Sieci Petriego

Andrzej M. Borzyszkowski

Instytut Informatyki
Uniwersytet Gdański

sem. letni 2022/2023

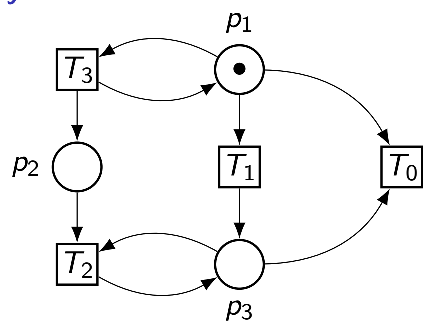
inf.ug.edu.pl/~amb/

Sieci miejsc i tranzycji c.d.

Stopnie żywości

- sieć $\mathcal{N} = (P, T, F, m_0)$, $t \in T$ pewna tranzycja, jest ona:
- L_1-żywa, jeśli jest wykonalna w pewnym osiągalnym zaznaczeniu m, tzn. $\exists\, \sigma \in T^*, m.\, m_0[\sigma\rangle m[t\rangle$
- L_2-żywa, jeśli jest wykonalna dowolną liczbę razy, tzn. $\forall\, k \in \mathbb{N}. \exists\, \sigma \in T^*.\, m_0[\sigma\rangle, P(t)(\sigma) \geqslant k$ – można znaleźć ciąg tranzycji, w którym wybrana t występuje co najmniej k razy
- L_3-żywa, jeśli jest wykonalna nieskończenie wiele razy, tzn. $\exists\, \sigma \in T^\omega.\, m_0[\sigma\rangle, P(t)(\sigma) = \infty$ – można znaleźć ciąg tranzycji, w którym wybrana t występuje nieskończoną liczbę razy
- L_4-żywa, jeśli jest żywa, tzn. w dowolnym osiągalnym zaznaczeniu może być w przyszłości wykonana – $\forall\, m \in [m_0\rangle. \exists\, \sigma \in T^*.\, m[\sigma\rangle, P(t)(\sigma) \geqslant 1$

Stopnie żywości, przykład



- tranzycja T_0 jest martwa, nigdy nie będzie wykonana
- tranzycja T_1 jest L_1-żywa, może być wykonana, ale tylko jeden raz
- tranzycja T_2 jest L_2-żywa, aby ją wykonać k razy, należy wykonać k razy tranzycję T_3 a potem jeden raz T_1 nie da się jednak wykonać jej więcej razy
- tranzycja T_3 jest L_3-żywa, w zaznaczenie początkowym może być wykonywalna dowolnie długo
- żadna tranzycja nie jest żywa (L_4-żywa), gdy miejsce p_1 jest puste T_1 i T_3 są niewykonalne, gdy p_2 jest puste, również T_2 jest niewykonalna czyli zakleszczenie

Stopnie żywości, hierarchia

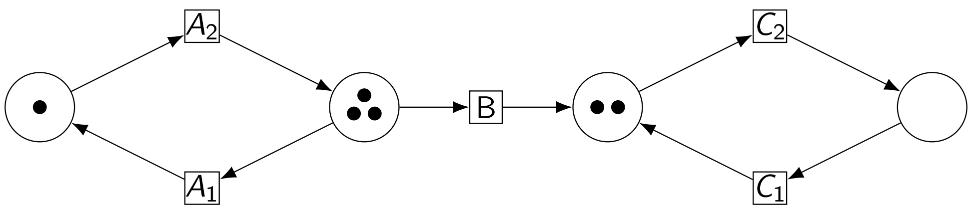
- Tw.: L_{i+1}-żywość implikuje L_i-żywość
- dw.: jeśli t jest wykonalna k razy, to w szczególności raz, $L_2 \Rightarrow L_1$
  jeśli t występuje nieskończenie wiele razy w ciągu σ takim, że $m_0[\sigma\rangle$, to może być wykonana k razy dla dowolnej liczby k, $L_3 \Rightarrow L_2$
  jeśli w każdym zaznaczeniu osiągalnym m istnieje ciąg σ zawierający t i $m[\sigma\rangle$, to startując od m_0 i σ_0 można przedłużać ten ciąg $m_0[\sigma_0\rangle m_1[\sigma_1\rangle \ldots$ do nieskończoności, każdy fragment zawiera t, czyli cały ciąg zawiera nieskończenie wiele wystąpień t, $L_4 \Rightarrow L_3$

Pułapki w odpływach

- $\mathcal{N} = (P, T, F, m_0)$, bez miejsc izolowanych
- Tw.: jeśli sieć $\mathcal{N}$ jest żywa, to każdy odpływ $A \subseteq P$ ma jakiś token w zaznaczeniu m_0
  dw: w przeciwnym przypadku tranzycje w $A^\bullet$ byłyby martwe
- $\mathcal{N} = (P, T, F, m_0)$, $T \neq \emptyset$, bez miejsc izolowanych, bez założeń o wagach i pojemnościach
- Tw.: jeśli każdy A odpływ zawiera w sobie pułapkę z tokenem w zaznaczeniu m_0, to sieć $\mathcal{N}$ nie ma zakleszczeń
  dw.: Niech m będzie zakleszczeniem, $A \subseteq P$ zbiorem miejsc bez tokenów w m. Nie da się wykonać żadnej tranzycji, nie zmieni tego inne zaznaczenie z większą liczbą tokenów (wagi są 1) ani z większą liczbą pustych miejsc (pojemność miejsc nie jest powodem uniemożliwienia tranzycji).
  Czyli zbiór A jest odpływem, po opróżnieniu z tokenów nigdy więcej ich nie będzie. Stoi to w sprzeczności z założeniem, że każdy odpływ zawiera pułapkę z tokenami, które już nigdy ich nie opuszczą.

Maszyny stanowe

- def.: maszyna stanowa, *state machine*, to sieć $\mathcal{N} = (P, T, F, m_0)$ bez założeń o wagach i pojemnościach taka, że $\forall\, t \in T.\, |{}^\bullet t| = |t^\bullet| = 1$



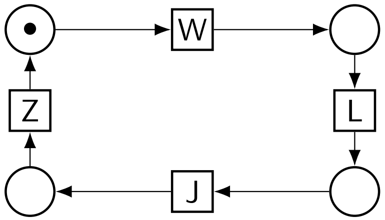
- Tw.: liczba tokenów w maszynie stanowej jest stała
- dw.: każda tranzycja przenosi jeden token z jednego miejsca na inne, z góry ustalone, bilans nie zmienia się
- wniosek: każde miejsce w maszynie stanowej ma ograniczoną pojemność

Maszyny stanowe c.d.

- Tw.: maszyna stanowa $\mathcal{N} = (P, T, F, m_0)$ jest żywa w.t.w. każda składowa spójności jest silnie spójna i ma token w m_0
- dw.: ($\Rightarrow$) jeśli składowa spójności nie ma tokenu w m_0, to żadna z tranzycji nie będzie mogła być nigdy wykonana, wbrew założeniu żywości
  jeśli składowa spójności nie jest silnie spójna, to zawiera w sobie odpływ, po usunięciu tokenów tranzycje nie będą mogły być wykonane, wbrew założeniu
  ($\Leftarrow$) token w miejscu p pozwoli wykonać dowolną tranzycję w $p^\bullet$, ta z kolei zaznaczy kolejne miejsce; wszystkie tranzycje w (silnej) składowej spójności będą mogły kiedyś być wykonane. Ponieważ nie ma innych tranzycji niż znajdujące się w silnych składowych spójności, więc sieć jest żywa.

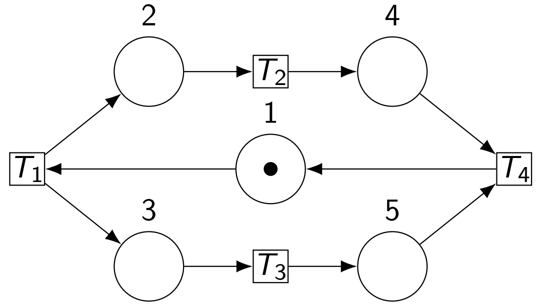
Maszyny stanowe inne pojęcie

- def.: maszyna stanowa, *state machine*, to sieć $\mathcal{N} = (P, T, F, m_0)$ bez założeń o wagach i pojemnościach taka, że $\forall\, t \in T.\, |{}^\bullet t| = |t^\bullet| = 1$ oraz m_0 ma jeden token
- konflikty są możliwe, ale współbieżność nie jest możliwa
- graf (osiągalnych) zaznaczeń, po ew. uporządkowaniu sieci, będzie izomorficzny z $\mathcal{N}$ – zaznaczenie ma jeden token wyznaczony przez miejsce, tranzycje będą łukami pomiędzy miejscami



Grafy znakowane

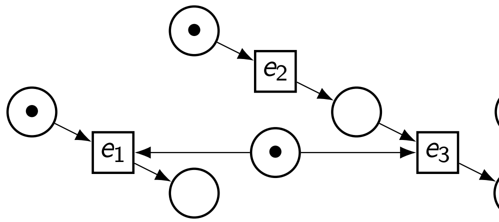
- def.: graf znakowany, *marked graph*, to sieć $\mathcal{N} = (P, T, F, m_0)$ bez założeń o wagach i pojemnościach taka, że $\forall\, p \in P.\, |{}^\bullet p| = |p^\bullet| = 1$
- w grafie znakowanym nie ma konfliktów, mogą być współbieżności

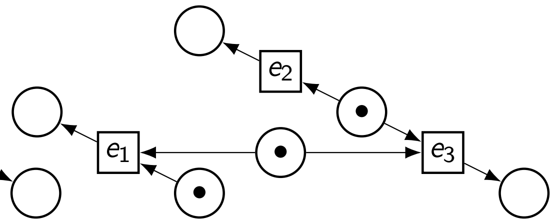


Grafy znakowane c.d.

- Tw.: liczba tokenów na każdym cyklu grafu znakowanego $\mathcal{N} = (P, T, F, m_0)$ jest stała
- dw.: jeśli $p_1, p_2, \ldots, p_n$ jest cyklem miejsc, to $p_1^\bullet = {}^\bullet p_2, \ldots, p_n^\bullet = {}^\bullet p_1$ tranzycja $t \in T$ albo nie należy do ${}^\bullet\{p_1, \ldots, p_n\}$ i wykonanie t nie zmienia tokenów na cyklu, albo $t = p_i^\bullet = {}^\bullet p_{i+1}$ (lub $t = p_n^\bullet = {}^\bullet p_1$), wówczas wykonanie t zabiera token w p_i i dodaje w p_{i+1}
- wniosek: graf znakowany jest żywy w.t.w. na każdym cyklu początkowe znakowanie m_0 ma co najmniej jeden token
- jest 1-bezpieczny w.t.w. na każdym cyklu jest dokładnie 1 token

Sieci wolnego wyboru





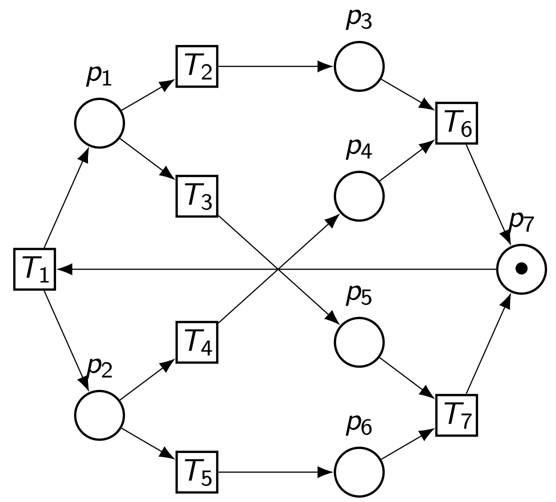
- konfuzja: dwie tranzycje można wykonać współbieżnie, ale wykonanie w różnej kolejności wpływa na inne tranzycje – konflikt zwiększa się lub zmniejsza
- sieć wolnego wyboru – nie ma sytuacji konfuzji, można dowolnie wybrać, którą z tranzycji w konflikcie wykonać
- def.: $\mathcal{N} = (P, T, F, m_0)$ jest siecią wolnego wyboru jeśli $\forall\, p \in P, t \in T.\, F(p, t) > 0 \Rightarrow p^\bullet = \{t\}$ lub ${}^\bullet t = \{p\}$

Sieci wolnego wyboru

- Tw.: następujące warunki równoważne
  1. $\forall\, p \in P, t \in T.\, F(p,t) > 0 \Rightarrow p^\bullet = \{t\}$ lub ${}^\bullet t = \{p\}$
  2. $\forall\, p \in P.\, |p^\bullet| > 1 \Rightarrow \forall\, t \in p^\bullet.\, {}^\bullet t = \{p\}$
  3. $\forall\, p_1 \neq p_2 \in P.\, p_1^\bullet \cap p_2^\bullet \neq \emptyset \Rightarrow \exists\, t \in T.\, p_1^\bullet = p_2^\bullet = \{t\}$
- dw.:
  - (1) $\rightarrow$ (2): jeśli $|p^\bullet| > 1$, $t \in p^\bullet$, to $p^\bullet \neq \{t\}$, stąd ${}^\bullet t = \{p\}$
  - (2) $\rightarrow$ (1): niech $F(p,t) > 0$, jeśli $|p^\bullet| = 1$, to $p^\bullet = \{t\}$, w.p.p. ${}^\bullet t = \{p\}$
  - (1) $\rightarrow$ (3): jeśli $p_1 \neq p_2$, $t \in p_1^\bullet \cap p_2^\bullet$, więc ${}^\bullet t \neq \{p_1\}$ i ${}^\bullet t \neq \{p_2\}$, stąd $p_1^\bullet = p_2^\bullet = \{t\}$
  - (3) $\rightarrow$ (1): niech $F(p,t) > 0$, być może ${}^\bullet t = \{p\}$, jeśli nie, to $\exists\, p' \in P, p' \neq p, p' \in {}^\bullet t$, stąd $p'^\bullet = p^\bullet = \{t\}$
- na kilka sposób mamy opisany zakazany wzorzec, gdy z miejsca p wychodzą dwie strzałki a tranzycje zależące od tego miejsca, zależą też od innych miejsc

Sieci wolnego wyboru

- Tw.: sieć wolnego wyboru $\mathcal{N} = (P, T, F, m_0)$ jest żywa w.t.w. każdy odpływ zawiera pułapkę, która ma token przy znakowaniu m_0



- $\{p_7, p_1, p_2, p_3, p_6\}$ jest odpływem, $\{p_4, p_5\}$ jest zakleszczeniem