.e. $(w,2) \models \text{sunny}$).

$(w,3) \models \text{cloudy}\,\mathsf{U}\,\text{shower}$ : it is cloudy until a day with showers.

$(w,1) \models \mathsf{F}\,\text{shower}$ : there is one day in the future with showers.
$(\mathsf{F}\varphi := \mathsf{True}\,\mathsf{U}\,\varphi)$

# Metric Temporal Logic (MTL)

Syntax of MTL:

$$p \mid \neg\varphi \mid \varphi \wedge \psi \mid \varphi \mathsf{U}_I \psi \mid \mathsf{X}_I \varphi$$

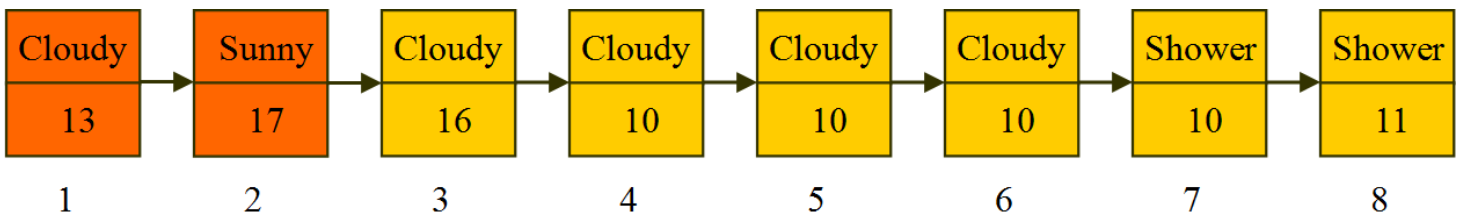
($p \in \mathsf{P}, I \subseteq \mathbb{Z}$ is an interval)

# Metric Temporal Logic (MTL)

Syntax of MTL:

$$p \mid \neg\varphi \mid \varphi \wedge \psi \mid \varphi \mathsf{U}_I \psi \mid \mathsf{X}_I \varphi$$

($p \in \mathsf{P}, I \subseteq \mathbb{Z}$ is an interval)



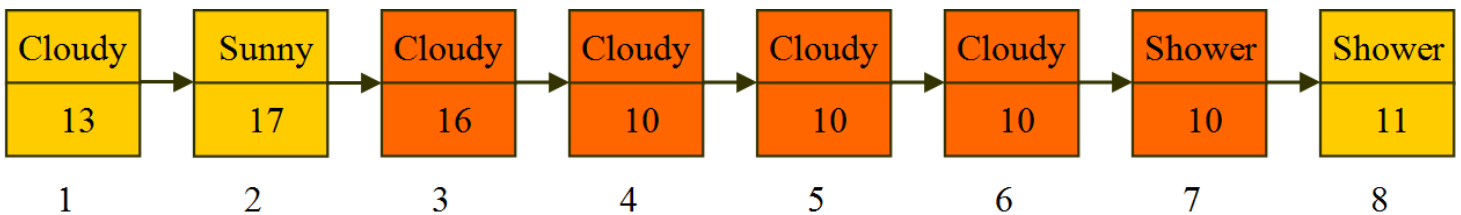

$(w,1) \models \mathsf{X}_{[2,6]}\,\text{sunny} : (w,2) \models \mathsf{X}\,\text{sunny}$ and $(17-13) \in [2,6]$.

# Metric Temporal Logic (MTL)

Syntax of MTL:

$$p \mid \neg\varphi \mid \varphi \wedge \psi \mid \varphi \mathsf{U}_I \psi \mid \mathsf{X}_I \varphi$$

($p \in \mathsf{P}, I \subseteq \mathbb{Z}$ is an interval)




$(w,1) \models \mathsf{X}_{[2,6]}\,\text{sunny} : (w,2) \models \mathsf{X}\,\text{sunny}$ and $(17-13) \in [2,6]$.

$(w,3) \models \text{cloudy}\,\mathsf{U}_{[-10,0]}\,\text{shower} : \ldots$ and $(10-16) \in [-10,0]$.

# Timed Propositional Temporal Logic (TPTL)

## Syntax of TPTL

$$p \mid \neg\varphi \mid \varphi \wedge \psi \mid \varphi \mathsf{U} \psi \mid \mathsf{X}\varphi \mid x.\varphi \mid x \in I$$

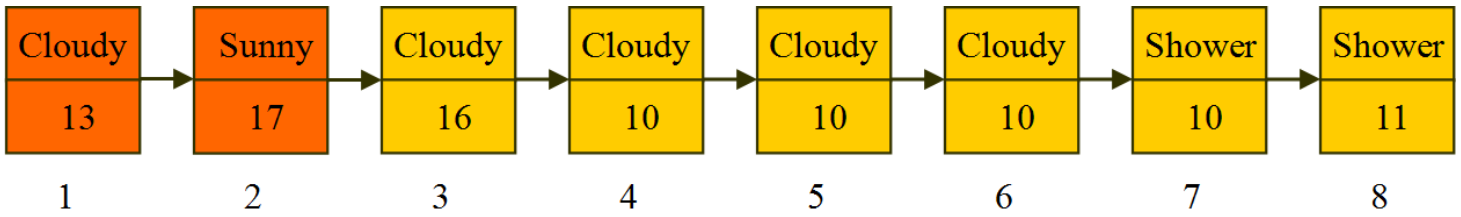
($x$ is a register variable, $I \subseteq \mathbb{Z}$ is an interval)

# Timed Propositional Temporal Logic (TPTL)

## Syntax of TPTL

$$p \mid \neg\varphi \mid \varphi \wedge \psi \mid \varphi \mathsf{U} \psi \mid \mathsf{X}\varphi \mid x.\varphi \mid x \in I$$

($x$ is a register variable, $I \subseteq \mathbb{Z}$ is an interval)



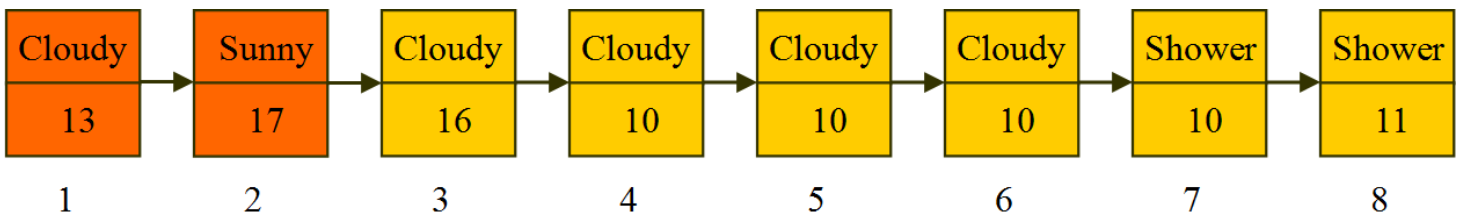

$\mathsf{X}_{[2,6]}$ sunny

# Timed Propositional Temporal Logic (TPTL)

## Syntax of TPTL

$$p \mid \neg\varphi \mid \varphi \wedge \psi \mid \varphi \mathsf{U} \psi \mid \mathsf{X}\varphi \mid x.\varphi \mid x \in I$$

($x$ is a register variable, $I \subseteq \mathbb{Z}$ is an interval)



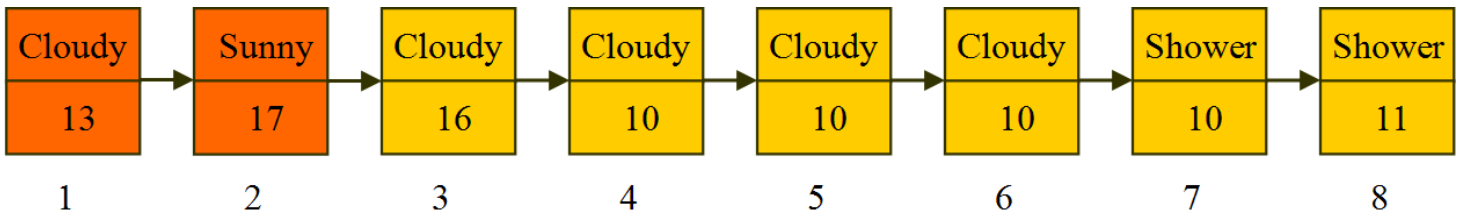

$x.\mathsf{X}(\text{sunny} \wedge x \in [2,6]) \equiv \mathsf{X}_{[2,6]}\,\text{sunny}$

# Timed Propositional Temporal Logic (TPTL)

## Syntax of TPTL

$$p \mid \neg\varphi \mid \varphi \wedge \psi \mid \varphi \mathsf{U} \psi \mid \mathsf{X}\varphi \mid x.\varphi \mid x \in I$$

($x$ is a register variable, $I \subseteq \mathbb{Z}$ is an interval)




$$(w,1) \models x.\mathsf{X}(\text{sunny} \wedge x \in [2,6])$$

store the current data value 13

check whether $17 - x \in [2,6]$

# Motivation

- Specification and verification of reactive systems

# Motivation

- Specification and verification of reactive systems
- Many extensions of LTL, e.g., CTL, MTL, TPTL

# Motivation

- Specification and verification of reactive systems
- Many extensions of LTL, e.g., CTL, MTL, TPTL
- Well-studied over monotonic data words

# Motivation

- Specification and verification of reactive systems
- Many extensions of LTL, e.g., CTL, MTL, TPTL
- Well-studied over monotonic data words
- Consider MTL and TPTL over all data words

# Motivation

- Specification and verification of reactive systems
- Many extensions of LTL, e.g., CTL, MTL, TPTL
- Well-studied over monotonic data words
- Consider MTL and TPTL over all data words
  - Expressive power

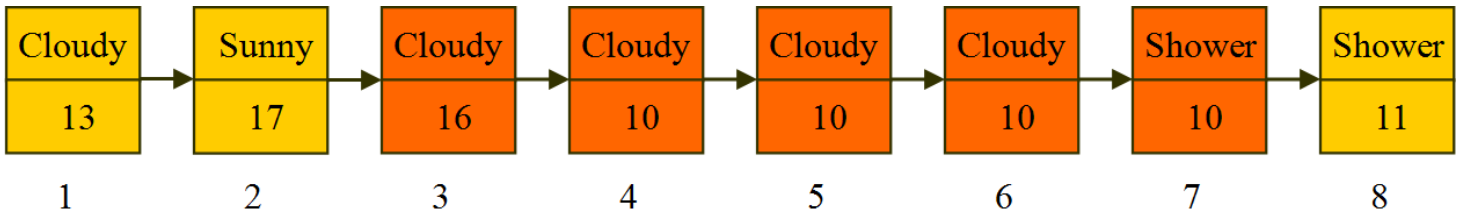
# Motivation

- Specification and verification of reactive systems
- Many extensions of LTL, e.g., CTL, MTL, TPTL
- Well-studied over monotonic data words
- Consider MTL and TPTL over all data words
  - Expressive power
  - Satisfiability

# Motivation

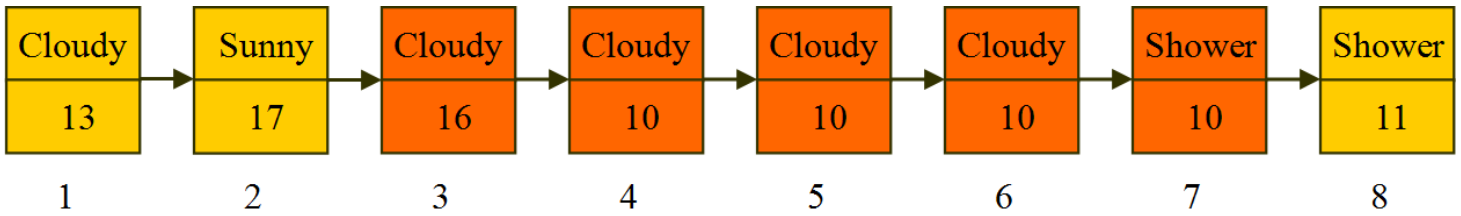
- Specification and verification of reactive systems
- Many extensions of LTL, e.g., CTL, MTL, TPTL
- Well-studied over monotonic data words
- Consider MTL and TPTL over all data words
  - Expressive power
  - Satisfiability
  - Complexity of path checking

# What is the expressive power?



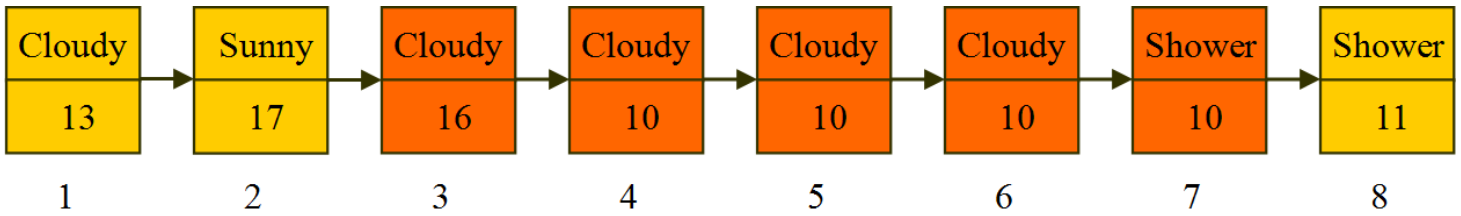

- It is cloudy until a day with showers and the temperature of each of the next few days is 6 degrees lower than that of the initial day.

# What is the expressive power?



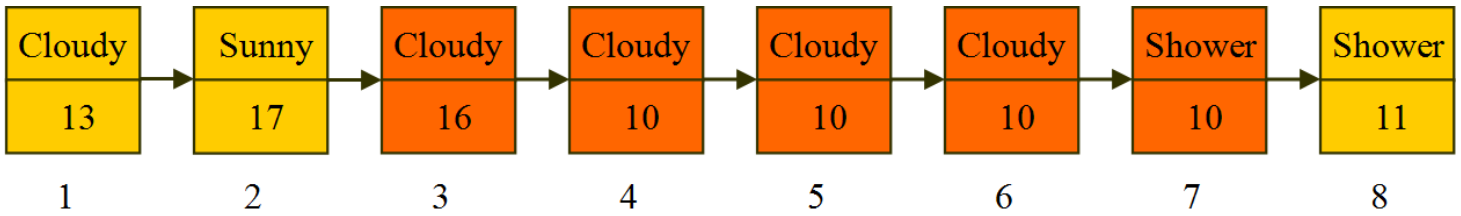

- It is cloudy until a day with showers and the temperature of each of the next few days is 6 degrees lower than that of the initial day.
- $\varphi = x.[(\text{cloudy} \wedge x \in [-6,-6])\,\mathsf{U}\,(\text{shower} \wedge x \in [-6,-6])]$

# What is the expressive power?

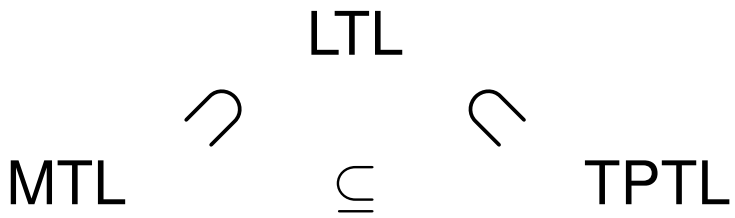


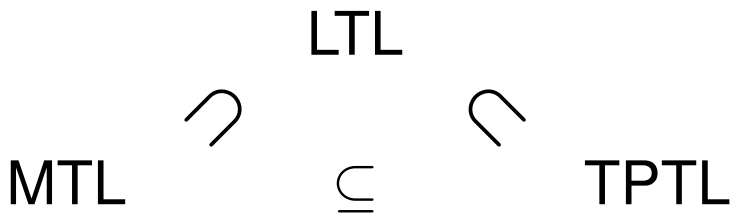

- It is cloudy until a day with showers and the temperature of each of the next few days is 6 degrees lower than that of the initial day.
- $\varphi = x.[(\text{cloudy} \wedge x \in [-6,-6])\,\mathsf{U}\,(\text{shower} \wedge x \in [-6,-6])]$
- Is $\varphi$ expressible in MTL?

# What is the expressive power?



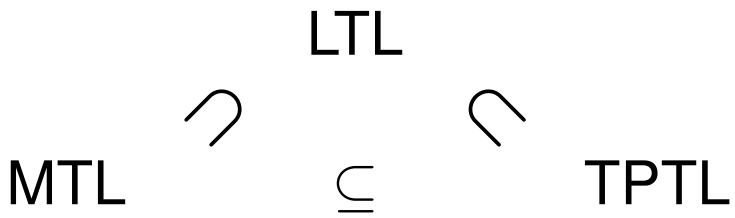

- It is cloudy until a day with showers and the temperature of each of the next few days is 6 degrees lower than that of the initial day.
- $\varphi = x.[(\text{cloudy} \wedge x \in [-6,-6])\,\mathsf{U}\,(\text{shower} \wedge x \in [-6,-6])]$
- Is $\varphi$ expressible in MTL?
- Do MTL and TPTL have the same expressive power?


LTL
MTL
⊆
TPTL




- MTL $\equiv$ TPTL on monotonic data words.

[Alur&Henzinger, 1993]

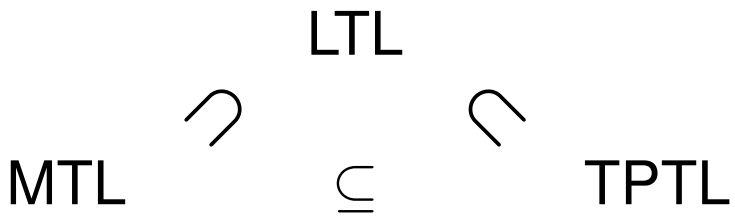



- MTL $\equiv$ TPTL on monotonic data words.

[Alur&Henzinger, 1993]

- MTL $\subsetneq$ TPTL on timed words.

[Bouyer et al, 2010]




- MTL $\equiv$ TPTL on monotonic data words.

  [Alur&Henzinger, 1993]

- MTL $\subsetneq$ TPTL on timed words.

  [Bouyer et al, 2010]

- MTL $\overset{?}{\equiv}$ TPTL on ***all*** data words?

Theorem: MTL $\subsetneq$ TPTL on all data words.

Theorem: MTL $\subsetneq$ TPTL on all data words.

Proof idea:

The TPTL formula $x.\mathrm{XX}(x = 0)$ cannot be expressed in MTL.

Theorem: MTL $\subsetneq$ TPTL on all data words.

Proof idea:

The TPTL formula $x.\mathsf{XX}(x = 0)$ cannot be expressed in MTL.

- $x.\mathsf{XX}(x = 0) \notin \mathsf{MTL}^{I}_{k}$, for each $k \in \mathbb{N}$ and finite $I \subseteq \mathbb{Z}$.

Theorem: MTL $\subsetneq$ TPTL on all data words.

Proof idea:

The TPTL formula $x.\mathsf{XX}(x = 0)$ cannot be expressed in MTL.

- $x.\mathsf{XX}(x = 0) \notin \mathsf{MTL}_k^I$, for each $k \in \mathbb{N}$ and finite $I \subseteq \mathbb{Z}$.
- $\mathsf{MTL}_k^I$ is the set of MTL formulas where
  - at most $k$ nested X and U operators
  - $\varphi_1 \mathsf{U}_{[a,b]} \varphi_2 \Rightarrow a, b \in I$

Theorem: MTL $\subsetneq$ TPTL on all data words.

Proof idea:

The TPTL formula $x.\mathsf{XX}(x = 0)$ cannot be expressed in MTL.

- $x.\mathsf{XX}(x = 0) \notin \mathrm{MTL}_k^I$, for each $k \in \mathbb{N}$ and finite $I \subseteq \mathbb{Z}$.
- $\mathrm{MTL}_k^I$ is the set of MTL formulas where
  - at most $k$ nested X and U operators
  - $\varphi_1 \mathsf{U}_{[a,b]} \varphi_2 \Rightarrow a, b \in I$
  - $\mathsf{X}(p \mathsf{U}_{[1,2]} q) \in \mathrm{MTL}_2^{\{1,2\}}$

# EF game for MTL

Played by: Spoiler • and Duplicator • on $w_0$ and $w_1$.

# EF game for MTL

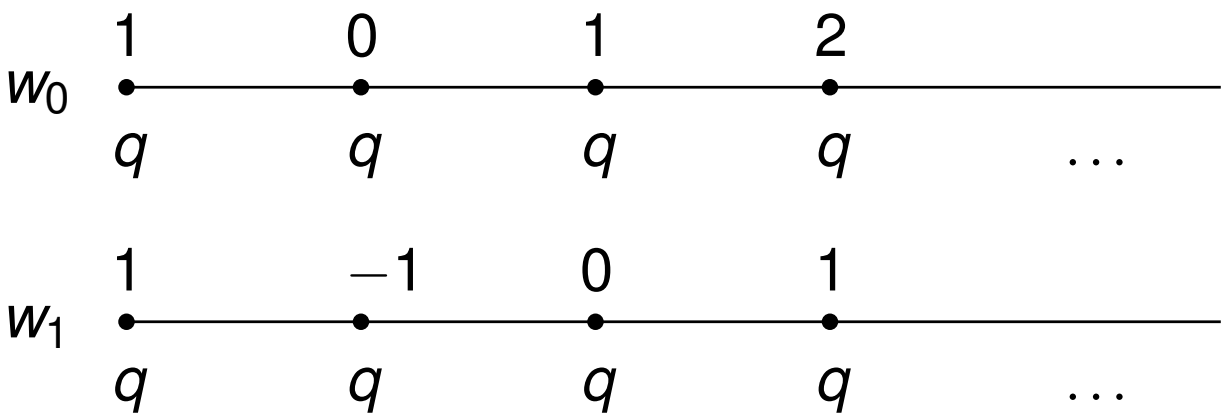
Played by: Spoiler • and Duplicator • on $w_0$ and $w_1$.

$x.\mathsf{XX}(x = 0) \notin \mathsf{MTL}_1^{\{0,1\}}$

# EF game for MTL

Played by: Spoiler • and Duplicator • on $w_0$ and $w_1$.

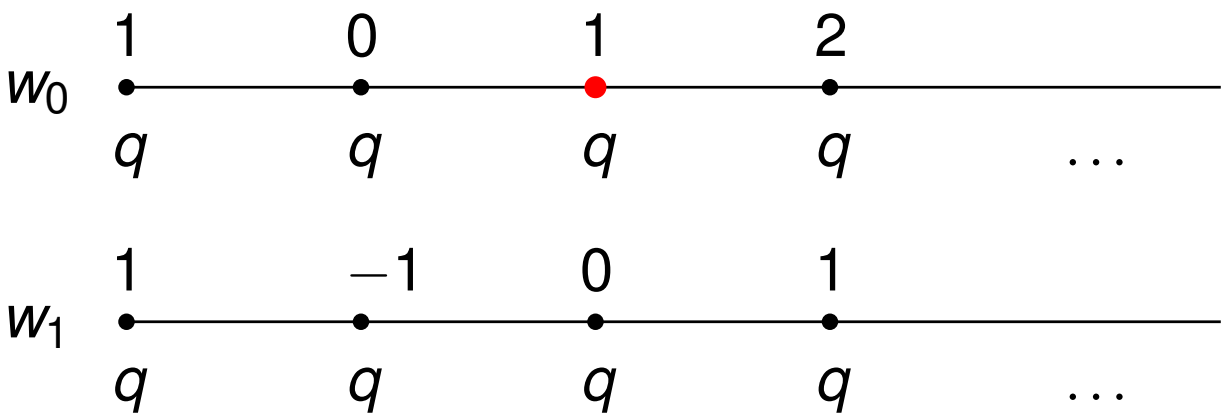
$x.\mathsf{XX}(x = 0) \notin \mathrm{MTL}_1^{\{0,1\}}$



# EF game for MTL

Played by: Spoiler • and Duplicator • on $w_0$ and $w_1$.

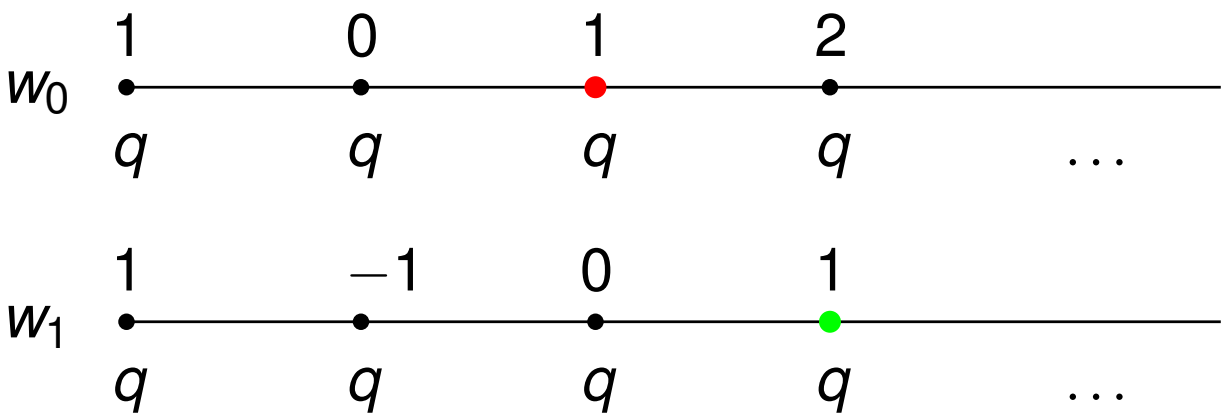
$x.\mathrm{XX}(x = 0) \notin \mathrm{MTL}_1^{\{0,1\}}$



# EF game for MTL

Played by: Spoiler • and Duplicator • on $w_0$ and $w_1$.

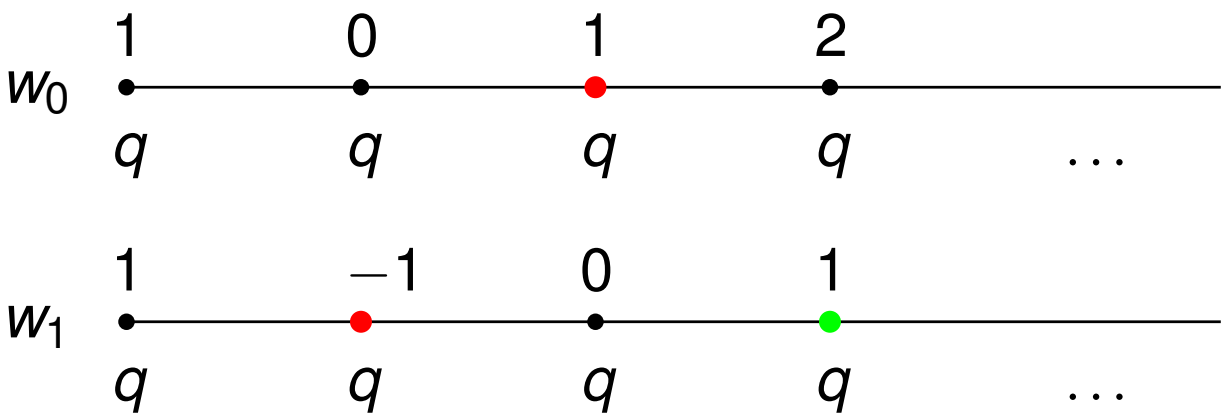
$x.\mathsf{XX}(x = 0) \notin \mathrm{MTL}_1^{\{0,1\}}$



# EF game for MTL

Played by: Spoiler • and Duplicator • on $w_0$ and $w_1$.

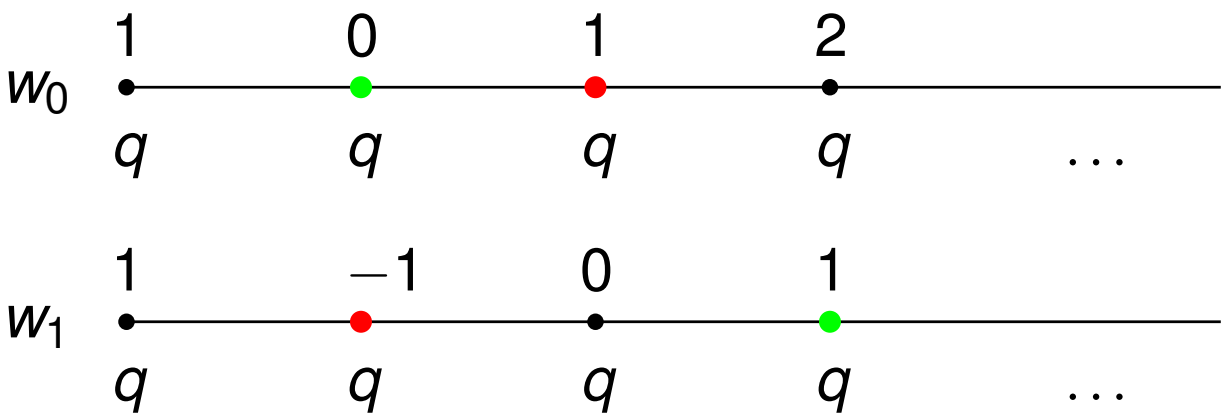
$x.\mathrm{XX}(x = 0) \notin \mathrm{MTL}_1^{\{0,1\}}$



# EF game for MTL

Played by: Spoiler • and Duplicator • on $w_0$ and $w_1$.

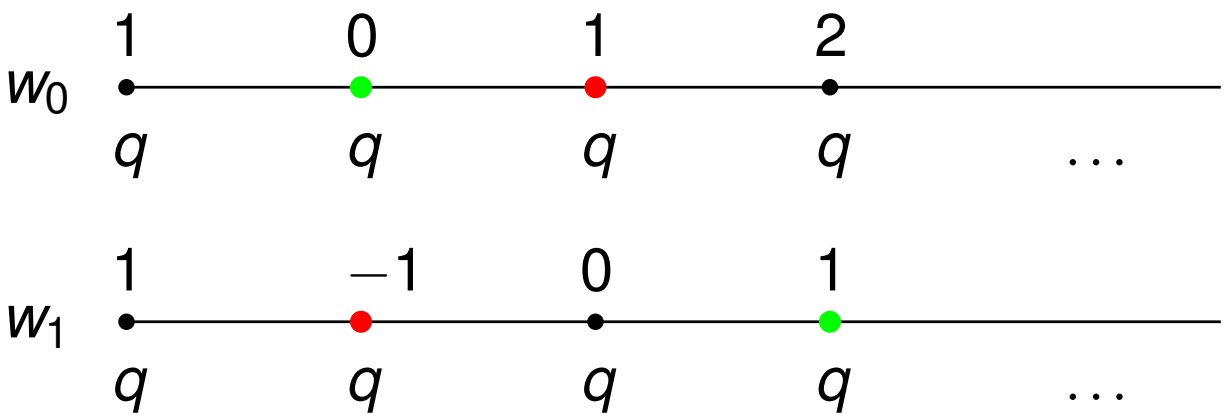
$x.\mathrm{XX}(x = 0) \notin \mathrm{MTL}_1^{\{0,1\}}$



# EF game for MTL

Played by: Spoiler • and Duplicator • on $w_0$ and $w_1$.

$x.\mathrm{XX}(x = 0) \notin \mathrm{MTL}_1^{\{0,1\}}$




The following are equivalent:

- Duplicator wins the first round on $w_0$ and $w_1$.
- $w_0$ and $w_1$ satisfy the same formulas in $\mathrm{MTL}_1^{\{0,1\}}$.

# Further Applications of the EF Game

- Theorem: The following MTL membership decision problem is undecidable:

  GIVEN: $\varphi \in \text{TPTL}$

  QUESTION: Is $\varphi$ equivalent to an MTL formula?

- Theorem: $\text{MTL} \subsetneq \text{TPTL}^1 \subsetneq \text{TPTL}^2$.

  $\text{TPTL}^k$: at most $k$ many different register variables are allowed in a formula.

# What is the satisfiability?

Satisfiability Problem (SAT)

INPUT: a formula $\varphi$

QUESTION: Is there a data word $w$ s.t. $w \models \varphi$?

# What is the satisfiability?

Satisfiability Problem (SAT)

INPUT: a formula $\varphi$

QUESTION: Is there a data word $w$ s.t. $w \models \varphi$?

- Finitary SAT: Is there a finite $w$...?
- Infinitary SAT: Is there an infinite $w$...?

# What is the satisfiability?

Satisfiability Problem (SAT)

INPUT: a formula $\varphi$

QUESTION: Is there a data word $w$ s.t. $w \models \varphi$?

- Finitary SAT: Is there a finite $w$...?
- Infinitary SAT: Is there an infinite $w$...?

Theorem[Alur & Henzinger, 94]:
Infinitary SAT of TPTL is undecidable ($\Sigma^1_1$-complete).

Theorem:

- Finitary SAT of MTL is undecidable ($\Sigma_1^0$-complete).
- Infinitary SAT of MTL is $\Sigma_1^1$-complete.

Theorem:

- Finitary SAT of MTL is undecidable ($\Sigma^0_1$-complete).
- Infinitary SAT of MTL is $\Sigma^1_1$-complete.

Also holds for the fragments of MTL and TPTL[1] where

- no propositions are allowed,
- only $X, F$ are allowed.

Theorem:

- Finitary SAT of MTL is undecidable ($\Sigma_1^0$-complete).
- Infinitary SAT of MTL is $\Sigma_1^1$-complete.

Also holds for the fragments of MTL and TPTL[1] where

- no propositions are allowed,
- only $X, F$ are allowed.

Proof idea:

- Finitary SAT: reduction from the halting problem of nondeterministic 2-counter machines.
- Infinitary SAT: reduction from the recurring problem of nondeterministic 2-counter machines.

Theorem: Infinitary SAT and finitary SAT coincide for positive TPTL and MTL, the SAT problem is $\Sigma_1^0$-complete.

"Positive": negation $\neg$ only applied to propositions.

$$p \,\mathsf{U}\, \neg q \text{ (positive)} \qquad \neg(p \,\mathsf{U}\, q) \text{ (not positive)}$$

Theorem: Infinitary SAT and finitary SAT coincide for positive TPTL and MTL, the SAT problem is $\Sigma_1^0$-complete.

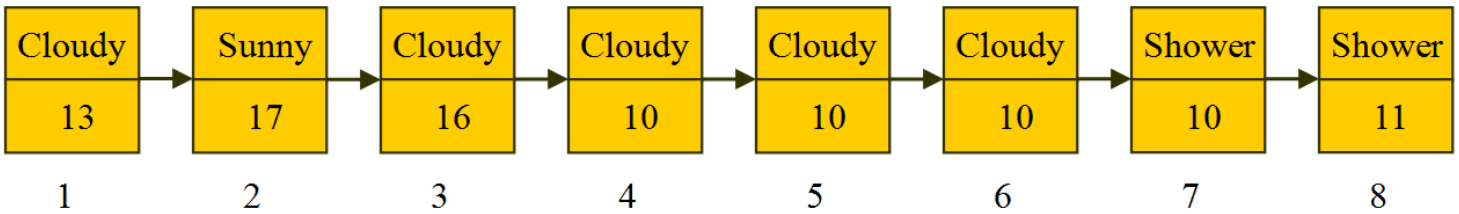
"Positive": negation $\neg$ only applied to propositions.

$$p \mathsf{U} \neg q \text{ (positive)} \qquad \neg(p \mathsf{U} q) \text{ (not positive)}$$

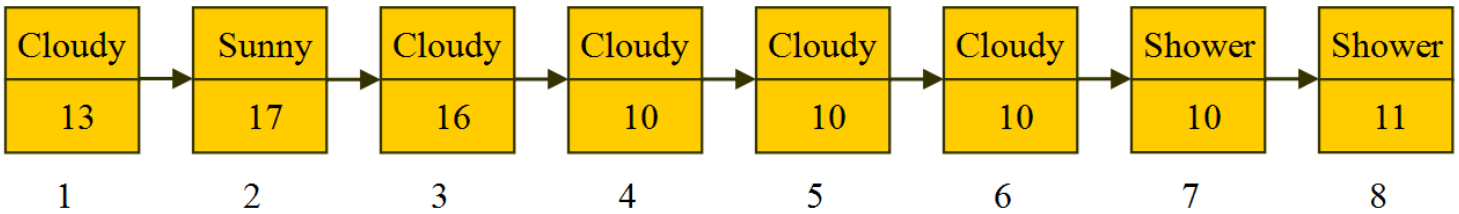
Theorem: SAT of positive $\mathrm{TPTL}_{\{\mathsf{F},\mathsf{X}\}}$ is NP-complete.

$\mathrm{TPTL}_{\{\mathsf{F},\mathsf{X}\}}$: the fragment of TPTL where only $\mathsf{F}, \mathsf{X}$ are allowed.

# What is path checking?

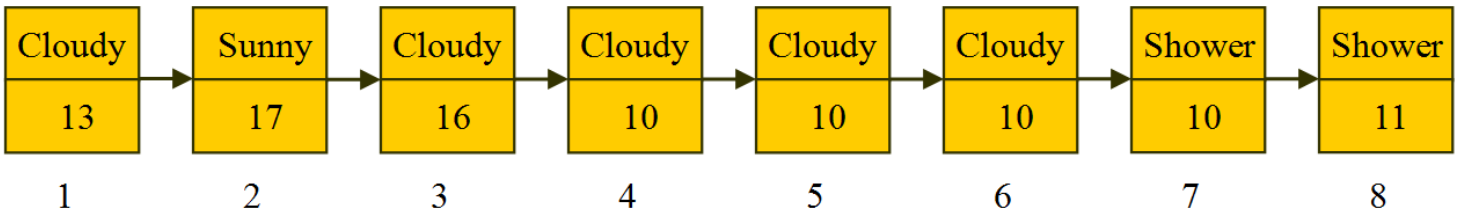


# What is path checking?



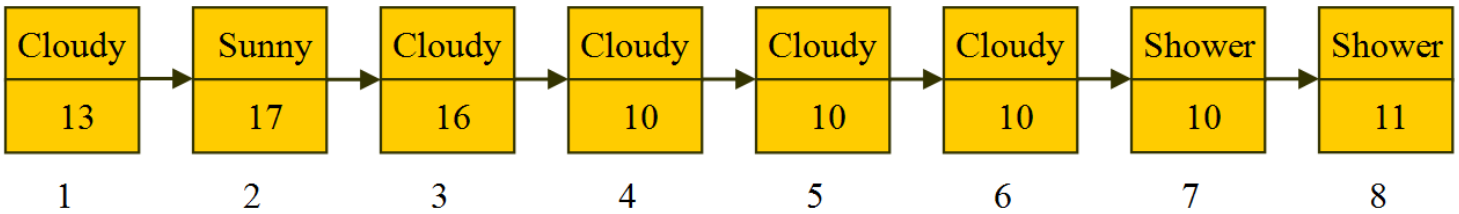

- Is Today cloudy, and there is some day in the future with showers where the temperature is 4 degrees higher than that of today?

# What is path checking?




- Is Today cloudy, and there is some day in the future with showers where the temperature is 4 degrees higher than that of today?
- Is $(\text{cloudy} \wedge F_{[0,4]}\text{shower})$ satisfied by the data word above?

# What is path checking?




- Is Today cloudy, and there is some day in the future with showers where the temperature is 4 degrees higher than that of today?
- Is (cloudy $\wedge$ $F_{[0,4]}$shower) satisfied by the data word above?

Path checking Problem

INPUT: data word $w$ and formula $\varphi$

QUESTION: whether $w \models \varphi$?

- Path checking for Freeze LTL over deterministic one-counter machine is PSPACE-complete.

[Demri, Lazic & Sangnier, 2008]

- Path checking for Freeze LTL over deterministic one-counter machine is PSPACE-complete.

  [Demri, Lazic & Sangnier, 2008]

- Finite path checking for LTL is in $\mathrm{AC}^1(\mathrm{logDCFL})$.

  [Kuhtz & Finkbeiner, 2009]

- Path checking for Freeze LTL over deterministic one-counter machine is PSPACE-complete.

  [Demri, Lazic & Sangnier, 2008]

- Finite path checking for LTL is in $\mathrm{AC}^1(\mathrm{logDCFL})$.

  [Kuhtz & Finkbeiner, 2009]

- Finite path checking for MTL over monotonic data words is in $\mathrm{AC}^1(\mathrm{logDCFL})$.

  [Bundala & Ouaknine, 2014]

- Path checking for Freeze LTL over deterministic one-counter machine is PSPACE-complete.

  [Demri, Lazic & Sangnier, 2008]

- Finite path checking for LTL is in $\mathrm{AC}^1(\mathrm{logDCFL})$.

  [Kuhtz & Finkbeiner, 2009]

- Finite path checking for MTL over monotonic data words is in $\mathrm{AC}^1(\mathrm{logDCFL})$.

  [Bundala & Ouaknine, 2014]

- Path checking for MTL and TPTL over ***all*** data words?

**Theorem**:
Path checking for MTL and TPTL[1] are P-complete.

Theorem:
Path checking for MTL and TPTL$^1$ are P-complete.

Proof idea:

Upper bound: Path checking for TPTL$^1$ is in P.

Theorem:
Path checking for MTL and TPTL$^1$ are P-complete.

Proof idea:

Upper bound: Path checking for TPTL$^1$ is in P.
Reduce this problem to the path checking for LTL.

Theorem:
Path checking for MTL and TPTL$^1$ are P-complete.

Proof idea:

Upper bound: Path checking for TPTL$^1$ is in P.
Reduce this problem to the path checking for LTL.

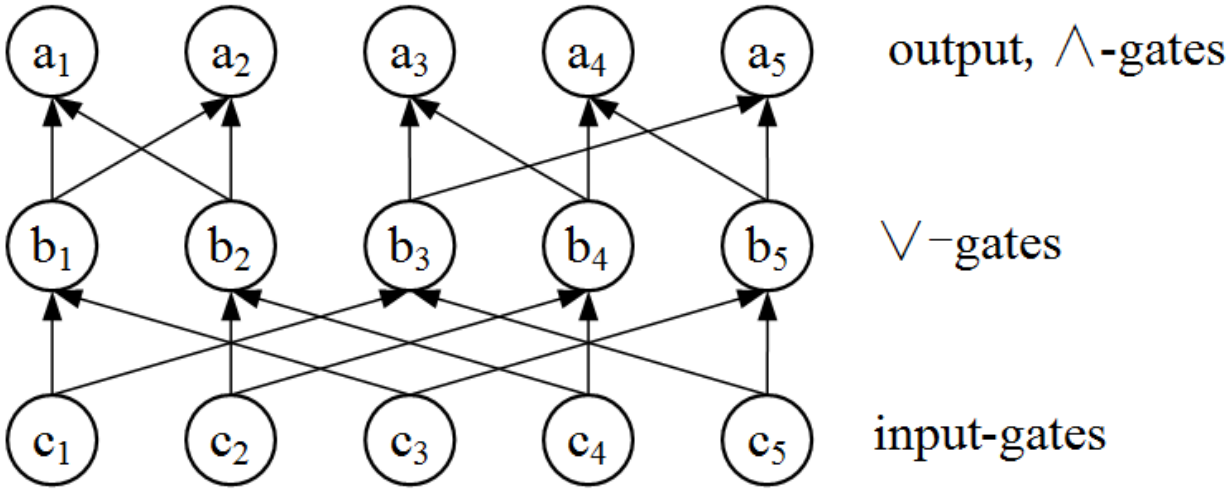
Lower bound: Path checking for MTL is P-hard.

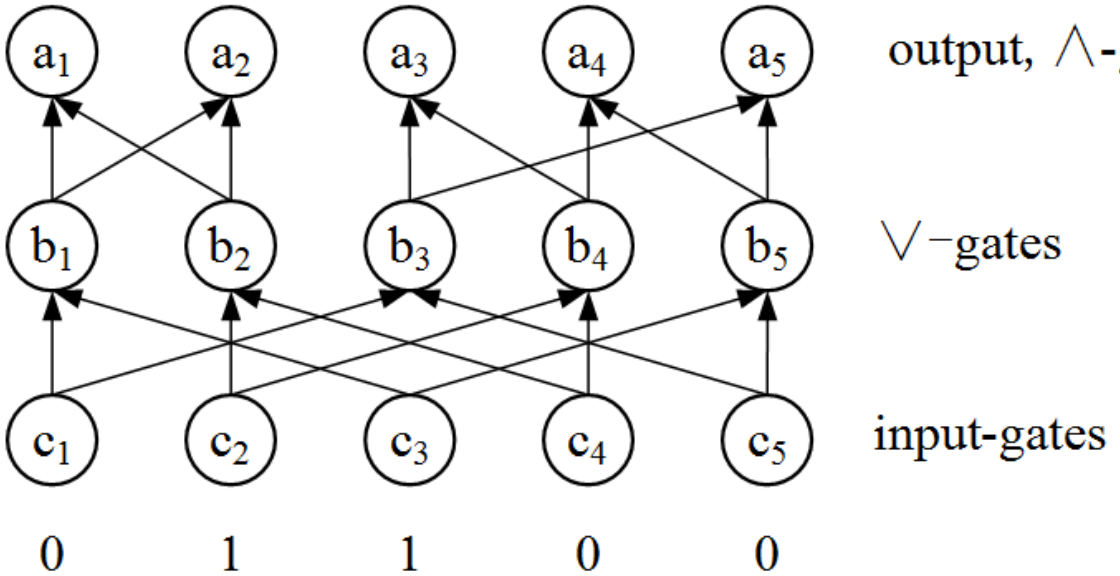
Theorem:
Path checking for MTL and TPTL$^1$ are P-complete.

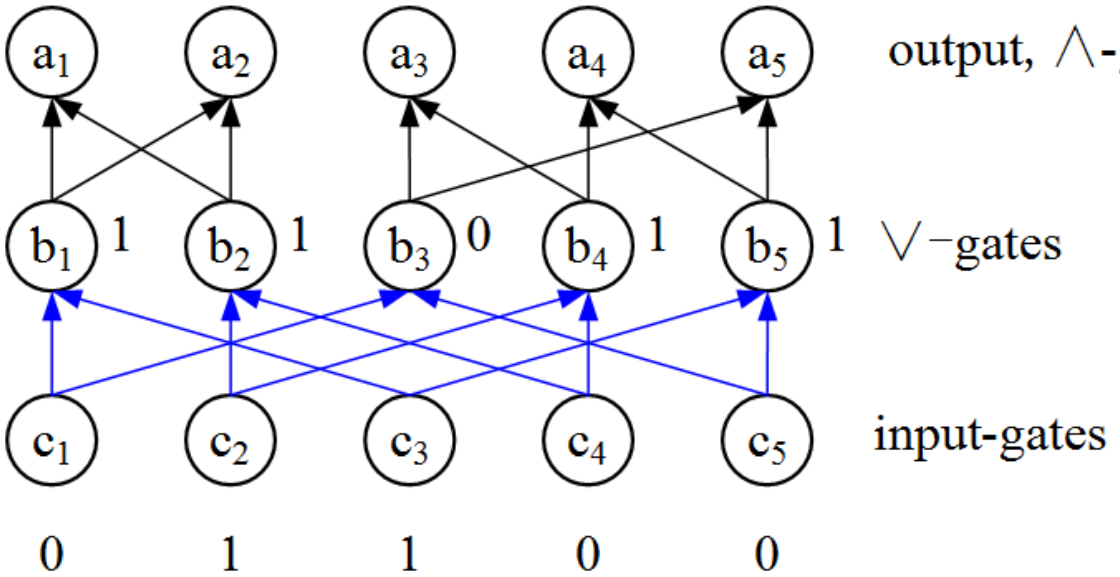
Proof idea:

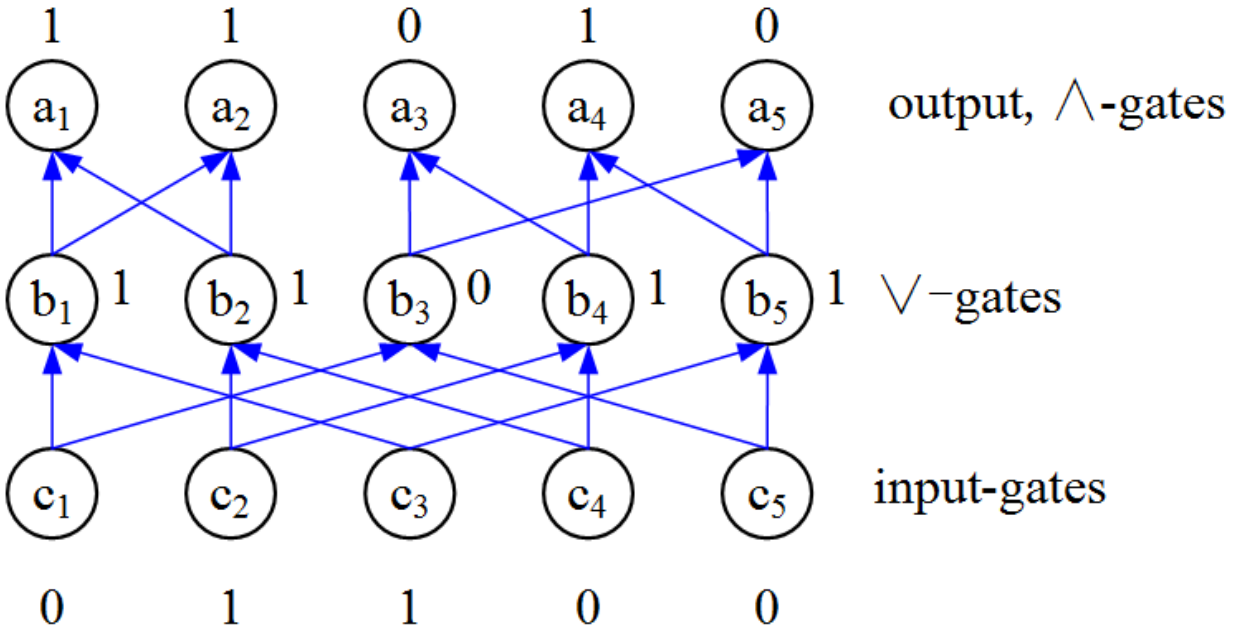
Upper bound: Path checking for TPTL$^1$ is in P.
Reduce this problem to the path checking for LTL.

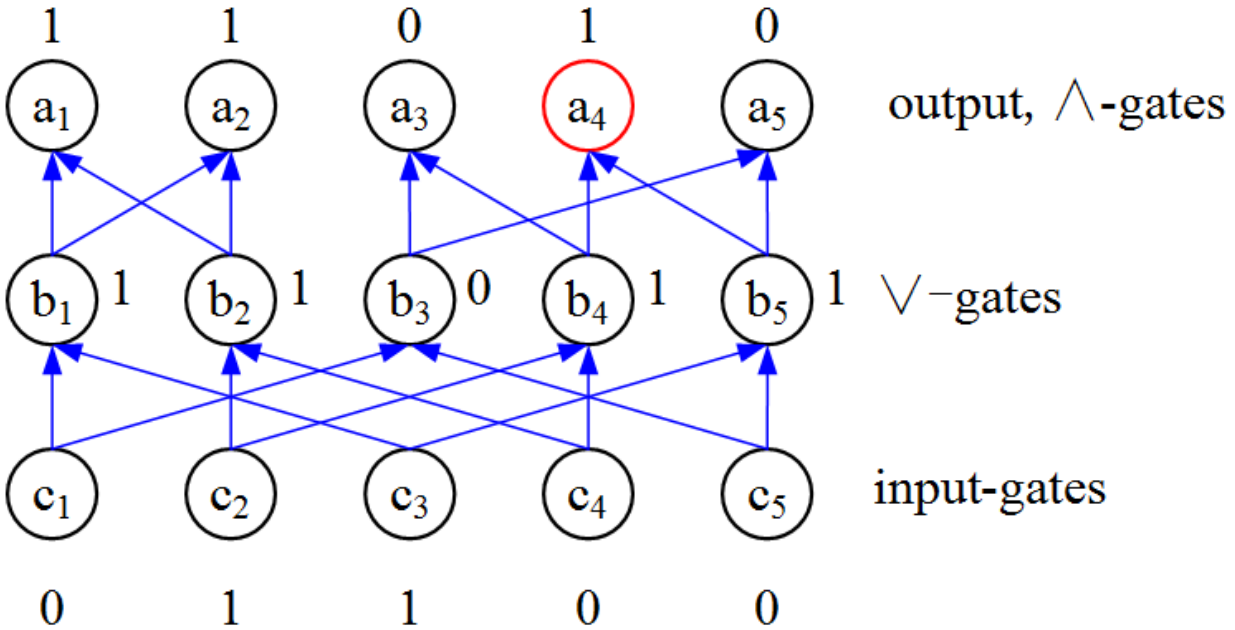
Lower bound: Path checking for MTL is P-hard.
Use a reduction from the circuit value problem for SAM2-circuits, which is P-complete.

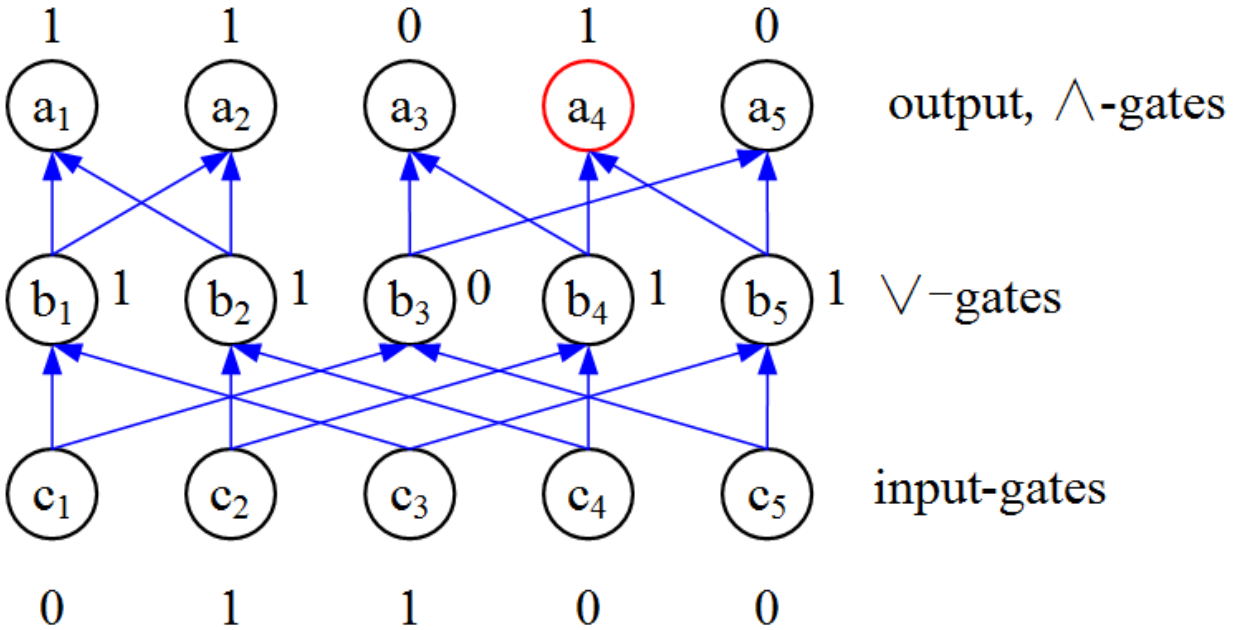

$a_1$ $a_2$ $a_3$ $a_4$ $a_5$
$b_1$ $b_2$ $b_3$ $b_4$ $b_5$
$c_1$ $c_2$ $c_3$ $c_4$ $c_5$
output, $\bigwedge$-gates
$\bigvee$−gates
input-gates


a1
a2
a3
a4
a5
b1
b2
b3
b4
b5
c1
c2
c3
c4
c5
0
1
1
0
0
output, ∧-gates
∨−gates
input-gates

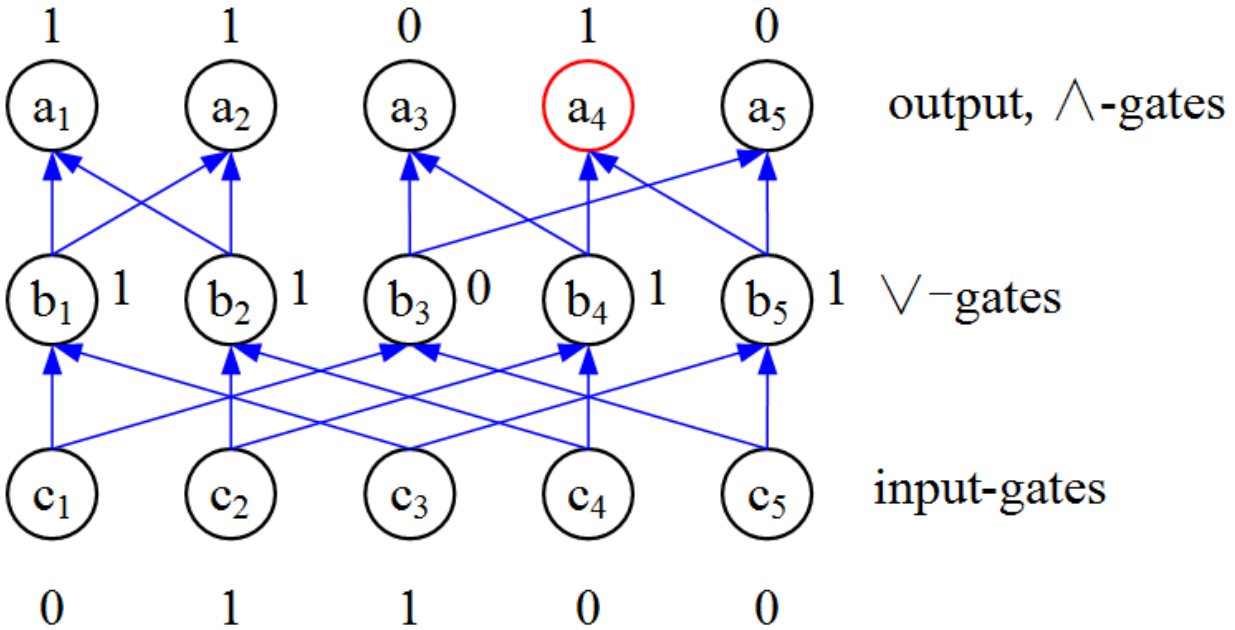

a1
a2
a3
a4
a5
output, ∧-gates
b1
1
b2
1
b3
0
b4
1
b5
1
∨−gates
c1
c2
c3
c4
c5
input-gates
0
1
1
0
0


1 1 0 1 0
a1 a2 a3 a4 a5
output, ∧-gates
b1 1 b2 1 b3 0 b4 1 b5 1
∨−gates
c1 c2 c3 c4 c5
input-gates
0 1 1 0 0

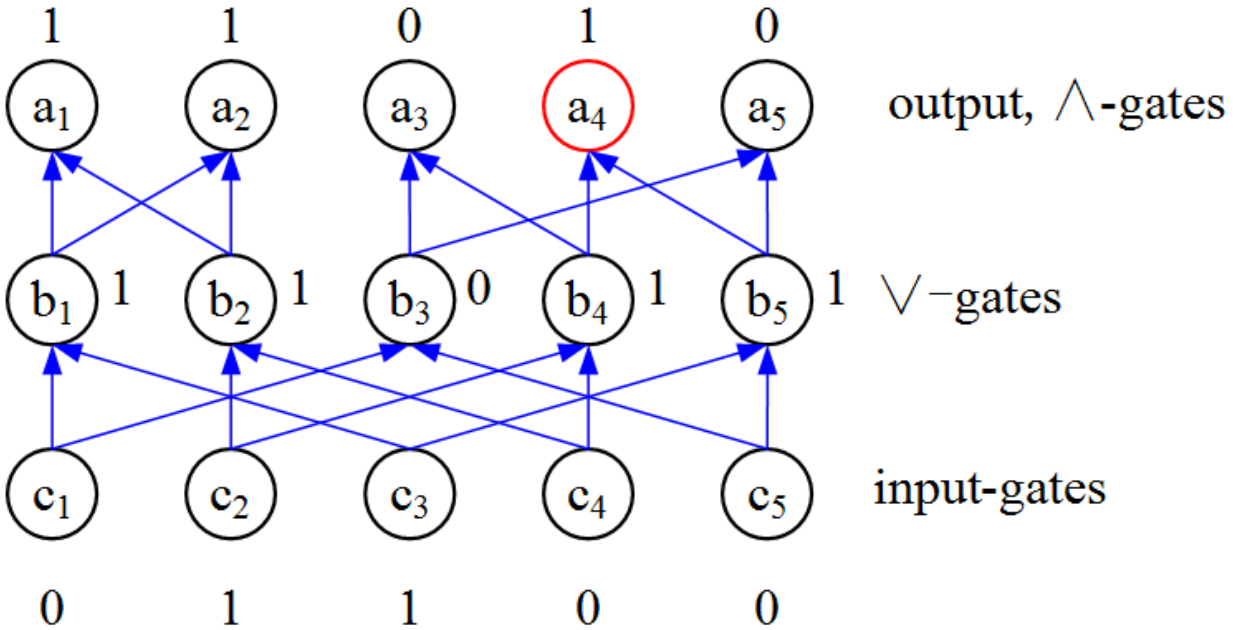

1 1 0 1 0
a1 a2 a3 a4 a5
output, ∧-gates
b1 1 b2 1 b3 0 b4 1 b5 1
∨−gates
c1 c2 c3 c4 c5
input-gates
0 1 1 0 0




- $w \models \varphi$ if and only if the output of $a_4$ is 1.




- $w \models \varphi$ if and only if the output of $a_4$ is 1.
- It is essential that $w$ is ***not*** monotonic.




- $w \models \varphi$ if and only if the output of $a_4$ is 1.
- It is essential that $w$ is ***not*** monotonic.
- If $\varphi$ is a TPTL$^1$ formula, then $w$ can be monotonic.

**Theorem**:
Path checking for TPTL is PSPACE-complete.

Theorem:
Path checking for TPTL is PSPACE-complete.

Proof idea:

Lower bound: Path checking for $\text{TPTL}^2$ is PSPACE-hard.
Use a reduction from quantified subset sum problem.

Theorem:
Path checking for TPTL is PSPACE-complete.

Proof idea:

Lower bound: Path checking for $\text{TPTL}^2$ is PSPACE-hard.
Use a reduction from quantified subset sum problem.

Upper bound: Path checking for TPTL is in PSPACE.

We use an alternating polynomial time algorithm (APTIME) to decide this problem, and APTIME=PSPACE.

# Conclusions

We consider MTL and TPTL over all data words.

# Conclusions

We consider MTL and TPTL over all data words.

- $\mathrm{MTL} \subsetneq \mathrm{TPTL}^1 \subsetneq \mathrm{TPTL}^2$.

# Conclusions

We consider MTL and TPTL over all data words.

- MTL $\subsetneq$ TPTL$^1$ $\subsetneq$ TPTL$^2$.
  - EF-game for MTL and TPTL.

# Conclusions

We consider MTL and TPTL over all data words.

- MTL $\subsetneq$ TPTL$^1$ $\subsetneq$ TPTL$^2$.
  - EF-game for MTL and TPTL.
  - until rank, set of constraint numbers,. . .

# Conclusions

We consider MTL and TPTL over all data words.

- MTL $\subsetneq$ TPTL$^1$ $\subsetneq$ TPTL$^2$.
  - EF-game for MTL and TPTL.
  - until rank, set of constraint numbers,. . .
- Most fragment of MTL and TPTL are undecidable.

# Conclusions

We consider MTL and TPTL over all data words.

- MTL $\subsetneq$ TPTL$^1$ $\subsetneq$ TPTL$^2$.
  - EF-game for MTL and TPTL.
  - until rank, set of constraint numbers,. . .
- Most fragment of MTL and TPTL are undecidable.
  - Reduction from nondeterministic 2-counter machines.

# Conclusions

We consider MTL and TPTL over all data words.

- MTL $\subsetneq$ TPTL$^1$ $\subsetneq$ TPTL$^2$.
  - EF-game for MTL and TPTL.
  - until rank, set of constraint numbers,. . .
- Most fragment of MTL and TPTL are undecidable.
  - Reduction from nondeterministic 2-counter machines.
  - Positive unary TPTL and MTL is NP-complete.

# Conclusions

We consider MTL and TPTL over all data words.

- MTL $\subsetneq$ TPTL$^1$ $\subsetneq$ TPTL$^2$.
  - EF-game for MTL and TPTL.
  - until rank, set of constraint numbers,. . .
- Most fragment of MTL and TPTL are undecidable.
  - Reduction from nondeterministic 2-counter machines.
  - Positive unary TPTL and MTL is NP-complete.
- Path checking is either P-complete or PSPACE-complete.

# Conclusions

We consider MTL and TPTL over all data words.

- MTL $\subsetneq$ TPTL$^1$ $\subsetneq$ TPTL$^2$.
  - EF-game for MTL and TPTL.
  - until rank, set of constraint numbers,...
- Most fragment of MTL and TPTL are undecidable.
  - Reduction from nondeterministic 2-counter machines.
  - Positive unary TPTL and MTL is NP-complete.
- Path checking is either P-complete or PSPACE-complete.
  - MTL, TPTL$^k$, TPTL,...

# Conclusions

We consider MTL and TPTL over all data words.

- MTL $\subsetneq$ TPTL$^1$ $\subsetneq$ TPTL$^2$.
  - EF-game for MTL and TPTL.
  - until rank, set of constraint numbers,...
- Most fragment of MTL and TPTL are undecidable.
  - Reduction from nondeterministic 2-counter machines.
  - Positive unary TPTL and MTL is NP-complete.
- Path checking is either P-complete or PSPACE-complete.
  - MTL, TPTL$^k$, TPTL,...
  - finite, infinite, monotonic, non-monotonic, unary, binary,...

# Conclusions

We consider MTL and TPTL over all data words.

- MTL $\subsetneq$ TPTL$^1$ $\subsetneq$ TPTL$^2$.
  - EF-game for MTL and TPTL.
  - until rank, set of constraint numbers,...
- Most fragment of MTL and TPTL are undecidable.
  - Reduction from nondeterministic 2-counter machines.
  - Positive unary TPTL and MTL is NP-complete.
- Path checking is either P-complete or PSPACE-complete.
  - MTL, TPTL$^k$, TPTL,...
  - finite, infinite, monotonic, non-monotonic, unary, binary,...
  - Path checking for deterministic 1-counter machines.

# Future work

- $\mathrm{TPTL}^1 \subsetneq \mathrm{TPTL}^2 \subsetneq \mathrm{TPTL}^3 \subsetneq \mathrm{TPTL}^4 \cdots$ is strict?
- Some decidable fragment of MTL and TPTL (e.g., is $\mathrm{MTL}_{\{F\}}$ decidable)?
- Over multi-data word, continuous semantics, ...