Elérhetőségi analízis
Petri hálók dinamikus
és strukturális tulajdonságai

dr. Bartha Tamás
BME Közlekedés- és Járműirányítási Tanszék

Dr. Pataricza András
BME Méréstechnika és Információs Rendszerek Tanszék

Petri hálók vizsgálata

Vizsgálati lehetőségek

Az elemzés mélysége szerint:

- Szimuláció
- Állapottér bejárása
  - elérhetőségi gráf analízis
  - dinamikus (viselkedési) tulajdonságok
- Strukturális tulajdonságok
  - invariáns analízis

egy trajektória bejárása

minden trajektória bejárása (kimerítő bejárás)

kezdőállapottól független (bármely kezdőállapotra)

ha mindez nem vezet eredményre

- Algebrai közelítés, részleges döntés

Petri hálók kezdőállapot-függő analízise

- Elérhetőségi analízis
  - az elérhetőségi gráf (állapottér!) konstrukciójával
  - **dinamikus** (viselkedési) tulajdonságok:
    - elérhetőség, fedhetőség, élőség, holtpontmentesség, korlátosság, fairség, megfordíthatóság
  - kezdőállapot-függő
    - nem általános érvényű tulajdonságok

ha az állapottér nem kezelhető

- Redukciós technikák
  - **tulajdonságmegtartó** transzformációk
    - a struktúra (és így az állapottér) szisztematikus csökkentése

Petri hálók kezdőállapot-független analízise

- Kezdőállapot-független tulajdonságok
  - meghatározhatók csak a struktúra alapján
    - vagy univerzális (minden működésre érvényes)
    - vagy egzisztenciális (lehetséges ilyen működés)
  - **strukturális** tulajdonságok
    - strukturális élőség, strukturális korlátosság, vezérelhetőség, konzervativitás, ismételhetőség, konzisztencia
- Invariáns tulajdonságok
  - tüzelési (T-, transition) invariáns
    - lehetséges olyan működés, hogy az állapot újra előálljon
  - hely (P-, place) invariáns
    - minden állapotban igaz és állandó
    - súlyozott tokenösszeg $\Rightarrow$ dinamikus egyensúly

Elérhetőség fogalma, tulajdonságai
Elérhetőségi analízis

Elérhetőség

Elérhetőségi analízis

- Kezdőállapotfüggő dinamikus viselkedés
  - Jelölés (marking) = állapot
  - Tokeneloszlás = állapotváltozó
  - Tüzelés = állapotátmenet
  - Tüzelési sorozatok hatására M_0, M_1, …, M_n állapotsorozat
- Állapotsorozat: trajektória az állapottérben
- M_n állapot *elérhető* az M_0 kiinduló állapotból, ha

$$\boxed{\exists \vec{\sigma} : M_0 [\vec{\sigma} > M_n}$$

- Elérhetőségi gráf: állapottér grafikus képe

Elérhetőségi analízis

Az M_0 kiinduló állapotból az N Petri hálóban

- Elérhető állapotok

$$\boxed{R(N,M_0)=\left\{M \mid \exists\vec{\sigma}: M_0\left[\vec{\sigma}>M\right.\right\}}$$

  - Állapot bázisú kérdések

- Végrehajtható tüzelési sorozatok

$$\boxed{L(N,M_0)=\left\{\exists\vec{\sigma} \mid M: M_0\left[\vec{\sigma}>M\right.\right\}}$$

  - Állapotátmenet (esemény) bázisú kérdések

Elérhetőségi probléma

Petri hálók elérhetőségi problémája:

– M_n állapot elérhető-e valamilyen M_0 kiinduló állapotból

$$\boxed{M_n \overset{?}{\in} R(N, M_0)}$$

Rész-tokeneloszlási probléma:

– helyek egy $P' \subset P$ részhalmazára korlátozva elérhető-e M_n állapot az adott helyekre megadott tokeneloszlással

$$\boxed{\overset{?}{\exists} M \in R(N, M_0) \wedge \forall p \in P' : M\left(p\right) = M_n\left(p\right)}$$

Elérhetőségi probléma megoldhatósága

- Az elérhetőségi probléma eldönthető
  - de exponenciális (hely) komplexitású általános esetben
- Míg az egyenlőségi probléma eldönthetősége általános esetben bizonyítottan nem lehetséges
  - feladat: két Petri háló (N, N') lehetséges tüzelési sorozatai azonosságának eldöntése

$$\boxed{L(N, M_0) \stackrel{?}{=} L(N', M_0')}$$

  - 1-korlátos (biztos) Petri hálók esetén exponenciális
    - processz algebra, biszimuláció

Petri hálók dinamikus (viselkedési) tulajdonságai

Dinamikus tulajdonságok

- Elérhetőséggel kapcsolatos tulajdonságok
  - Függenek a kiinduló állapottól (kezdő jelöléstől)
    - ⇒ **strukturális** tulajdonságok: kiinduló állapottól <u>függetlenek</u>
  - Nem csak elérhetőségi analízissel határozhatók meg
- Dinamikus tulajdonságok:
  - Korlátosság
  - Élő tulajdonság
    - Holtpontmentesség
  - Megfordíthatóság
  - Visszatérő állapot
  - Fedhetőség
  - Perzisztencia
  - Fair tulajdonság
    - Korlátozott fairség
    - Globális fairség

Dinamikus tulajdonságok: végesség

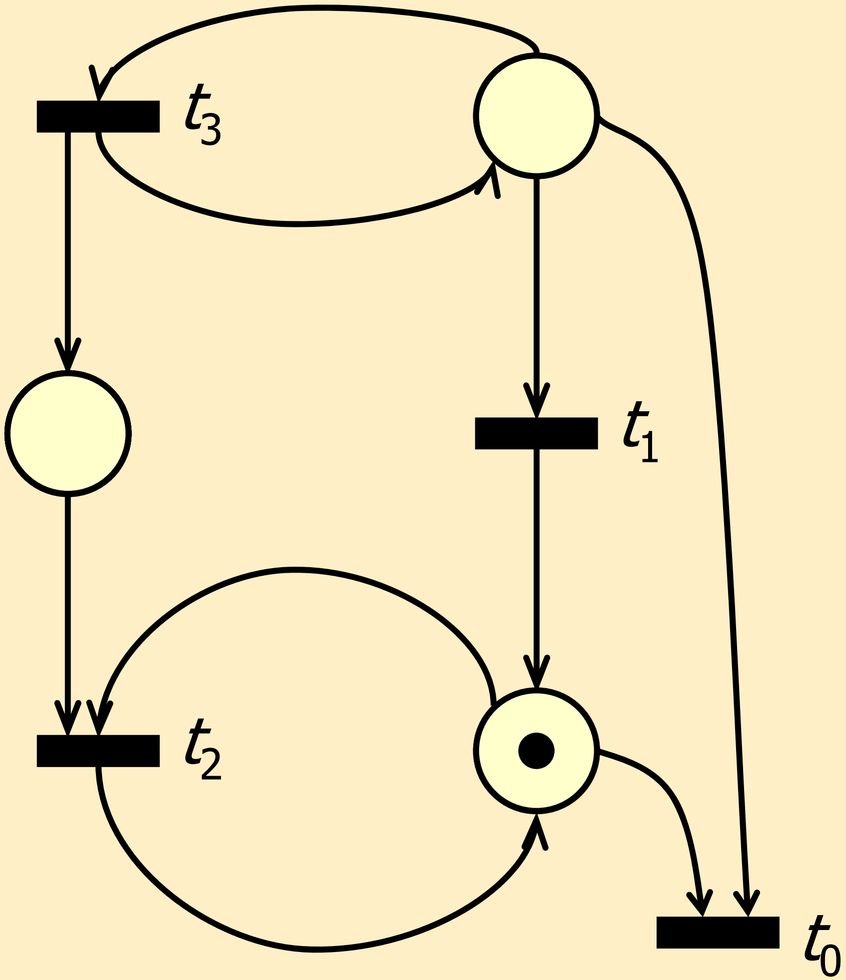
- *k*-korlátosság (korlátosság)
  - bármely állapotban minden helyen helyenként maximum *k* token lehet (M_0 kiinduló állapot függő!)
    - Biztos Petri háló**:** korlátosság speciális esete (k = 1)
  - végesség kifejezése
    - korlátosság $\Leftrightarrow$ véges állapottér
  - erőforrás használat, illetve általános értelemben vett feladatkezelés modellezése
    - (részben) konzervativitás $\overset{?}{\Rightarrow}$ korlátosság
  - a rendszerben a feladatok elvégzése garantált-e?

Dinamikus tulajdonságok: élőség

- Holtpont (deadlock) -mentesség
  - Minden állapotban legalább egy tranzíció tüzelhető
- Élő tulajdonság
  - Tranzíció egyszer/többször/végtelenszer tüzelhet-e?
  - Gyenge élő tulajdonságok egy t tranzícióra:
    - L_0-élő (halott): t sohasem tüzelhető egyetlen
    - L_1-élő: t legalább egyszer tüzelhető valamely
    - L_2-élő: bármely véges $k > 1$ egészre t legalább k–szor tüzelhető valamely
    - L_3-élő: t végtelen sokszor tüzelhető valamely

    $\vec{\sigma} \in L(N, M_0)$ állapot-trajektóriában
  - L_4-élő: t L_1-élő bármely $M_n \in R(N, M_0)$ állapotban

Élő tulajdonság: példa



- t_0 tranzíció: L_0-élő (halott)
- t_1 tranzíció: L_1-élő
- t_2 tranzíció: L_2-élő
- t_3 tranzíció: L_3-élő

Dinamikus tulajdonságok: élőség (folyt.)

- Egy (P, T, M_o) Petri háló L_x-élő
  - ha minden $t \in T$ tranzíció L_x-élő
  - L_4-től L_1-ig az élő tulajdonságok tartalmazzák egymást
- Egy (P, T, M_o) Petri háló élő
  - ha L_4-élő, azaz minden $t \in T$ tranzíció L_4-élő
  - Bejárási úttól függetlenül garantáltan holtpontmentes
    - köztes állapottól függetlenül minden tranzíció újra tüzelhető
    - holtpontmentesség $\Leftarrow$ élőség
  - Bizonyítása költséges lehet
    - szerencsés esetben nem (invariánsok!)
    - ideális rendszert tételez fel

Dinamikus tulajdonságok: ciklikusság

- Megfordíthatóság
  - A kezdőállapot bármely követő állapotból elérhető

$$\boxed{\forall M \in R(N, M_0) \Rightarrow M_0 \in R'(N, M)}$$

  - Gyakran ciklikus működésű hálózat
- Visszatérő állapot
  - Van olyan, a kezdőállapotból elérhető állapot, amely bármely őt követő állapotból elérhető

$$\boxed{\exists M_n \in R(N, M_0) : \forall M \in R'(N, M_n) \Rightarrow M_n \in R''(N, M)}$$

  - Gyakran ciklikus (rész)hálózat inicializáló szekvenciával

Dinamikus tulajdonságok: ciklikusság (folyt.)

- Fedhetőség
  - Létrejön-e korábbi működést magában foglaló állapot?
  - M' állapot fedi M állapotot, ha $M' \in R(N, M_0) \wedge M' \geq M$
    - M állapot fedhető M' állapottal
    - $M' \geq M$ jelentése: $\forall p \in P : \mathrm{m}'(p) \geq \mathrm{m}(p)$
      - gyenge fedhetőség esetén az azonos állapot is fed, ha elérhető
      - erős fedhetőség: $\exists p \in P : \mathrm{m}''(p) > \mathrm{m}(p)$
  - μ a t tranzíciót engedélyező minimális tokeneloszlás
    - t akkor és csak akkor nem L_1-élő, ha μ nem fedhető le
      - gyenge fedhetőség elég
    - fordítva: μ lefedhetősége garantálja t L_1-élő voltát

Dinamikus tulajdonságok: kölcsönhatás

- Perzisztencia
  - Két tetszőleges engedélyezett tranzíció közül egyik tüzelése sem tiltja le a másik engedélyezettségét
    - engedélyezett tranzíció engedélyezve marad tüzelésig!
  - Rendszerbeli funkcionális dekompozíció megmarad-e?
  - Párhuzamos működések befolyásolják-e egymást?
- Egy (P, T, M_o) Petri háló perzisztens, ha
  - bármely két $t_1, t_2 \in T$ tranzíciója az összes lehetséges tüzelési szekvenciában perzisztens

Dinamikus tulajdonságok: kölcsönhatás (folyt.)

- Fair tulajdonság
  - Párhuzamos folyamatok nem tartják-e fel egymást?
  - Valamennyi folyamat végbemegy-e (előbb-utóbb)?
- Nem egységes a fairség definíciója
  - Korlátozott fairség (B-fairség)
  - Globális fairség (korlátlan fairség)
- Egy tüzelési szekvencia korlátozottan fair (B-fair)
  - ha bármely tranzíció maximum korlátos sokszor tüzelhet anélkül, hogy egy másik tranzíció tüzelne
- Egy (P, T, M_0) Petri háló korlátozottan fair (B-fair)
  - ha minden $t \in T$ tranzíció az összes lehetséges tüzelési szekvenciában korlátozottan fair (B-fair)

Dinamikus tulajdonságok: kölcsönhatás (folyt.)

- Globális fairség
  - Egy tüzelési szekvencia globálisan fair, ha
    - véges, vagy
    - az összes tranzíció végtelen sokszor szerepel benne
  - Egy (P, T, M_0) Petri háló globálisan (korlátlanul) fair
    - ha a háló összes lehetséges tüzelési szekvenciája globálisan (korlátlanul) fair

Állapottér reprezentációk:
az elérhetőségi és fedési gráf

Állapottér reprezentációk: elérhetőségi gráf

- Elérhetőségi gráf
  - M_0 kezdőállapotból induló állapotgráf
    - csomópontok: állapotok → címkézés: tokeneloszlások
    - állapotátmenetek: irányított élek → címkézés: tüzelések
    - legfeljebb annyi új csomópont, ahány engedélyezett tranzíció
      - kevesebb, ha prioritásos a Petri háló
    - csomópont, amiből nem indul ki él: holtpont
  - Nem korlátos a Petri háló → végtelen sok állapot
    - korlátosság ⇔ véges állapottér
  - Szélességi típusú bejárás az állapotból tüzelések mentén
    - mélységi bejárás nem korlátos állapottérben rossz ötlet…
    - viszont a részleges sorrendezési redukció mélységi kereséssel

Állapottér reprezentációk: fedési gráf

- Petri háló mérete/bonyolultsága és az állapottér mérete közti korreláció?
  - pl. megállási probléma
- Végesség hiányát kezelni kell
- Fedési gráf: végtelen állapottér esetére is
  - Hasonló felépítés: M_0 kezdőállapot, élek: tüzelések
  - Kritikus részek: token „túlszaporodás”
    - trajektória: $M_0 \ldots M'' \ldots M, M'$ és $M'' \leq M' \rightarrow$ fedett állapotok!
    - $p \in P : \mathrm{m}'(p) > \mathrm{m}''(p) \rightarrow$ fedett helyek (erős fedhetőség)
    - fedett helyekre speciális szimbólum: ω a végtelenség kifejezője

Fedési fa generáló algoritmus

$L_{\text{vizsgálandó}} \leftarrow \{ M_0 \}$

`MAIN:` **`if`** $L_{\text{vizsgálandó}} \neq \emptyset$

A következő $M \in L_{\text{vizsgálandó}}$ állapot kiválasztása

`if` *M* a gyökértől idáig vezető úton már szerepelt

`then` *M* -et „régi állapotként” jelöljük

`goto` `MAIN` // ciklus

`if` *M*-ben nincs engedélyezett tüzelés

`then` *M* -et „végállapotként” (halott állapot) jelöljük

`goto` `MAIN` // ciklus

Fedési fa generáló algoritmus (folyt.)

else // (van M -ben engedélyezett tranzíció)

for all t engedélyezett tranzícióra:

Az M' rákövetkező állapot meghatározása: $M\left[\vec{e}_t > M'\right.$

if létezik az M_0-tól M -ig vezető úton olyan M'', amelyet M' fed

$$\boxed{M' \neq M'' \wedge \forall p \in P : \mathrm{m}'(p) \geq \mathrm{m}''(p) \wedge \exists p \in P : \mathrm{m}'(p) > \mathrm{m}''(p)}$$

then M'' fedett állapot:

az M' állapotot jelölő tokeneloszlásban
a fedett helyek jelöléseit ω-val helyettesítjük

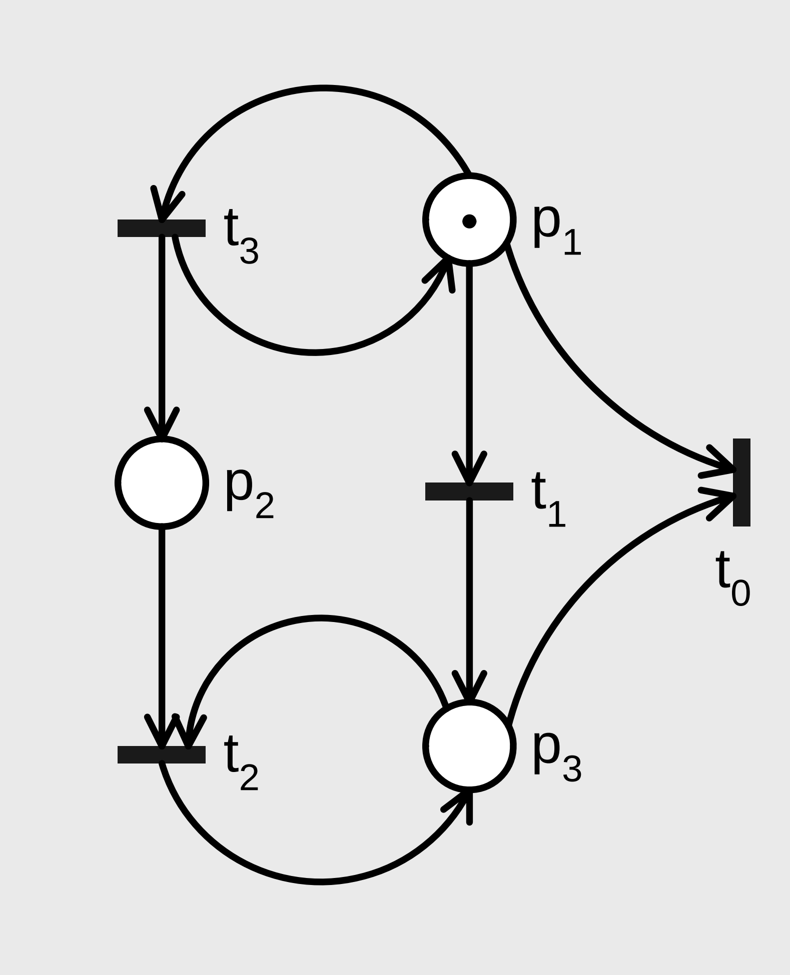
$$\boxed{\forall p \in P : \mathrm{m}'(p) > \mathrm{m}''(p) \rightarrow \mathrm{m}'(p) = \omega}$$

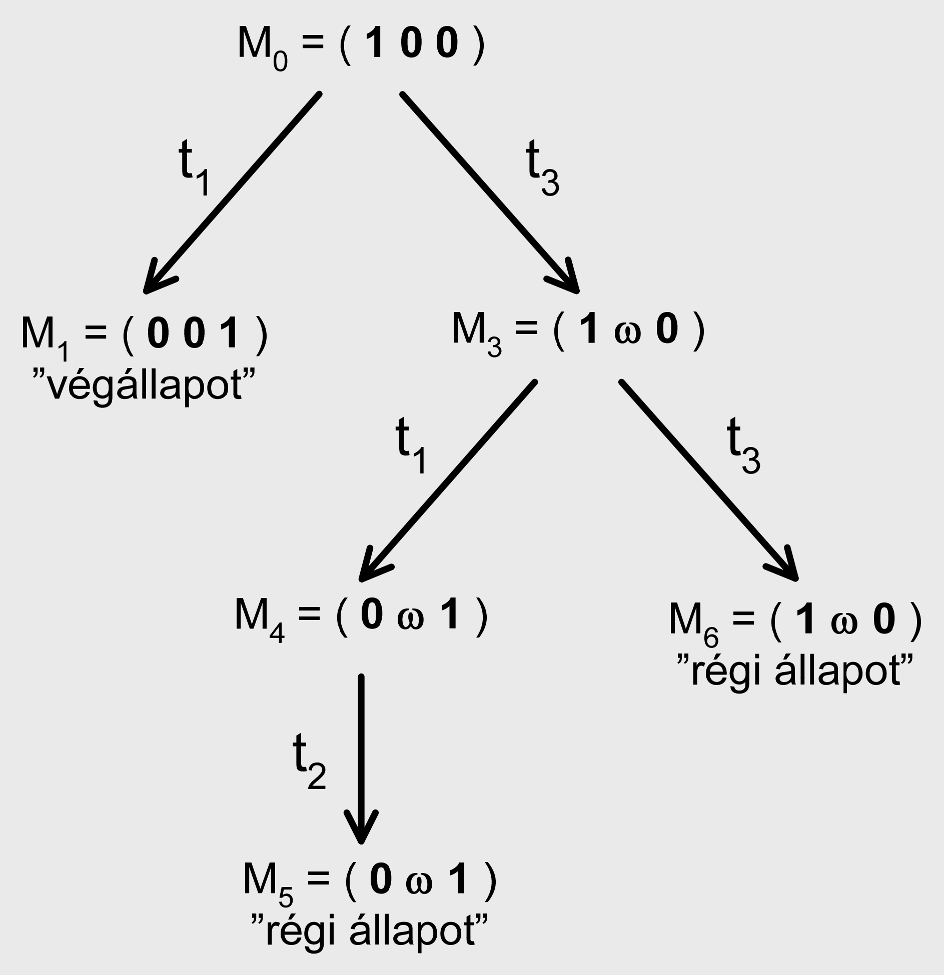
else M' új állapot: $L_{\text{vizsgálandó}} \leftarrow L_{\text{vizsgálandó}} \cup M'$

M -ből M'-höz egy t -vel jelölt élet húzunk

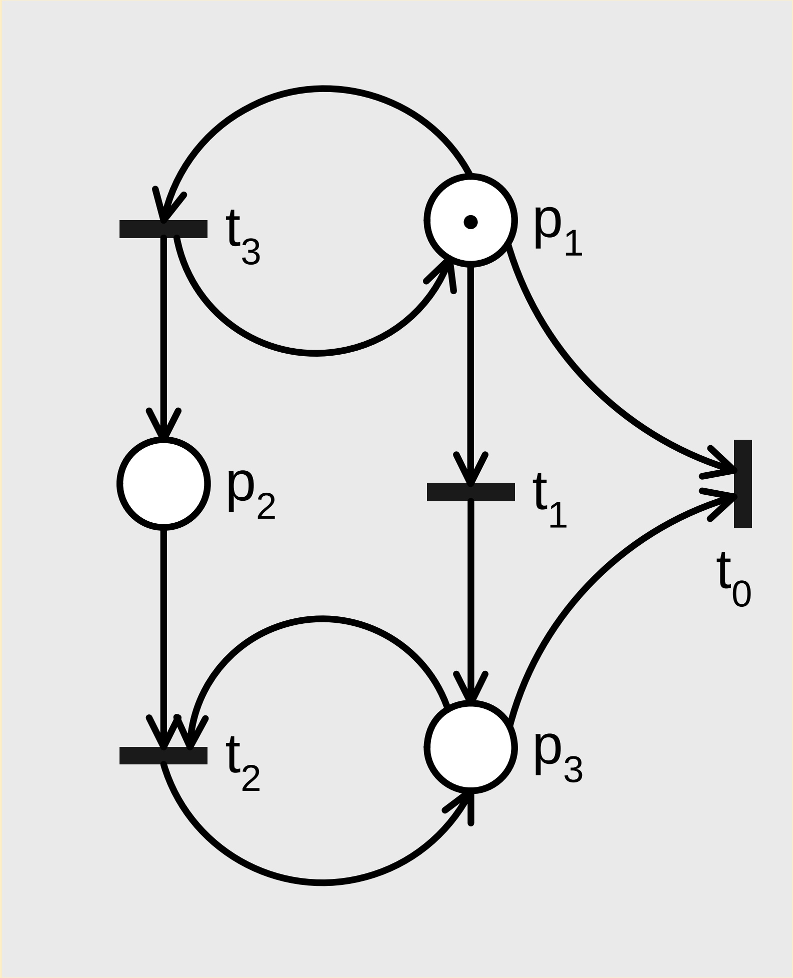
goto `MAIN` // ciklus

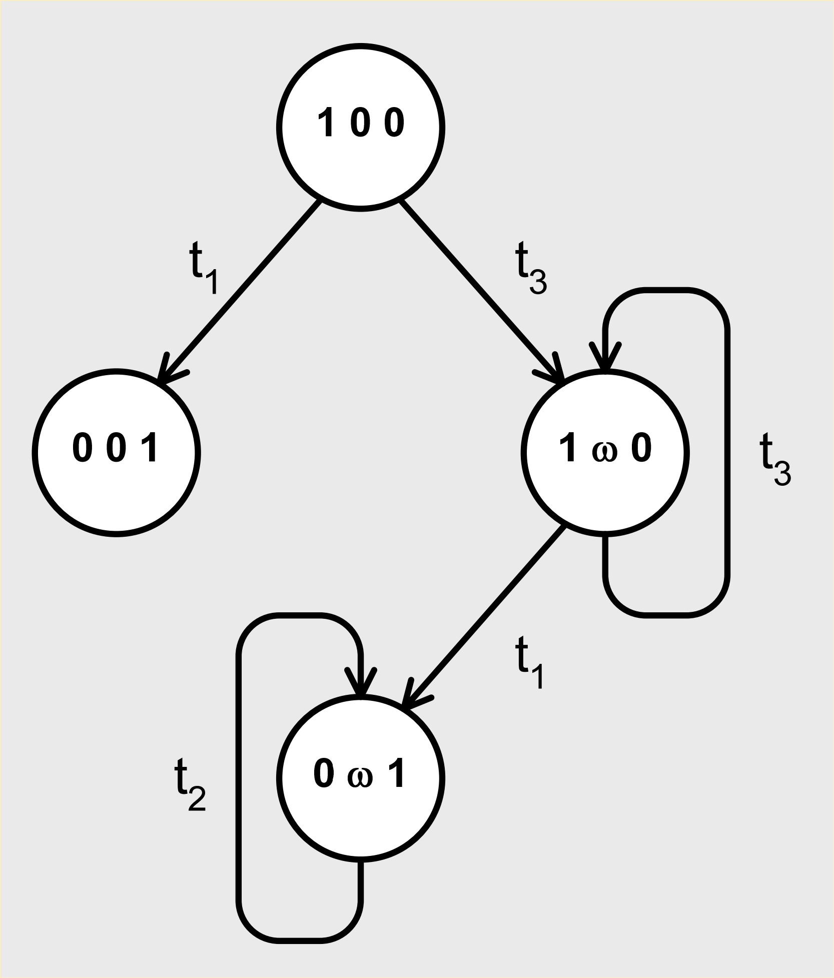
A példa és annak fedési fája





A példa és annak fedési gráfja





Petri hálók fedési fájának analízise

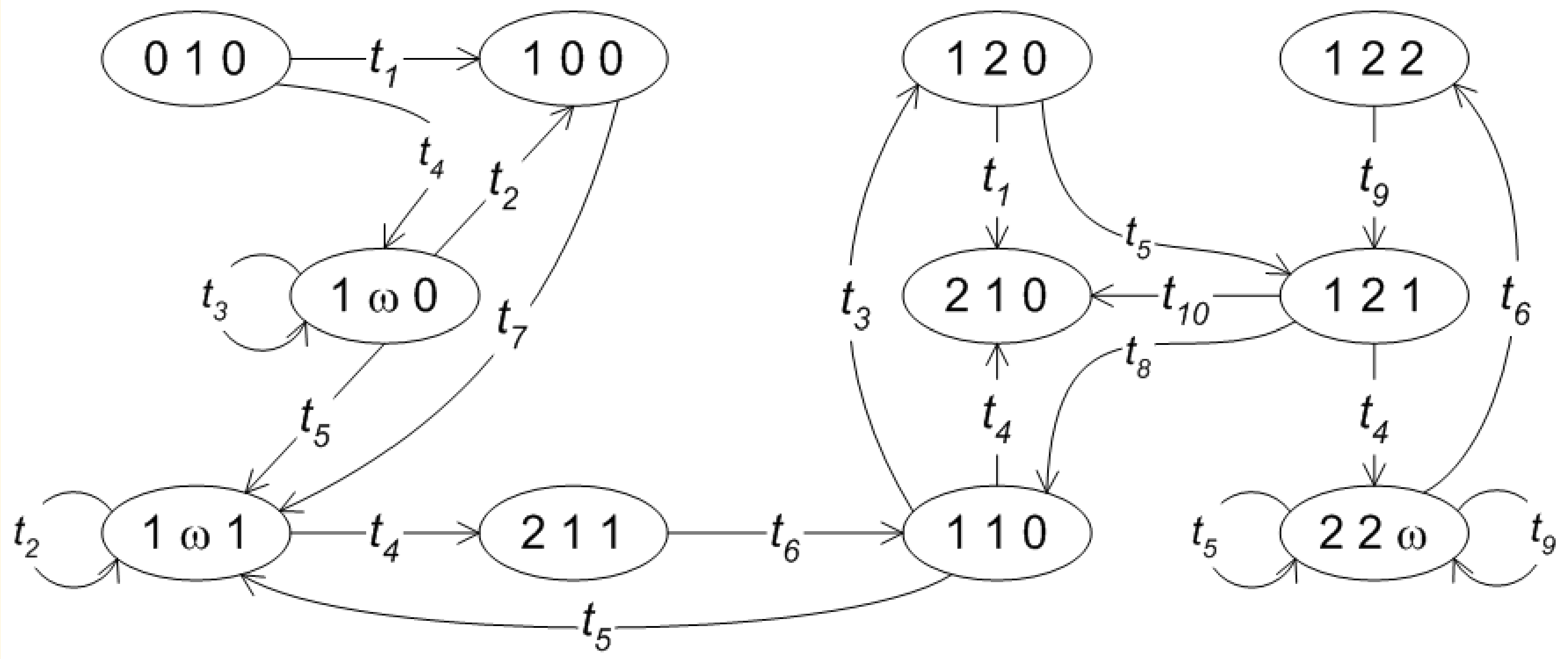
Közvetlenül is leolvasható tulajdonságok:

- Petri háló korlátos $\Leftrightarrow$ $R(N, M_0)$ elérhetőségi gráfja véges $\Leftrightarrow$ Fedési fában ω nem jelenik meg címkeként
- Petri háló biztonságos $\Leftrightarrow$ Csak 0 és 1 jelenik meg csomópont címkeként a fedési fában
- Petri háló egy tranzíciója halott $\Leftrightarrow$ tranzícióhoz tartozó tüzelés nem jelenik meg élcímkeként a fedési fában

Dinamikus tulajdonságok vizsgálata az állapotérben

Tipikus feladat

Az ábra egy Petri háló állapotterét mutatja be fedési gráf alakban. A hálóban 10 darab tranzíció található, amelyeket t_1, ..., t_{10} címkékkel jelölünk. Az állapotokat a token eloszlás vektorral címkéztük meg, tehát 0 1 0 jelentése: $m(p_1) = 0$, $m(p_2) = 1$ és $m(p_3) = 0$.



Tipikus kérdések

1. A Petri háló élő?
2. A háló (deadlock) holtpontmentes?
3. t_6 tranzíció L_3-élő?
4. t_7 tranzíció L_2-élő?
5. A (2 2 1) állapot fedhető?
6. A (2 1 0) állapot fedhető?
7. A háló perzisztens?
8. A háló korlátos?
9. A háló megfordítható?
10. A hálóban létezik visszatérő állapot?
11. t_4 és t_6 tranzíció korlátos fair tulajdonságú?
12. t_5 és t_8 tranzíció korlátos fair tulajdonságú?
13. Létezik P-invariáns?
14. Létezik T-invariáns?

Élőség vizsgálata az állapottérben

- Ellenpéldát találni (szinte mindig) a leggyorsabb és legegyszerűbb megoldás!
- Élőség:
  - L_4-élőség
    - végig kell nézni, hogy mindig tüzelhetővé válik-e?
    - minden trajektóriára teljesülnie kell!
  - L_3-élőség (és a többi)
    - elég egy trajektóriát találni, ahol teljesül!
  - A háló akkor élő, ha minden tranzíciója élő!
    - ha találunk akár egy tranzíció esetében ellenpéldát $\rightarrow$ nem élő
  - Ha holtpontmentes, akkor még nem biztos, hogy élő is!

Korlátosság vizsgálata az állapottérben

- Korlátosság:
  - lásd a fedési gráfnál tanultakat!
    - „Petri háló korlátos $\Leftrightarrow$ $R(N, M_0)$ elérhetőségi gráfja véges $\Leftrightarrow$ fedési fában ω nem jelenik meg állapot címkében”
  - biztosság:
    - „Petri háló biztos $\Leftrightarrow$ csak 0 és 1 jelenik meg állapot címkében a fedési fában”

További dinamikus tulajdonságok

- Megfordíthatóság:
  - Az elérhetőségi gráf egyetlen erősen összekötött komponens?
- Visszatérő állapot:
  - Van az elérhetőségi gráfban erősen összekötött komponens?
  - Az adott állapot része ennek?
- Fairség:
  - „Az egyik tranzíció korlátos sokszor tüzelhet, mielőtt a másik tüzelne”
  - Van-e olyan ciklus, amiben az egyik tranzíció benne van és a másik nincs?
    - →van: ellenpélda, hiszen létezik tüzelési szekvencia, amiben korlátlan sokszor tüzel
- Perzisztencia:
  - „A tranzíció mindaddig engedélyezett marad, amíg nem tüzel”
    - →ha több engedélyezett és nem ő tüzelt, akkor a következő állapotban is engedélyezett marad
    - →ha engedélyezett maradt, akkor meg is jelenik élcímkeként (prioritás?)

Petri hálók redukciós módszerei

Elérhetőségi probléma kezelése

- Struktúra redukálása
  - Tulajdonságmegtartó transzformáció redukált modellre
- Hierarchikus modellezés
  - Részhálózatok összevonása egyetlen csomóponttá
  - Petri hálók nemdeterminizmusa $\rightarrow$ modellabsztrakció
    - Keresési tér behatárolása durvább modellen
    - Részletes analízis egy finomított modellen
- Kompozícionális verifikáció
  - Rendszerek $\Leftarrow$ részrendszerek + interfészek + együttműködés
  - Részrendszerek analízise és az együttműködések vizsgálata
  - A teljes rendszer analízise a részrendszerekre kapott eredményekből

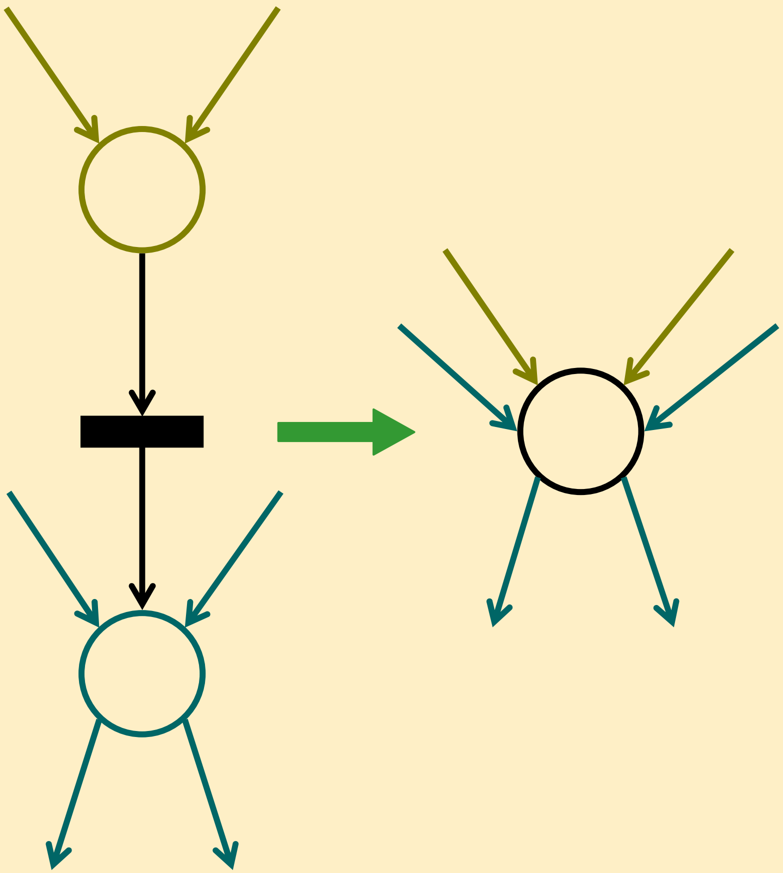
Elérhetőségi probléma egyszerűsítése

- Redukció:
  - Érthető modellből kompakt modell
    - redundancia eliminálása
  - További egyszerűsítés: modell kifejezőereje csökken
    - ellenőrzött változtatások, de a funkcionalitás megváltozik
    - a kiválasztott tulajdonságokat megőrzi!
    - eredeti modellt a tulajdonságok szerint „fedő” modell jön létre
  - Sokféle tulajdonságmegőrző transzformáció létezik

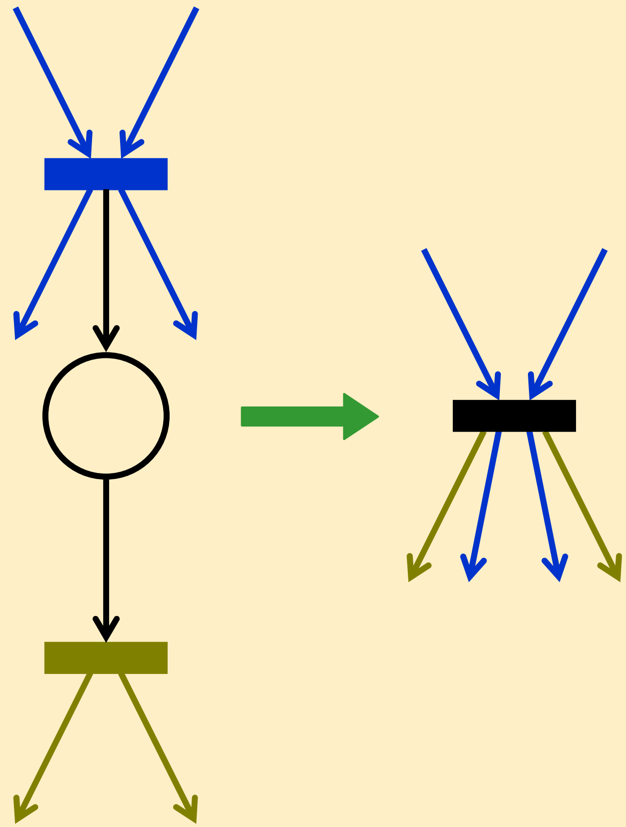
Transzformációk

- Egyszerű tulajdonságmegőrző transzformációk:
  - soros helyek összevonása
  - soros tranzíciók összevonása
  - párhuzamos helyek összevonása
  - párhuzamos tranzíciók összevonása
  - önhurkot alkotó helyek törlése
  - önhurkot alkotó tranzíciók törlése
- Megőrzik az élő, korlátos és biztos tulajdonságot

Soros összevonások

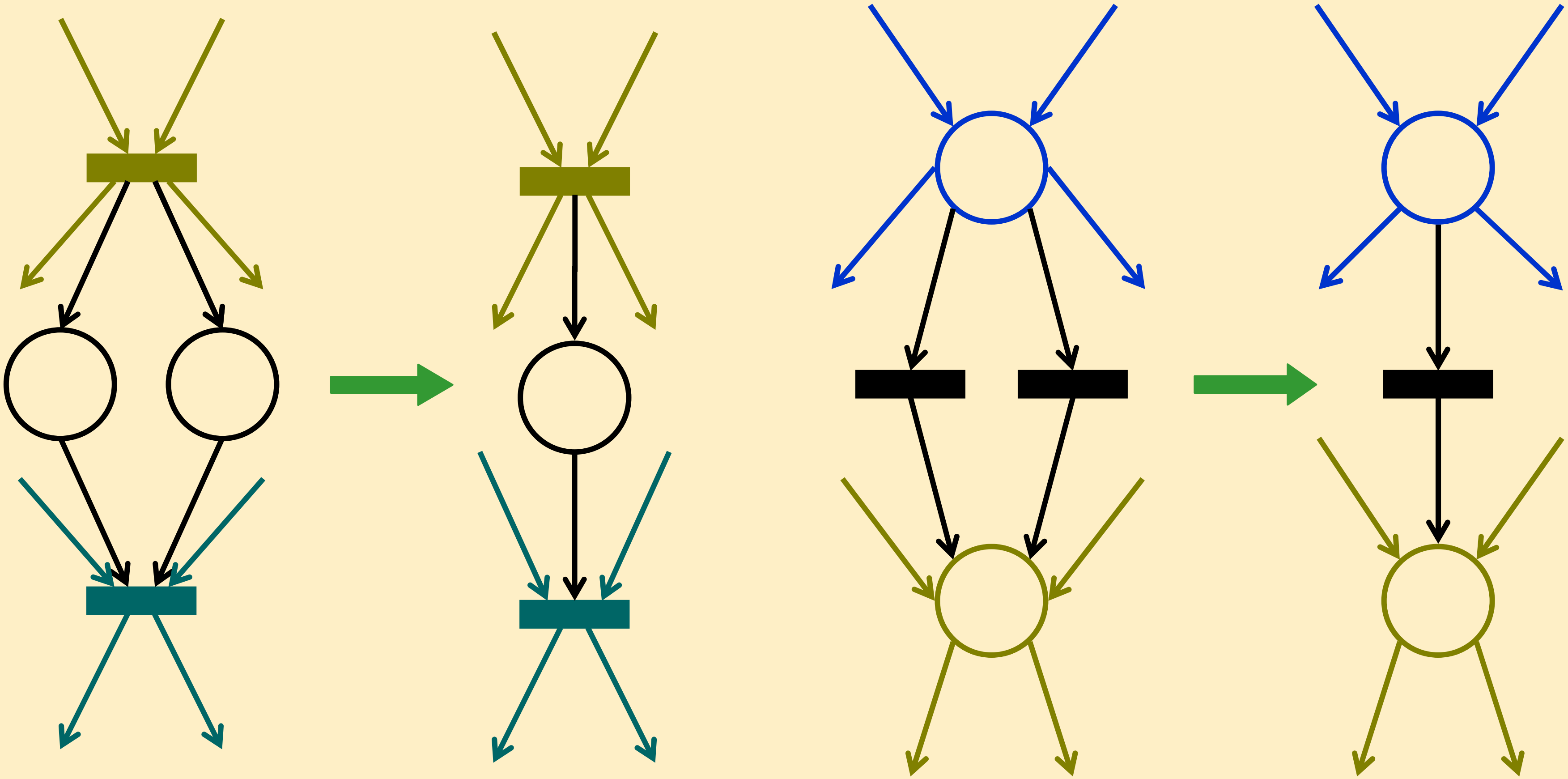


soros helyek összevonása (FSP)



soros tranzíciók összevonása (FST)

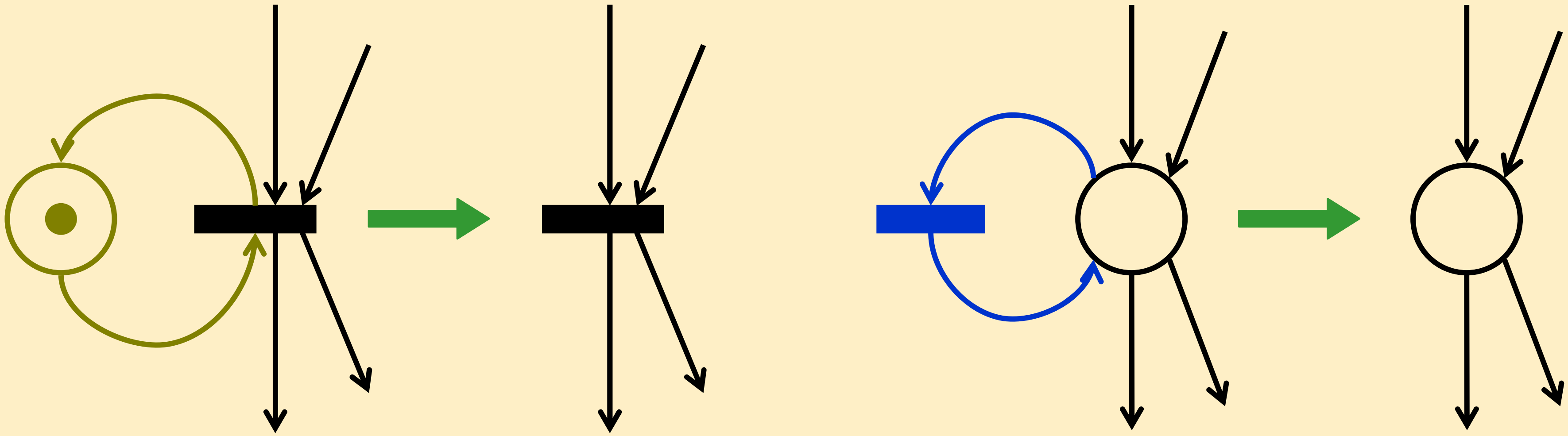
Párhuzamos összevonások



párhuzamos helyek
összevonása (FPP)

párhuzamos tranzíciók
összevonása (FPT)

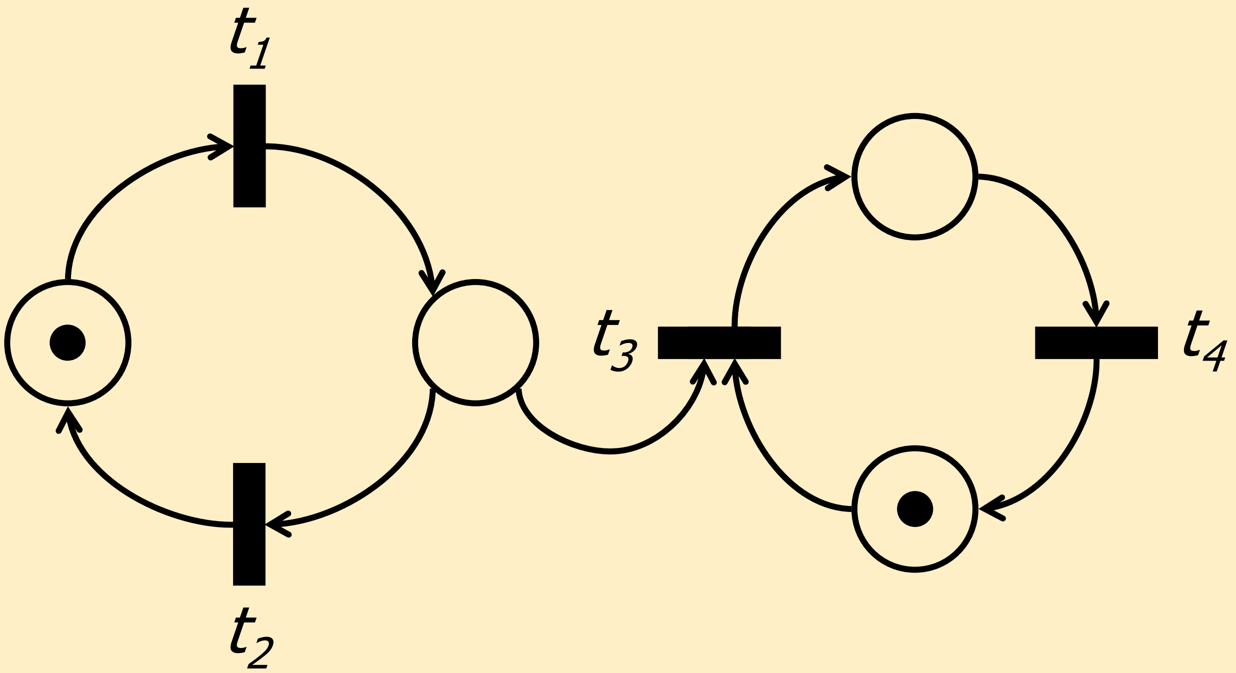
Önhurkok törlése



önhurkot alkotó helyek
törlése (ESP)

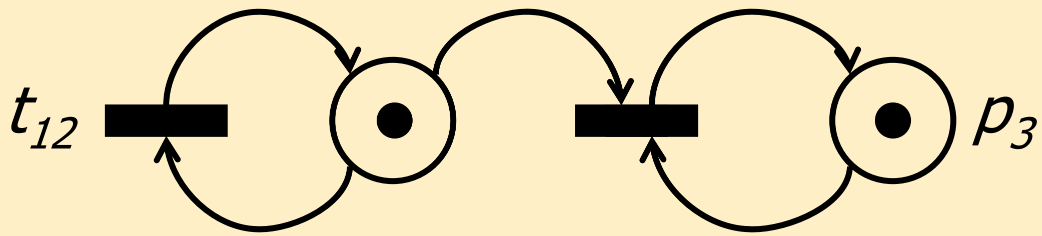
önhurkot alkotó tranzíciók
törlése (EST)

Példa



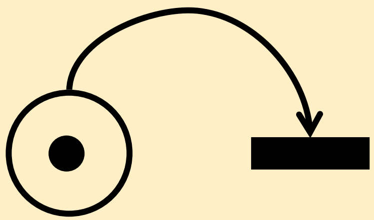
- t_1 tüzelése
- t_1 és t_2 összevonása (soros tranzíciók) $\rightarrow$ t_{12}
- t_3 és t_4 összevonása (soros tranzíciók) $\rightarrow$ t_{34}

Példa: 2. lépés



- t_{12} törlése (önhurkot alkotó tranzíció)
- p_3 törlése (önhurkot alkotó hely)

Példa: eredmény



a példa korlátos, de nem élő (és nem megfordítható)

Egyszerű struktúrájú Petri hálók viselkedése

Konfliktus, konkurencia

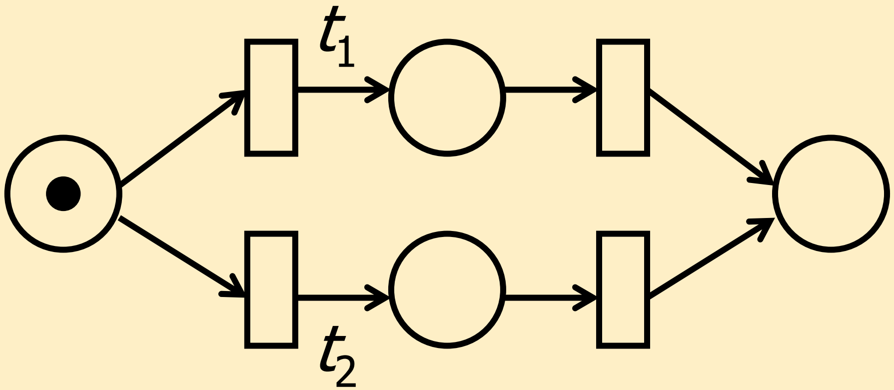
Konfliktus: kizáró események, választás

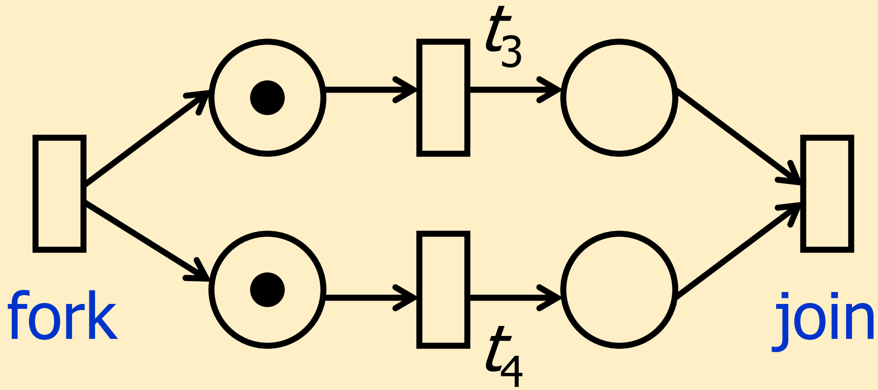
e_1 vagy e_2 esemény következik be, de csak az egyik

Konkurencia: párhuzamos események

e_1, e_2 esemény kauzálisan független: $e_1 \not\to e_2 \wedge e_2 \not\to e_1$

- reflexív: e_1 co e_1 és e_2 co e_2
- szimmetrikus: e_1 co $e_2 \Rightarrow e_2$ co e_1
- nem tranzitív: e_1 co $e_2 \wedge e_2$ co $e_3 \not\Rightarrow e_1$ co e_3

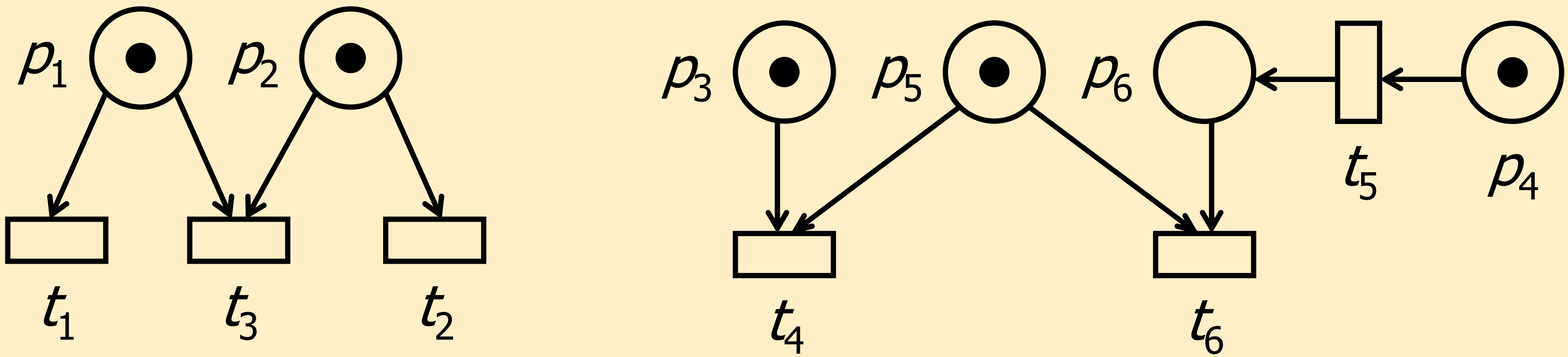




Konfúzió

Konfúzió: konkurencia és konfliktus is jelen van

- Szimmetrikus: egyaránt konkurens és konfliktusos
  - t_1 és t_2 konkurens, mindkettő konfliktusban van t_3 átmenettel
- Aszimmetrikus: tüzelési szekvenciától függően
  - t_4 és t_5 konkurens, de ha t_5 tüzel előbb, akkor t_4 konfliktusba kerül t_6 átmenettel



Petri háló alosztályok

A továbbiakban végig **közönséges Petri hálókról** beszélünk!

Alosztályok

- Állapotgép (State Machine, SM)
  - minden átmenetnek pontosan egy be- és kimeneti helye

    $$\boxed{\forall t \in T : \left|\bullet t\right| = \left|t \bullet\right| = 1}$$

  - van konfliktus, nincs szinkronizáció
- Jelölt gráf (Marked Graph, MG)
  - minden helynek pontosan egy be- és kimeneti tüzelése

    $$\boxed{\forall p \in P : \left|\bullet p\right| = \left|p \bullet\right| = 1}$$

  - van szinkronizáció, nincs konfliktus

Alosztályok (folyt.)

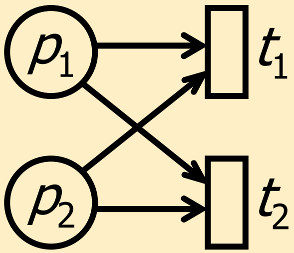
- Szabad választású háló (Free-Choice Net, FC)
  - közönséges Petri háló, melyben minden helyből kifelé mutató él vagy egyedüli kimenő él, vagy egyedüli bemenő él egy átmenetbe

$$\boxed{\begin{array}{l}\forall p \in P : \left| p\bullet \right| \geq 1 \Rightarrow \bullet\left(p\bullet \right) = \left\{ p \right\}, \\ \forall p_1, p_2 \in P : p_1\bullet \cap p_2\bullet \neq \varnothing \Rightarrow \left| p_1\bullet \right| = \left| p_2\bullet \right| = 1\end{array}}$$

  - lehet konkurencia és konfliktus is, de nem egyszerre
    - nincs benne konfúzió
  - dekomponálható MG és SM komponensekre

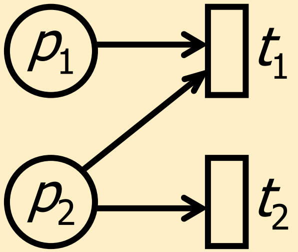
Alosztályok (folyt.)

- Kiterjesztett szabad választású háló (EFC)
  - többszörös szinkronizáció is lehetséges



$$\boxed{\forall p_1, p_2 \in P : p_1 \bullet \cap p_2 \bullet \neq \varnothing \Rightarrow p_1 \bullet = p_2 \bullet}$$

- Aszimmetrikus választású háló (AC)
  - lehetséges az (aszimmetrikus) konfúzió is

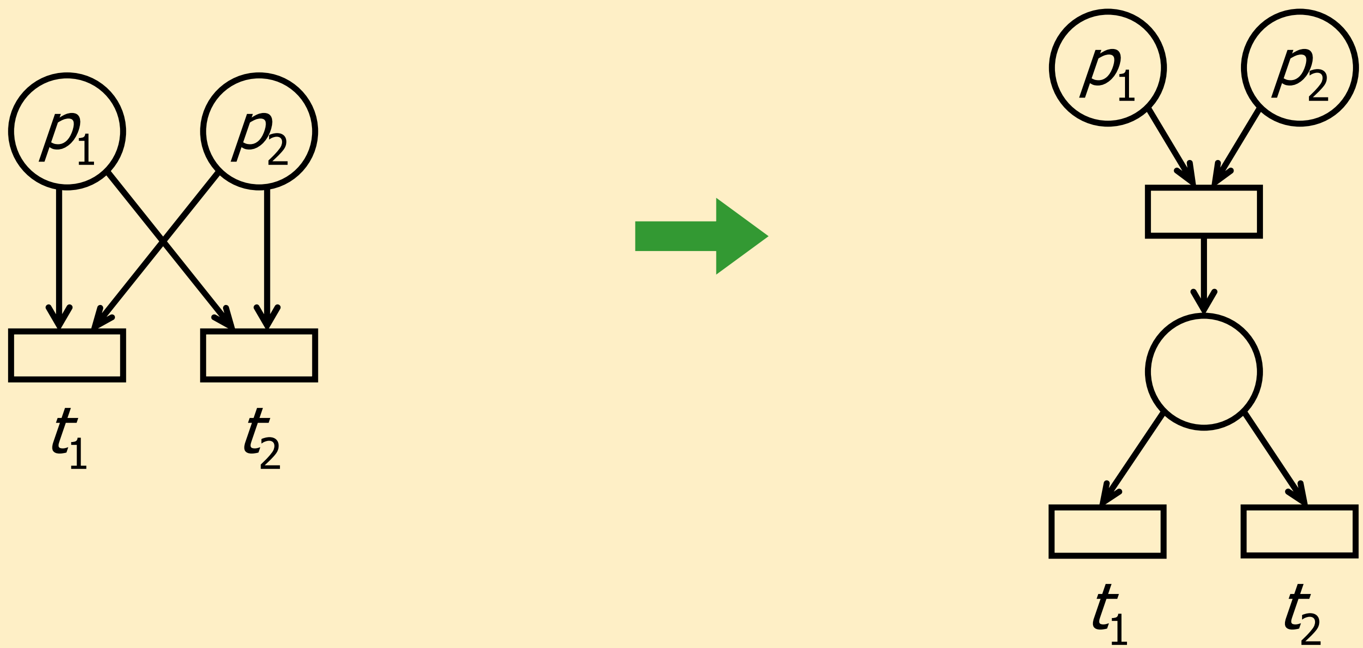


$$\boxed{\forall p_1, p_2 \in P : p_1 \bullet \cap p_2 \bullet \neq \varnothing \Rightarrow p_1 \bullet \subseteq p_2 \bullet \vee p_2 \bullet \subseteq p_1 \bullet}$$

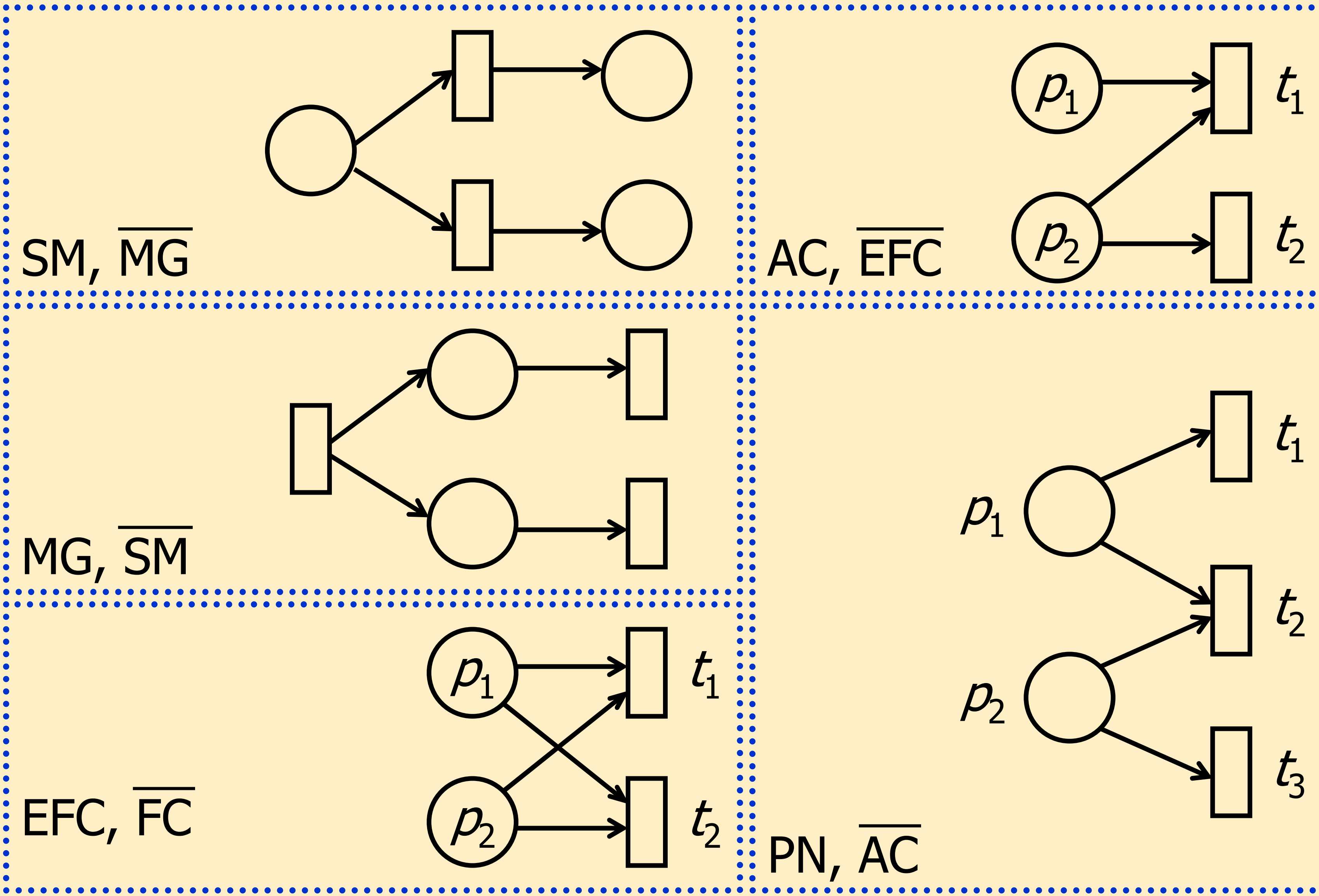
- Nincs szimmetrikus választású $\rightarrow$ a többi PN

EFC transzformációja FC-re

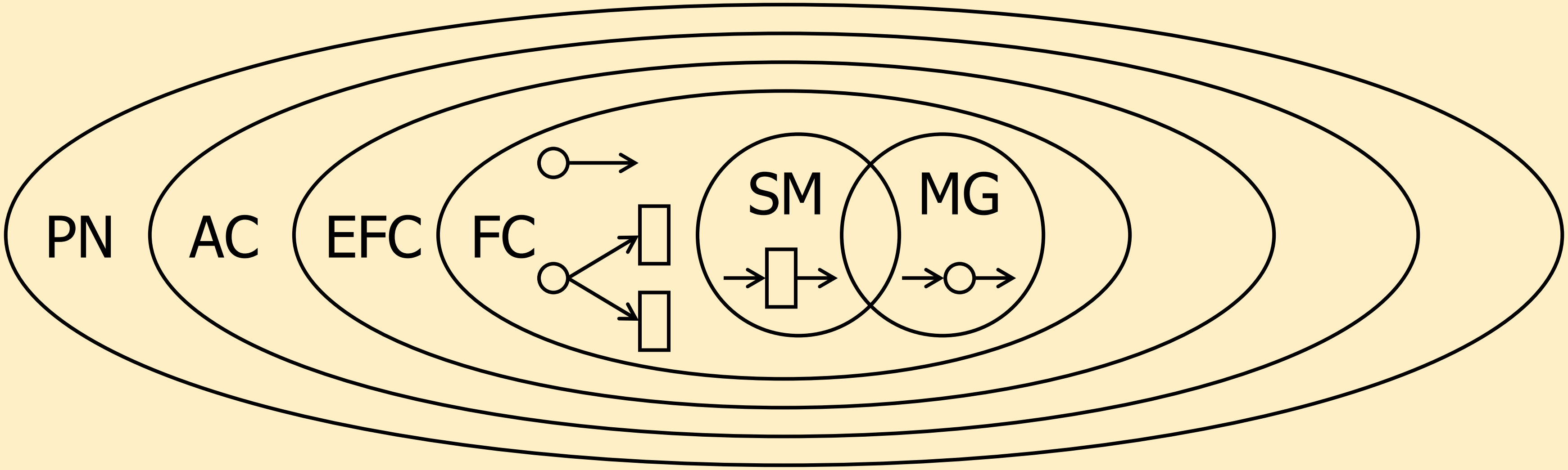
- FC és EFC közös tulajdonsága:
  - ha t_1 és t_2 $\exists$ közös bemeneti helye, akkor nincs olyan állapot, melyben az egyik engedélyezett és a másik nem
  - EFC transzformálható tulajdonságmegtartó módon FC-re



Alapstruktúrák



Az alosztályok viszonya



- Állapotgép (SM) nem enged meg szinkronizációt
- Jelölt gráf (MG) nem enged meg választást
- Szabad választású háló (FC) nem enged meg konfúziót
- Aszimmetrikus vál. háló (AC) megenged aszimmetrikus konfúziót, de nem enged meg szimmetrikus konfúziót

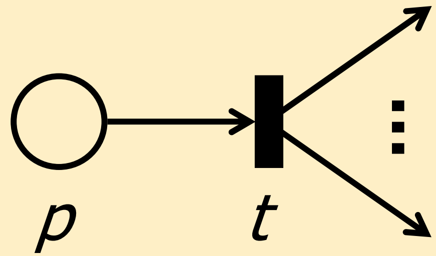
Élő és bizt(onság)os tulajdonság kritériumai

Szükséges feltétel általános esetben:

- Ha egy (N, M_0) Petri háló élő és biztos, akkor N erősen (szigorúan) összekötött gráf topológia
  - bármely csomópontból bármely másikba vezet irányított út
  - sem forrás, sem nyelő csomópontok: $\forall n \in P \cup T : \bullet n \neq \emptyset \neq n \bullet$
- Elégséges feltételek alosztályokra fogalmazhatók meg

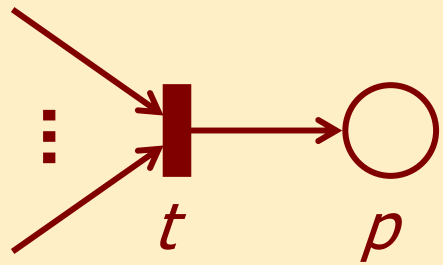
Miért nem lehetnek forrás és nyelő csomópontok?

$\bullet p = \emptyset$
(forrás hely)



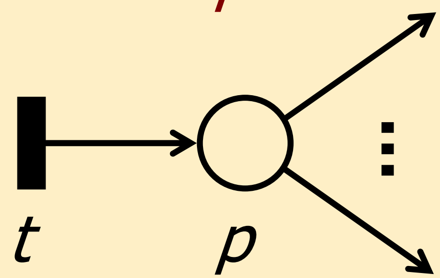
t nem élő

$p\bullet = \emptyset$
(nyelő hely)



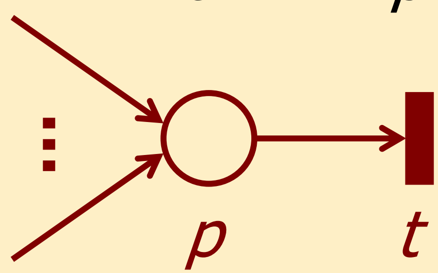
p nem biztos, ha *t* élő

$\bullet t = \emptyset$
(forrás tranzíció)



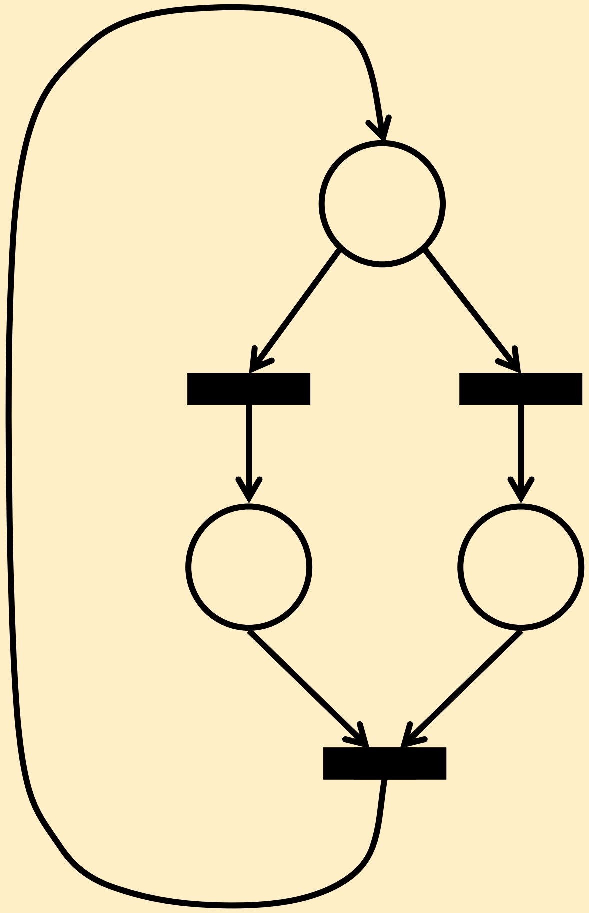
p nem biztos

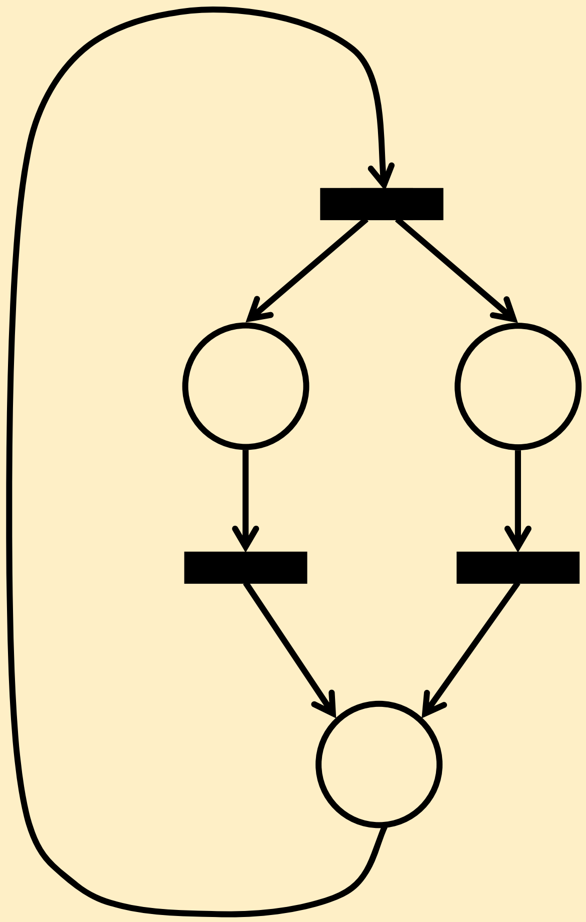
$t\bullet = \emptyset$
(nyelő tranzíció)



t nem élő, ha *p* biztos

Miért nem elégséges az összekötöttség?





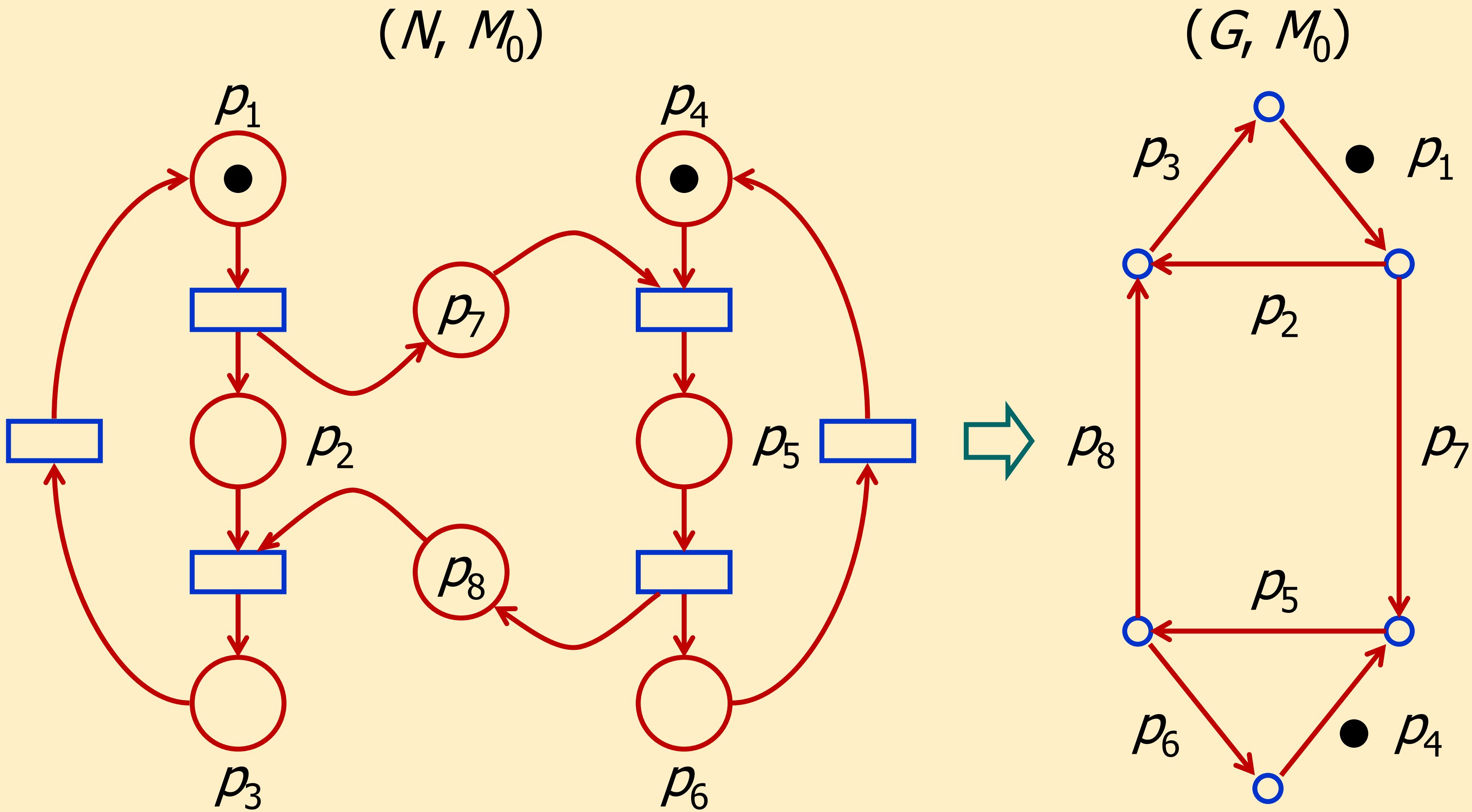
nincs élő jelölése

nincs nemüres biztos jelölése

Élő és biztos tulajdonság az SM hálókban

1. Egy (N, M_0) állapotgép a.cs.a. **élő**, ha N erősen összefüggő és M_0-ban van legalább egy token
   - triviális, hiszen minden tüzelés csak egy tokent mozgat
2. Egy (N, M_0) állapotgép a.cs.a. **biztos**, ha M_0-ban van legfeljebb egy token
3. Egy élő (N, M_0) állapotgép a.cs.a. **biztos**, ha M_0-ban pontosan egy token van

Jelölt gráf reprezentáció



Élő tulajdonság az MG hálókban

4. Egy (G, M_0) jelölt gráfban a tokenek száma minden C irányított körben állandó

$$\boxed{\forall M \in R(N, M_0), \forall C \subseteq G : M(C) = M_0(C)}$$

   - közönséges Petri háló: egyszeres élek; minden csomóponthoz a körben egy bemenő és egy kimenő él

5. Egy (G, M_0) jelölt gráf a.cs.a. **élő**, ha M_0 állapotban minden G-beli irányított körben van legalább egy token

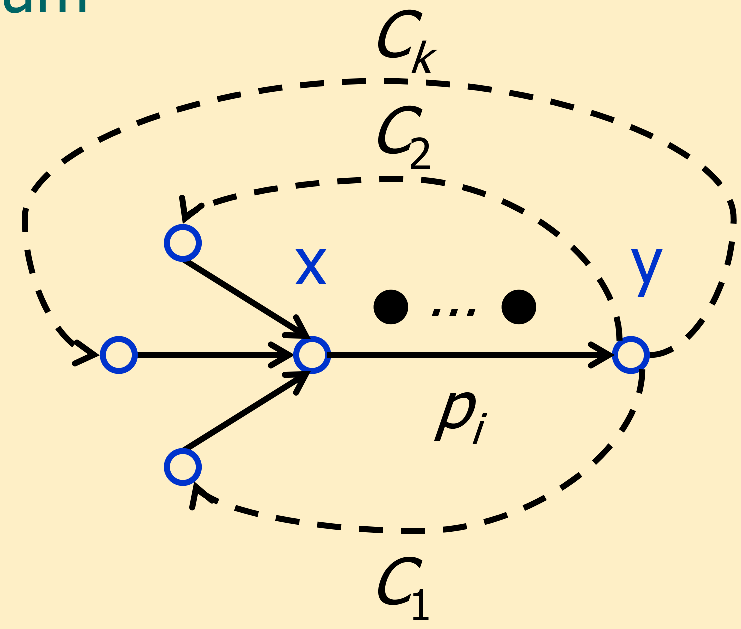
Biztos tulajdonság az MG hálókban

6. Egy (G, M_0) jelölt gráfban egy élt jelölő tokenek maximális száma egyenlő az élt tartalmazó bármely irányított körön az M_0 állapotban levő tokenek minimális számával

- y elsütése nélkül x-et maximum

$$\min_{\forall C_j \subseteq G | p_i \in C_j} \left(M_0(C_j) \right)$$

alkalommal lehet elsütni



Biztos tulajdonság az MG hálókban

7. Egy élő (G, M_0) jelölt gráf a.cs.a. **biztos**, ha minden él (hely) olyan C irányított körben van, amelyre $M_0(C) = 1$
8. Egy G irányított gráfban a.cs.a. létezik **élő** és **biztos** jelölt gráfot létrehozó M_0 állapot, ha G erősen összefüggő gráf
   - a feltétel triviálisan szükséges
   - elégséges is, hiszen van legalább egy irányított kör, és minden irányított körbe elég egy tokent tenni

Biztos tulajdonság az MG hálókban

- Visszacsatoló élhalmaz (Feedback Arc Set, FAS)
  - Egy E′ élhalmaz visszacsatoló élhalmaz, ha elhagyásával a G erősen összefüggő gráf irányított kör mentessé válik, azaz a $G' = (V, E - E')$ körmentes
    - minimális FAS: egyetlen valódi részhalmaza sem FAS
    - minimum FAS: egyetlen más FAS sem tartalmaz kevesebb élt

9. Egy erősen összekötött élő (G, M_0) jelölt gráf a.cs.a. **biztos**, ha az M_0 kezdőállapotból elérhető minden $M \in R(G, M_0)$ állapotban a jelölt élek halmaza minimális visszacsatoló élhalmaz

Szifon és csapda

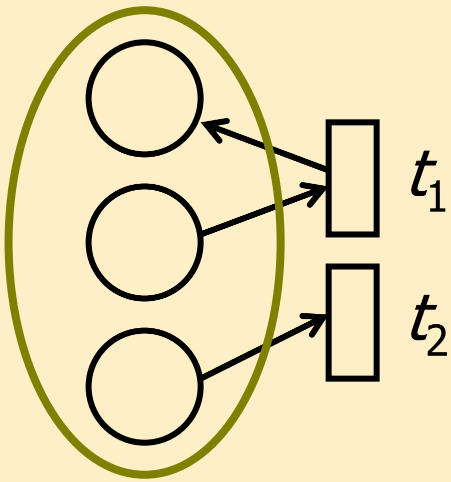
Egy S helyhalmaz **szifon**, ha $\bullet S \subseteq S\bullet$

- minden S-beli kimeneti helyhez bemeneti hely S-ben
- egy állapotban tokenmentes $\rightarrow$ követő állapotokban is

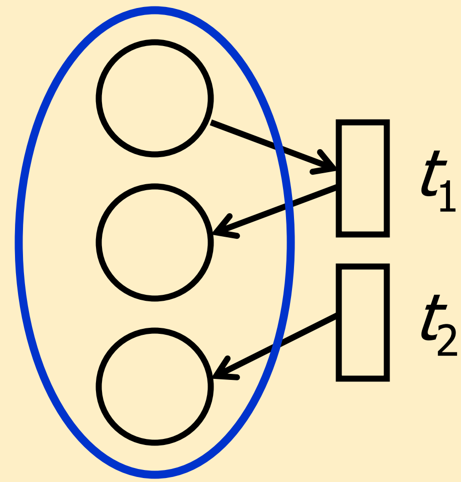
Egy Q helyhalmaz **csapda**, ha $Q\bullet \subseteq \bullet Q$

- minden Q-beli bemeneti helyhez kimeneti hely Q-ban
- egy állapotban jelölt $\rightarrow$ követő állapotokban is

$\bullet S = \{t_1\}$
$S\bullet = \{t_1, t_2\}$
$\bullet S \subseteq S\bullet$



$\bullet Q = \{t_1, t_2\}$
$Q\bullet = \{t_1\}$
$\bullet Q \supseteq Q\bullet$



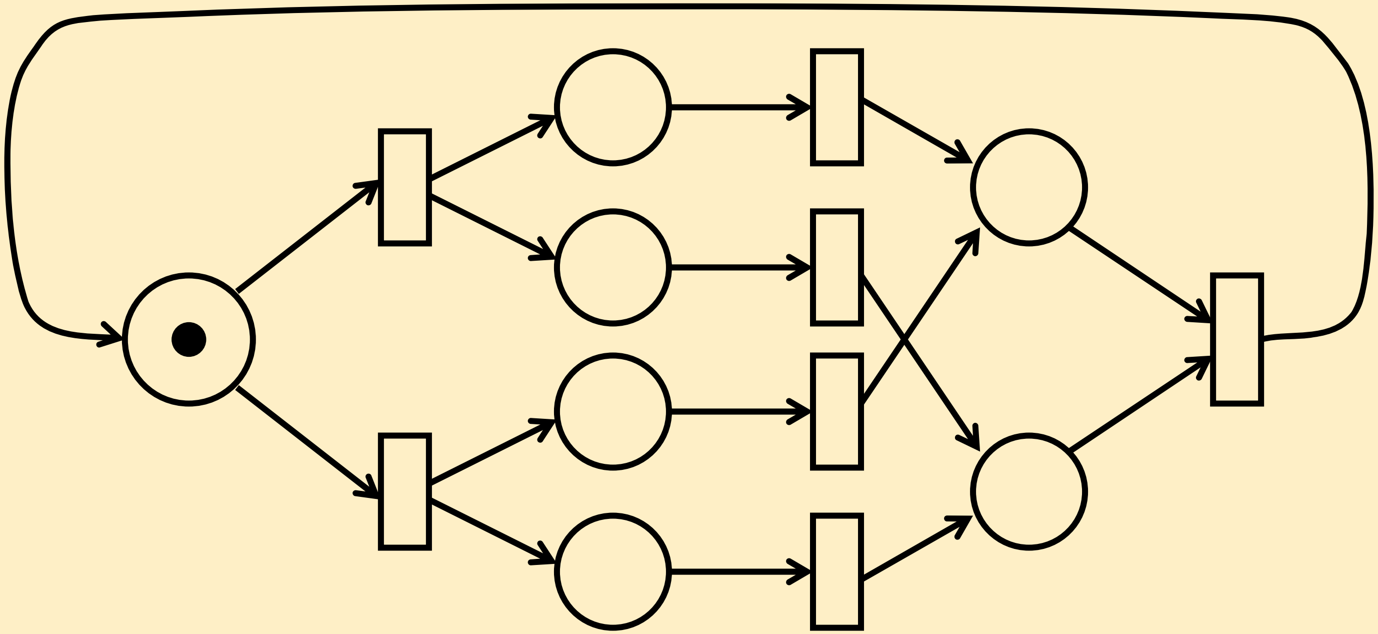
Élő és biztos tulajdonság az FC hálókban

10. Egy (N, M_0) szabad választású háló a.cs.a. **élő**, ha minden N–beli szifon tartalmaz jelölt csapdát
11. Egy élő (N, M_0) szabad választású háló a.cs.a. **biztos**, ha N lefedhető egy tokent tartalmazó erősen összekötött SM komponensekkel
12. Ha (N, M_0) élő és biztos szabad választású háló, akkor N lefedhető erősen összekötött MG komponensekkel. Létezik olyan $M \in R(N, M_0)$, hogy minden (N_1, M_1) komponens élő és biztos MG háló, ahol M_1 az M-nek N_1-re vett rész tokeneloszlása

Példa: élő és biztos FC háló

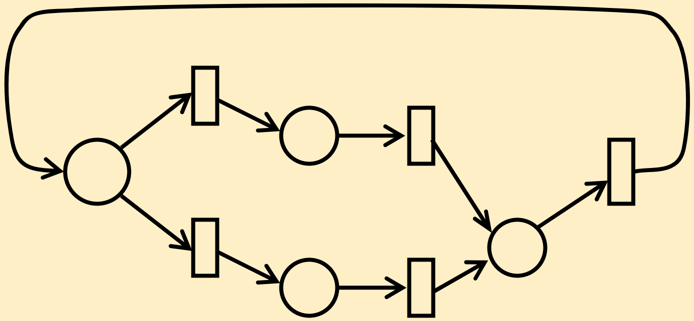
Élő és biztos a jelölt kezdőállapottal, mert

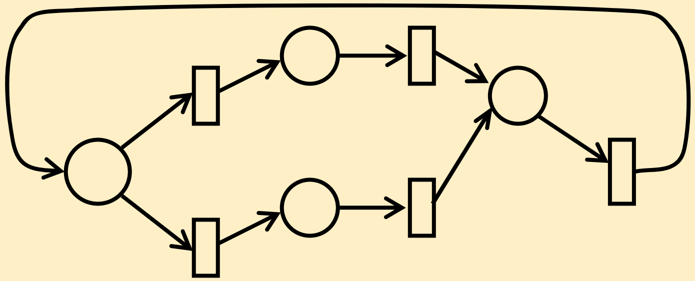
- minden SM komponens erősen összekötött, egy tokent tartalmaz, és a komponensek lefedik a teljes hálót
- minden MG komponens erősen összekötött, egy tokent tartalmaz, és a komponensek lefedik a teljes hálót



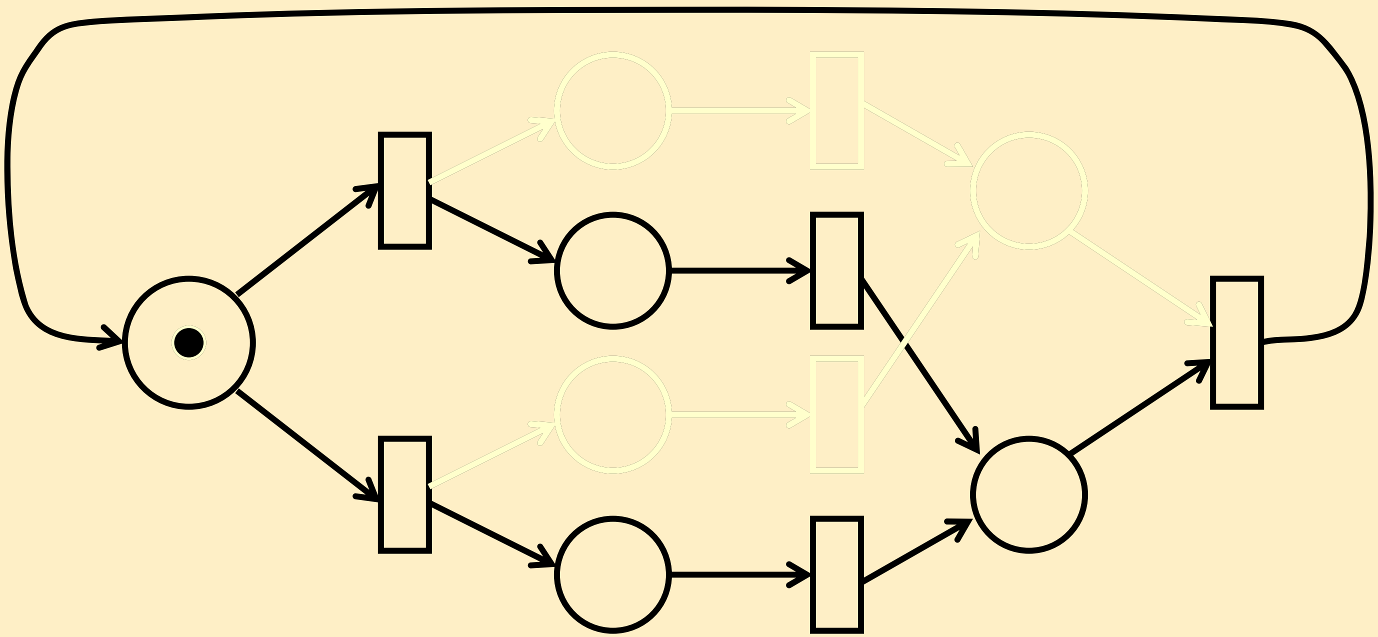
SM dekompozíció

- Allokáció: választás a szinkronizációban részt vevő tevékenységek közt
- Redukció menete
  1. Allokáció választása
  2. A nem allokált helyek törlése
  3. Minden kimeneti hely nélküli átmenet törlése
  4. Minden törölt kimeneti tüzeléssel bíró hely törlése



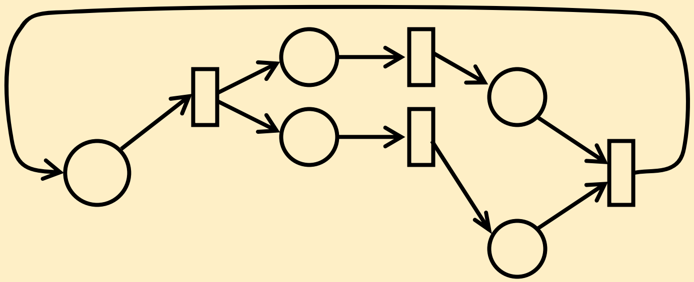


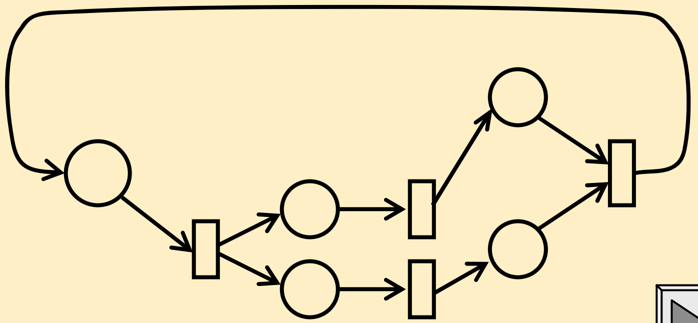
A példa FC háló SM dekompozíciója



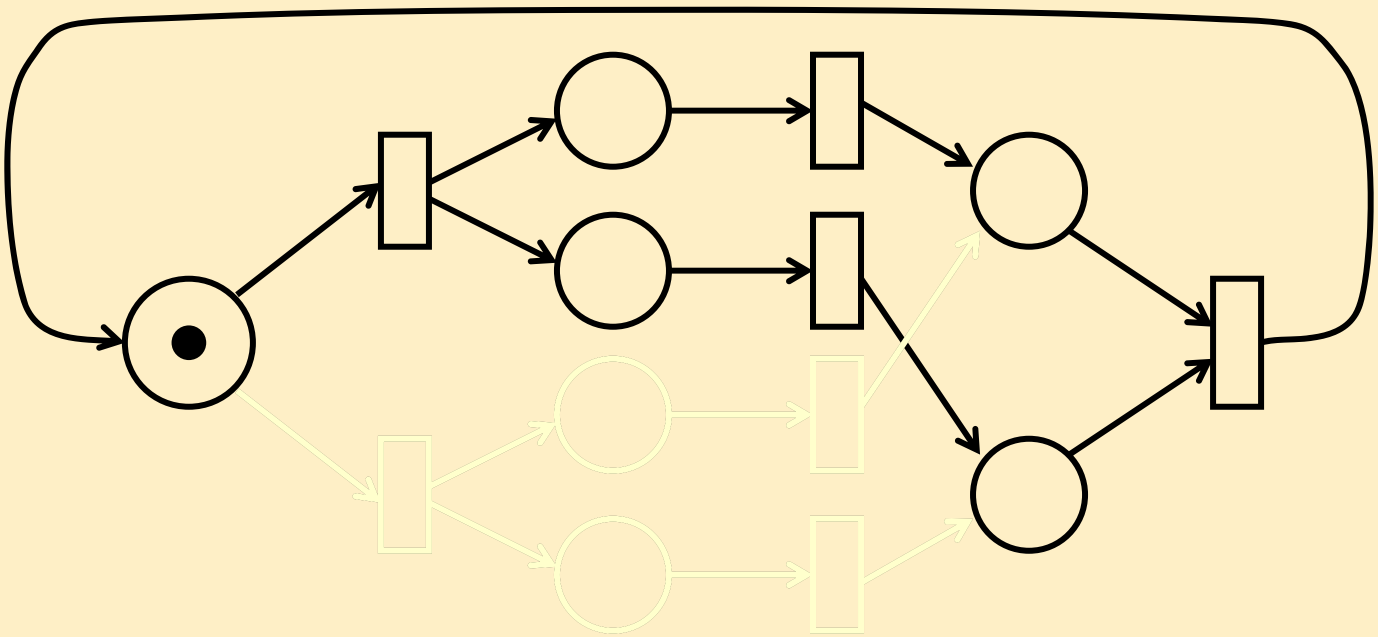
MG dekompozíció

- Allokáció: választás a konfliktusban levő átmenetek közt
- Redukció menete
  1. Allokáció választása
  2. A nem allokált átmenetek törlése
  3. Minden bemeneti tüzelés nélküli hely törlése
  4. Minden törölt bemeneti hellyel bíró átmenet törlése





A példa FC háló MG dekompozíciója



Petri hálók strukturális tulajdonságai

Strukturális tulajdonságok

Petri hálók kezdőállapot-független tulajdonságai:

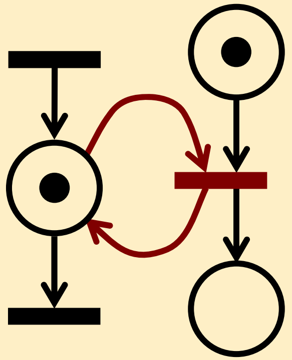
- Strukturális korlátosság
- Vezérelhetőség
- Konzervativitás
  - Hely invariáns,
    P- (place) invariáns
- Strukturális élőség
- Ismételhetőség
- Konzisztencia
  - Tüzelési invariáns,
    T- (transition) invariáns

Csak a háló struktúra határozza meg őket:

- vagy $\forall$ korlátos kezdő tokeneloszlásra igazak,
- vagy $\exists$ olyan korlátos kezdő tokeneloszlás, amire igazak

Állapotegyenlet

- Petri háló dinamikája: tokenek áramlanak
  - Kirchhoff egyenletekhez hasonló egyensúlyi egyenletek
- Előfeltétel (egyértelműség): tiszta Petri háló
  - Nincs olyan átmenet, amely egyazon helynek egyszerre bemenő és kimenő átmenete: $\forall t \in T : \bullet t \cap t \bullet = \varnothing$
  - Hurokél
    - a tüzeléskor a tokeneloszlás nem változik, de
    - a tüzelési feltételben szerepet játszik



Állapotegyenlet

- Tüzelési szekvencia: $M_{i_0}\left[\vec{\sigma} > M_{i_n}\right.$

$$\boxed{\vec{\sigma} = \left\langle M_{i_0} t_{i_1} M_{i_1} \ldots t_{i_n} M_{i_n} \right\rangle = \left\langle t_{i_1} \ldots t_{i_n} \right\rangle}$$

- **Engedélyezés:**
  - $t_{i,j}$ átmenet minden $p \in \bullet t_{i,j}$ bemenő helyén **elég** token

$$\boxed{\forall t_{i_j} \in \vec{\sigma}, \forall p \in \bullet t_{i_j} : M_{i_{j-1}}(p) \geq w^{-}(p, t_{i_j}) = \mathbf{W}^{-\mathrm{T}} \vec{e}_{i_j}}$$

Állapottrajektóriák

- **Állapotátmenet:**
  - $t_{i,j}$ átmenet engedélyezett $\rightarrow$ tüzel
    - minden $p \in \bullet t_{i,j}$ bemenő helyéről $w^-(p, t_{i,j})$ tokent vesz el
    - minden $p \in t_{i,j}\bullet$ kimenő helyére $w^+(p, t_{i,j})$ tokent tesz ki

$$\boxed{M_{i_j} = M_{i_{j-1}} - \mathbf{W}^{-^{\mathrm{T}}} \vec{e}_{i_j} + \mathbf{W}^{+^{\mathrm{T}}} \vec{e}_{i_j} = M_{i_{j-1}} + \mathbf{W}^{\mathrm{T}} \vec{e}_{i_j}}$$

    - összeadva és átrendezve:

$$\boxed{M_{i_0} [\vec{\sigma} > M_{i_n} \Rightarrow M_{i_n} - M_{i_0} = \mathbf{W}^{\mathrm{T}} \vec{\sigma}_T}$$

- Tüzelési szám vektor: az egyes tranzíciók tüzeléseinek száma a tüzelési szekvenciában

Állapotegyenlet levezetése

$$M_1 = M_0 + \mathbf{W}^{\mathrm{T}} \vec{e}_{t_1}$$

$$M_2 = M_1 + \mathbf{W}^{\mathrm{T}} \vec{e}_{t_2} = \overbrace{M_0 + \mathbf{W}^{\mathrm{T}} \vec{e}_{t_1} + \mathbf{W}^{\mathrm{T}} \vec{e}_{t_2}}^{M_1 \text{ behelyettesítésével}}$$

$$\ldots$$

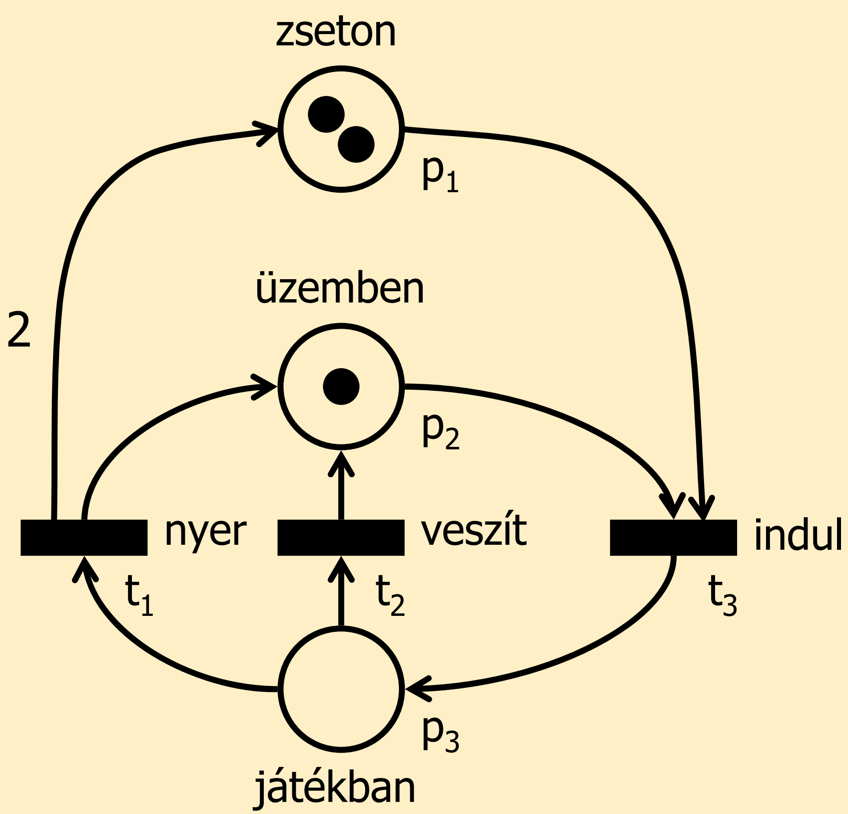
$$M_{n+1} = M_n + \mathbf{W}^{\mathrm{T}} \vec{e}_{t_{n+1}} = M_0 + \mathbf{W}^{\mathrm{T}} \vec{e}_{t_1} + \mathbf{W}^{\mathrm{T}} \vec{e}_{t_2} + \ldots + \mathbf{W}^{\mathrm{T}} \vec{e}_{t_{n+1}}$$

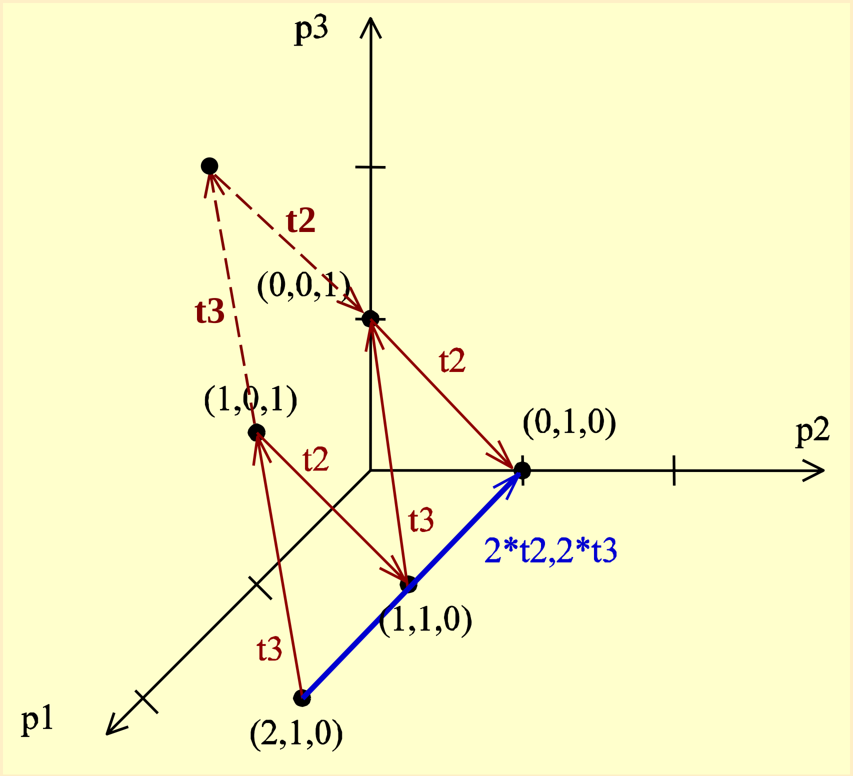
$$\ldots \qquad \Downarrow$$

$$M_m = M_0 + \underbrace{\mathbf{W}^{\mathrm{T}} \vec{e}_{t_1} + \mathbf{W}^{\mathrm{T}} \vec{e}_{t_2} + \ldots + \mathbf{W}^{\mathrm{T}} \vec{e}_{t_m}}_{\text{összevonva}} = M_0 + \mathbf{W}^{\mathrm{T}} \sum_{i=1}^{m} \vec{e}_{t_i}$$

$$M_m = M_0 + \mathbf{W}^{\mathrm{T}} \vec{\sigma}_T \Rightarrow \boxed{M_m - M_0 = \mathbf{W}^{\mathrm{T}} \vec{\sigma}_T}$$

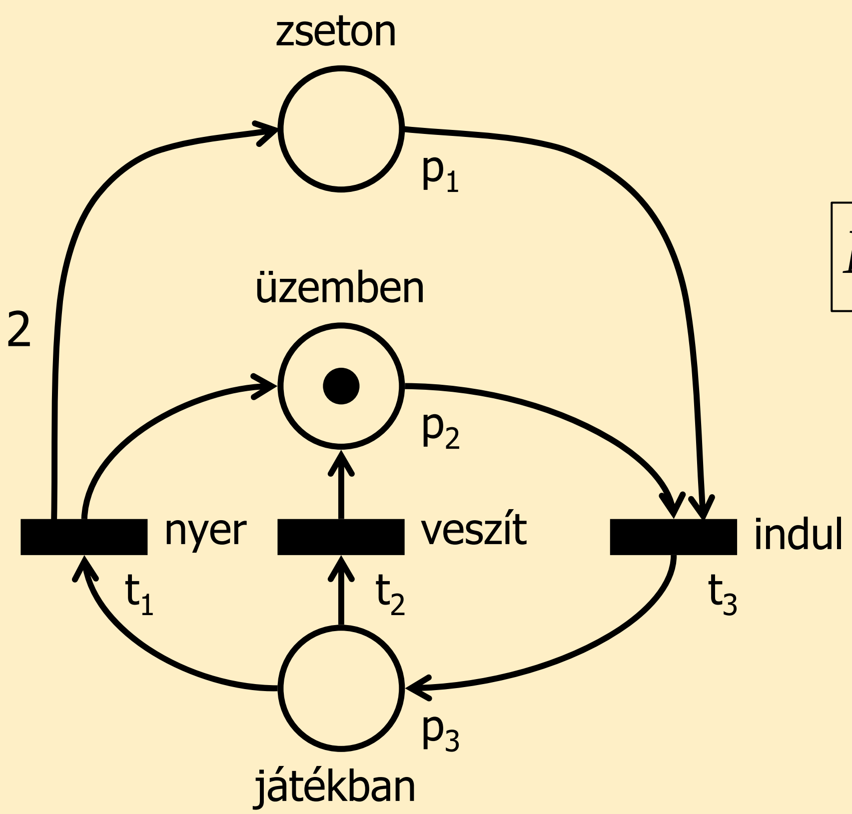
Állapotegyenlet és elérhetőség





- A **tüzelési szám vektor**ban kevesebb az információ, mint a **tüzelési szekvenciá**ban!

Állapotegyenlet és elérhetőség



- Tokenmérleg teljesülése a tüzelésnek csak a **szükséges** feltétele!

$$\boxed{M_{i_0}[\vec{\sigma} > M_{i_n} \Rightarrow M_{i_n} - M_{i_0} = \mathbf{W}^{\mathrm{T}} \vec{\sigma}_T}$$

$$\mathbf{W}^{\mathrm{T}} = \begin{array}{c c} & \begin{array}{ccc} \mathbf{t_1} & \mathbf{t_2} & \mathbf{t_3} \end{array} \\ \begin{array}{c} \mathbf{p_1} \\ \mathbf{p_2} \\ \mathbf{p_3} \end{array} & \begin{pmatrix} 2 & 0 & -1 \\ 1 & 1 & -1 \\ -1 & -1 & 1 \end{pmatrix} \end{array}$$

$$\boxed{(1,1,0)^{\mathrm{T}} - (0,1,0)^{\mathrm{T}} = \mathbf{W}^{\mathrm{T}} \cdot (1,0,1)^{\mathrm{T}}}$$

t_1 és t_3 sem nem tüzelhető a (0,1,0) kezdőállapotban!

Invariánsok fogalma
Tüzelési és hely invariáns

Tüzelési, avagy T-invariáns

A σ tüzelési szekvencia végrehajtása nem változtatja meg a tokeneloszlást:

$$\boxed{\mathbf{W}^{\mathrm{T}} \vec{\sigma}_T = 0}$$

- Ciklus az állapottérben: $M_i \left[\vec{\sigma}_T > M_i \right.$
  - ha σ_T szekvencia az M_i állapotból végrehajtható!

$$\boxed{\forall t_{i_j} \in \vec{\sigma}, \forall p \in \left\{ \bullet t_{i_j} \right\} : m_{i_{j-1}}\left(p\right) \geq w^{-}\left(p, t_{i_j}\right) = \mathbf{W}^{-\mathrm{T}} \cdot \vec{e}_{i_j}}$$

- Megjegyzés: bármely σ tüzelési szekvenciához található olyan M_0 kezdőállapot, amelyből σ végrehajtható
  - Pl. $M_0 \geq \mathbf{W}^{-^{\mathrm{T}}} \vec{\sigma}$ esetén induláskor annyira „teletömött”, hogy a σ tüzelési szekvencia által termelt tokenekre már nincs szükség!

T-invariánsok halmaza

$$\boxed{\mathbf{W}^{\mathrm{T}}\vec{\sigma}_T = 0}$$

Homogén, lineáris egyenletrendszer

- egy megoldás többszöröse is megoldás
  - ha tüzelhető, akkor többször is befutja a ciklust
- megoldás összege is megoldás
  - ha tüzelhető, akkor több ciklus kombinációját futja be
- megoldások lineáris kombinációi is megoldások

Keressünk **BÁZIST**!

- az összes megoldást előállító minimális halmaz

Minimális T-invariáns

- A σ tüzelési szekvencia $\sup(\sigma)$ **alapja**
  - azon átmenetek $T' = \{t_i \mid \sigma_i > 0\}$ részhalmaza, amelyek σ szekvenciában előfordulnak
- A σ_T tüzelési invariáns **minimális alapú**
  - ha nincs olyan T-invariáns, amelynek alapja σ_T alapjának valódi részhalmaza, vagy
  - ha részhalmaza azonos, annak tüzelési számai kisebbek

$$\boxed{\forall \sigma_T^1 : \mathbf{W}^{\mathrm{T}} \sigma_T^1 = 0 \Rightarrow \left(\sigma_T^1 \geq \sigma_T\right) \vee \left(\sup(\sigma_T) \not\subseteq \sup(\sigma_T^1)\right)}$$

Hely, avagy P-invariáns

A μ_P súlyvektor által kijelölt helyeken a tokenek súlyozott összege nem változik:

$$\boxed{\vec{\mu}_P^{\mathrm{T}} M = \text{állandó}}$$

– A tokenek (egy része) a helyek egy részhalmazában kering (pl. erőforrások nem fogynak, nem keletkeznek)

$$\left.\begin{array}{l} M = M_0 + \mathbf{W}^{\mathrm{T}} \vec{\sigma} \\[2ex] \underbrace{\vec{\mu}_P^{\mathrm{T}} M = \vec{\mu}_P^{\mathrm{T}} M_0}_{\vec{\mu}_P^{\mathrm{T}} M = \vec{\mu}_P^{\mathrm{T}} M_0 = \text{állandó}} + \vec{\mu}_P^{\mathrm{T}} \mathbf{W}^{\mathrm{T}} \vec{\sigma} \\[2ex] \vec{\mu}_P^{\mathrm{T}} \mathbf{W}^{\mathrm{T}} \vec{\sigma} = 0 \Rightarrow \vec{\mu}_P^{\mathrm{T}} \mathbf{W}^{\mathrm{T}} \underset{\forall \vec{\sigma}}{\equiv} 0 \end{array}\right\} \qquad \boxed{\mathbf{W} \vec{\mu}_P = 0}$$

Invariánsok alkalmazásai

- T-invariánsok alkalmazásai
  - Termelési folyamat meghatározása
  - Szabályalapú rendszerek, diagnosztikai problémák
  - Dinamikus tulajdonságok
    - ciklikusan tüzelhető ➲ megfordíthatóság, visszatérő állapot
    - később is tüzelhető ➲ élő tulajdonság, holtpontmentesség
- P-invariánsok alkalmazásai
  - Véges automaták keresése → dekompozíció
  - Dinamikus tulajdonságok
    - token nem vész el ➲ élő tulajdonság, holtpontmentesség
    - token nem termelődik ➲ korlátosság

Invariánsok számítása

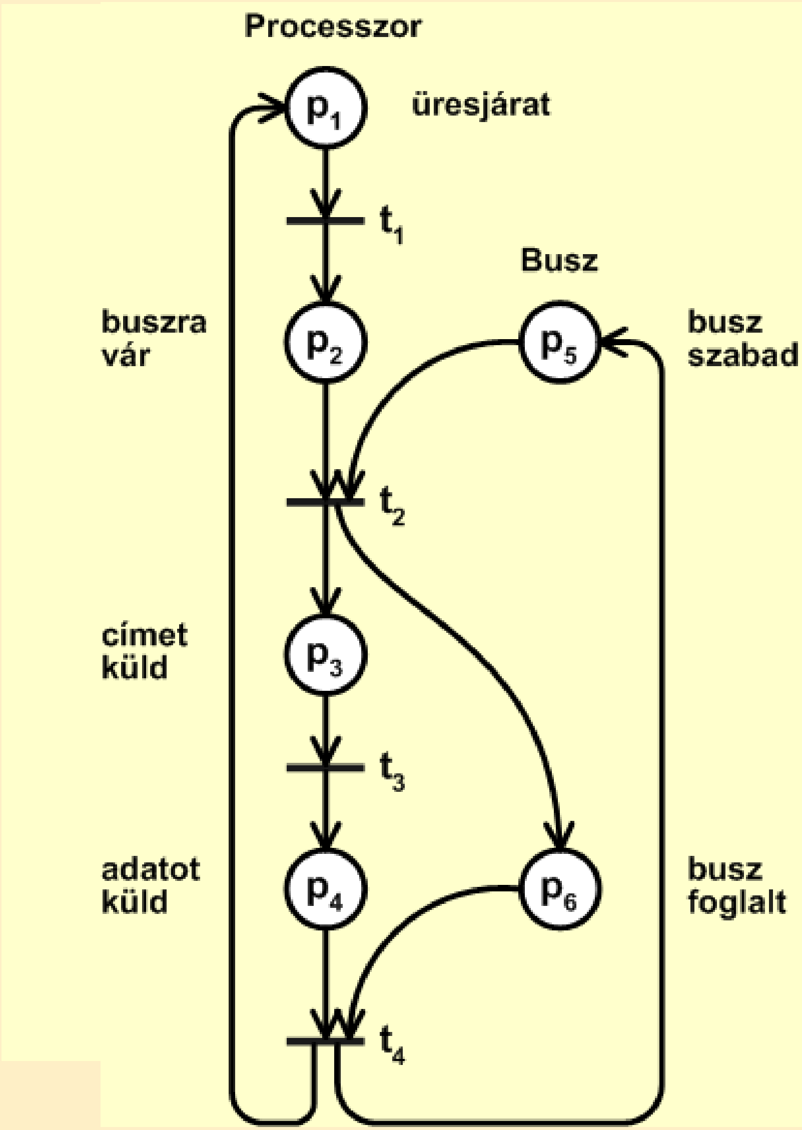
Megoldási módszerek

Kérdések:

- σ_T bázis komponenseinek értelmezési tartománya?
- a lineáris kombinációk együtthatóinak értelmezési tartománya?

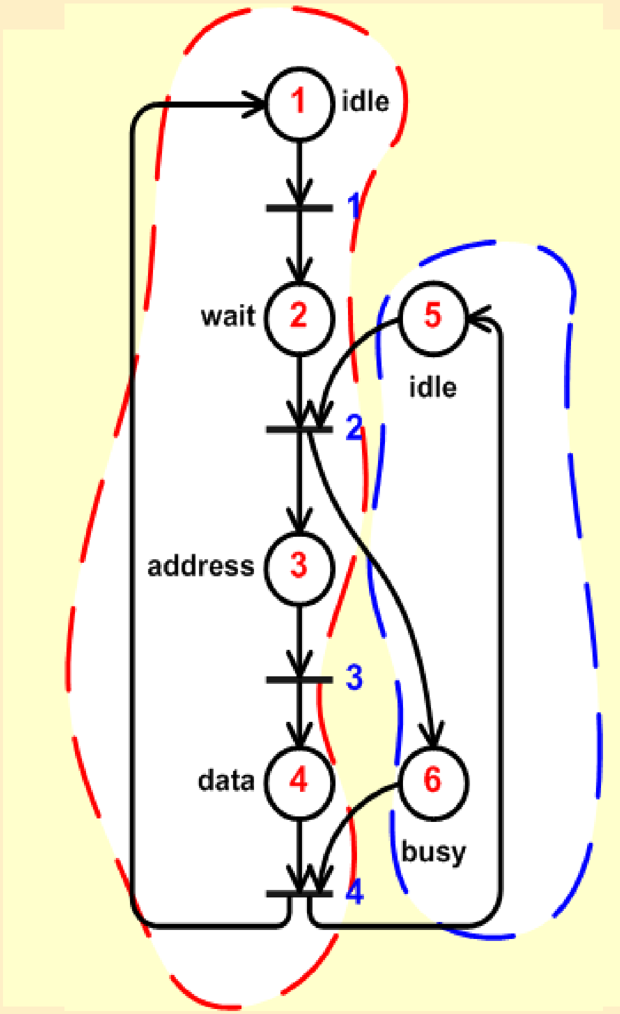
Szint	Tartomány	Együttható	Lineárisan független?	Egyértelmű?	Algoritmus
1	$x \in \mathbf{Z}$	$\mathbf{Q}$	Igen	Nem	Gauss elimináció
2	$x \in \mathbf{Z}$	$\mathbf{Z}$	Igen	Nem	Hermite redukció
3	$x \in \mathbf{N}_0$	$\mathbf{Q}_0$	Nem biztos	Igen	Martinez-Silva
4	$x \in \mathbf{N}_0$	$\mathbf{N}_0$	Nem biztos	Igen	Pascoletti
5	$x \in \mathbf{B}$	$\mathbf{B}$	Nem biztos	Igen	Jaxy

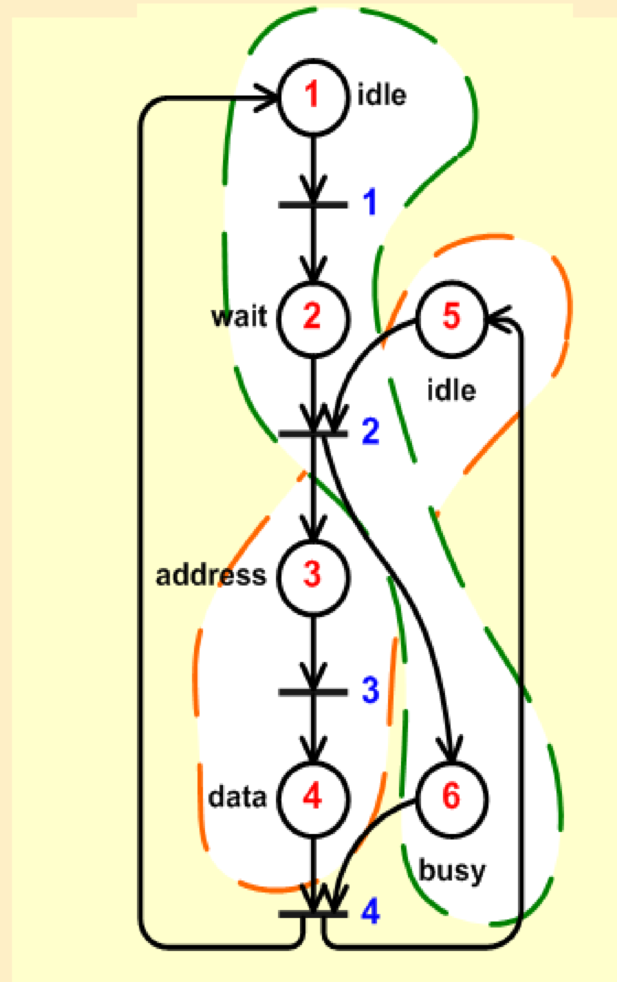
Példa: processzor adatátvitel



- Processzor
  - várakozik (idle - üresjárat)
  - busz hozzáférési jogot kér
  - címet tesz ki a címbuszra
  - adatot tesz ki az adatbuszra
- Busz(ok)
  - szabad (nem használja senki)
  - foglalt (processzor/periféria)
- Petri háló
  - n = 4 darab átmenet
  - m = 6 darab hely

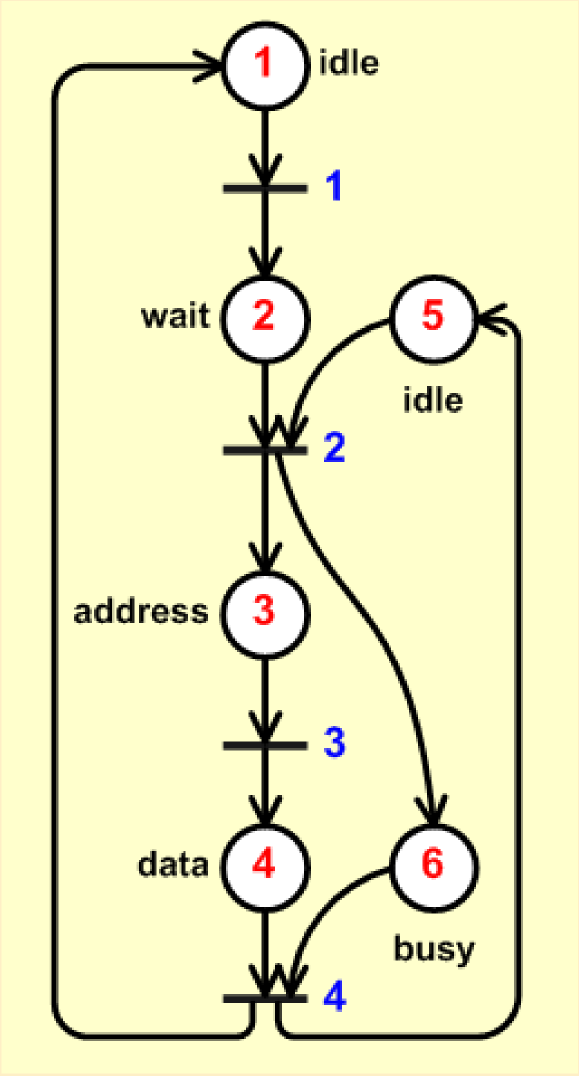
Keressük meg fejben a megoldást!





Négy P invariáns található

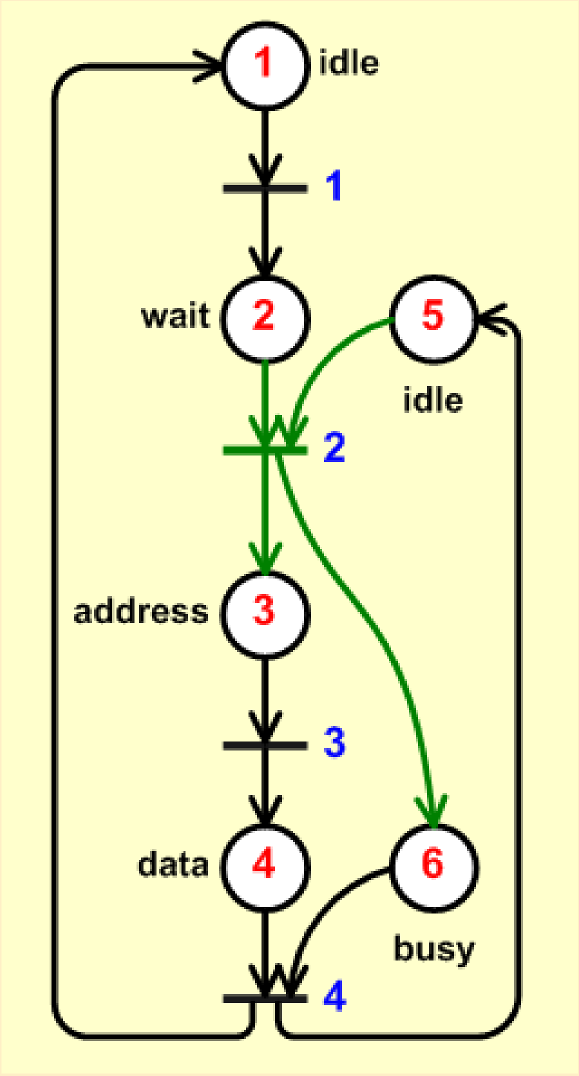
Szomszédossági mátrixok



$$
W^{-} = \begin{array}{c} \begin{matrix} p_1 & p_2 & p_3 & p_4 & p_5 & p_6 \end{matrix} \\ \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \end{bmatrix} \end{array} \begin{array}{c} \\ t_1 \\ t_2 \\ t_3 \\ t_4 \end{array}
$$

$$
W^{+} = \begin{array}{c} \begin{matrix} p_1 & p_2 & p_3 & p_4 & p_5 & p_6 \end{matrix} \\ \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \end{array} \begin{array}{c} \\ t_1 \\ t_2 \\ t_3 \\ t_4 \end{array}
$$

Szomszédossági mátrixok



$$
W^T = \begin{array}{c} \\ \\ \\ \\ \\ \\ \\ \end{array}
\left[\begin{array}{ccccc}
t_1 & t_2 & t_3 & t_4 & \\
-1 & 0 & 0 & 1 & p_1 \\
1 & -1 & 0 & 0 & p_2 \\
0 & 1 & -1 & 0 & p_3 \\
0 & 0 & 1 & -1 & p_4 \\
0 & -1 & 0 & 1 & p_5 \\
0 & 1 & 0 & -1 & p_6
\end{array}\right]
$$

Martinez-Silva algoritmus: inicializálás

$i \leftarrow 1$

$T_i \leftarrow \{\, t \in T \,\}$

$\mathbf{A} \leftarrow \mathbf{W}^{\mathrm{T}}$, $\mathbf{D} \leftarrow \mathbf{1}_n$ // $n = |P|$

$\mathbf{Q}_i \leftarrow [\mathbf{D} \mid \mathbf{A}]$ // egységmátrix + szomszédossági mátrix

$L_p \leftarrow$ a $\mathbf{Q}_i$ mátrix p. sora

$T_1 = \{\, t_1, t_2, t_3, t_4 \,\}$

$$\mathbf{Q_1} = \begin{array}{c} \begin{array}{cccccc|cccc} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 & t_1 & t_2 & t_3 & t_4 \end{array} \\ \left[\begin{array}{cccccc|cccc} 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & -1 \end{array}\right] \end{array} \begin{array}{c} \\ p_1 \\ p_2 \\ p_3 \\ p_4 \\ p_5 \\ p_6 \end{array}$$

Martinez-Silva algoritmus: ciklus

while $\mathbf{A_i} \neq 0$

if $t_j \in T_i$ // válasszunk egy eddig nem vizsgált oszlopot

$T_{i+1} \leftarrow T_i \setminus \{ t_j \}$

$L_{delete} \leftarrow \varnothing$

$\mathbf{Q}_{i+1} \leftarrow \mathbf{Q}_i$

for all $u, v : A_i(u, j) \neq 0 \wedge A_i(v, j) \neq 0 \wedge$
$\exists \lambda_u, \lambda_v \in \infty^+ : \lambda_u A_i(u, j) + \lambda_v A_i(v, j) = 0$

$\mathbf{Q}_{i+1}$-hez adjuk hozzá a $\lambda_u L_u + \lambda_v L_v$ sort

$L_{delete} \leftarrow L_{delete} \cup \{ L_u, L_v \}$

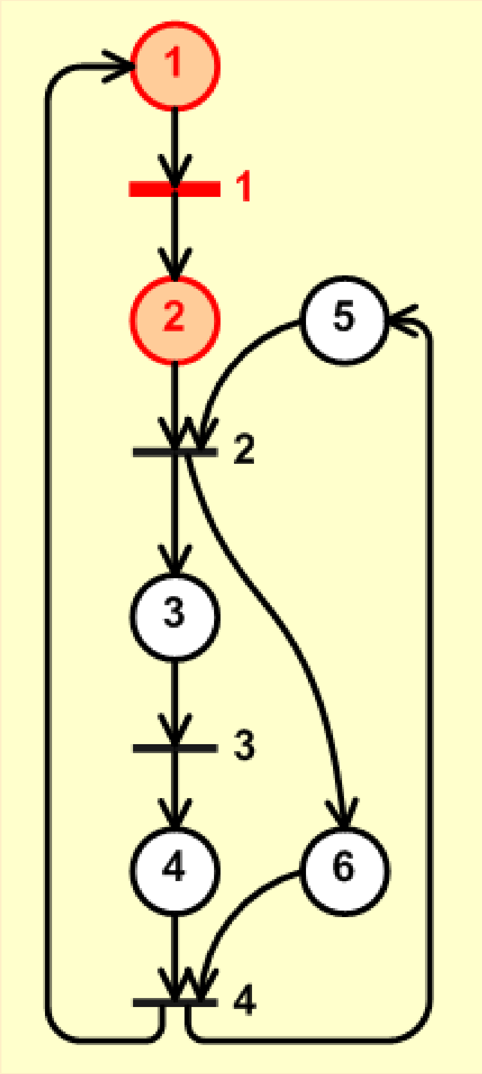
end for

$\mathbf{Q}_{i+1}$-ből töröljük az L_{delete} halmazbeli sorokat

$i \leftarrow i + 1$

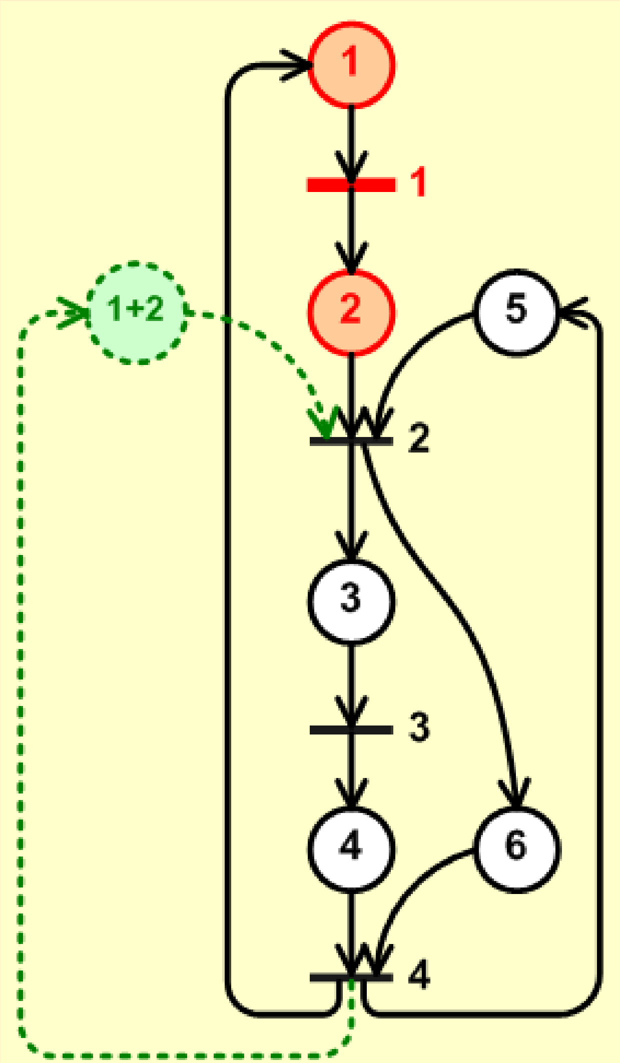
end while

Martinez-Silva algoritmus: 1-1. lépés



$$Q_1 = \begin{array}{cc} \begin{matrix} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 & t_1 & t_2 & t_3 & t_4 \end{matrix} & \\ \left[\begin{array}{cccccc|cccc} 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & -1 \end{array}\right] & \begin{matrix} p_1 \\ p_2 \\ p_3 \\ p_4 \\ p_5 \\ p_6 \end{matrix} \end{array}$$

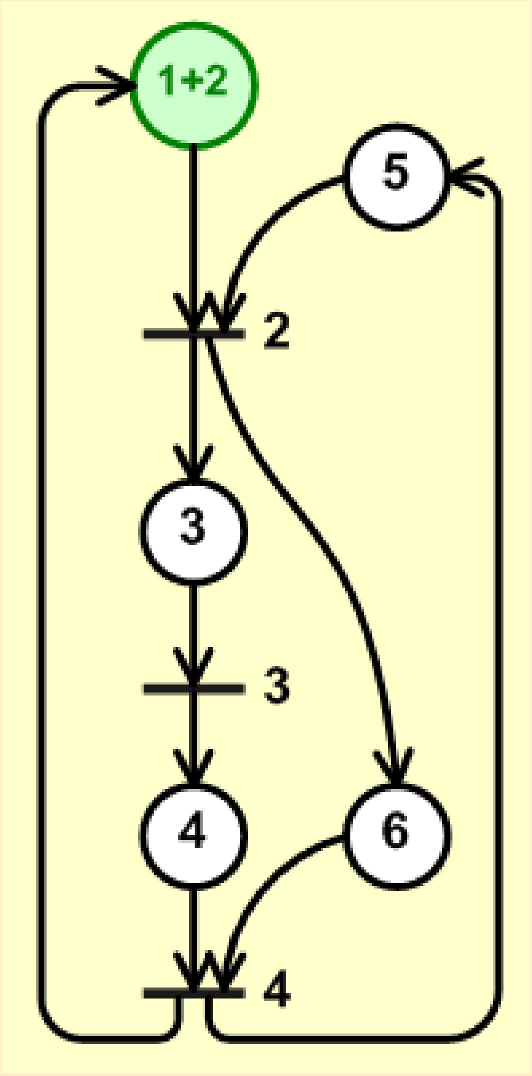
Martinez-Silva algoritmus: 1-2. lépés



$$Q_1 = \begin{array}{c}\begin{array}{cccccc|cccc} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 & t_1 & t_2 & t_3 & t_4 \end{array} \\ \left[\begin{array}{cccccc|cccc} 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & -1 \end{array}\right]\end{array} \begin{array}{c} \\ p_1 \\ p_2 \\ p_3 \\ p_4 \\ p_5 \\ p_6 \end{array}$$

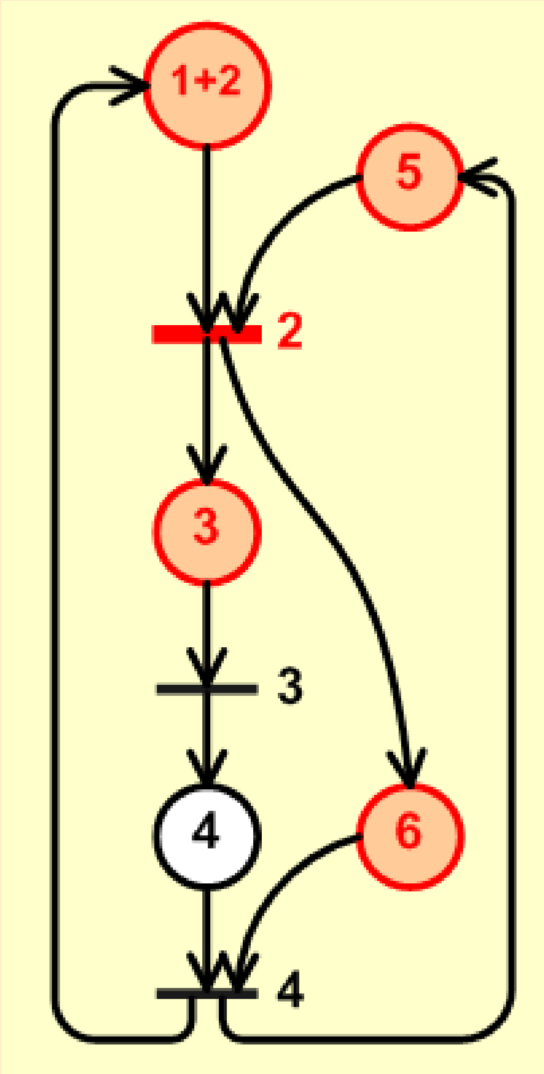
$$Q_1' = \left[\begin{array}{cccccccccc} 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & -1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 \end{array}\right] \begin{array}{c} p_1 \\ p_2 \\ p_3 \\ p_4 \\ p_5 \\ p_6 \\ p_{1+2} \end{array}$$

Martinez-Silva algoritmus: 1. részeredmény



$$Q_1'' = \begin{array}{cccccc|ccccc} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 & t_1 & t_2 & t_3 & t_4 & \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & p_3 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & -1 & p_4 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & 1 & p_5 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & -1 & p_6 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 & p_{1+2} \end{array}$$

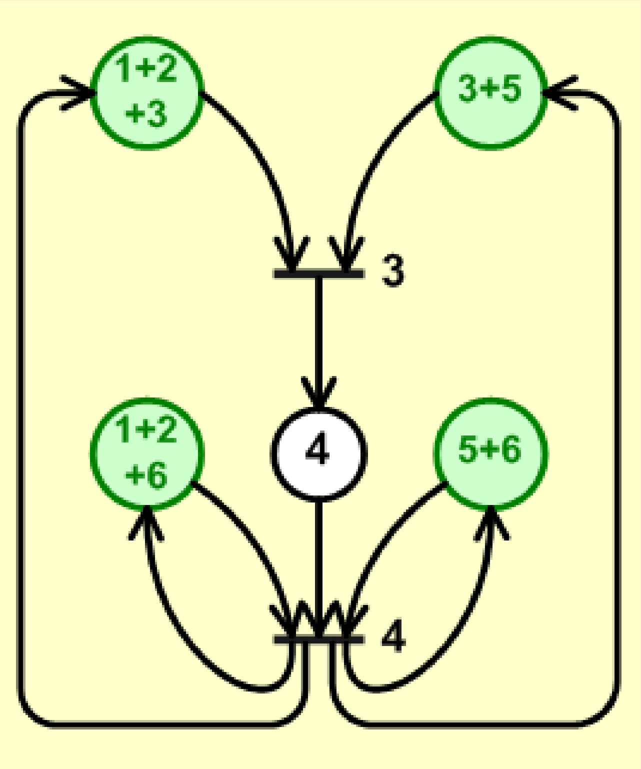
Martinez-Silva algoritmus: 2-1, 2-2. lépés



$$Q_2 = \begin{array}{cc}
\begin{matrix} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 & t_1 & t_2 & t_3 & t_4 \end{matrix} & \\
\left[\begin{array}{cccccc:cccc}
0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & -1 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & -1 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & -1 \\
1 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1
\end{array}\right] & \begin{matrix} p_3 \\ p_4 \\ p_5 \\ p_6 \\ p_{1+2} \end{matrix}
\end{array}$$

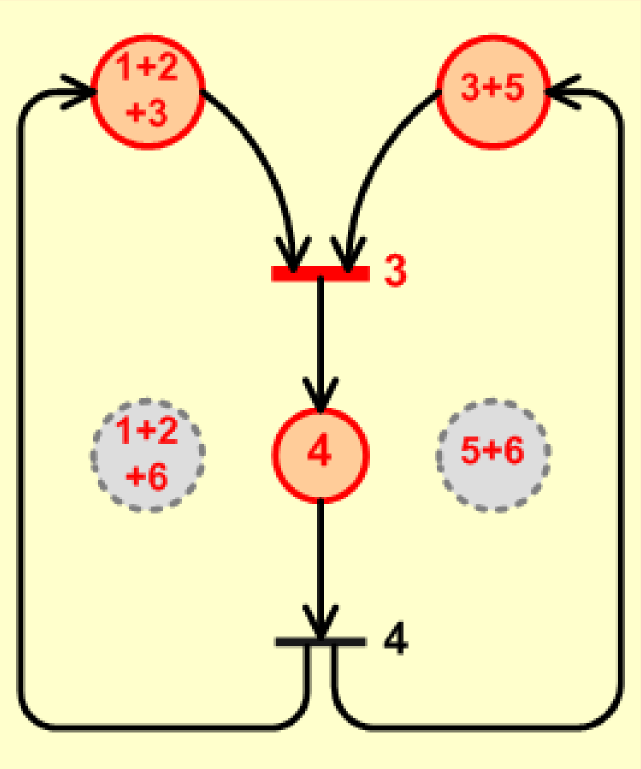
$$Q_2' = \begin{array}{cc}
\left[\begin{array}{cccccccccc}
0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & -1 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & -1 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & -1 \\
1 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 \\
1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 1 \\
0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & -1 & 1 \\
1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0
\end{array}\right] & \begin{matrix} p_3 \\ p_4 \\ p_5 \\ p_6 \\ p_{1+2} \\ p_{1+2+3} \\ p_{3+5} \\ p_{1+2+6} \\ p_{5+6} \end{matrix}
\end{array}$$

Martinez-Silva algoritmus: 2. részeredmény



$$Q_2'' = \begin{array}{cccccc:cccc|l}
e_1 & e_2 & e_3 & e_4 & e_5 & e_6 & t_1 & t_2 & t_3 & t_4 & \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & -1 & p_4 \\
1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & p_{1+2+3} \\
0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & -1 & 1 & p_{3+5} \\
1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & p_{1+2+6} \\
0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & p_{5+6}
\end{array}$$

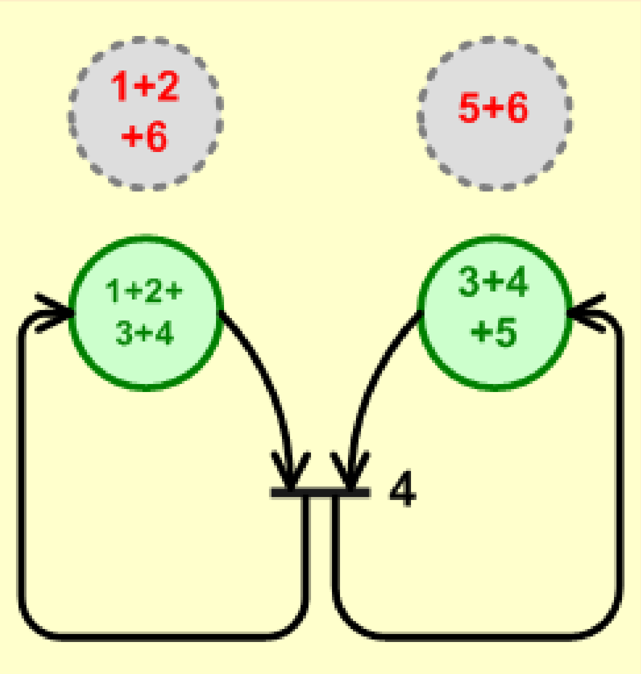
Martinez-Silva algoritmus: 3-1, 3-2. lépés



$$Q_3 = \begin{array}{c} \begin{array}{cccccc|cccc} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 & t_1 & t_2 & t_3 & t_4 \end{array} \\ \left[\begin{array}{cccccc|cccc} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & -1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & -1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \end{array}\right] \end{array} \begin{array}{l} \\ p_4 \\ p_{1+2+3} \\ p_{3+5} \\ p_{1+2+6} \\ p_{5+6} \end{array}$$

$$Q_3' = \left[\begin{array}{cccccccccc} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & -1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & -1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \end{array}\right] \begin{array}{l} p_4 \\ p_{1+2+3} \\ p_{3+5} \\ p_{1+2+6} \\ p_{5+6} \\ p_{1+2+3+4} \\ p_{3+4+5} \end{array}$$

Martinez-Silva algoritmus: végeredmény



$$Q_3'' = \begin{array}{cc} \begin{array}{cccccccccc} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 & t_1 & t_2 & t_3 & t_4 \end{array} & \\ \left[\begin{array}{cccccc|cccc} 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \end{array}\right] & \begin{array}{l} p_{1+2+6} \\ p_{5+6} \\ p_{1+2+3+4} \\ p_{3+4+5} \end{array} \end{array}$$

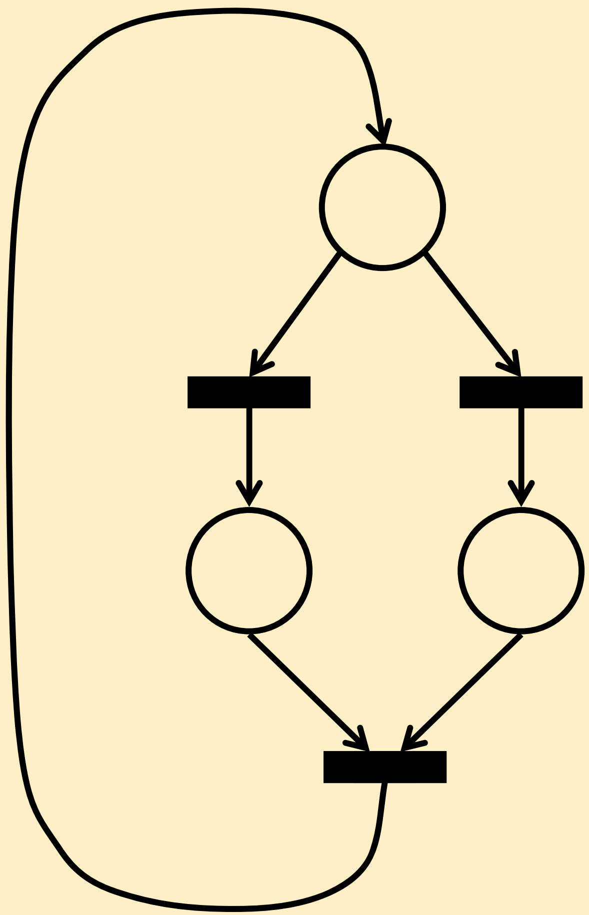
- Invariánsok:
  - a végső $Q_m = [D_m|0]$ mátrix alapján a D_m mátrix soraiban található együtthatók
- Kiszámított P (hely) -invariánsok:
  1. $m(p_1)+m(p_2)+m(p_6) = 1$
  2. $m(p_5)+m(p_6) = 1$
  3. $m(p_1)+m(p_2)+m(p_3)+m(p_4) = 1$
  4. $m(p_3)+m(p_4)+m(p_5) = 1$

További strukturális tulajdonságok

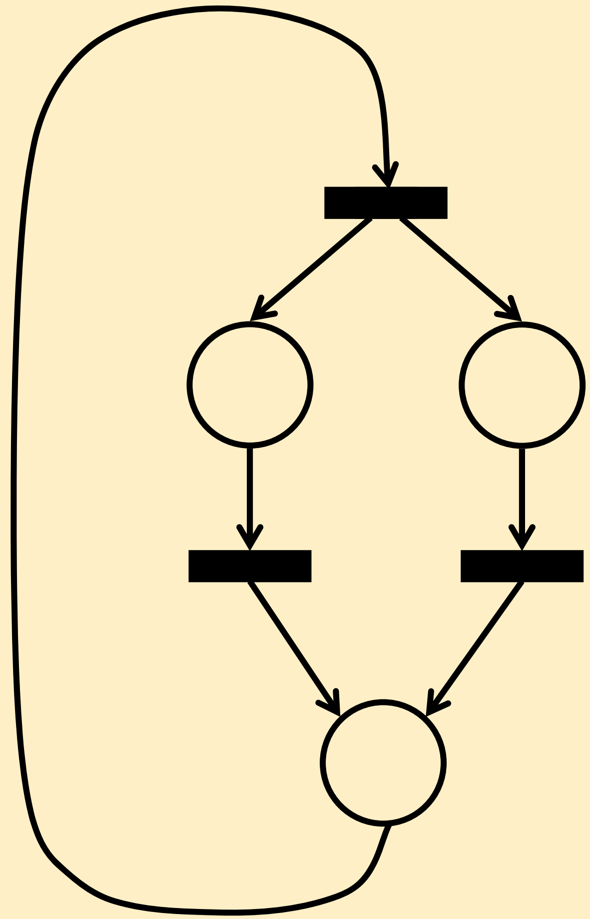
Strukturális élőség

- Egy N Petri háló strukturálisan élő, ha létezik olyan M_0 kezdőállapota, amelyben (N, M_0) (L_4-)élő
  - Szükséges feltétel: erősen összekötött gráf struktúra
  - Jelölt gráfok: egy (G, M_0) jelölt gráf a.cs.a. élő, ha M_0 állapotban minden G-beli irányított körben van legalább egy token $\rightarrow$ minden jelölt gráf strukturálisan élő
  - FC hálók: egy szabad választású háló strukturálisan élő, ha minden N–beli szifon tartalmaz csapdát
  - Általános (közönséges) Petri hálókra a strukturálisan élőség jellemzése (még) nem ismert

Strukturális élőség, korlátosság?



nincs élő jelölése



nincs nemüres biztos jelölése

Vezérelhetőség

- Egy *N* Petri háló teljesen vezérelhető, ha bármely korlátos M_0 kezdőállapot esetén:

$$\boxed{\forall M_i, M_j : M_i, M_j \in R(N, M_0) \Rightarrow M_i \in R(N, M_j) \wedge M_j \in R(N, M_i)}$$

  – azaz bármely állapot elérhető bármely más állapotból

- Elégséges feltétel: $\boxed{\mathrm{rang}(\mathbf{W}^{\mathrm{T}}) = m}$

  – mert $M_1 = M_0 + \mathbf{W}^{\mathrm{T}}\vec{\sigma} \rightarrow \mathbf{W}^{\mathrm{T}}\vec{\sigma} = \Delta M$

  – rangfeltétel: $\mathrm{rang}(\mathbf{W}^{\mathrm{T}}) = \mathrm{rang}(\mathbf{W}^{\mathrm{T}} \mid \Delta M) = m$

  ahol m a helyek száma.

- Ue. szükséges feltétel is jelölt gráfok esetén

Strukturális korlátosság

- Egy *N* Petri háló **strukturálisan korlátos**, ha bármely korlátos M_0 kezdőállapotra korlátos marad
- Feltétele: létezik egy m pozitív komponensű $\vec{\mu}$

$$\boxed{\vec{\mu} > 0, \mathbf{W}\vec{\mu} \le 0}$$

- Szükségesség: $M \in R(N, M_0) \rightarrow M = M_0 + \mathbf{W}^{\mathrm{T}}\vec{\sigma}, \vec{\sigma} \ge 0$
  - átrendezve: $M^{\mathrm{T}}\vec{\mu} = M_0^{\mathrm{T}}\vec{\mu} + \underbrace{\vec{\sigma}^{\mathrm{T}}\mathbf{W}\vec{\mu}}_{\mathbf{W}\vec{\mu}\le 0, \vec{\sigma}\ge 0}$ ← felhasználjuk a feltételt

    (belső szorzat)
  - felső korlát: $M^{\mathrm{T}}\vec{\mu} \le M_0^{\mathrm{T}}\vec{\mu} \Rightarrow M(p) \le \dfrac{M_0^{\mathrm{T}}\vec{\mu}}{\mu_p}$

Strukturális korlátosság: elégségesség

- $\vec{\mu} > 0, \mathbf{W}^{\mathrm{T}} \vec{\mu} \leq 0$ feltétel elégséges is, mert
  - egyébként $\exists \vec{\sigma} \geq 0 : \mathbf{W}^{\mathrm{T}} \vec{\sigma} \underset{\neq}{\geq} 0$

    $$\exists M, M_0 : M - M_0 = \mathbf{W}^{\mathrm{T}} \vec{\sigma} \underset{\neq}{\geq} 0 \rightarrow M \underset{\neq}{\geq} M_0$$

  - ekkor megfelelő M_0 választásával $\vec{\sigma}$ tetszőlegesen sokszor végrehajtható és N nem korlátos

Lineáris mátrixegyenlőtlenségek

vagy az egyik, vagy a másik megoldható

Lemma	Rendszer$_1$ $\oplus$	Rendszer$_2$
Minkowski-Farkas	$\mathbf{W}^{\mathrm{T}}\vec{\sigma} \underset{\neq}{\geq} \vec{b}$, $\vec{\sigma}$ tetszőleges	$\mathbf{W}\vec{\mu} = 0, \vec{\mu} \geq 0, \vec{\mu}^{\mathrm{T}}\vec{b} > 0$
Minkowski-Farkas	$\mathbf{W}^{\mathrm{T}}\vec{\sigma} \underset{\neq}{\geq} \vec{b}, \vec{\sigma} \geq 0$	$\mathbf{W}\vec{\mu} \leq 0, \vec{\mu} \geq 0, \vec{\mu}^{\mathrm{T}}\vec{b} > 0$
Stiemke	$\mathbf{W}^{\mathrm{T}}\vec{\sigma} \underset{\neq}{\geq} 0$, $\vec{\sigma}$ tetszőleges $\rightarrow$ nem konzervatív	$\mathbf{W}\vec{\mu} = 0, \vec{\mu} > 0$ $\rightarrow$ konzervatív
Farkas	$\mathbf{W}^{\mathrm{T}}\vec{\sigma} \underset{\neq}{\geq} 0, \vec{\sigma} \geq 0$ $\rightarrow$ nem strukturálisan korlátos	$\mathbf{W}\vec{\mu} \leq 0, \vec{\mu} > 0$ $\rightarrow$ strukturálisan korlátos

Konzervativitás

- Egy *N* Petri háló (részlegesen) konzervatív, ha bármely korlátos M_0 és $M \in R(N, M_0)$ állapotra minden (néhány) $p \in P$ helyhez található egy μ_p pozitív egész súlytényező, hogy $M\vec{\mu} = M_0\vec{\mu} = \text{állandó}$
- Szükséges és elégséges feltétel: $\boxed{\exists \vec{\mu} \underset{\neq}{\geq} 0 : \mathbf{W}\vec{\mu} = 0}$

Ismételhetőség

- Egy *N* Petri háló (részlegesen) ismételhető, ha létezik olyan M_0 kezdőállapot és M_0-ből induló σ tüzelési szekvencia, hogy minden (néhány) $t \in T$ tranzíció végtelen sokszor tüzel σ-ban
- Szükséges és elégséges feltétel: $\boxed{\exists \vec{\sigma} \underset{\neq}{\geq} 0 : \mathbf{W}^{\mathrm{T}} \vec{\sigma} \geq 0}$
  - Bizonyítás: $\exists \vec{\sigma} > 0 \text{: } \mathbf{W}^{\mathrm{T}} \vec{\sigma} \geq 0$

    $$\exists M, M_0 : M - M_0 = \mathbf{W}^{\mathrm{T}} \vec{\sigma} \geq 0 \rightarrow M \geq M_0$$
  - ekkor megfelelő M_0 választásával $\vec{\sigma}$ tetszőlegesen sokszor végrehajtható

Konzisztencia

- Egy *N* Petri háló (részlegesen) konzisztens, ha létezik olyan M_0 kezdőállapot és M_0-ből induló és M_0-ba visszavezető σ tüzelési szekvencia, hogy minden (néhány) $t \in T$ tranzíció legalább egyszer tüzel σ-ban
- Szükséges és elégséges feltétel: $\boxed{\exists \vec{\sigma} \underset{\neq}{\geq} 0 : \mathbf{W}^{\mathrm{T}} \vec{\sigma} = 0}$
  - Bizonyítás: ismételhetőség feltételénél látott módon

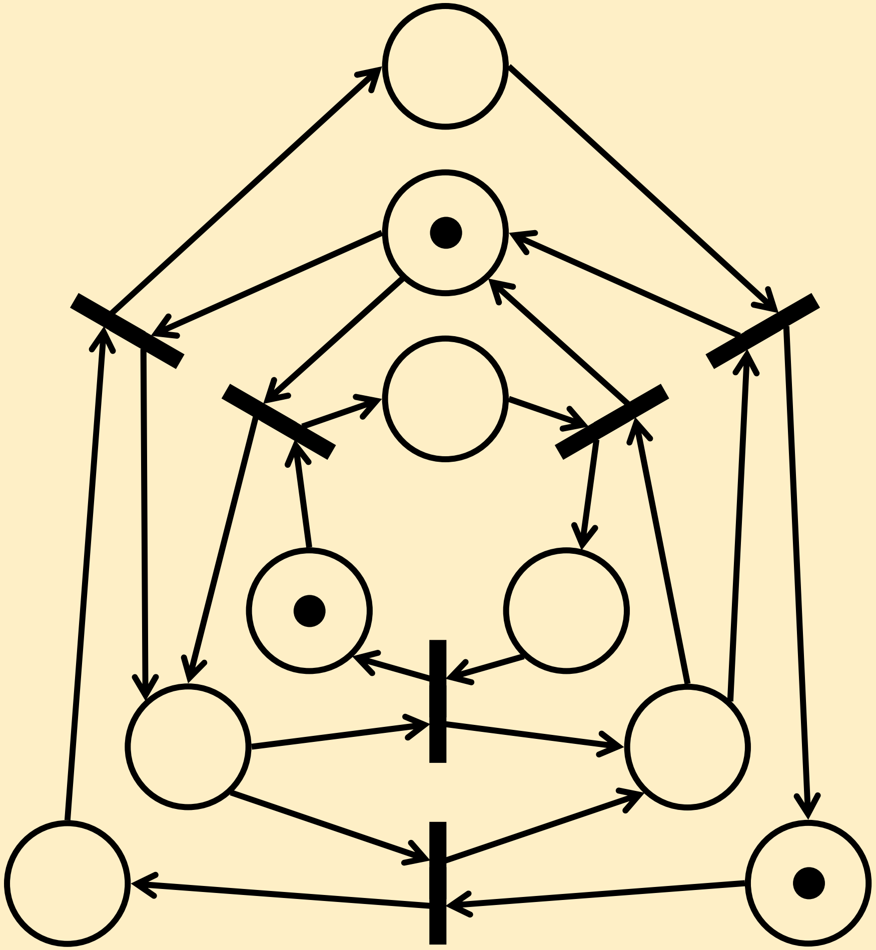
Strukturális B-fairség

- Két tranzíció **strukturálisan B-fair**, ha bármely M_0 kezdőállapot esetén B-fair (korlátos fair) relációban állnak.
- Egy *N* Petri háló B-fair, ha bármely két tranzíciója esetén a B-fair reláció teljesül
- Egy *N* Petri háló strukturálisan B-fair, ha bármely M_0 kezdőállapotra a háló B-fair
  - B-fair reláció ekvivalencia reláció $\rightarrow$ tranzíciókat ekvivalencia osztályokba csoportosítja
  - Strukturális B-fair reláció $\underset{\not\Leftarrow}{\Rightarrow}$ B-fair reláció

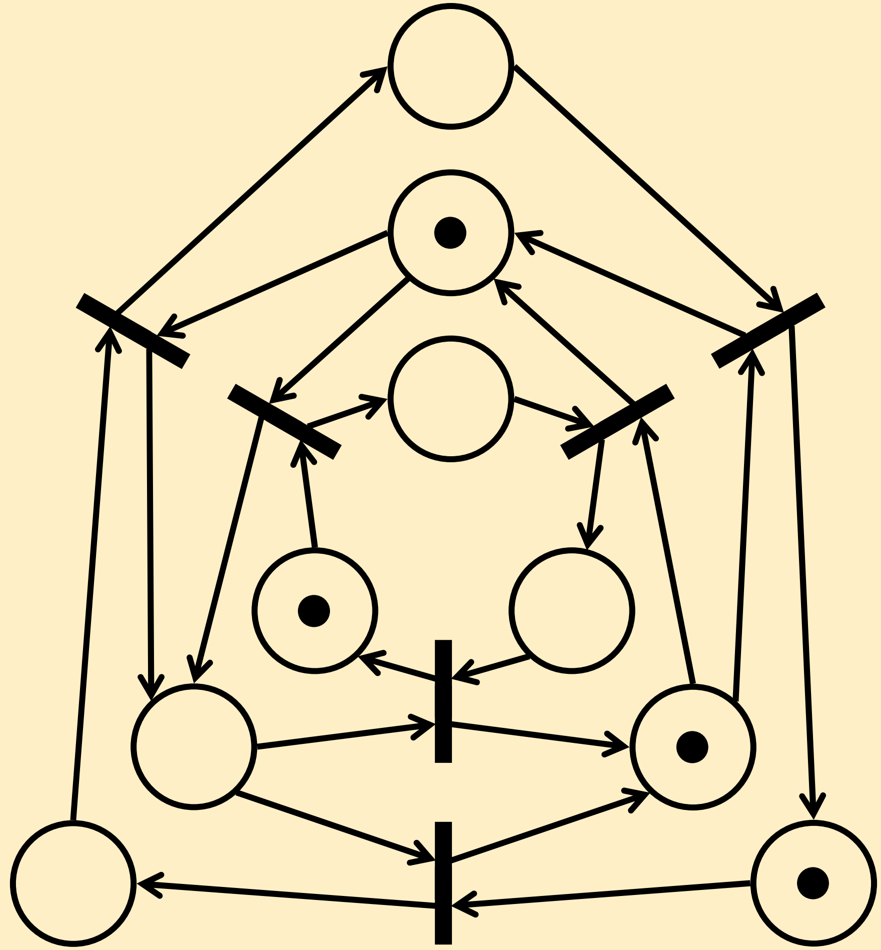
Strukturális B-fairség feltétele

- Egy strukturálisan korlátos Petri háló a.cs.a. strukturálisan B-fair, ha
  - konzisztens és <u>csak egy</u> minimális nemnegatív T-invariánsa van, vagy
  - nem konzisztens és nincs minimális nemnegatív T-invariánsa
- Minden erősen összekötött jelölt gráf strukturálisan B-fair

B-fair, de nem strukturálisan B-fair háló



élő és B-fair M_0



élő, de nem B-fair M_0

Összefoglalás

	Tulajdonság	Szükséges és elégséges felt.
SB	Strukturálisan korlátos	$\exists\vec{\mu} > 0, \mathbf{W}\vec{\mu} \le 0 \;(\text{vagy} \nexists\vec{\sigma} > 0, \mathbf{W}^{\mathrm{T}}\vec{\sigma} \underset{\neq}{\ge} 0)$
CN	Konzervatív	$\exists\vec{\mu} > 0, \mathbf{W}\vec{\mu} = 0 \quad (\text{vagy} \nexists\vec{\sigma}, \mathbf{W}^{\mathrm{T}}\vec{\sigma} \underset{\neq}{\ge} 0)$
PCN	Részlegesen konzervatív	$\exists\vec{\mu} \underset{\neq}{\ge} 0, \mathbf{W}\vec{\mu} = 0$
RP	Ismételhető	$\exists\vec{\sigma} > 0, \mathbf{W}^{\mathrm{T}}\vec{\sigma} \ge 0$
PRP	Részlegesen ismételhető	$\exists\vec{\sigma} \underset{\neq}{\ge} 0, \mathbf{W}^{\mathrm{T}}\vec{\sigma} \ge 0$
CS	Konzisztens	$\exists\vec{\sigma} > 0, \mathbf{W}^{\mathrm{T}}\vec{\sigma} = 0 \quad (\text{vagy} \nexists\vec{\mu}, \mathbf{W}\vec{\mu} \underset{\neq}{\ge} 0)$
PCS	Részlegesen konzisztens	$\exists\vec{\sigma} \underset{\neq}{\ge} 0, \mathbf{W}^{\mathrm{T}}\vec{\sigma} = 0$

További strukturális tulajdonságok

Ha	Akkor
N strukturálisan korlátos és strukturálisan élő	*N* konzervatív és konzisztens.
$\exists \vec{\mu} \geq 0, \mathbf{W}\vec{\mu} \underset{\neq}{\leq} 0$	Létezik nem élő M_0 *N*–hez. *N* nem konzisztens.
$\exists \vec{\mu} \geq 0, \mathbf{W}\vec{\mu} \underset{\neq}{\geq} 0$	(*N*, M_0) nem korlátos egy élő M_0 esetén. *N* nem konzisztens.
$\exists \vec{\sigma} \geq 0, \mathbf{W}^{\mathrm{T}}\vec{\sigma} \underset{\neq}{\leq} 0$	Létezik nem élő M_0 strukturálisan korlátos *N*–hez. *N* nem konzisztens.
$\exists \vec{\sigma} \geq 0, \mathbf{W}^{\mathrm{T}}\vec{\sigma} \underset{\neq}{\geq} 0$	*N* nem strukturálisan korlátos. *N* nem konzervatív.

Példa Petri háló modell analízisére
Alternáló bit protokol

A modellezési feladat

Alternating Bit Protocol

- Átviteli protokoll veszteséges csatornához
  - üzenet elveszhet (véges számú alkalommal)
  - üzenet tartalma nem változhat
- Cél: a protokoll biztosítsa, hogy minden üzenet véges időn belül eljusson a vevőhöz

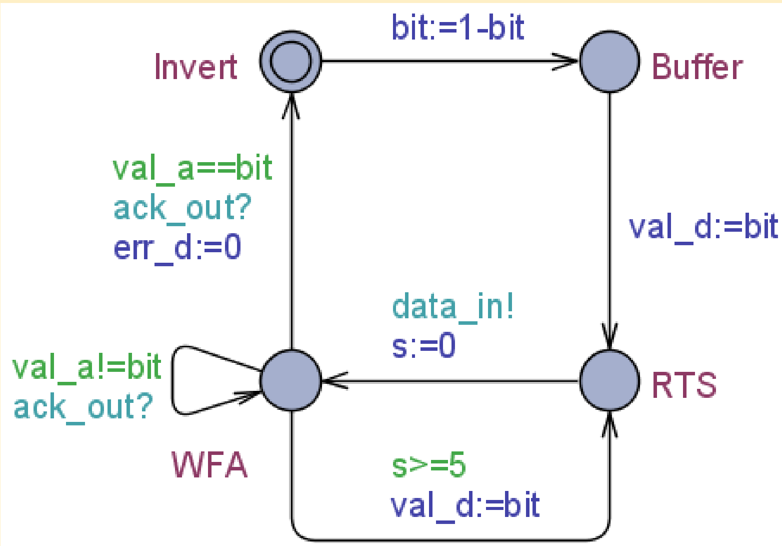
Küldő folyamat

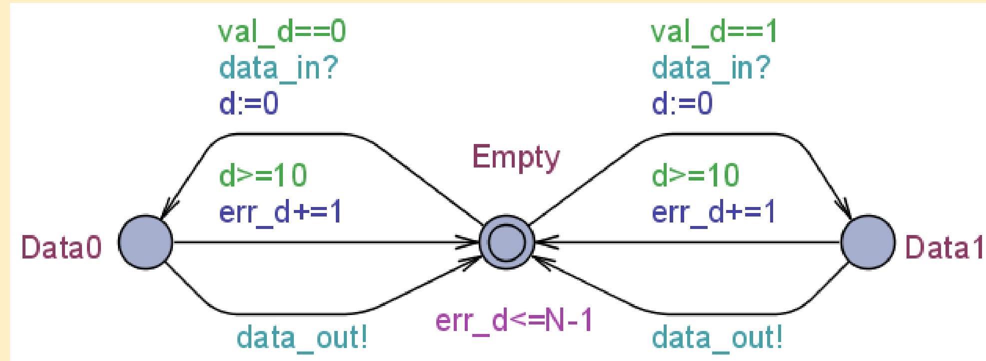
- Üzenetekhez egy ellenőrző bitet kapcsol
- Az üzenetek megérkezését nyugta jelzi
- A nyugta tartalmazza az ellenőrző bitet
- Első üzenethez csatolt bit: $\mathbf{b^0}$
  - ha az üzenet elvész, a folyamat időtúllepéssel észleli a nyugta hiányát $\rightarrow$ újra küldi
  - ha a folyamat $\mathbf{b^0}$ bittel ellátott nyugtát kap (ilyet várt), akkor a következő üzenethez $\mathbf{b^1} = \neg\, \mathbf{b^0}$ bitet köt
  - ha a folyamat $\mathbf{b^X}$ bittel ellátott nyugtát vár és $\mathbf{b^Y}$ bittel ellátott nyugtát kap $\rightarrow$ egyszerűen eldobja

Fogadó folyamat

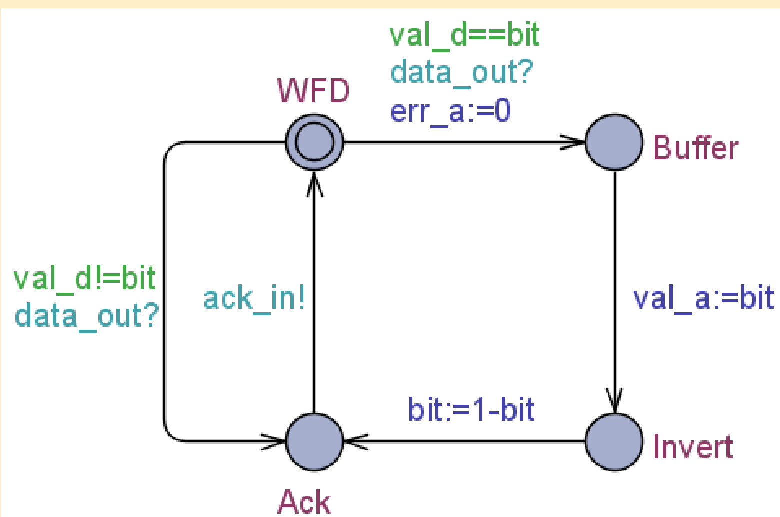
- Első vétel: **b^0** ellenőrző bittel jelölt üzenetet kap
- Az üzenetet feldolgozza, a vételt a kapott bit visszaküldésével nyugtázza
  - ha a következő üzenetben az ellenőrző bit értéke **b^1** (helyesen), akkor az új üzenetet is feldolgozza és a **b^1** bit visszaküldésével nyugtázza
  - ha a következő üzenetben az ellenőrző bit értéke **b^0** (nem megfelelő), akkor az üzenetet eldobja (korábban már feldolgozta), de nyugtát küld

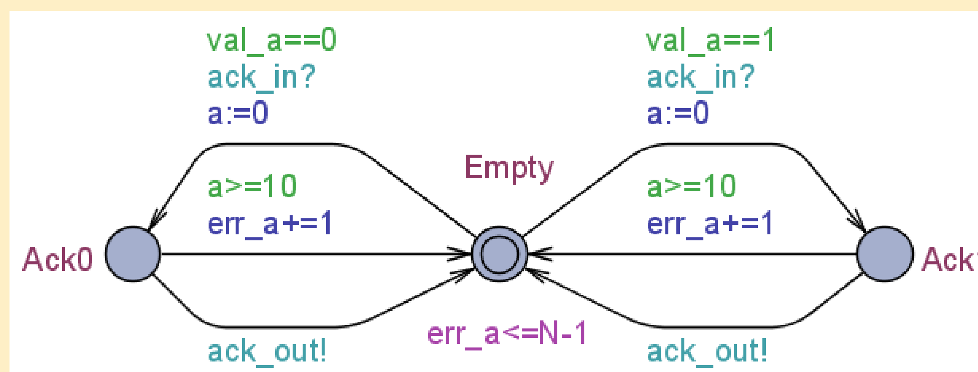
UPPAAL modell



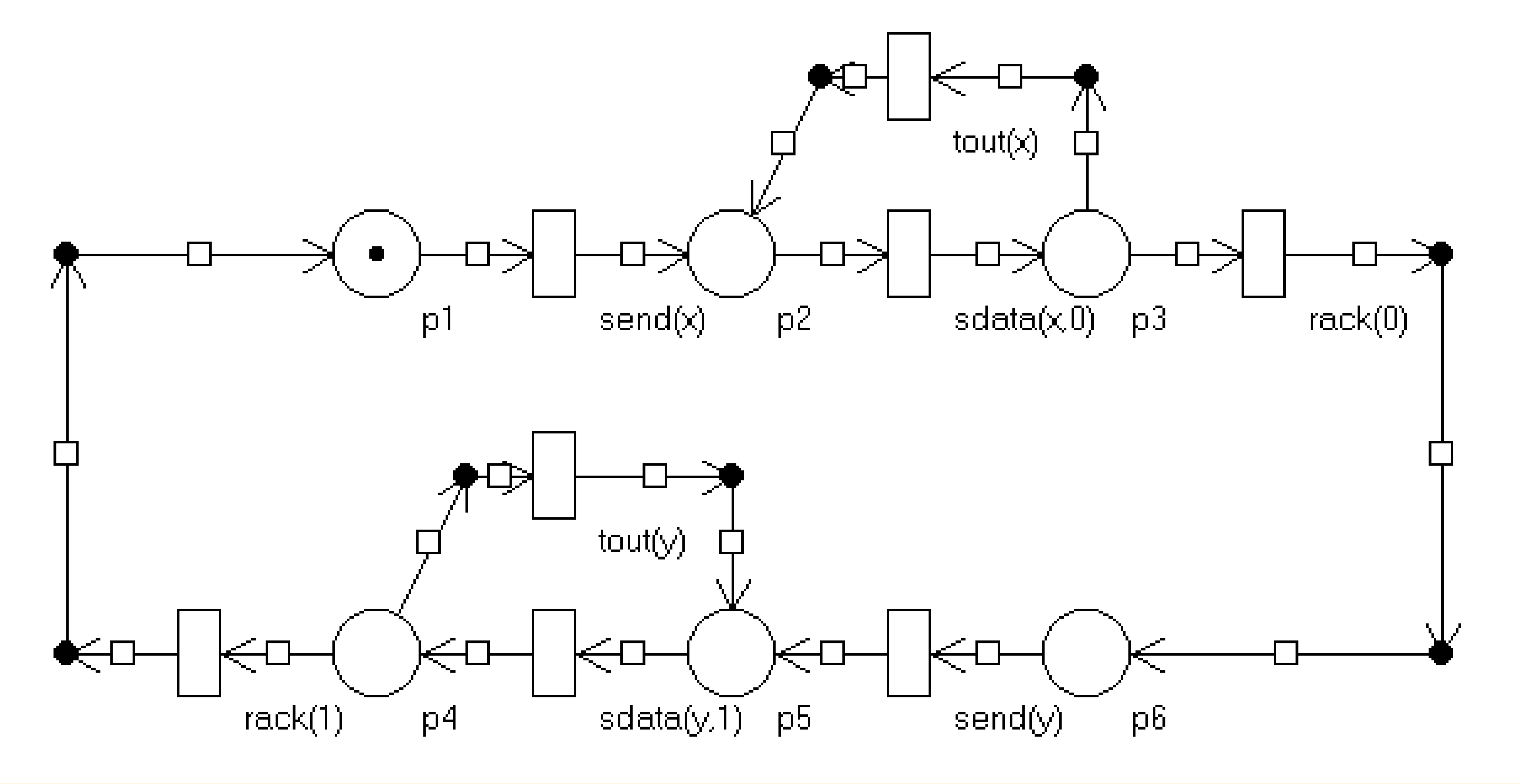



```
const int N=10;

typedef int[0,1] bit_t;
typedef int[0,2] value_t;
typedef int[0,N] error_t;

value_t val_d, val_a;
error_t err_d, err_a;

chan data_in, ack_in;
urgent chan data_out, ack_out;
```

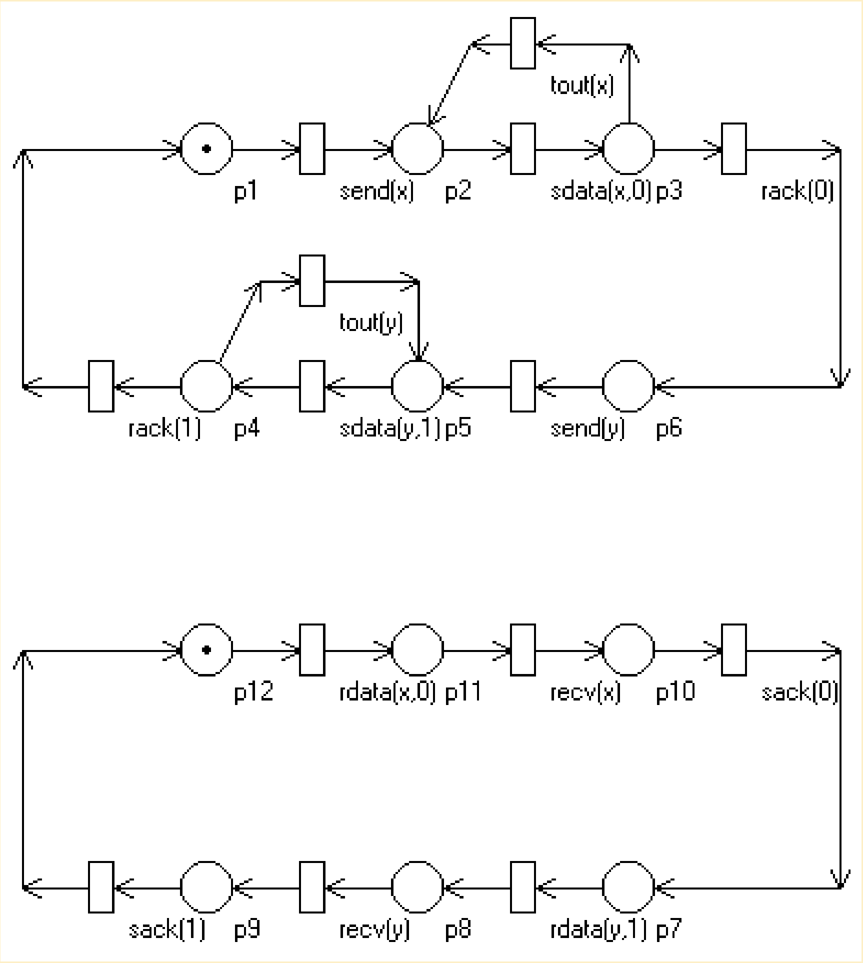




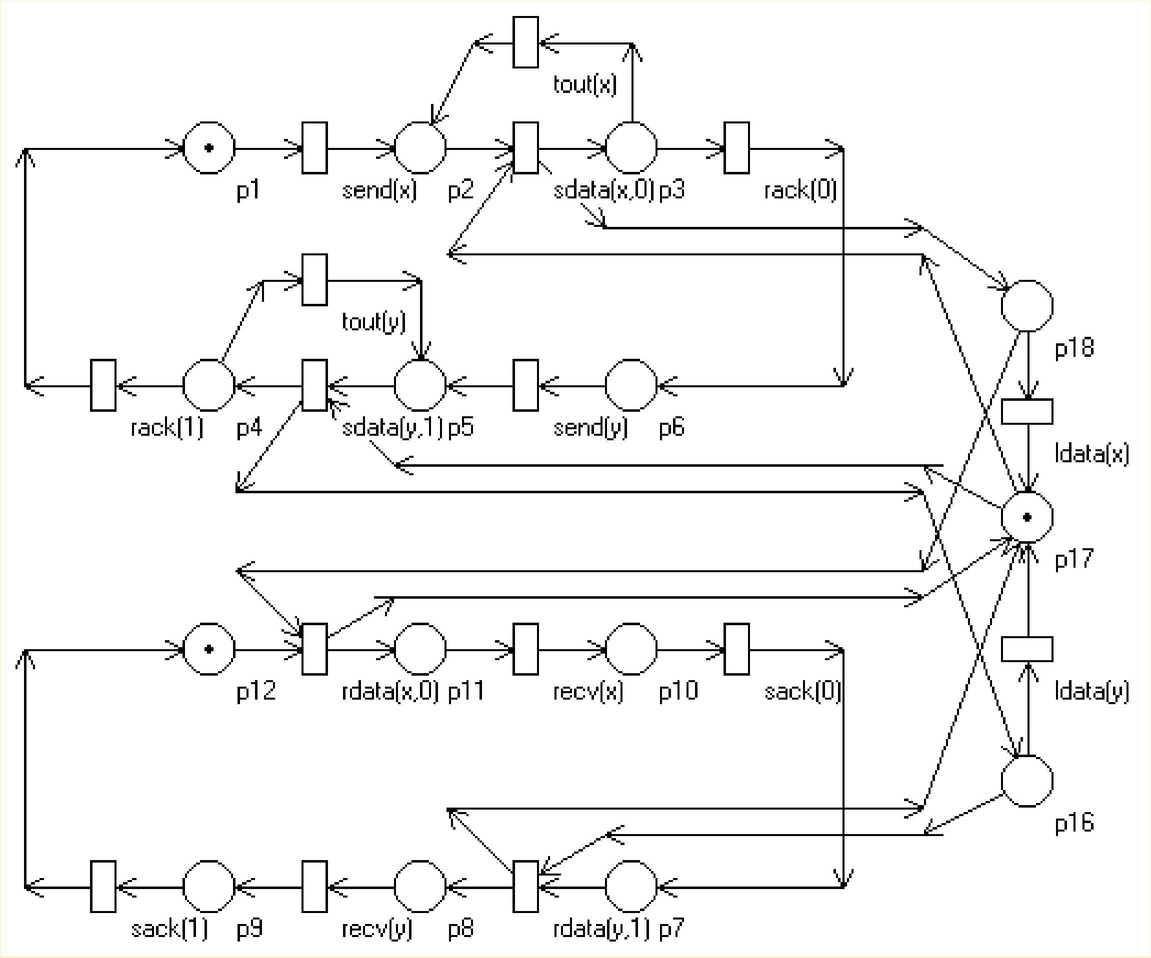
Küldő folyamat Petri háló modellje



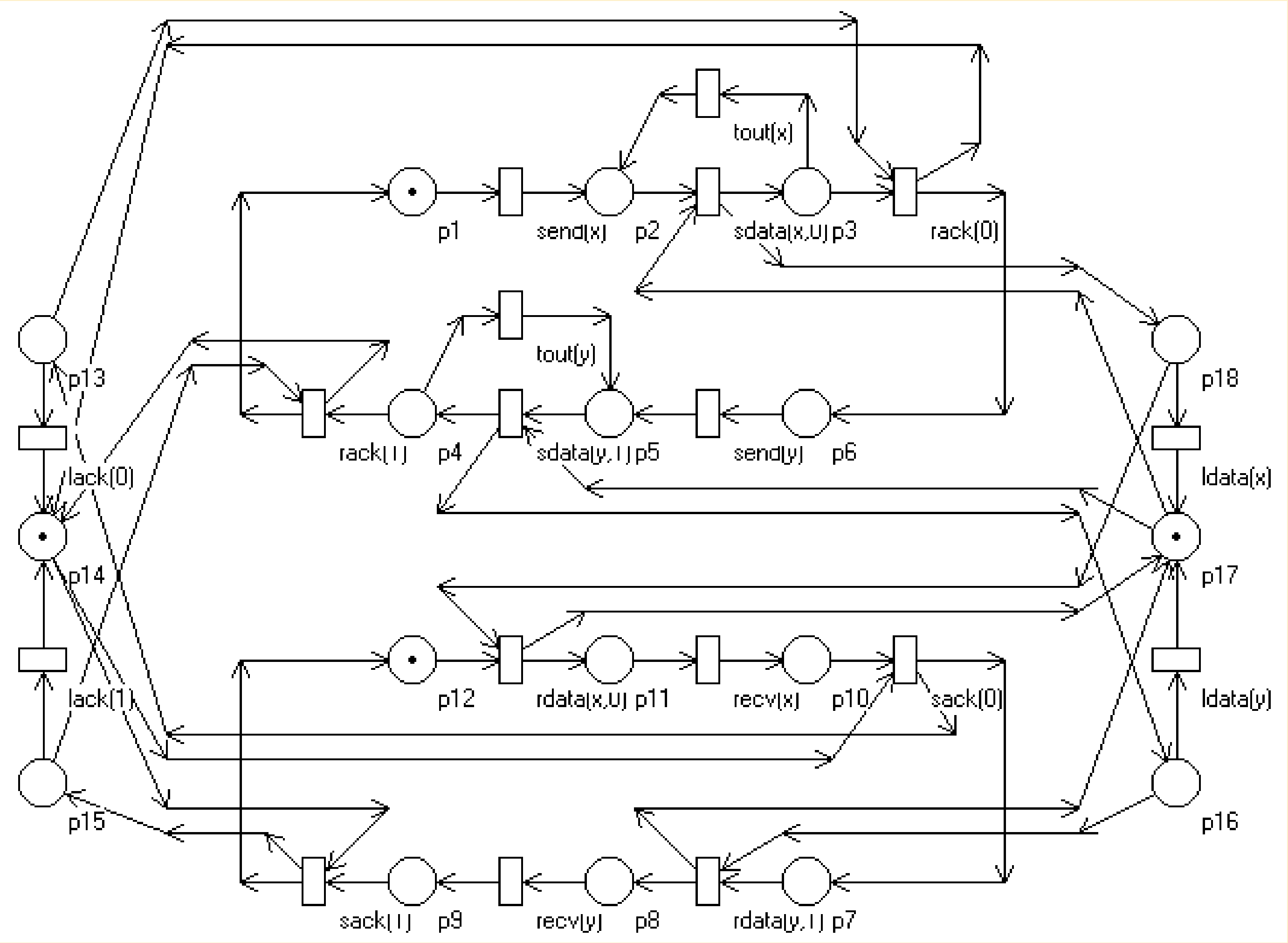
Fogadó folyamat Petri háló modellje



Adat csatorna és az átvitel

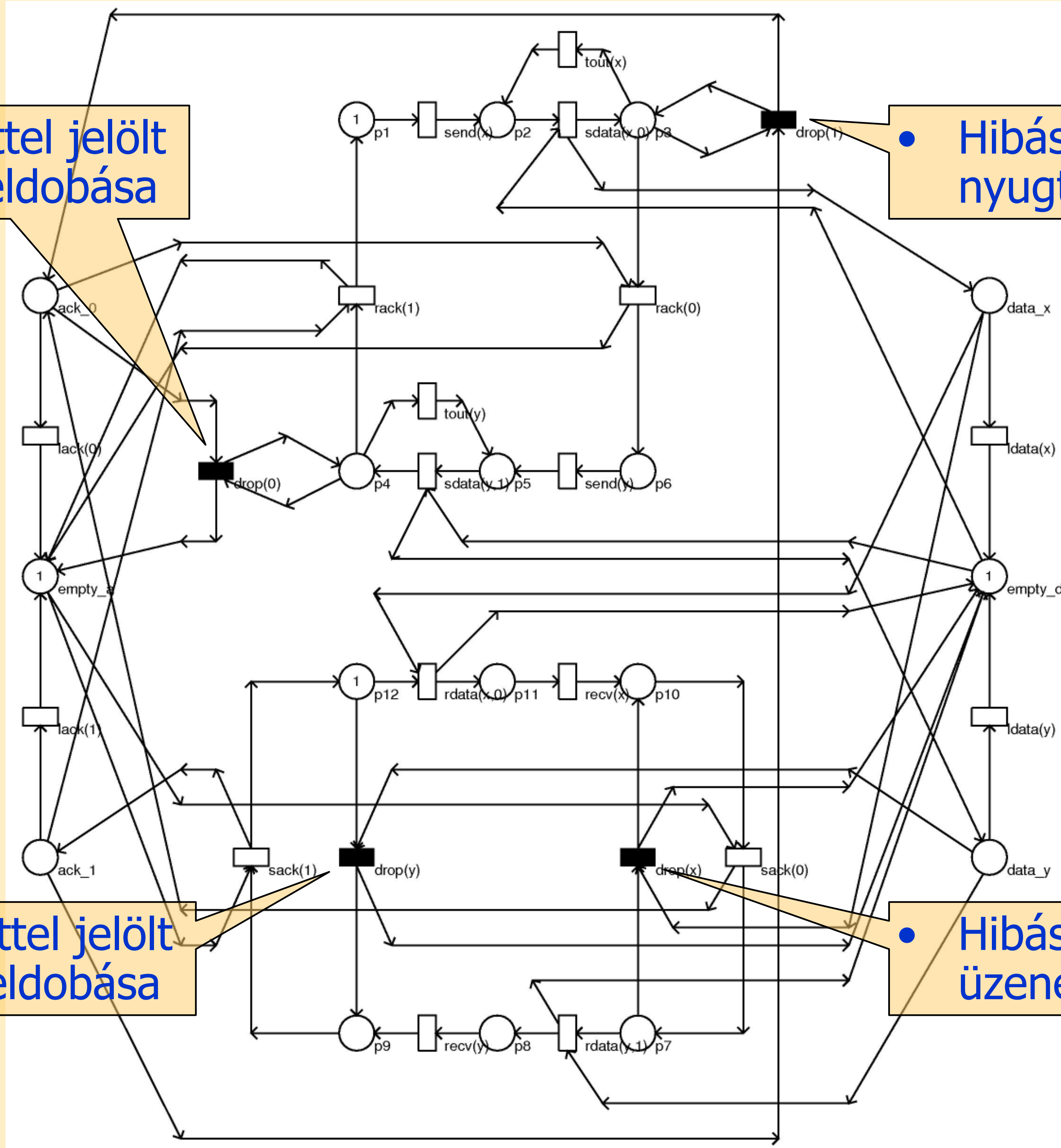


Nyugtázó csatorna



- Hibás bittel jelölt nyugta eldobása

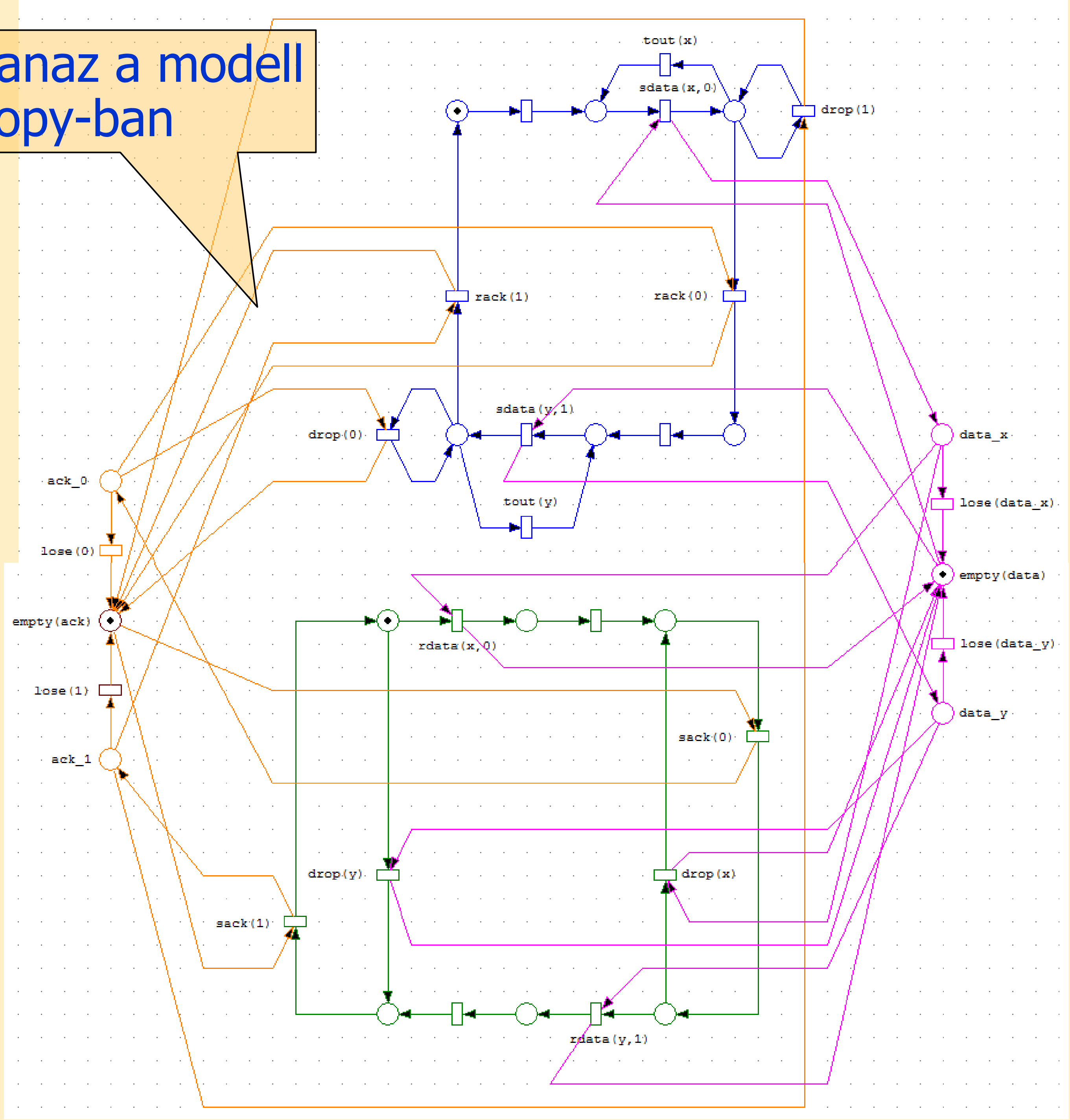
- Hibás bittel jelölt nyugta eldobása



- Hibás bittel jelölt üzenet eldobása

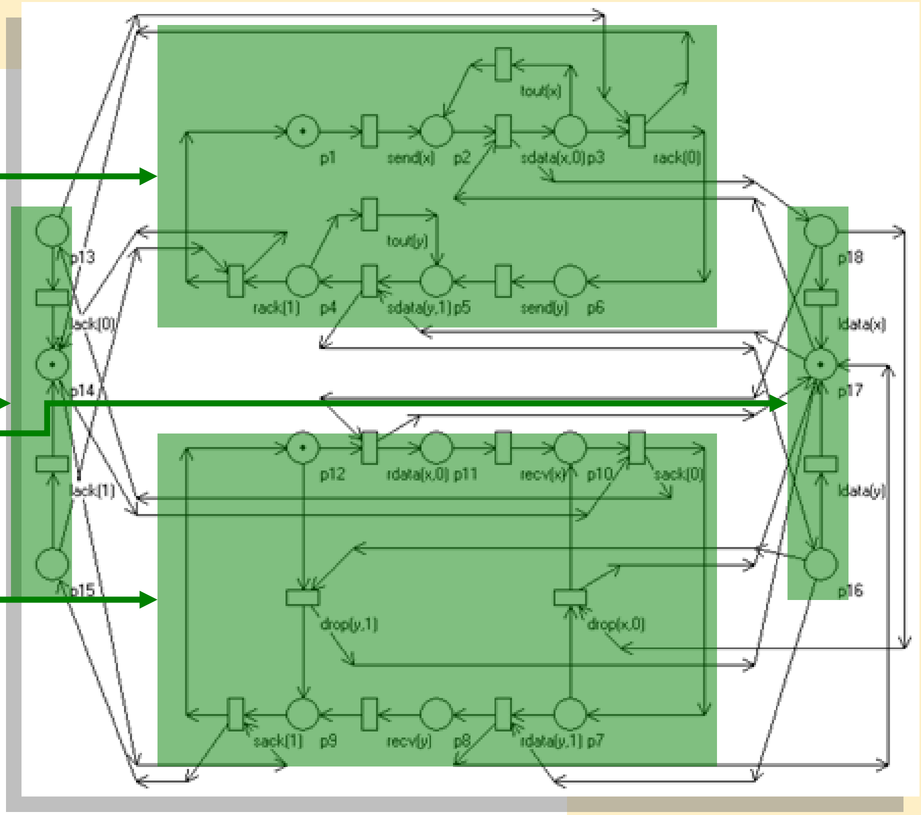
- Hibás bittel jelölt üzenet eldobása

- Ugyanaz a modell Snoopy-ban



DNAnet: Hely invariánsok

```
P-invariants:

1 0 0 0     (main.p1)
1 0 0 0     (main.p2)
1 0 0 0     (main.p3)
1 0 0 0     (main.p4)
1 0 0 0     (main.p5)
1 0 0 0     (main.p6)
0 1 0 0     (main.p7)
0 1 0 0     (main.p8)
0 1 0 0     (main.p9)
0 1 0 0     (main.p10)
0 1 0 0     (main.p11)
0 1 0 0     (main.p12)
0 0 1 0     (main.p13)
0 0 1 0     (main.p14)
0 0 1 0     (main.p15)
0 0 0 1     (main.p16)
0 0 0 1     (main.p17)
0 0 0 1     (main.p18)

ie.

M(main.p1) + M(main.p2) + M(main.p3) + M(main.p4) + M(main.p5) + M(main.p6)
M(main.p7) + M(main.p8) + M(main.p9) + M(main.p10) + M(main.p11) + M(main.p12)
M(main.p13) + M(main.p14) + M(main.p15)
M(main.p16) + M(main.p17) + M(main.p18)

All places are covered by P-invariants.
```



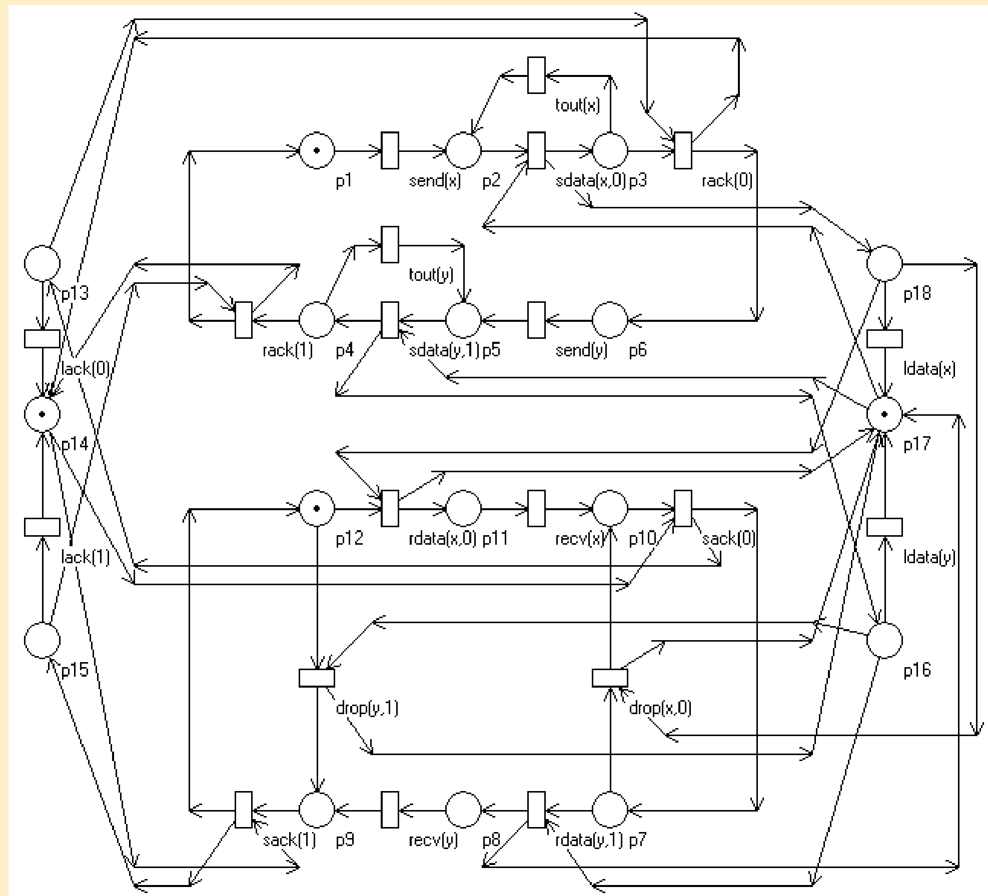
DNAnet: Tüzelési invariánsok

nyugtázás vesztés

adatvesztés

T invariants:

1	0	0	0	0	0	1	(main.send(x))
1	0	1	1	1	0	1	(main.sdata(x,0))
1	0	0	0	0	0	1	(main.rack(0))
1	0	0	0	0	0	1	(main.send(y))
1	1	0	0	1	1	1	(main.sdata(y,1))
1	0	0	0	0	0	1	(main.rack(1))
0	0	1	1	1	0	0	(main.tout(x))
0	1	0	0	1	1	0	(main.tout(y))
1	0	0	0	1	1	1	(main.sack(1))
1	0	0	0	1	0	0	(main.recv(y))
1	0	0	0	1	0	0	(main.rdata(y,1))
1	0	0	1	1	0	1	(main.sack(0))
1	0	0	0	1	0	0	(main.recv(x))
1	0	0	0	1	0	0	(main.rdata(x,0))
0	0	0	1	1	0	0	(main.lack(0))
0	0	0	0	1	1	0	(main.lack(1))
0	1	0	0	0	0	0	(main.ldata(y))
0	0	1	0	0	0	0	(main.ldata(x))
0	0	0	1	0	0	1	(main.drop(x,0))
0	0	0	0	0	1	1	(main.drop(y,1))



elkésett nyugta feldolgozás

kétszeres nyugtázás vesztés (nem tüzelhető)

hibátlan végrehajtás