Systemy hybrydowe

Joanna Iwaniuk

2 listopada 2010

Plan prezentacji

1. Co to jest system hybrydowy?
2. Liniowe systemy hybrydowe
3. Analiza systemu hybrydowego
   - Wyznaczanie stanów osiągalnych
   - Model checking
4. Podsumowanie

System hybrydowy

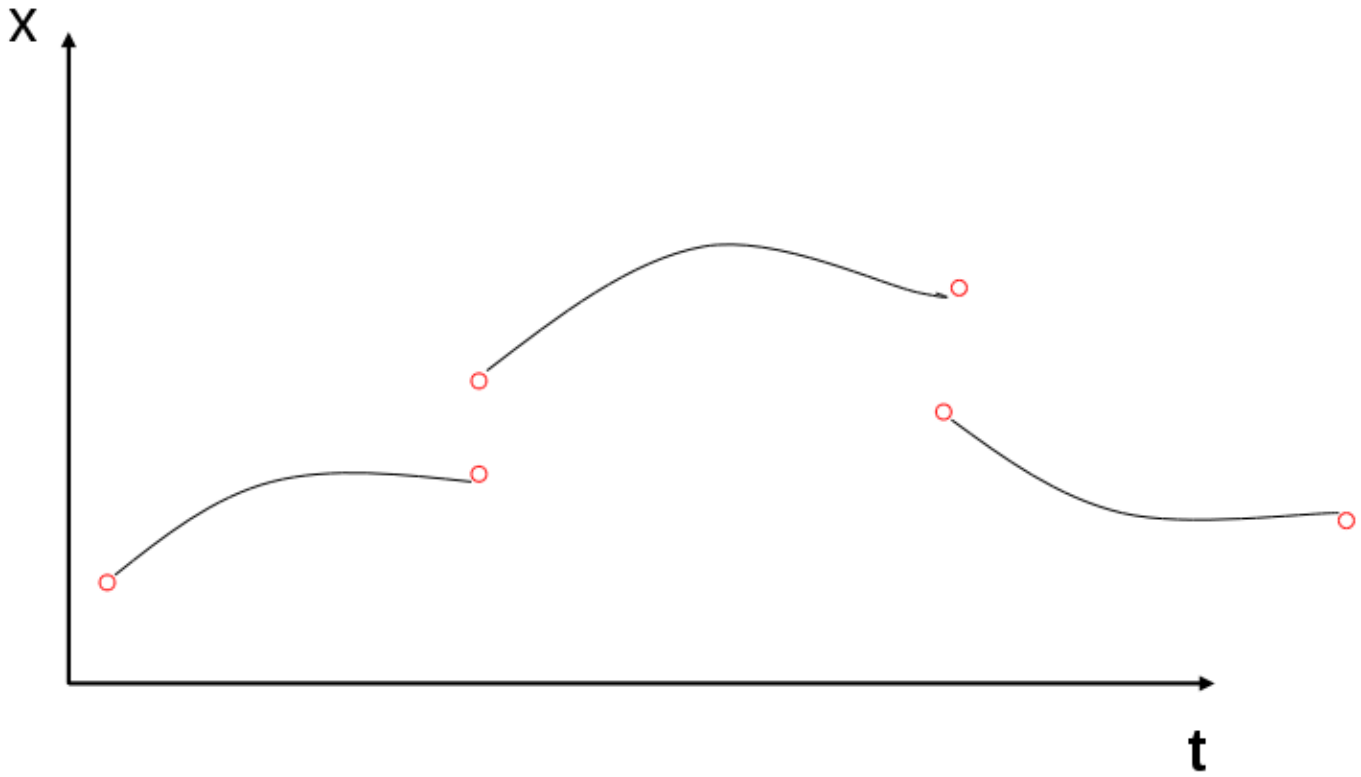
Komponent dyskretny

Komponent ciągły

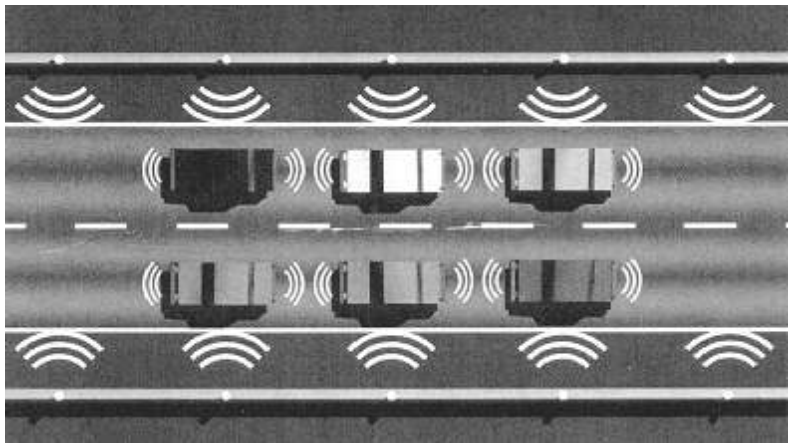
System hybrydowy

Komponent dyskretny

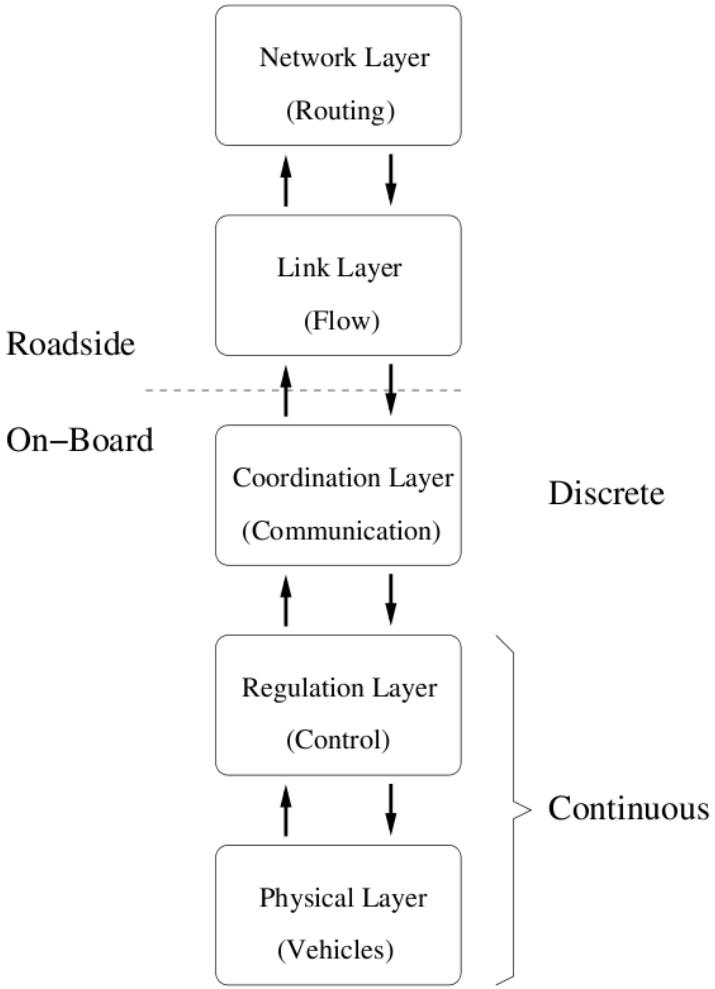
Komponent ciągły



Przykład – Automated Highway System



Przykład – Automated Highway System



Przykład – STARMAC

`http://hybrid.eecs.berkeley.edu/`

„The purpose of this work was to demonstrate that some of the theories from hybrid systems, and in particular, the analytical tools of reachability, could be applied to a complex real-world system."

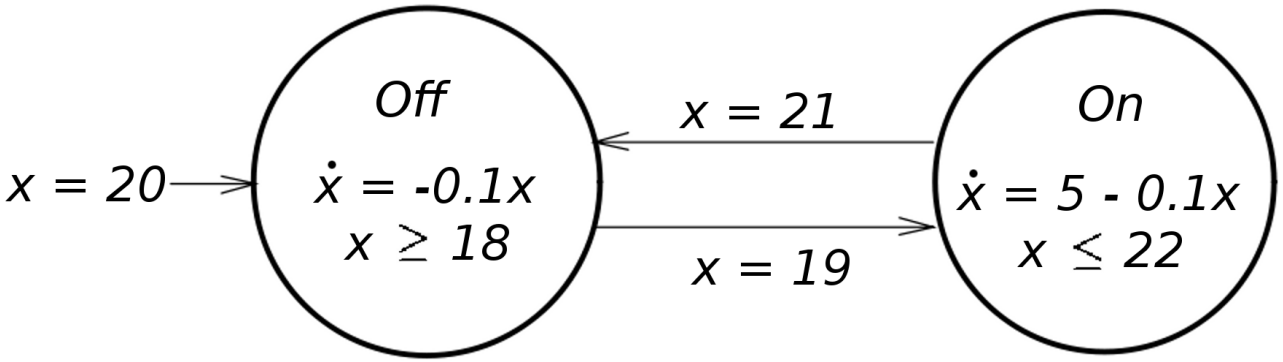
Po co analizować systemy hybrydowe?

- różnorodność zastosowań
- zagadnienia bezpieczeństwa

Automaty hybrydowe

$$\approx \text{zwykłe automaty}$$
$$+$$
$$\text{zbiór zmiennych rzeczywistych}$$
$$+$$
$$\text{reguły ich zmienności}$$

Przykład automatu hybrydowego



Rysunek: Automat dla termostatu. x – temperatura

Formalny model automatu hybrydowego

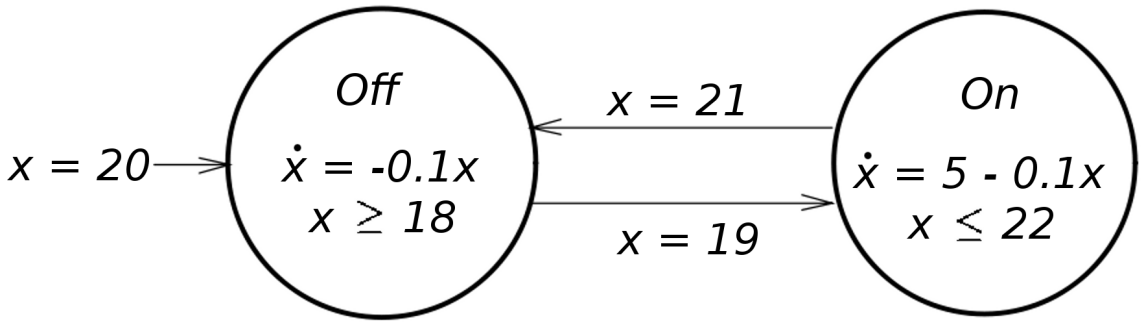
$X = \{x_1, x_2, ..., x_n\}$	Loc	$(l, v) \in \Sigma$
$transition\ successor$	Var	$v(x) \in \mathbb{R}$
$\tau \in Lab$	$control\ modes$	$init(v)$
Edg	$e = (l, a, \mu, l')$	$\mu \subseteq V^2$
$Con \subseteq Var$	$flow(x)$	$jump\ condition$
(l, τ, Id_{Con}, l)	$(v, v') \in Id_{Con}$	Act
$control\ switches$	$inv(v)$	$(f + t) \in Act(l)$
$stutter\ transition$	$f^x(t) = f(t)(x)$	$Inv(l) \subseteq V$
$\varphi_l[v]$	$\{\dot{x}_1, \dot{x}_2, ..., \dot{x}_n\}$	$jump(e)$
(V, E)	$event: E \to \Sigma$	$synchronization\ labels$
$X' = \{x'_1, x'_2, ..., x'_n\}$	...	

Stany

- Loc – zbiór wierzchołków
- Var – zbiór zmiennych
- V – zbiór wartościowań zmiennych
- stan to para (l, v), $l \in Loc$, $v \in V$

Stany

- $Loc = \{On, Off\}$
- $Var = \{x\}$
- przestrzeń stanów: $\{On, Off\} \times \mathbb{R}$



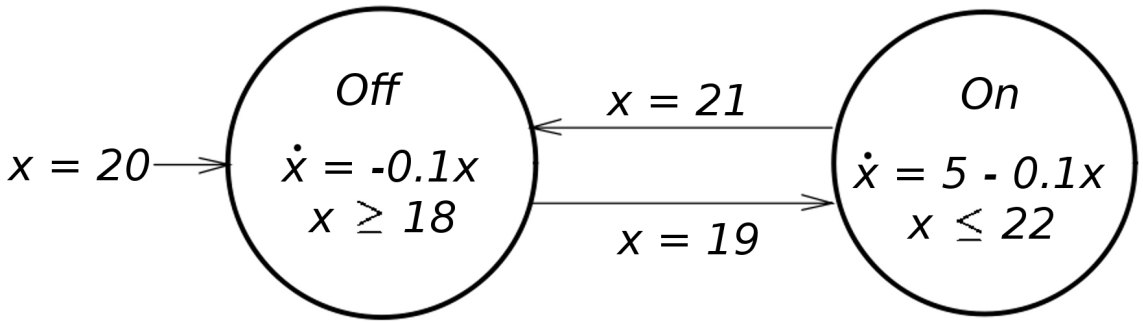
Przejścia

- Lab – zmienne synchronizacyjne
- Edg – zbiór krawędzi
- $\mu \subseteq V^2$ – relacja przejścia
- $e = (l, a, \mu, l')$, gdzie $l, l' \in Loc$, l – źródło, l' – cel, $a \in Lab$

Przejścia

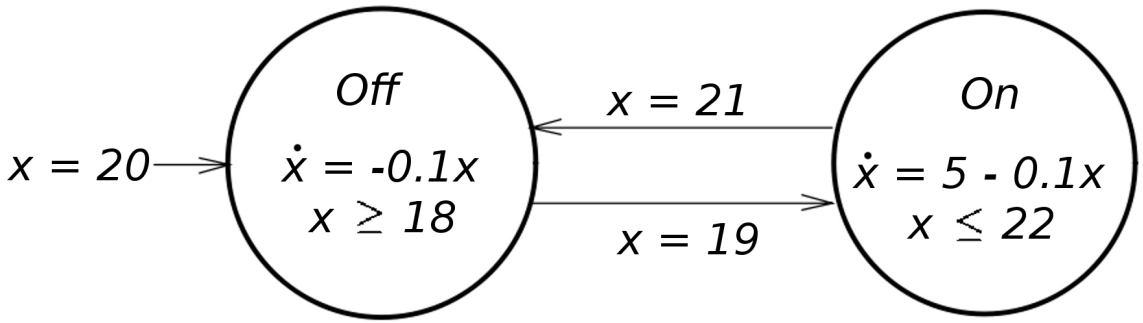
W automacie reprezentującym termostat są dwie krawędzie:

- z *Off* do *On* jeśli $x = 19$, bez zmiany wartości x,
- z *On* do *Off* jeśli $x = 21$, bez zmiany wartości x



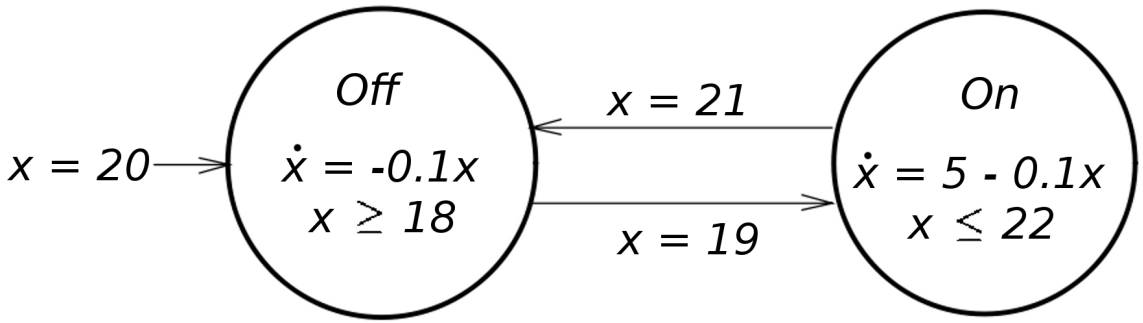
Niezmienniki

- funkcja $Inv : Loc \rightarrow P(V)$
- $Inv(Off) : x \geq 18$
- $Inv(On) : x \leq 22$



Czynności *(activities)*

- funkcja $Act : \mathbb{R}_{\geq 0} \to V$
- w jaki sposób zmienia się wartość zmiennych w czasie
- np. $\dot{x} = -0.1x$



Automat hybrydowy – podsumowanie

Składniki automatu hybrydowego:

- zbiór wierzchołków Loc
- zbiór zmiennych Var i zbiór ich wartościowań V
- zbiór zmiennych synchronizacyjnych Lab
- zbiór krawędzi Edg i relacja przejścia μ
- funkcja czynności $Act : \mathbb{R}_{\geq 0} \to V$
- funkcja niezmienników $Inv : Loc \to P(V)$

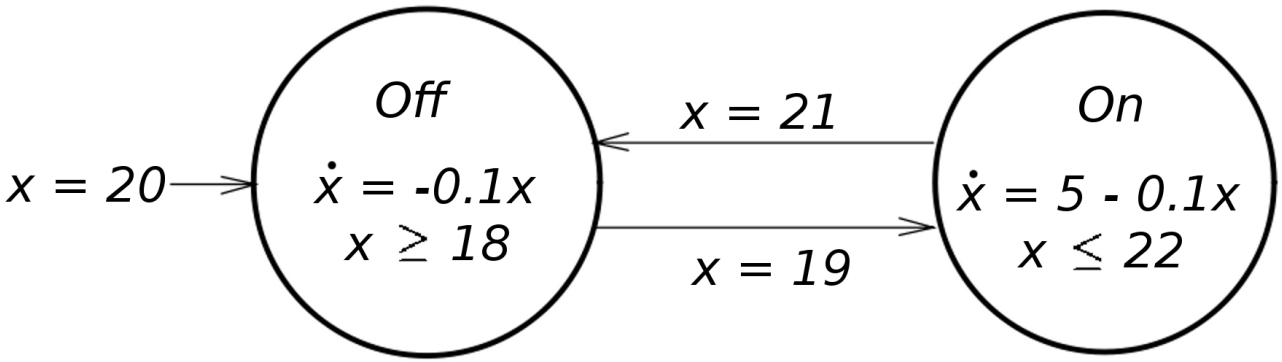
Liniowy automat hybrydowy

- automaty hybrydowe w ogólności są zbyt skomplikowane, żeby móc je analizować

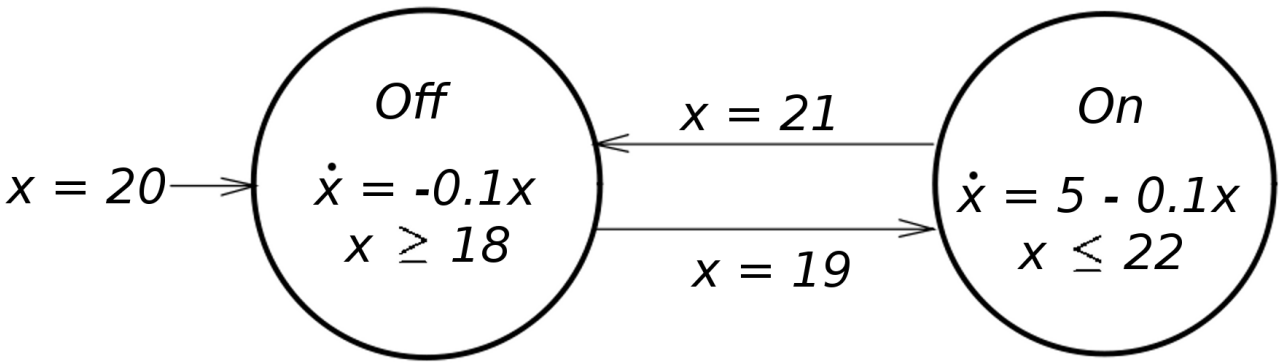
Liniowy automat hybrydowy – definicja

- czynności mają postać $\dot{x} = k_x$, gdzie $k_x \in \mathbb{Z}$
- niezmienniki są zdefiniowane przez formuły liniowe, np. $x \leq 2y$, $x < 10$
- każda relacja przejścia μ jest zdefiniowana przez formuły liniowe

Termostat – liniowy system hybrydowy?



Termostat – liniowy system hybrydowy?

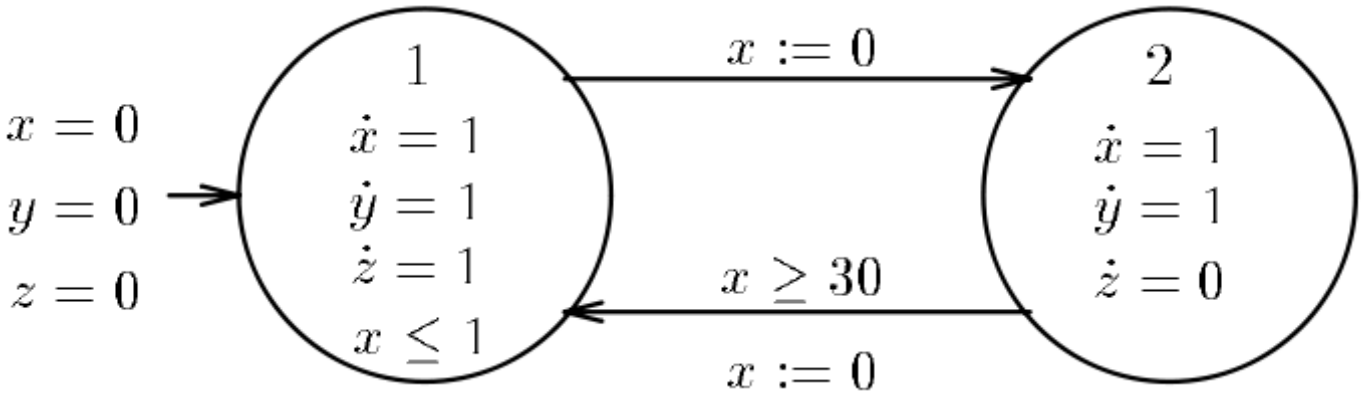


Nie, bo $\dot{x} \neq k_x$

Palnik

- z palnika może ulatniać się gaz
- $\leq$ 1s ulatniania
- $\geq$ 30s przerwy

Automat dla palnika



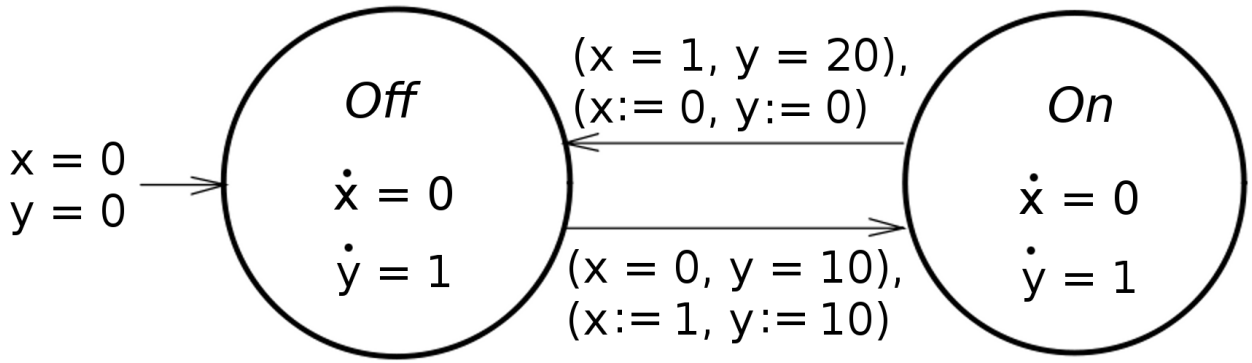
- wierzchołek 1 – ulatnianie gazu, 2 – brak ulatniania
- y – mierzy czas od początku działania
- z – mierzy kumulatywny czas ulatniania
- x – czas spędzony w aktualnym wierzchołku

Przypadki szczególne liniowych automatów hybrydowych

automaty czasowe i ich modyfikacje

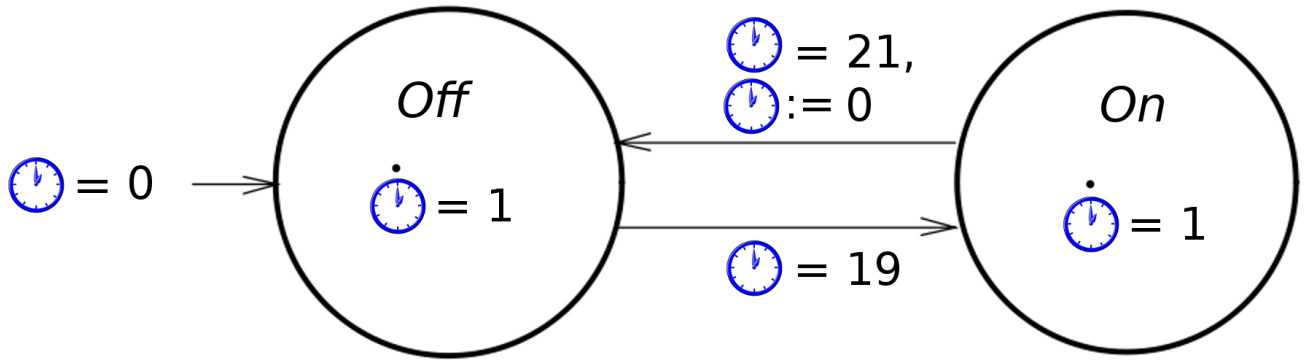
Zmienne zero-jedynkowe *(propositions)*

- zmieniają się tylko w wyniku przejść dyskretnych
- przyjmują wartości 0 lub 1



Zegary

- w każdym wierzchołku $\dot{x} = 1$
- po zmianie wierzchołka x nie zmienia wartości lub nowa wartość wynosi 0

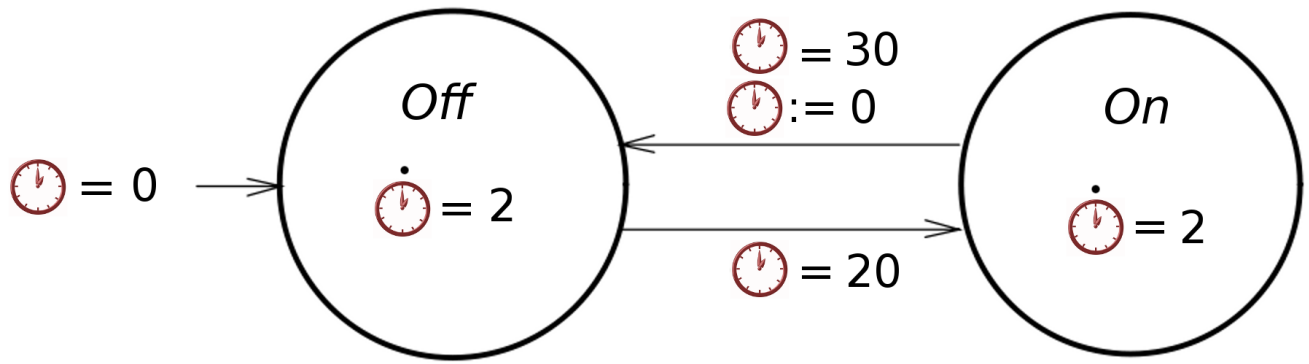


Automaty czasowe

- każda zmienna jest zero-jedynkowa lub jest zegarem
- wyrażenia liniowe w niezmiennikach i relacjach przejścia są postaci $x \# c$ lub $x - y \# c$, gdzie $c \in \mathbb{Z}_+$, $\# \in \{<, \leq, =, \geq, >\}$

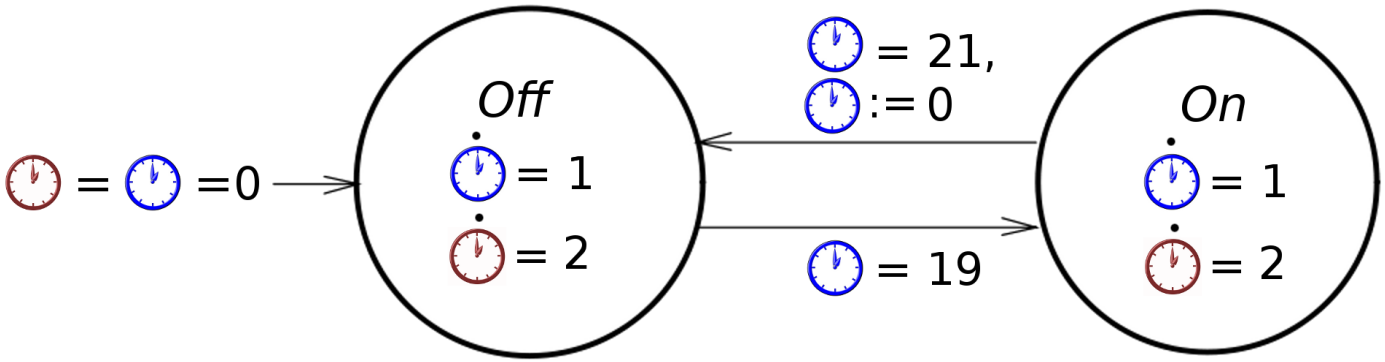
Zaburzone zegarki *(skewed clocks)*

- w każdym wierzchołku $\dot{x} = k$, gdzie $k \in \mathbb{Z}_+$
- po zmianie wierzchołka x nie zmienia wartości lub nowa wartość wynosi 0
- zaburzony zegarek $\equiv$ zegar, który może szybciej chodzić



Wielowspółczynnikowe automaty czasowe *(multirate timed systems)*

- każda zmienna jest zero-jedynkowa lub jest zaburzonym zegarkiem
- automat n-współczynnikowy – jest n różnych „prędkości tykania”



Proste automaty hybrydowe

- niezmienniki i przejścia zdefiniowane przez formuły postaci $x \leq k$, $k \leq x$, gdzie $x \in Var$, $k \in \mathbb{Z}$

Liniowe automaty hybrydowe – podsumowanie

- $\dot{x} = k_x$
- niezmienniki i relacja przejścia – formuły liniowe
- w szczególności automaty czasowe i wielowspółczynnikowe automaty czasowe

Problem osiągalności

- $\sigma' \in \Sigma$ jest osiągalne z $\sigma \in \Sigma$ (ozn. $\sigma \mapsto^* \sigma'$) w automacie hybrydowym H jeśli istnieje przebieg H zaczynający się w σ i kończący w σ'
- Chcemy obliczyć stany osiągalne z danej konfiguracji początkowej I, tzn. zbiór $(I \mapsto^*)$ t. że: $\sigma \in (I \mapsto^*)$ wtw. $\exists_{\sigma' \in I} \sigma' \mapsto^* \sigma$

Problem osiągalności

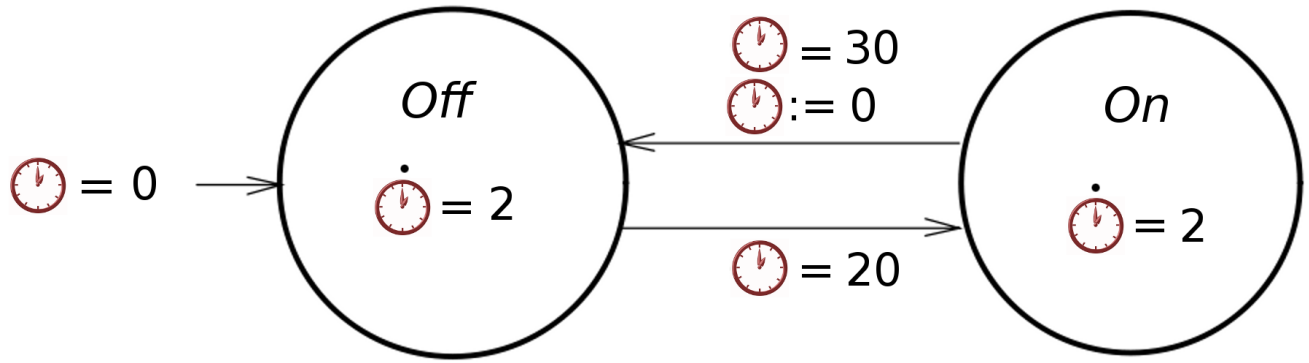
- jest nierozstrzygalny dla automatów hybrydowych, a nawet dla liniowych automatów hybrydowych
- w pewnych szczególnych przypadkach jest rozstrzygalny

Wielowspółczynnikowe proste automaty czasowe

- problem rozstrzygalny
- można sprowadzić do prostego automatu czasowego

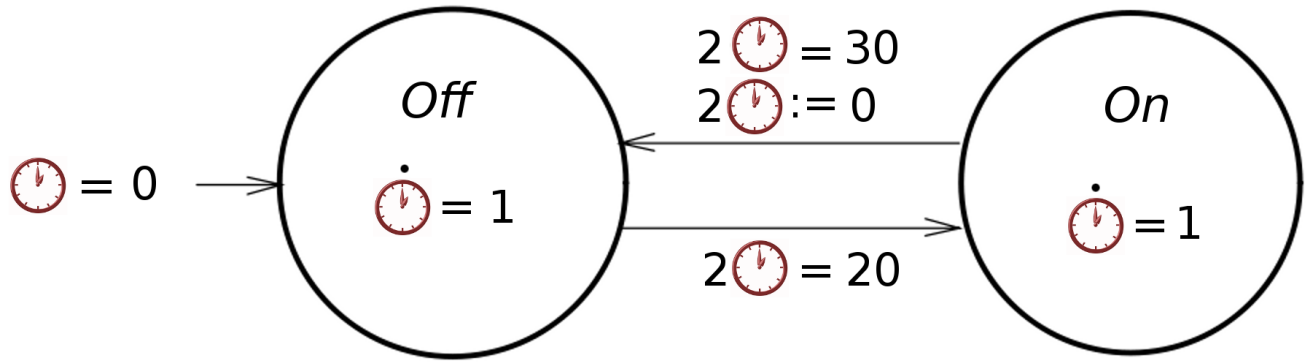
Wielowspółczynnikowe proste automaty czasowe

- problem rozstrzygalny
- można sprowadzić do prostego automatu czasowego



Wielowspółczynnikowe proste automaty czasowe

- problem rozstrzygalny
- można sprowadzić do prostego automatu czasowego



2-współczynnikowe automaty czasowe

- problem nierozstrzygalny
- równoważny z problemem stopu dla maszyny z 2 licznikami

Nierozstrzygalność – szkic dowodu

- wartość licznika n w i-tej konfiguracji maszyny = wartość zegara $1/2^n$ po upływie i jednostek czasu
- reset zegara w odpowiednim momencie odpowiada operacjom na wartości licznika
- wyznaczenie tych momentów wymaga porównywania wartości zmiennych

Szukanie stanów osiągalnych

- analiza w przód *(forward analysis)*
- analiza w tył *(backward analysis)*
- obie zawodne

Analiza w przód

Ogólna idea:

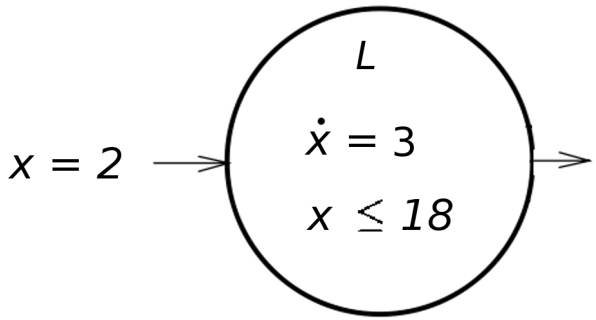
- mamy zbiór początkowy wartościowań zmiennych
- dodajemy kolejne wartościowania, które można osiągnąć z dotychczasowych w wyniku przejścia lub upływu czasu

Analiza w przód – definicje i oznaczenia

- $\langle P \rangle_I^{\nearrow}$ (*forward time closure*) – zbiór wartościowań zmiennych osiągalnych z $P \subseteq V$ pod wpływem upływu czasu, bez zmiany wierzchołka

Analiza w przód – definicje i oznaczenia

- $\langle P \rangle_l^{\nearrow}$ (*forward time closure*) – zbiór wartościowań zmiennych osiągalnych z $P \subseteq V$ pod wpływem upływu czasu, bez zmiany wierzchołka



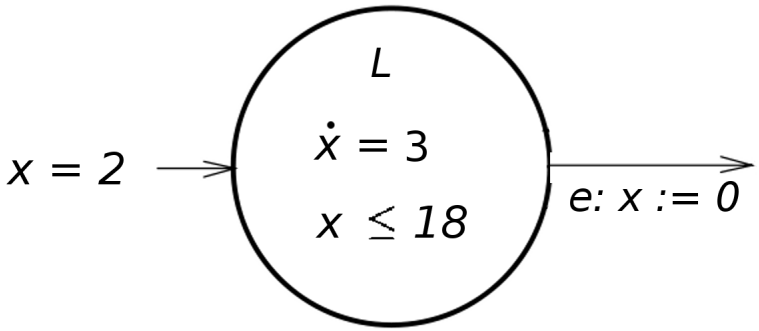
$$P : x = 2$$
$$\langle P \rangle_L^{\nearrow} = \{x \in [2, 18]\}$$

Analiza w przód – definicje i oznaczenia

- $post_e[P]$ (*postcondition*) – wartościowania, które można osiągnąć z $P \subseteq V$ w wyniku przejścia e

Analiza w przód – definicje i oznaczenia

- $post_e[P]$ (*postcondition*) – wartościowania, które można osiągnąć z $P \subseteq V$ w wyniku przejścia e



$$P : x = 2$$
$$post_e[P] = \{x = 0\}$$

Analiza w przód – definicje i oznaczenia c.d.

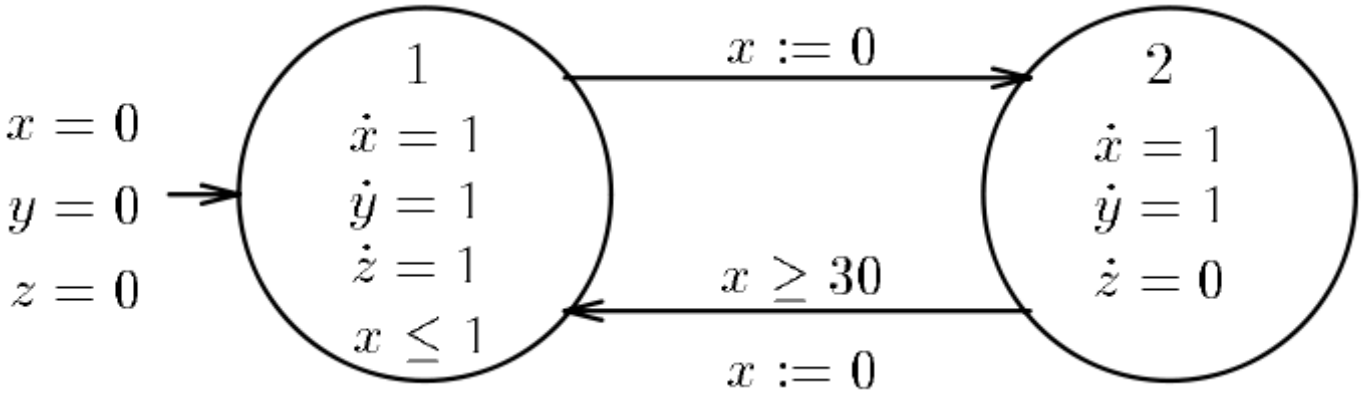
Rozszerzenie definicji na zbiory stanów:

- $\langle R \rangle^{\nearrow} = \bigcup_{l \in Loc} (l, \langle R_l \rangle_l^{\nearrow})$
- $post[R] = \bigcup_{e=(l,l') \in Edg} (l', post_e[R_l])$
- $R_l \subseteq V$, $R \subseteq Loc \times V$

Analiza w przód – metoda

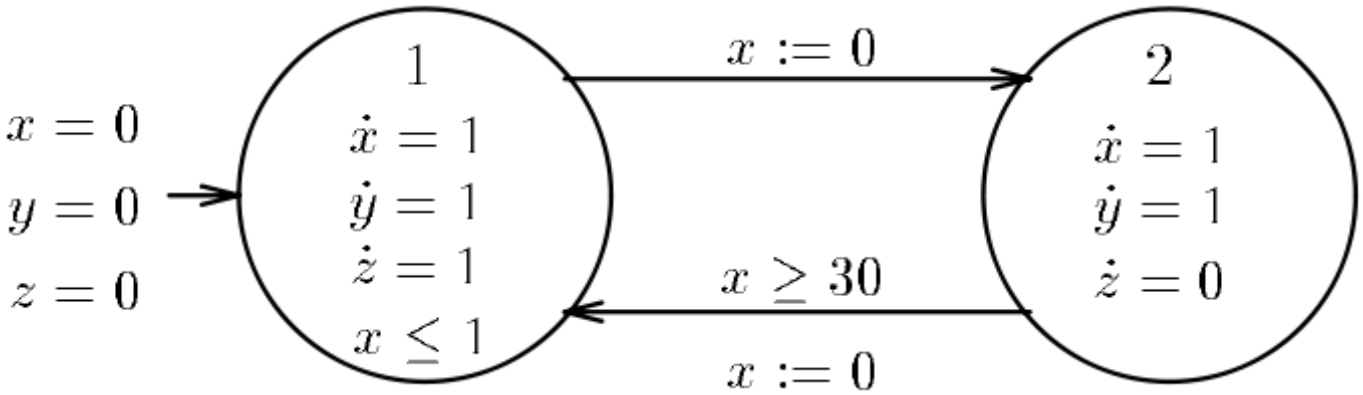
Twierdzenie: Osiągalny zbiór stanów ($I \mapsto^*$) jest najmniejszym punktem stałym równania $X = \langle I \cup post[X] \rangle^{\nearrow}$

Przykład – wyznaczanie stanów osiąglanych dla palnika



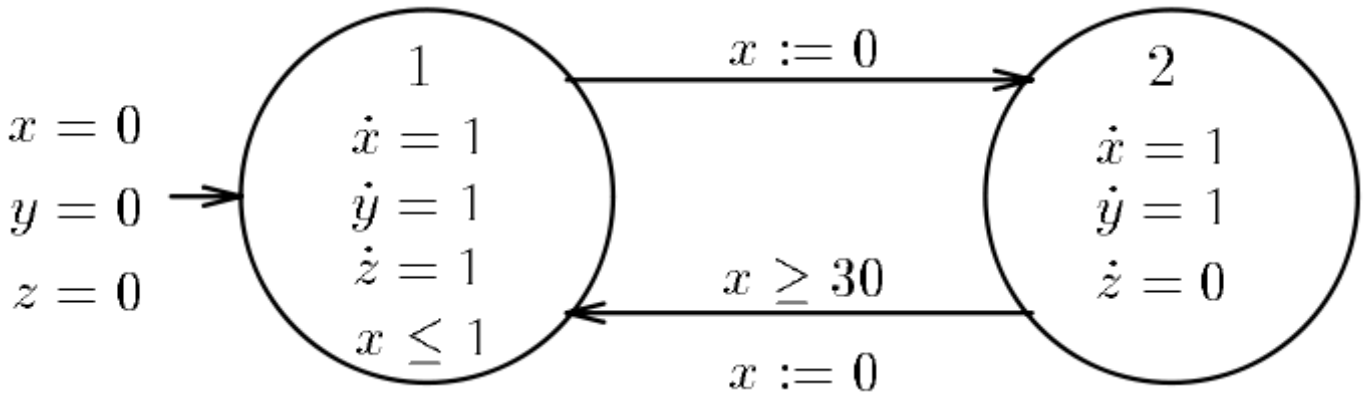
- I – zbiór stanów początkowych zdefiniowany przez formułę: $\psi_I = (pc = 1 \wedge x = y = z = 0)$, gdzie $pc \in Loc$ – zmienna pomocnicza

Wyznaczanie stanów osiągalnych dla palnika cd.



- stany osiągalne = najmniejszy punkt stały „$X = \langle I \cup post[X] \rangle$”
- $\psi_1 = \langle x = y = z = 0 \vee post_{(2,1)}[\psi_2] \rangle$
- $\psi_2 = \langle false \vee post_{(1,2)}[\psi_1] \rangle$

Wyznaczanie stanów osiągalnych dla palnika cd.



- $\psi_{2,0} = \mathit{false}$
- $\psi_{2,1} = \psi_{2,0} \vee \langle \mathit{post}_{(1,2)}[\psi_{1,0}] \rangle_2^{\nearrow} =$
  $\langle \mathit{post}_{(1,2)}[x \leq 1 \wedge x = y = z] \rangle_2^{\nearrow} =$
  $\langle x = 0 \wedge y \leq 1 \wedge y = z \rangle_2^{\nearrow} =$
  $(z \leq 1 \wedge y = z + x)$

Wyznaczanie stanów osiągalnych dla palnika cd.

Stosując indukcję można pokazać, że:

- $\psi_1 = (x \leq 1 \wedge x = y = z) \vee (x \leq 1 \wedge x \leq z \wedge y + 30x \geq 31z)$
- $\psi_2 = (z \leq 1 \wedge y = x + z \wedge x \geq 0) \vee (y \geq x + 31z - 30)$

Wyznaczanie stanów osiągalnych dla palnika – wynik

- $\psi_1 = (x \leq 1 \wedge x = y = z) \vee (x \leq 1 \wedge x \leq z \wedge y + 30x \geq 31z)$
- $\psi_2 = (z \leq 1 \wedge y = x + z \wedge x \geq 0) \vee (y \geq x + 31z - 30)$
- $\psi = (pc = 1 \wedge \psi_1) \vee (pc = 2 \wedge \psi_2)$ jest niezmiennikiem systemu
- z tego wynika, że $y \geq 20z$ dla $y \geq 60$

Model checking liniowych systemów hybrydowych

- czy system hybrydowy spełnia daną formułę?
- nierozstrzygalne, oprócz pewnych przypadków (np. wielowspółczynnikowe automaty czasowe)

TCTL (Timed Computation Tree Logic)

$$\psi ::= \phi \mid \neg\psi \mid \psi_1 \vee \psi_2 \mid z.\psi \mid \psi_1 \exists_U \psi_2 \mid \psi_1 \forall_U \psi_2$$

- $\psi_1 \exists_U \psi_2$ spełnione w σ – istnieje przebieg z σ do σ' t. że σ' spełnia ψ_2 i w trakcie przebiegu cały czas $\psi_1 \vee \psi_2$
- $\psi_1 \forall_U \psi_2$ – każdy przebieg prowadzi do stanu, w którym ψ_2, reszta analogicznie
- można zbudować równoważne zdania z $\Diamond p$ (możliwe że p) i $\Box p$ (zawsze p)

Operator $\triangleright$

- $R \triangleright R' \subseteq \Sigma$ dla $R, R' \subseteq \Sigma$
- operator $\triangleright$ posłuży do iteracyjnego wyznaczania zbiorów spełniających $\exists_U$, $\forall_U$
- miła własność: jeśli R, R' są liniowe to $R \triangleright R'$ też jest liniowy

Operator $\triangleright$ – definicja

$$(l, v) \in (R \triangleright R') \text{ wtw.}$$

$$\exists_{(l',v')\in R', t\in\mathbb{R}_{\geq 0}}((l, v) \mapsto^t (l', v') \wedge \forall_{0\leq t'\leq t}(l, v + t') \in (R \cup R')),$$

gdzie R, R' – zbiory stanów

Operator $\triangleright$ – definicja

$$(l, v) \in (R \triangleright R') \text{ wtw.}$$

$$\exists_{(l',v')\in R', t\in\mathbb{R}_{\geq 0}}((l, v) \mapsto^t (l', v') \wedge \forall_{0\leq t'\leq t}(l, v + t') \in (R \cup R')),$$

gdzie R, R' – zbiory stanów

- w jednym kroku można dojść do R', tak że przez cały czas jesteśmy w $R \cup R'$

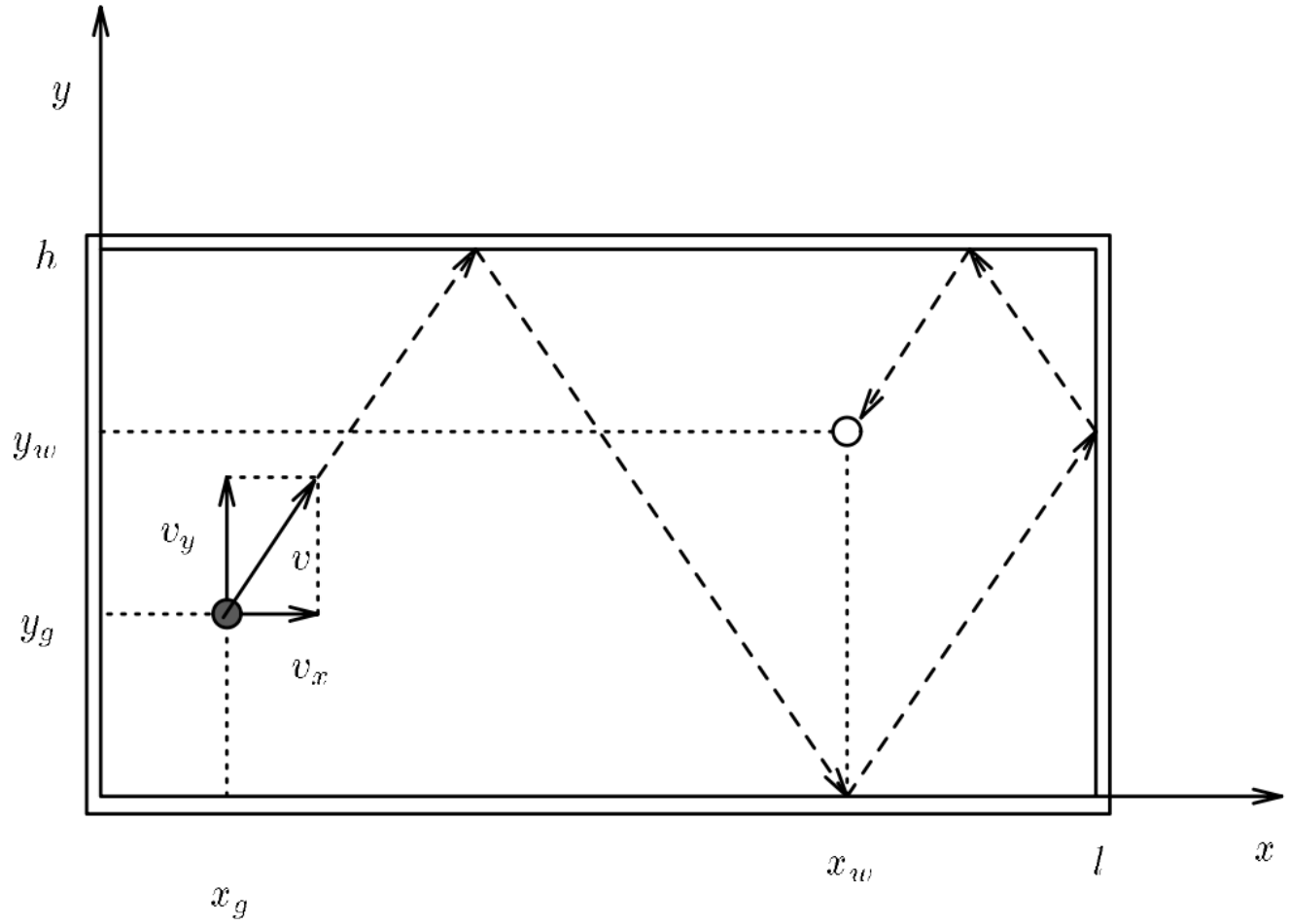
Obliczanie zbioru spełniającego $\phi \exists_U \phi'$

- niech R, R' – wyznaczanie przez formuły TCTL ϕ i ϕ'
- $\phi \exists_U \phi'$ wyznaczamy iteracyjnie jako $\bigcup_i R_i$
- $R_0 = R'$
- $R_{i+1} = R_i \cup (R \triangleright R_i)$

Czy wierzchołki startowe są zawarte w $\forall\Box\phi$?

- można obliczyć iteracyjnie jako $\bigcap_i R_i$
- $R_0 =$ zbiór stanów spełniających ϕ
- $R_{i+1} = R_i \cap \neg(true \triangleright \neg R_i)$

Przykładowy wynik model checkingu



Przykładowy wynik model checkingu

periodT $\neg(\neg(x = x_w \wedge y = y_w)\exists_{U_{>T}}(x = x_w \wedge y = y_w)$

touch $\exists\Diamond(x = x_w \wedge y = y_w)$

touchT $\exists\Diamond_{\leq T}(x = x_w \wedge y = y_w)$

Przykładowy wynik model checkingu

parameters								*formula*	*number of iterations*	*running times*
l	h	v_x	v_y	x_g	y_g	x_w	y_w			
13	10	2	1	0	0	10	8	$[periodT]$	55	7.77
								$[touch]$	55	6.69
								$[touchT]$	55	8.17
4	2	5	1	0	0	1	1	$[periodT]$	24	1.97
								$[touch]$	24	1.58
								$[touchT]$	24	1.90
3	8	1	2	0	0	1	6	$[periodT]$	10	0.56
								$[touch]$	10	0.40
								$[touchT]$	10	0.48

Rysunek: Wyniki model checkingu dla gry w bilard, przeprowadzonej programem KRONOS

Podsumowanie

- komponent dyskretny i ciągły
- automaty hybrydowe
- liniowe automaty hybrydowe
- weryfikacja na ogół jest bardzo trudna lub niemożliwa