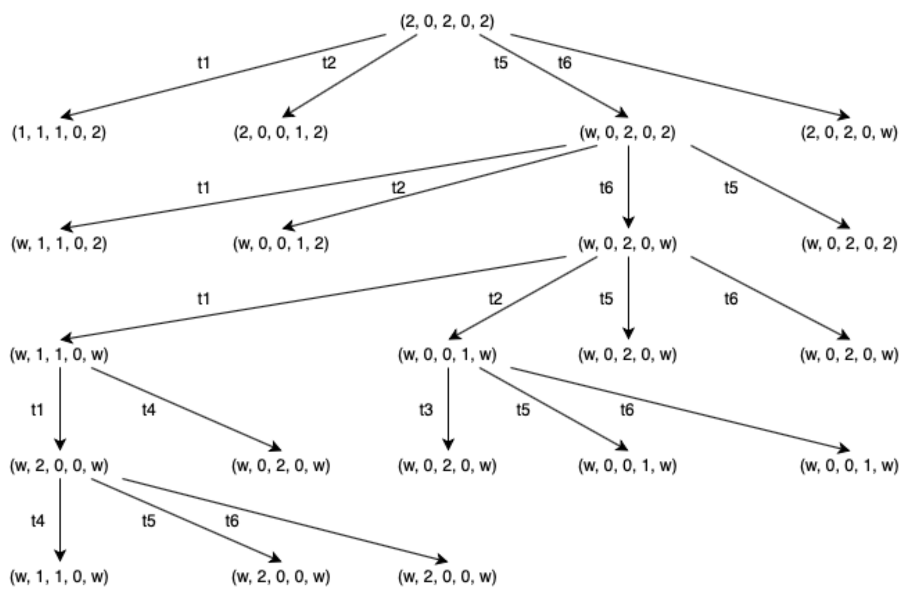
# MBA - 1. úloha

Tomáš Lapšanský (xlapsa00)

March 26, 2020

## 1 Príklad




### 1.1

a) Sieť nie je obmedzená. Aby sieť bola obmedzená, muselo by platiť, že všetky miesta v sieti sú obmedzené. Ako môžeme vidieť na strome dosiahnuteľných značení, existujú tu miesta, ktoré maju označnie $w$, teda môžu dosiahnúť neobmedzený počet značiek.

b) Sieť nie je bezpečná, kedže nie je obmedzená. Vyplýva to z definície bezpečnosti siete.

c) Značenie $M_1 = (3, 0, 1, 1, 2)$ nie je pokryteľné. Ak by bolo pokryteľné, videli by sme ta na niektorom uzle stromu.

d) Sieť je živá. Každý jej prechod je živý, keďže evolúciou ľubovoľného dosiahnuteľného značenie sme schopný sa dostať späť k vykonaniu tohoto prechodu.

## 1.2

| | t1 | t2 | t3 | t4 | i1 | i2 | i3 | M0 |
|---|---|---|---|---|---|---|---|---|
| p1 | -1 | 0 | 0 | 1 | 1 | 0 | 0 | 3 |
| p2 | 1 | 0 | 0 | -1 | 1 | 0 | 1 | 0 |
| p3 | -1 | -2 | 2 | 1 | 0 | 0 | 1 | 1 |
| p4 | 0 | 1 | -1 | 0 | 0 | 1 | 2 | 1 |
| p5 | 0 | -1 | 1 | 0 | 0 | 1 | 0 | 2 |

a) $v_1 = (1,2,1,3,1)$ dokážeme zostrojiť pomocou $i_1 + i_2 + i_3$. Je teda P-invariantov.

$v_2 = (1,2,1,2,1)$ nie je P-invariantov. Nedokážeme to zostrojiť pomocou aritmetických úprav, a zároveň to dokážeme overiť pomocou vzorca $N^T.v_2$.

b) Sieť nie je striktne konzervatívna, keďže neplatí rovnosť súčtu prvkov v značení. Napríklad pri značení $M_0 = (3,0,1,1,2)$ je súčet prvkov 6, zatiaľ čo pri značení po aplikácií $t3-> M_1 = (3,0,2,0,3)$ je súčet prvkov 8. Sieť je konzervatívna ku váhovému vektoru $v = (1,2,1,2,0)$.

c) Značenie $M_2 = (3,0,1,1,2)$ je dosiahnuteľné aplikáciou prechodov $t_1$ a $t_4$.

d) Invariant $i_1$ označuje to, že súčet procesov v stavoch $p_1$ a $p_2$ je vždy rovný 3. Počet čitateľov sa teda nemení a ich počet je konštantný.

Invariant $i_2$ označuje to isté ako u čitateľov, ale pre procesi $p_4$ a $p_5$, teda konštantný počet a jasne daný stav.

Invariant $i_3$ označuje to, že kapacita miesta $p_3$ je rovná dvom, a teda že čitatelia zaberajú pomyselné 2 miesta, no jeden pisár zaberie sám celý priestor (2 miesta), je teda možné aby bol buď jeden pisár alebo čitatelia, nie spoločne.

## 1.3

a) Vektor $v_1 = (30,20,20,30)$ je T-invariantov, dokážeme ho zostrojiť ako $v_1 = 30 * i_2 + 20 * i_1$. Vektor $v_2 = (1,2,1,2,1)$ nie je T-invariantov, pretože ho takto nedokážeme zostrojiť.

b) Živá sieť musí byť pokrytá T-invariantami. Ak nie je pokrytá T-invariantami, nie je živá a nie je ani obmedzená.

| | t1 | t2 | t3 | t4 | M0 |
|---|---|---|---|---|---|
| | -1 | 0 | 0 | 1 | 3 |
| | 1 | 0 | 0 | -1 | 0 |
| | -1 | -2 | 2 | 1 | 2 |
| | 0 | 1 | -1 | 0 | 0 |
| | 0 | -1 | 1 | 0 | 3 |
| i1 | 0 | 1 | 1 | 0 | |
| i2 | 1 | 0 | 0 | 1 | |

# 2 Príklad

## 2.1

Protokol na základe dosiahnuteľných stavov garantuje vzájomné vylúčenie. Keďže je vstup do kritickej sekcie modelovaný ako "obslužná linka", do kritickej sekcie nikdy nevstúpi viac ako jeden proces. Postrádam však význam tejto sekcie, keďže na rovnaké miesto v pamäti zapisujú procesy aj mimo tejto kritickej sekcie.

Pre zistenie, či sieť garantuje nemožnosť uviaznutia, musíme analyzovať sieť a zistiť, či je daná sieť živá. Analýzou stromu dosiahnuteľných značení vieme dokázať, že sieť nie je živá, pretože: Ak procesy začnú v rovnakom čase a teda spolu prevedú riadok 2 a riadok 3, jeden z procesov (proces 0) sa zasekne na riadku 4 kde sa nachádza funkcia *waituntil*. Tu bude čakať až do momentu, než druhý proces nastaví flag. K tomuto stavu však nikdy nedôjde, pretože flag procesu 1 je nastavený tak, že nikdy nevôjde do kritickej sekcie a bude stále opakovať svôj cyklus.

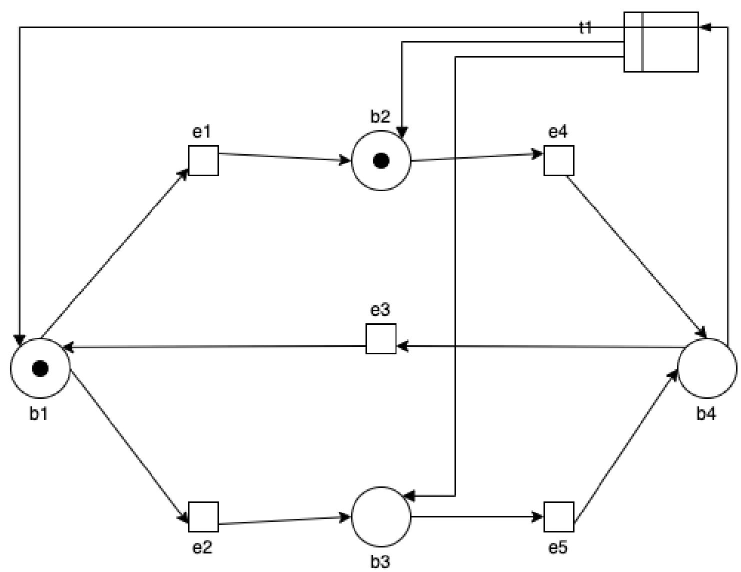
## 2.2

Ak upravíme sieť tak, aby do jednotlivých vetiev mohlo pristupovať neobmedzené množstvo procesov, sieť bude neobmedzená. Dokážeme vygenerovať teoreticky neobmedzené množstvo dosiahnuteľných stavov.

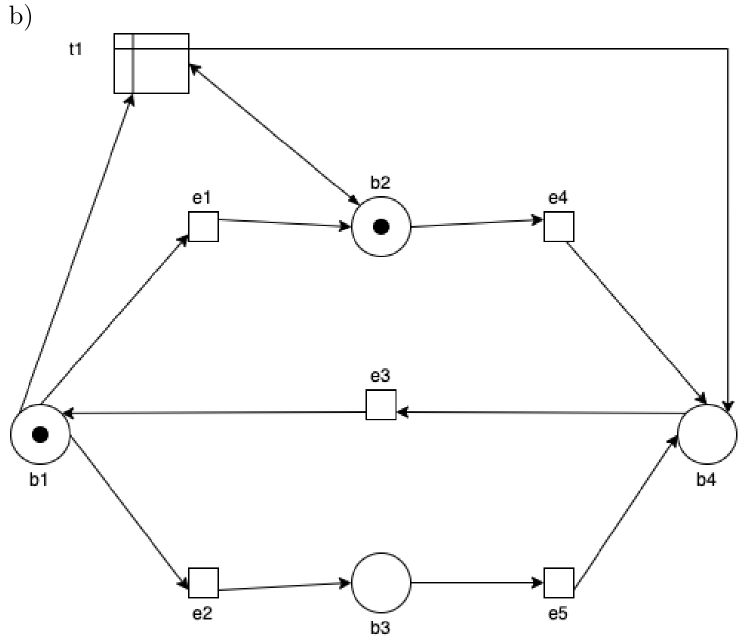
# 3 Príklad

## 3.1

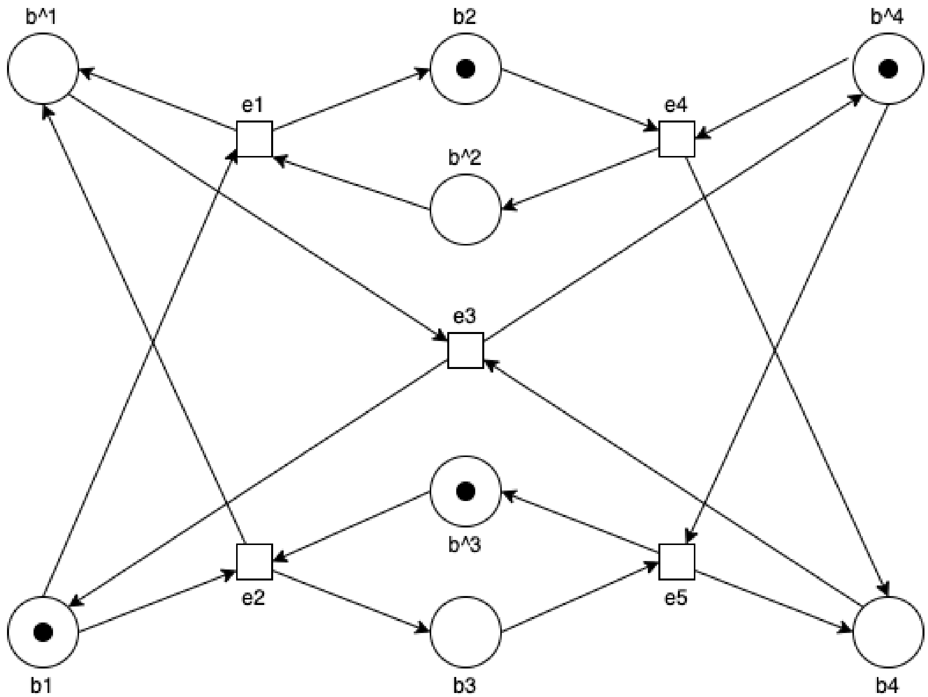
a)



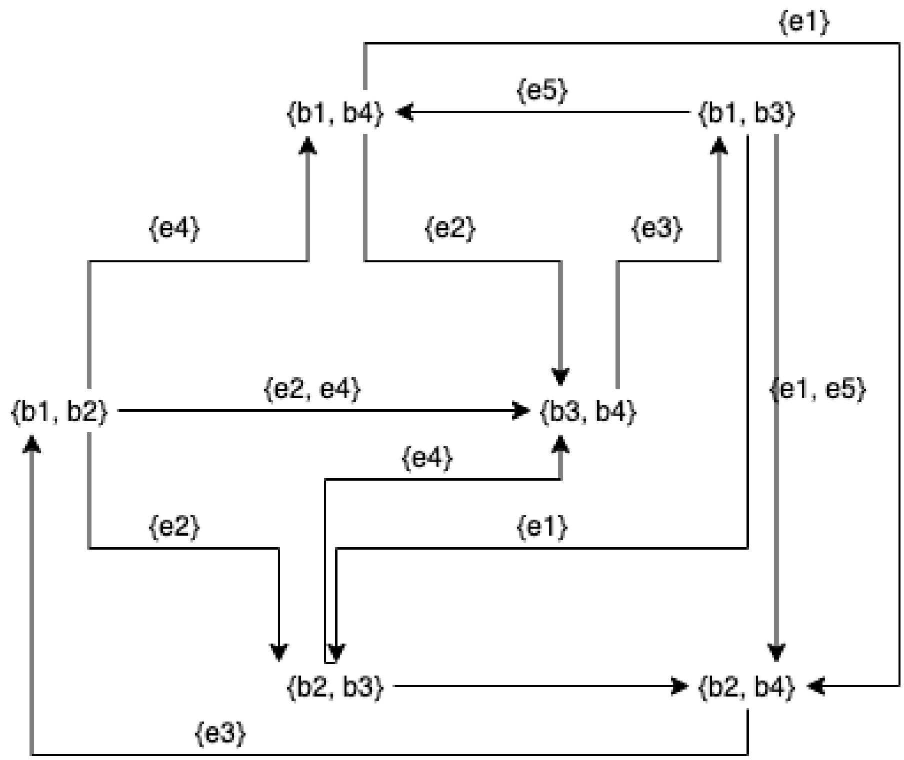

b)



### 3.2



**3.3**


{e1}
{e5}
{b1, b4}
{b1, b3}
{e4}
{e2}
{e3}
{b1, b2}
{e2, e4}
{b3, b4}
{e1, e5}
{e4}
{e2}
{e1}
{b2, b3}
{b2, b4}
{e3}