Modelowanie i Analiza Systemów Informatycznych

Logika Temporalna i Automaty Czasowe

(9)

Inne rodzaje logiki temporalnej

© Paweł Głuchowski, Politechnika Wrocławska

wersja 2.3

Treść wykładu

Wybrane rodzaje logiki temporalnej

- Logika punktowa
- Logika przedziałowa

Treść wykładu

Logika RTCTL

- Charakterystyka logiki
- Składnia i semantyka
- Ograniczenie czasowe dla operatorów temporalnych
- Przykładowe formuły
- Porównanie z logiką CTL

Treść wykładu

Logika PRTCTL

- Charakterystyka logiki
- Składnia i semantyka
- Porównanie z logiką RTCTL

Treść wykładu

Logika TCTL

- Charakterystyka logiki
- Składnia i semantyka
- Ograniczenie czasowe dla operatorów temporalnych
- Przykładowe formuły
- Porównanie z logiką RTCTL

Treść wykładu

Logika CTL*

- Charakterystyka logiki
- Składnia i semantyka
- Przykładowe formuły

Wybrane rodzaje logiki temporalnej

- Logika punktowa
- Logika przedziałowa

Wybrane rodzaje logiki temporalnej

Logika punktowa

- **RTCTL** (*Real-Time Computation Tree Logic*)

  — wersja CTL, gdzie wartości czasu dane są ilościowo jako stałe;

- **PRTCTL** (*Parameterised Real-Time Computation Tree Logic*)

  — wersja CTL, gdzie wartości czasu dane są ilościowo jako zmienne;

- **TCTL** (*Timed Computation Tree Logic*)

  — wersja CTL, gdzie wartości czasu dane są ilościowo jako stałe;

- **CTL*** (*Full Branching Computation Tree Logic*)

  — wersja CTL o większej ekspresji, zawierającej też ekspresję LTL.

Wybrane rodzaje logiki temporalnej

Logika przedziałowa

- **ITL** (*Interval Temporal Logic*)

  — czas: dyskretny, skończony lub nieskończony, liniowy, przedziałowy;

- **DC** (*Duration Calculus*)

  — wersja ITL, gdzie mierzy się prawdziwość wyrażenia w odcinku czasu.

Logika przedziałowa pozwala określać prawdziwość wyrażenia dla przedziału czasu, a nie dla momentu (jak w logice punktowej).

W ramach przedmiotu „Modelowanie i Analiza Systemów Informatycznych” Logika przedziałowa nie zostanie omówiona.

Logika RTCTL

- Charakterystyka logiki
- Składnia i semantyka
- Ograniczenie czasowe dla operatorów temporalnych
- Przykładowe formuły
- Porównanie z logiką CTL

Logika RTCTL

Charakterystyka logiki

Real-Time Computation Tree Logic (RTCTL)

– rozszerzenie logiki CTL, w którym zakres każdego operatora temporalnego jest ograniczony.

Czas w RTCTL:

- punktowy,
- rozgałęziony (lewostronnie liniowy),
- lewostronnie skończony
  (rozciągający się od teraźniejszego momentu do przyszłości),
- określony ilościowo
  (w CTL czas jest określony jakościowo).

Logika RTCTL

Charakterystyka logiki

Zastosowanie RTCTL:

- takie samo jak dla CTL,
- do opisu zależności czasowych nie tylko jakościowo (jak w CTL), ale też ilościowo, np.:
  - aby podać maksymalną odległość „czasu” między dwoma wyrażeniami na każdej / pewnej ścieżce stanów,
  - aby podać maksymalny „czas” prawdziwości własności systemu.

Logika RTCTL

Składnia i semantyka

Struktura czasowa (struktura Kripkego) – jak w CTL.

Niech *WA* będzie zbiorem wyrażeń atomowych; niech $i \in \{0,1,...\}$.

Formułę generuje się wg reguł S1, S2, S3 i **S4**:

S1: każde wyrażenie $p \in WA$ jest formułą:

$s_i \models p$ jeśli $L(s_i,p) = prawda$

S2: jeśli p i q są formułami, to $p \wedge q$ i $\neg p$ też są formułami:

$s_i \models p \wedge q$ jeśli $s_i \models p$ i $s_i \models q$ $\qquad$ $s_i \models \neg p$ jeśli nie $s_i \models p$

Logika RTCTL

Składnia i semantyka

S3: jeśli *p* i *q* są formułami, to A(*p*U*q*), E(*p*U*q*), AX*p* i EX*p* też są formułami:

$s_i \models A(pUq)$ jeśli dla <u>każdej</u> ścieżki $s_i, s_{i+1}, \ldots$
$(\exists k \geq i)\ ((s_k \models q) \wedge (\forall j, i \leq j < k)\ (s_j \models p))$

$s_i \models E(pUq)$ jeśli dla <u>pewnej</u> ścieżki $s_i, s_{i+1}, \ldots$
$(\exists k \geq i)\ ((s_k \models q) \wedge (\forall j, i \leq j < k)\ (s_j \models p))$

$s_i \models AXp$ jeśli dla <u>każdej</u> ścieżki $s_i, s_{i+1}, \ldots$ $s_{i+1} \models p$

$s_i \models EXp$ jeśli dla <u>pewnej</u> ścieżki $s_i, s_{i+1}, \ldots$ $s_{i+1} \models p$

Logika RTCTL

Składnia i semantyka

S4: jeśli *p* i *q* są formułami, $m \in N_{\geq 0}$, to $A(pU^{\leq m} q)$ i $E(pU^{\leq m} q)$ też są formułami:

- $s_i \models A(pU^{\leq m} q)$ jeśli dla każdej ścieżki $s_i, s_{i+1}, \ldots$
  $(\exists k, m \geq k \geq i)((s_k \models q) \wedge (\forall j, i \leq j < k)(s_j \models p))$

- $s_i \models E(pU^{\leq m} q)$ jeśli dla pewnej ścieżki $s_i, s_{i+1}, \ldots$
  $(\exists k, m \geq k \geq i)((s_k \models q) \wedge (\forall j, i \leq j < k)(s_j \models p))$

m *– ograniczenie czasowe operatora temporalnego (krótko: ograniczenie).*

Logika RTCTL

Ograniczenie czasowe dla operatorów temporalnych

Operator temporalny F z ograniczeniem czasowym:

- $s_i \models AF^{\leq m} p \equiv A(\textit{prawda}\ U^{\leq m}\ p)$
  — *p* jest prawdziwe na każdej ścieżce w pewnym stanie z zakresu od s_i do s_{i+m}, włącznie;
- $s_i \models EF^{\leq m} p \equiv E(\textit{prawda}\ U^{\leq m}\ p)$
  — *p* jest prawdziwe na pewnej ścieżce w pewnym stanie z zakresu od s_i do s_{i+m}, włącznie.

Logika RTCTL

Ograniczenie czasowe dla operatorów temporalnych

Operator temporalny G z ograniczeniem czasowym:

- $s_i \models AG^{\leq m}\, p \equiv A(p\ U^{\leq m}\ \textit{fałsz})$
  — *p* jest prawdziwe na każdej ścieżce w każdym stanie z zakresu od s_i do s_{i+m}, włącznie;
- $s_i \models EG^{\leq m}\, p \equiv E(p\ U^{\leq m}\ \textit{fałsz})$
  — *p* jest prawdziwe na pewnej ścieżce w każdym stanie z zakresu od s_i do s_{i+m}, włącznie.

Logika RTCTL

Ograniczenie czasowe dla operatorów temporalnych

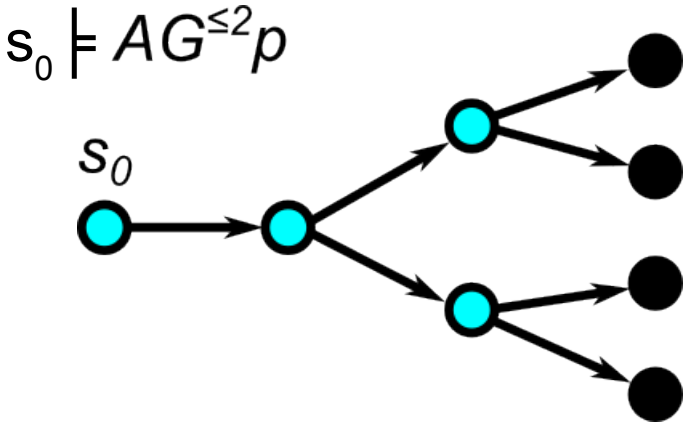
- Ograniczenie m (np. $E(pU^{\leq m} q)$) reprezentuje maksymalną liczbę <u>przejść</u> na ścieżce stanów zanim formuła z tak ograniczonym operatorem temporalnym zostanie spełniona, (licząc od stanu, w którym ta formuła jest prawdziwa).
- Przyjmujemy, że 1 przejście zajmuje 1 jednostkę czasu.
- Jeśli ograniczenie nie jest podane, to przyjmujemy, że może wynosić nieskończoność, np.:

$$A(pUq) \equiv (\exists m)(A(pU^{\leq m} q))$$

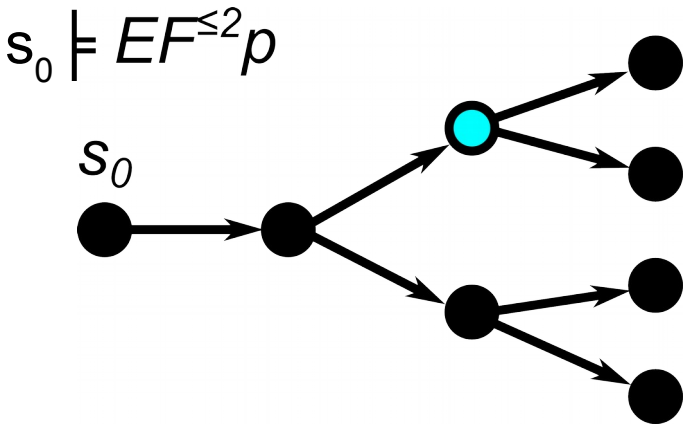
Logika RTCTL

Przykładowe formuły

- $s_0 \models AG^{\leq 2}\, p$
  – *p* jest prawdziwe w każdym stanie (momencie) każdej możliwej ścieżki (przyszłości): od s_0 do s_2.



- $s_0 \models EF^{\leq 2}\, p$
  – *p* jest prawdziwe w pewnym stanie (momencie) pewnej możliwej ścieżki (przyszłości): od s_0 do s_2.

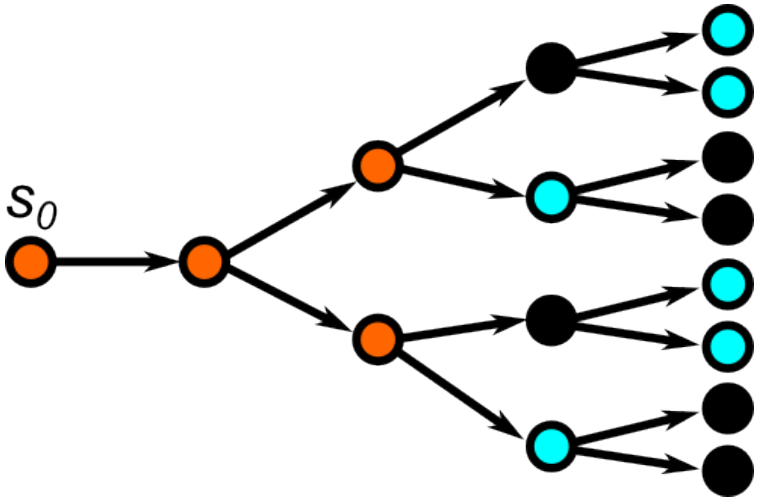


niebieski punkt: $s_i \models p$
czarny punkt: $s_i \models$ *prawda*

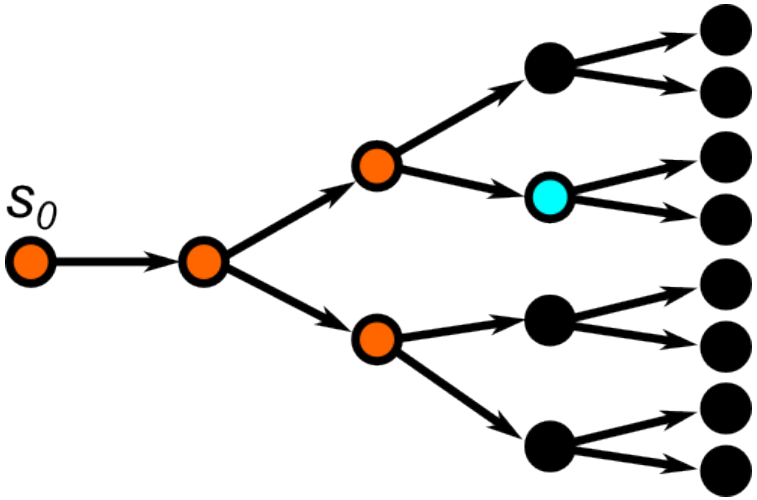
Logika RTCTL

Przykładowe formuły

- $s_0 \models AF^{\leq 4}\, p \wedge AG^{\leq 2}\, \neg p$
  – *p* jest prawdziwe w każdym stanie (momencie) każdej możliwej ścieżki (przyszłości): od s_3 do s_4, wcześniej nie.



- $s_0 \models EF^{\leq 4}\, p \wedge AG^{\leq 2}\, \neg p$
  – *p* jest prawdziwe w pewnym stanie (momencie) pewnej możliwej ścieżki (przyszłości): od s_3 do s_4, wcześniej nie (na każdej ścieżce).



niebieski punkt: $s_i \models p$
pomarańczowy punkt: $s_i \models \neg p$
czarny punkt: $s_i \models$ *prawda*

Logika RTCTL

Porównanie z logiką CTL

- RTCTL można zastosować aby określić zależności czasowe nie tylko jakościowo (jak w CTL), ale też ilościowo.
- **Przykład jakościowej zależności:** $AG(p \Rightarrow EF\ q)$

  – jeśli *p* jest prawdziwe, to *q* będzie prawdziwe w pewnym momencie pewnego wariantu przyszłości.
- **Przykład ilościowej zależności:** $AG(p \Rightarrow EF^{\leq 25}\ q)$

  – jeśli *p* jest prawdziwe, to *q* będzie prawdziwe w pewnym momencie <u>(nie później niż w ciągu następnych 25 momentów)</u> pewnego wariantu przyszłości.

Logika PRTCTL

- Charakterystyka logiki
- Składnia i semantyka
- Porównanie z logiką RTCTL

Logika PRTCTL

Charakterystyka logiki

Parameterized Real-Time Computation Tree Logic (PRTCTL)

– rozszerzenie logiki RTCTL, w którym ograniczenia operatorów temporalnych dane są jako kwantyfikowane zmienne.

Czas w PRTCTL jest zdefiniowany tak samo jak w RTCTL.

Zastosowanie PRTCTL:

- aby określić zależności czasowe:
  - jakościowo (jak w CTL),
  - ilościowo dla ustalonych ograniczeń operatorów temporalnych (jak w RTCTL),
  - ilościowo dla zmiennych ograniczeń operatorów temporalnych.

Logika PRTCTL

Składnia i semantyka

Struktura czasowa (struktura Kripkego) – jak w CTL i RTCTL.

Formułę PRTCTL generuje się z formuły vRTCTL:

> **vRTCTL** rozszerza RTCTL o możliwość użycia zmiennych (np.: v,w) i ich liniowych kombinacji (np.: $2v+w$) dla ograniczeń operatorów temporalnych, np: $EF^{\leq 2v+w}\, p$.

Niech $f(x)$ będzie formułą vRTCTL lub PRTCTL, gdzie x jest wolną zmienną, i niech $a \in N$.

Formuły PRTCTL:
- $\forall x\, f(x)$,
- $\forall x \leq a\, f(x)$,
- $\exists x\, f(x)$,
- $\exists x \leq a\, f(x)$.

Logika PRTCTL

Składnia i semantyka

Struktura czasowa (struktura Kripkego) – jak w CTL i RTCTL.

Niech *f(x)* będzie formułą vRTCTL lub PRTCTL, gdzie *x* jest wolną zmienną, i niech $a \in N$. Wówczas:

- $M, s_i \models \forall x\, f(x)$ jeśli dla każdego x $M, s_i \models f(x)$
- $M, s_i \models \forall x \leq a\, f(x)$ jeśli dla każdego $x \leq a$ $M, s_i \models f(x)$
- $M, s_i \models \exists x\, f(x)$ jeśli dla pewnego x $M, s_i \models f(x)$
- $M, s_i \models \exists x \leq a\, f(x)$ jeśli dla pewnego $x \leq a$ $M, s_i \models f(x)$

Logika PRTCTL

Porównanie z logiką RTCTL

- **Przykład dla RTCTL:** $AG(p \Rightarrow EF^{\leq 25} q)$

  – jeśli *p* jest prawdziwe, to *q* będzie prawdziwe w pewnym momencie (nie później niż dokładnie w ciągu następnych 25 momentów) w pewnym wariancie przyszłości.

- **Przykład dla PRTCTL:** $\exists x \leq 25 \quad AG(p \Rightarrow EF^{\leq x} q)$

  – jeśli *p* jest prawdziwe, to *q* będzie prawdziwe w pewnym momencie (nie później niż najwyżej w ciągu następnych 25 momentów) w pewnym wariancie przyszłości.

Logika TCTL

- Charakterystyka logiki
- Składnia i semantyka
- Ograniczenie czasowe dla operatorów temporalnych
- Przykładowe formuły
- Porównanie z logiką RTCTL

Logika TCTL

Charakterystyka logiki

Timed Computation Tree Logic (TCTL)

– rozszerzenie logiki CTL, w którym ograniczenie operatora temporalnego dane jest jako para: ograniczenie dolne i ograniczenie górne.

Struktura czasowa w TCTL jest taka sama jak w CTL.

Zastosowanie TCTL:

- aby określić zależności czasowe:
  - jakościowo (jak w CTL),
  - ilościowo dla ustalonych ograniczeń operatorów temporalnych (podobnie do RTCTL, ale dla liczb rzeczywistych), podając:
    - dolne ograniczenie i górne ograniczenie, czyli przedział czasu.

Logika TCTL

Składnia i semantyka

Struktura czasowa (struktura Kripkego) – jak w CTL.

Niech *WA* będzie zbiorem wyrażeń atomowych; niech $i \in \{0,1,...\}$.

Formułę generuje się wg reguł S1, S2 i **S3**:

S1: każde wyrażenie $p \in WA$ jest formułą:

$s_i \models p$ jeśli $L(s_i,p) = prawda$

S2: jeśli *p* i *q* są formułami, to $p \wedge q$ i $\neg p$ też są formułami:

$s_i \models p \wedge q$ jeśli $s_i \models p$ i $s_i \models q$ $\qquad$ $s_i \models \neg p$ jeśli nie $s_i \models p$

Logika TCTL

Składnia i semantyka

Niech I będzie przedziałem nieujemnych liczb rzeczywistych ($R_{\geq 0}$) o postaci: I=[a,b], I=(a,b], I=[a,b) lub I=(a,b), gdzie $a,b \in N_{\geq 0}$.

I nazywamy ograniczeniem czasowym operatora temporalnego.

S3: jeśli *p* i *q* są formułami, to $A(pU_I q)$ i $E(pU_I q)$ też są formułami:

$s_i \models A(pU_I q)$ jeśli dla <u>każdej</u> ścieżki $s_i, s_{i+1}, \ldots$
$(\exists k \geq i, \mathbf{k\text{-}i} \in \mathbf{I})\ ((s_k \models q) \wedge (\forall j, i \leq j < k)(s_j \models p))$

$s_i \models E(pU_I q)$ jeśli dla <u>pewnej</u> ścieżki $s_i, s_{i+1}, \ldots$
$(\exists k \geq i, \mathbf{k\text{-}i} \in \mathbf{I})\ ((s_k \models q) \wedge (\forall j, i \leq j < k)(s_j \models p))$

Logika TCTL

Ograniczenie czasowe dla operatorów temporalnych

Operator temporalny F z ograniczeniem czasowym:

- $s_i \models AF_I\, p \equiv A(\mathit{prawda}\ U_I\, p)$
  – p jest prawdziwe na każdej ścieżce w pewnym stanie s_{i+j}, gdzie $j \in I$;
- $s_i \models EF_I\, p \equiv E(\mathit{prawda}\ U_I\, p)$
  – p jest prawdziwe na pewnej ścieżce w pewnym stanie s_{i+j}, gdzie $j \in I$.

Operator temporalny G z ograniczeniem czasowym:

- $s_i \models EG_I\, p \equiv \neg\, AF_I \neg p$
- $s_i \models AG_I\, p \equiv \neg\, EF_I \neg p$

Logika TCTL

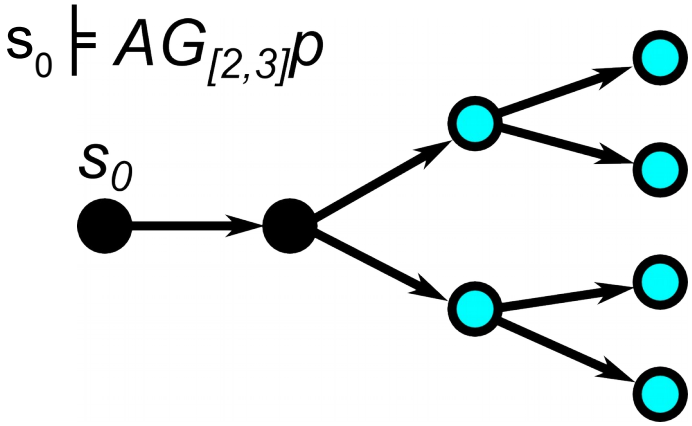
Przykładowe formuły

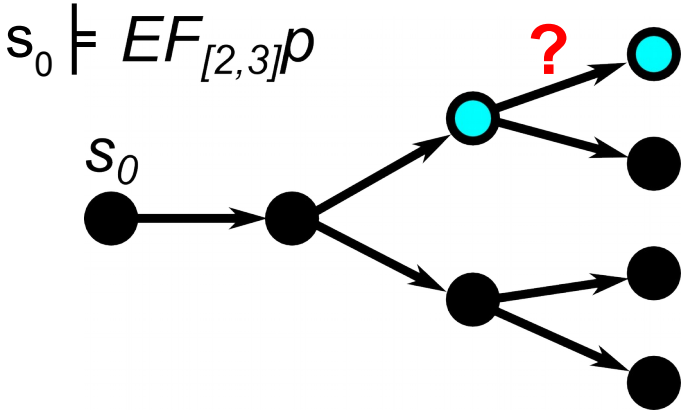
- $s_0 \models AG_{[2,3]}\, p$
  – *p* jest prawdziwe w każdym momencie każdej możliwej ścieżki (przyszłości) między stanami s_{0+2} a s_{0+3}, włącznie.

- $s_0 \models EF_{[2,3]}\, p$
  – *p* jest prawdziwe w pewnym momencie pewnej możliwej ścieżki (przyszłości) między stanami s_{0+2} a s_{0+3}, włącznie.

Uwaga: czas $\in \mathbf{R}_{\geq 0}$

niebieski punkt: $s_i \models p$ **?**
czarny punkt: $s_i \models$ *prawda*





Logika TCTL

Porównanie z logiką RTCTL

- **Przykład dla RTCTL:**

  $s_0 \models AF^{\leq 4}\, p \wedge AG^{\leq 2} \neg p$
  — p jest prawdziwe w każdym stanie (momencie) każdej możliwej ścieżki (przyszłości): od s_3 do s_4; <u>wcześniej nie</u>.

- **Przykład dla TCTL:**

  $s_0 \models AF_{[3,4]}\, p$
  — p jest prawdziwe w każdym stanie (momencie) każdej możliwej ścieżki (przyszłości): od s_3 do s_4.

  $s_0 \models AF_{[3,4]}\, p \wedge AG_{[0,2]} \neg p$
  — p jest prawdziwe w każdym stanie (momencie) każdej możliwej ścieżki (przyszłości): od s_3 do s_4; <u>wcześniej nie</u>.

Logika CTL*

- Charakterystyka logiki
- Składnia i semantyka
- Przykładowe formuły

Logika CTL*

Charakterystyka logiki

***Full Branching* Computation Tree Logic (CTL*)**

– rozszerzenie logiki CTL łączące ekspresję języków LTL i CTL.

- Nie każdy operator temporalny musi mieć swego operatora ścieżkowego, np.:
  - formuła AFAG*p* jest poprawna w CTL i CTL*;
  - formuła AFG*p* jest poprawna tylko w CTL*.

Czas w CTL* jest zdefiniowany tak samo jak w CTL.

Zastosowanie CTL* – jak dla CTL i LTL.

Logika CTL*

Składnia i semantyka

Struktura czasowa (struktura Kripkego) – jak w CTL.

Niech *WA* będzie zbiorem wyrażeń atomowych; niech $i \in \{0,1,...\}$.

Formułę (punktową) generuje się wg reguł S1, S2 i S3:

S1: każde wyrażenie $p \in WA$ jest formułą punktową:

$s_i \models p$ jeśli $L(s_i,p) = prawda$

S2: jeśli p i q są formułami punktowymi, to $p \wedge q$ i $\neg p$ też nimi są:

$s_i \models p \wedge q$ jeśli $s_i \models p$ i $s_i \models q$ $\qquad$ $s_i \models \neg p$ jeśli nie $s_i \models p$

S3: jeśli p jest formułą ścieżkową, to Ap i Ep są formułami punktowymi:

$s_i \models Ep$ jeśli $\exists x=(s_i,s_{i+1},...)\ x \models p$ $\qquad$ $s_i \models Ap$ jeśli $\forall x=(s_i,s_{i+1},...)\ x \models p$

Logika CTL*

Składnia i semantyka

Formułę ścieżkową generuje się wg reguł P1, P2 i P3:

P1: każda formuła punktową jest też formułą ścieżkową:

$x \models p$ jeśli $s_i \models p$ dla $x=(s_i,s_{i+1},...)$

P2: jeśli p i q są formułami ścieżkowymi, to $p \wedge q$ i $\neg p$ też nimi są:

$x \models p \wedge q$ jeśli $x \models p$ i $x \models q$ dla $x=(s_i,s_{i+1},...)$
$x \models \neg p$ jeśli nie $x \models p$ dla $x=(s_i,s_{i+1},...)$

P3: jeśli p i q są formułami ścieżkowymi, to pUq i Xp też nimi są:

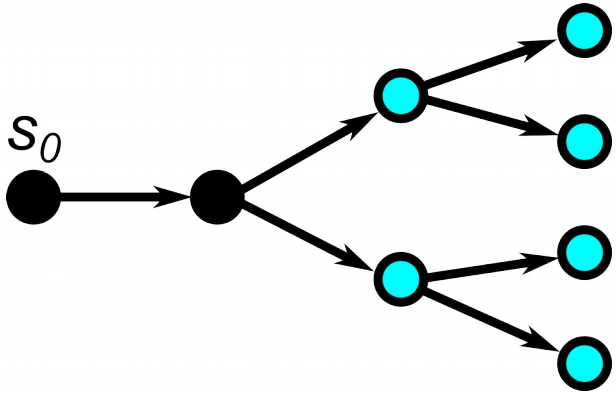
$x \models (pUq)$ jeśli $(\exists k \geq i)((x^k \models q) \wedge (\forall j, i \leq j < k)(x^j \models p))$
dla $x=(s_i,s_{i+1},...)$, $x^k=(s_{i+k},...)$, $x^j=(s_{i+j},...)$
$x \models Xp$ jeśli $x^1 \models p$ dla $x=(s_i,s_{i+1},...)$, $x^1=(s_{i+1},...)$
gdzie $x^n=(s_{i+n},...)$ jest podścieżką ścieżki x: $x=(s_i,...x^n)$.

Logika CTL*

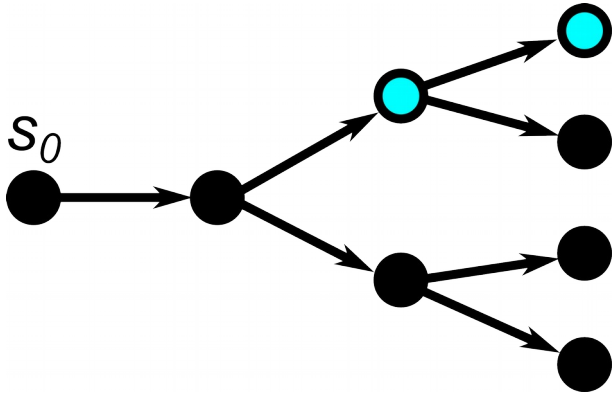
Przykładowe formuły

(te formuły CTL nie należą do CTL)*

- $s_0 \models$ AFG*p*
  – *p* jest prawdziwe w każdym momencie, począwszy od pewnego momentu, na każdej możliwej ścieżce.



- $s_0 \models$ EFG*p*
  – *p* jest prawdziwe w każdym momencie, począwszy od pewnego momentu, na pewnej możliwej ścieżce.



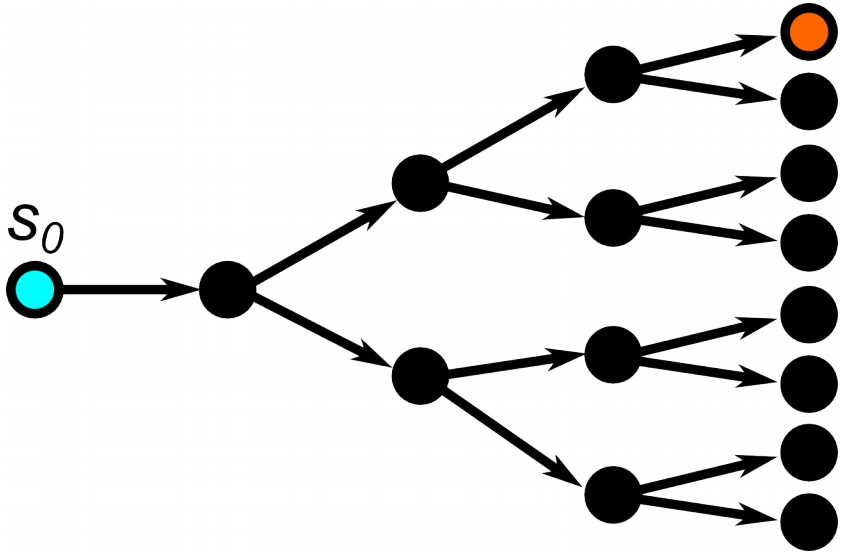
niebieski punkt: $s_i \models p$
czarny punkt: $s_i \models$ *prawda*

Logika CTL*

Przykładowe formuły

(te formuły CTL nie należą do CTL)*

- $s_0 \models E(p \wedge (Fp \Rightarrow Fq))$
  – na pewnej ścieżce: *p* jest prawdziwe i, jeśli kiedyś *p* jest prawdziwe, to kiedyś *q* też jest prawdziwe.



Ta koniunkcja i implikacja są prawdziwe <u>na tej samej ścieżce</u>!

niebieski punkt: $s_i \models p$
pomarańczowy punkt: $s_i \models q$
czarny punkt: $s_i \models$ *prawda*

Koniec

Literatura:

- R. Alur et al. *„Model-Checking in Dense Real-Time”*, 1993
- A. Cau et al. *„Interval Temporal Logic”*, 2007
- Z. Chaochen *„Duration Calculus. A formal approach to Real-Time Systems”*, 2004
- E.A. Emerson *„Temporal and modal logic”*, 1995
- E.A. Emerson et al. *„Parametric Quantitative Temporal Reasoning”*, 1999
- E.A. Emerson et al. *„Quantitative temporal reasoning”*, 1992
- R. Klimek *„Zastosowanie logiki temporalnej w specyfikacji i weryfikacji oprogramowania – w stronę czasu rzeczywistego”*, w: „Wykłady zaproszone. Systemy Czasu Rzeczywistego 2000”, 2001