

Petri-hálók alkalmazása termelési és logisztikai folyamatok modellezésére

Dr. Benkő János, egyetemi tanár

SZIE GTK, Regionális Gazdaságtani és Vidékfejlesztési Intézet, 2100 Gödöllő, Páter K. u.1.

Benkő János

Okleveles mezőgazdasági gépészmérnök (GATE, 1973), a közgazdaságtudomány kandidátusa, egyetemi tanár, a SZIE Gazdaságtudományi Karán a Logisztikai menedzsment mesterképzési szak vezetője. Az anyagmozgatás és a logisztika területén közel 40 éve végez oktató- és kutatómunkát. Főbb szakterületei: anyagmozgató gépek tervezésének elméleti problémái, logisztikai folyamatok matematikai modellezése és szimulációja. Számos cikk, szoftver, egyetemi jegyzet és könyv szerzője. Több, az említett területeken működő bizottság és testület tagja. E-mail: benko.janos@gtk.szie.hu

Abstract: A Carl A. Petriről elnevezett Petri hálókat, ismertté válásuk (1964) óta, elsősorban konkurens és párhuzamos rendszerek struktúrájának és szabályozásának tanulmányozására fejlesztették tovább. Az elmúlt évtizedekben a Petri hálókkal kapcsolatban a legtöbb publikáció a kommunikáció protokollok és a számítógépes rendszerek területén jelent meg, azonban a közelmúltban már a gyártási rendszerek területén is széleskörű érdeklődés mutatkozott irántuk. A tanulmányban a Petri hálók elméletének rövid bevezetése után a műhelyrendszerű és a rugalmas gyártó rendszerek (FMS) modellezésére és elemzésére egyaránt használható alkalmazást mutatunk be. Ezek a hálók jól használhatók gyártási rendszerek működésének tanulmányozására, és olyan fontos információk meghatározását teszik lehetővé, mint a termelékenység, a minimális anyagmozgatógép igény, a gépek kihasználtsága és így tovább.

Kulcsszavak: időzített Petri háló, esemény gráf, folyamatkör, vezérlőkör, műhelyrendszerű gyártás, gépkihhasználás

1. Bevezetés

A Petri hálók szakirodalma az alkalmazások és a modellek széles választékát foglalja magában. A Petri hálók hasznosnak bizonyultak a diszkrét gyártási rendszerek modellezésére is, és alkalmasak e rendszerek validálására, elemzésére és ellenőrzésére. Ez elsősorban annak köszönhető, hogy képesek leírni e rendszerek lényeges jellemzőit. Ilyenek például a konkurencia vagy párhuzamosság (több művelet párhuzamosan folyik), az aszinkron tevékenységek (a műveletek elvégzése változó mennyiségű időt igényel), az osztott erőforrások miatt jelentkező konfliktusok (ugyanaz a gép különböző műveleteket végez), és a diszkrét-esemény folyamatok (a gyártási folyamat úgy tekinthető, mint diszkrét események egy sorozata). A tanulmányban elemzett integrált gyártási és logisztikai probléma modellezésére a Petri hálók egy speciális osztályát, az ún. időzített esemény gráfokat fogjuk használni. Először azonban a tanulmány megértéséhez rövid áttekintést adunk a Petri hálók elméleti alapjairól.

1.1. A Petri hálók terminológiája

A Petri háló egy irányított, súlyozott páros gráf, amely két csomóponttípust tartalmaz. Az egyik csomóponttípus a helyeket és a másik típus az átmeneteket jelöli. A helyeket és az átmeneteket irányított élek kötik össze. Formálisan egy Petri háló három véges halmazból áll, $N=(P, T, F)$, ahol a $P=\{p_1, p_2, \dots, p_n\}$ a **helyek** véges halmaza, $T=\{t_1, t_2, \dots, t_m\}$ az **átmenetek** véges halmaza és $(P \times T) \cup (T \times P) \supset F$ az **irányított élek** halmaza.

Grafikusan a helyeket körök jelzik, és az átmeneteket álló téglalapok, ahogy azt az 1. ábra mutatja. Egy átmenet bemeneti, illetve kimeneti helyeit az átmenet bejövő, illetve kimenő élei kapcsolják össze. A t_j , ($j=1, 2, \dots, m$) átmenet bemeneti helyeinek halmazát $Inp(t_j)$ jelöli. Hasonlóképpen $Out(t_j)$ jelöli a t_j átmenet kilépő helyeinek halmazát.

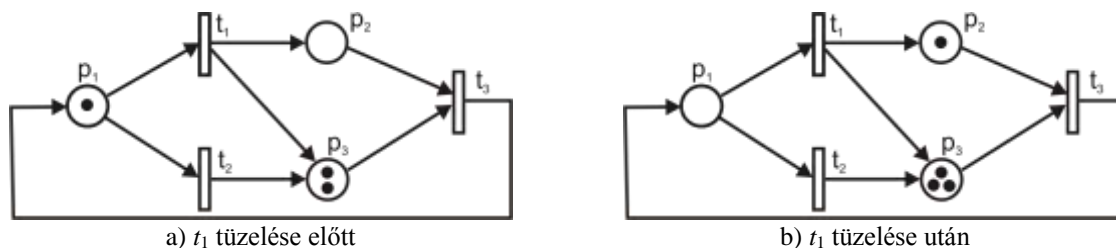
Egy Petri háló N **jelölése** (M) egy függvény:

$$M : P \rightarrow \{0, 1, 2, \dots\} \text{ (pozitív egészs számok halmaza),}$$

amely a helyek állapotát jelző **tokeneket** (jeleket), nem negatív számokat rendel a hálózat minden p_i , ($i=1, 2, \dots, n$) helyéhez. Speciálisan $M^0(p_i)$ jelenti a kezdeti jelölést. A tokeneket a

helyeket ábrázoló körökbe rajzolt pontok reprezentálják, például az 1. ábrán az $M^0(p_3)=2$.

Az i -edik hely állapotát a benne lévő tokenek száma jelzi, és a hálózat állapota az egyes helyek állapotainak az összessége. A Petri háló különleges tulajdonsága, hogy a jelölés (a tokenek száma) az átmenetek „tüzelésével” megváltoztatható.



1. ábra: Egy jelölt Petri háló változása a t_1 tüzelés hatására (Forrás: saját szerkesztés)

A t_j átmenetről akkor, és csak is akkor mondjuk, hogy az M jelöléssel engedélyezett (vagy tüzelhető), ha minden $p_i \in \text{Inp}(t_j)$ bemeneti helye legalább egy tokenet tartalmaz, vagyis az $M(p_i) > 0$. Amikor a t_j átmenet tüzel, akkor egy tokenet eltávolítunk minden bemeneti helyéről és egy tokenet hozzáadunk minden kimeneti helyéhez. Ez a változás határozza meg a hálózat új jelölését, az elérhető átmenetek új halmazát, és így tovább.

Például az 1/a. ábrán látható Petri hálón a t_1 és t_2 is engedélyezett az $M^0 = [1, 0, 2]$ jelöléssel, de a t_3 nem, mivel annak egyik bemeneti helye, a p_2 nem tartalmaz tokenet. A t_1 tüzelése eltávolít egy tokenet a p_1 bemeneti helyről, és hozzáad egy tokenet minden kimeneti helyhez, p_2 -hez és p_3 -hoz. Az eredményként kapott új jelölést, $M^1 = [0, 1, 3]$ az 1/b. ábra szemlélteti.

Fontos tudni, hogy egy hálózat fejlődése radikálisan különböző lehet, attól függően, hogy melyik átmenetet választjuk ki tüzelésre. Azt is hangsúlyozni kell, hogy a Petri hálón nem feltétlenül garantált a tokenek számának a megőrzése, mivel minden tüzelés után a tokenek száma a hálózatban megváltozhat.

Az átmenetek tüzelésével elérhető ún. „token játék” teszi lehetővé a dinamikus viselkedésű rendszerek modellezését a Petri hálókkal. Egy átmenet tüzelése úgy értelmezhető, mint egy esemény bekövetkezése, ami elnyel bizonyos erőforrásokat és létrehoz másokat. Ebből a szempontból a tokenek erőforrásokat reprezentálnak, amelyek bizonyos helyeken egy bizonyos állapotban vannak.

1.2. Időzített esemény gráfok

Az **időzített** Petri háló bevezetése Ramchandani nevéhez köthető [8]. Ebben a modellben a tüzelési idő függvény definíciója: $\tau: T \rightarrow \mathbf{R}^+$ (pozitív racionális számok halmaza), amely egy pozitív (racionális) számot $\tau(t)$ társít minden egyes t átmenethez.

A korábban leírtakhoz képest a tüzelési szabályok az időzített Petri hálókon nem változnak, kivéve, hogy minden t_j átmenethez tartozik $\tau(t_j)$ valós tüzelési időtartam. A tüzelési időről azt feltételezzük, hogy determinisztikus. Megjegyezzük, hogy léteznek sztochasztikus Petri háló is, a tanulmányban azonban csak a determinisztikus modellre koncentrálnak.

Esemény gráfnak (más néven a döntésmentes Petri hálónak) nevezzük azt a Petri hálót, amelynek minden p_i helyéhez pontosan egy bemeneti és egy kimeneti átmenet tartozik. Ennek eredményeként, a Petri háló működése teljesen determinisztikus, mivel a tüzelési folyamathoz nem kell döntést hozni (egy engedélyezett átmenetet nem lehet kikapcsolni egy másik átmenet tüzelésével). Ez a tulajdonság az átmenetek időzítésével lehetővé teszi a hálózat dinamikus viselkedésének teljes jellemzését.

Egy esemény gráfon az **elemi kör** egy irányított út, amely valamely csomópontból (hely vagy

átmenet) indul és visszavezet az indulási csomóponthoz úgy, hogy a csomópontokat csak egyszer érinti. A továbbiakban az elemi köröket $\beta_k=(p_1, t_1, \dots, p_n, t_n)$ -vel jelöljük.

Az esemény gráfok fontosabb tulajdonságai [1]:

1. tulajdonság: A tokenek száma egy elemi körben invariáns (változatlan) bármely átmenet tüzelésére.

2. tulajdonság: Egy esemény gráf élő, ha minden elemi körben legalább egy token létezik.

Az 1. tulajdonság azt jelenti, hogy minden M lehetséges jelöléshez tartozó összes tokenek száma a β_k körökben invariáns, ezért egyenlő a kezdeti jelöléssel, vagyis

$$M(\beta_k) = M^0(\beta_k).$$

Azonnali következmény az is, hogy ha az esemény gráf erősen összefüggő, ami azt jelenti, hogy bármelyik csomópontból bármely más csomópontba vezet egy irányított út, akkor a gráf korlátos. Ténylegesen minden egyes p_i hely szükségszerűen legalább egy elemi kör része, és az 1. tulajdonság miatt, egy adott p_i helyen, az egyidejűleg jelenlévő tokenek száma soha nem haladhatja meg a körben lévő összes tokenek számát.

1.3. Az időzített esemény gráfok analízise

Most összefoglaljuk a legfontosabb eredményeket, amelyeket később használni fogunk gyártási probléma tanulmányozásakor. Emlékeztetünk arra, hogy $\tau(t_j)$ (pozitív racionális szám) jelöli a t_j átmenet a tüzelési idejét. A β_k körben az átmenetek összes tüzelési ideje legyen $\tau(\beta_k)$, azaz

$$(1) \quad \tau(\beta_k) = \sum_{j=1}^m \tau(t_{kj}) \text{ (időegység).}$$

$M(\beta_k)$ jelölje a β_k körben a tokenek számát:

$$(2) \quad M(\beta_k) = \sum_{i=1}^n M^0(p_{ki}) \text{ (token),}$$

ahol M^0 a kezdeti jelölés. Végül vezessük be az egy tokenre eső tüzelési időt, amit a β_k kör ciklusidejének nevezünk:

$$(3) \quad C(\beta_k) = \frac{\tau(\beta_k)}{M(\beta_k)} \text{ (időegység/token).}$$

Adott kezdeti jelölésről tegyük fel most, hogy az engedélyezett átmenetek tüzelése biztosítja a rendszer folyamatos működését. Amint korábban említettük, a „token játék” az esemény gráfokon teljesen determinisztikus, amely garantálja az előredefiniált viselkedést. A következő két tulajdonság az erősen összefüggő esemény gráfokra vonatkozik:

3. tulajdonság: A rendszer véges idő után állandósul, periodikus működésűvé válik.

4. tulajdonság: A ciklusidőt állandósult állapotban az elemi körök ciklusidőinek a maximális értéke határozza meg, azaz

$$(4) \quad C = \text{Max}\{C(\beta_k)\} \text{ (időegység/token).}$$

Állandósult állapotban alternatív módon használható a **termelési ráta** is, amely az időegységre eső tüzelhető átmenetek száma:

$$(5) \quad \lambda = 1/C = \text{Min}\{1/C(\beta_k)\} \text{ (token/időegység).}$$

Azt a β_k^* kört, amelyre a ciklusidő, $C = C(\beta_k^*)$ maximum, kritikus körnek nevezzük. A kritikus kör ciklusideje határozza meg a rendszerben átbocsátóképességének felső határát. Megjegyezzük, hogy ha egy körben nincs token ($M(\beta_k)=0$), akkor a teljesítménye nulla (a ciklusidő végtelen), ez nem meglepő, hiszen a háló a 2. tulajdonság szerint halott.

Miután minden elemi kör ismert, egyértelműen kiszámítható az állandósult állapot teljesítménye. A (3)-(4) egyenletek szerint a teljesítmény függ a kezdeti állapottól (mivel a kezdeti jelölés az egyes körökben egyedileg határozza meg a tokenek számát), és természetesen az átmenetek tüzelési idejétől is. Megjegyezzük, a periodikus működés (állandósult állapot) általában csak az átmenetek K számú, egymást követő tüzelése után érhető el.

A továbbiakban ezeket az eredményeket felhasználjuk a műhelyrendszerű termelés teljesítményelemzésére.

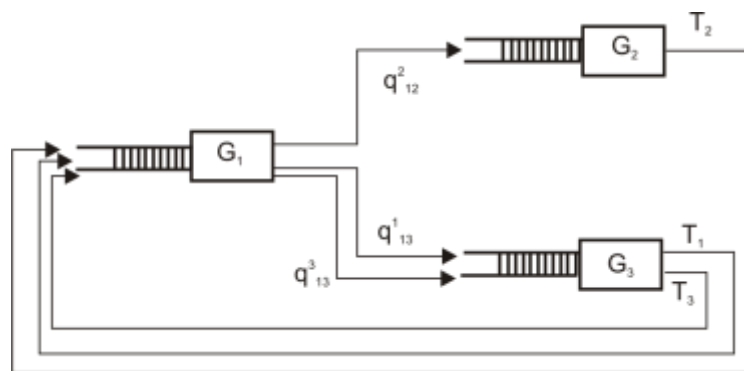
2. A műhelyrendszerű termelés Petri háló modellje

2.1. A modell leírása

Tekintsünk egy integrált gyártási és logisztikai feladatot, amelynek elvégzéséhez G_1, G_2, \dots, G_m -mel szimbólumokkal jelölt, m számú gép áll rendelkezésünkre, és segítségükkel T_1, T_2, \dots, T_n -nel jelölt, n számú, különböző terméktípust kell előállítani meghatározott, $\alpha_1, \alpha_2, \dots, \alpha_n$ arányban. A rendszerben feltételezzük, hogy egy terméktípus előállítása előírt sorrendben, több gépen elvégzendő műveletekből áll, és a műveleti idők adottak, valamint determinisztikusak. A terméktípusok termelési útvonala, ami a műveletek sorrendjét vagyis a gépek felkeresési sorrendjét jelenti, egyedileg meghatározott. A terméktípusok gyártásáról feltételezzük az ismétlődést vagy ciklikusságot, továbbá, hogy meg kell felelniük az előírt $\alpha_1, \alpha_2, \dots, \alpha_n$ termékösszetételnek.

A leírt feladat egy lehetséges változatát a 2. ábra szemlélteti. A vizsgált gyártási rendszer a G_1, G_2 és G_3 gépekből áll, amelyek T_1, T_2 és T_3 terméktípusokon előírt sorrend szerint különböző műveleteket végeznek. Az 1. táblázat foglalja össze, hogy a T_i terméktípus megmunkálása a G_j gépen mennyi időt igényel. A táblázatból a **műveleti sorrendek**, vagyis a termelési útvonalak is kiolvashatók:

(I) $T_1: G_1, G_3; \quad T_2: G_1, G_2; \quad T_3: G_1, G_3.$



2. ábra: A vizsgált termelési rendszer (Forrás: saját szerkesztés)

1. táblázat

A gyártási rendszer műveleti idői

Munkadarab	Termékösszetétel (α_i)	Műveleti idő (τ_{ij})		
		G_1	G_2	G_3
T_1	1/3	1	–	2
T_2	1/3	3	8	–
T_3	1/3	2	–	4

A gyártási rendszerben a gépek között targoncák mozgatják a raklapokra helyezett terméktípusokat (egy targonca egy raklapot, egy raklap egy terméktípust hordoz). Azt is feltételezzük, hogy a terméktípusok gyártása azonos arányban ($\alpha_1=\alpha_2=\alpha_3$) történik. Így a termékösszetétel-

ben minden terméktípus $1/3$ részt képvisel.

Vezessük be a G_j ($j=1, 2, 3$) géphez tartozó v_j **igénybevételi ráta** fogalmát, ami azt mutatja, hogy a terméktípusok milyen arányban keresik fel a G_j gépet. Például, a G_2 gépet a három terméktípus közül csak a T_2 veszi igénybe (*1. táblázat*), így a $v_2=1/3$. Az igénybevételi ráták külön-külön:

$$v_1 = 1, \quad v_2 = \frac{1}{3}, \quad v_3 = \frac{1}{3} + \frac{1}{3} = \frac{2}{3}.$$

A G_j ($j=1, 2, 3$) gép egy terméktípusra eső **átlagos műveleti ideje** a μ_j átlagos kiszolgálási ráta reciprok értéke, azaz $1/\mu_j$, amely ugyancsak az *1. táblázatból* számítható. Például a G_3 gépet csak a T_1 és T_3 terméktípusok gyártásához használjuk 2 és 4 műveleti időkkel, így az $1/\mu_3=(2+4)/2=3$. Ezek alapján az átlagos műveleti vagy kiszolgálási idők:

$$\frac{1}{\mu_1} = \frac{1+3+2}{3} = 2, \quad \frac{1}{\mu_2} = 8, \quad \frac{1}{\mu_3} = \frac{2+4}{2} = 3 \text{ időegység.}$$

A modell fontos jellemzője a **minimális terméktípus halmaz** (MTH) fogalma, amely a termékösszetételnek megfelelő arányban és a legkisebb számban tartalmazza a terméktípusokat. A példánkban az MTH egyszerűen:

$$S = \{T_1, T_2, T_3\},$$

mivel az összes terméktípust azonos arányban gyártjuk. Világos, hogy a terméktípus halmaz ciklikus (ismétlődő) gyártása garantálja, hogy a termékösszetétel minden esetben teljesüljön.

Megjegyezzük, hogy a termékösszetételtől függően ugyanaz a terméktípus többször is megjelenhet az MTH-ban. Például a következő termelési arányok ($1/2, 1/4, 1/4$) szerint a T_1 -ből kétszer annyit termelünk, akkor az ennek megfelelő MTH $\{T_1, T_1, T_2, T_3\}$.

A termelési cél eléréséhez elegendő az MTH-ban felsorolt valamennyi terméktípust az előírt sorrendben ciklikusan gyártani. Például a G_1, G_2 és G_3 gépekre választhatjuk a következő terméktípus sorrendet:

$$(II) \quad S_1 = \{T_1, T_2, T_3\}, \quad S_2 = \{T_2\}, \quad S_3 = \{T_1, T_3\}$$

A (II) szekvenciák a terméktípus gyártásának egy lehetséges alternatíváját képviselik, ténylegesen a munkák elvégzésének összes permutációja elfogadható. Ezekre a szekvenciákra a továbbiakban, mint a **gépszekvenciákra** hivatkozunk.

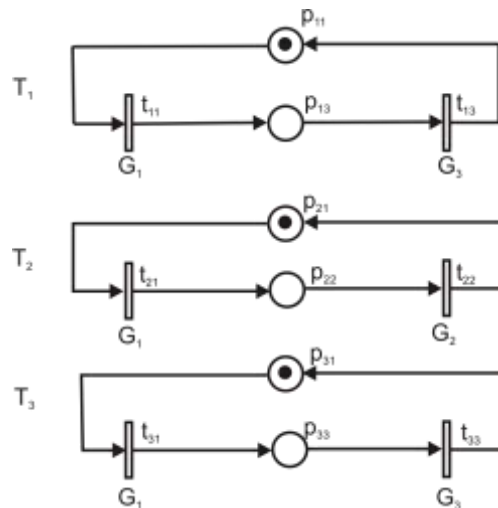
Végül a modellezéshez a termelési rendszerünket két egyértelműen szétválasztható szegmensre bontjuk: **operatív szegmens**, amely a műveleti sorrendekkel (termelési útvonalakkal), és **vezérlő szegmens**, amely a gépszekvenciák által meghatározott [2]. A Petri háló hasznos eszköz ahhoz, hogy leírja e rendszer alapvető jellemzőit. Először az operatív szegmens modellezési lehetőségeinek tárgyalásával kezdünk, majd a vezérlő részt is bemutatjuk.

2.2. Az operatív szegmens

Az operatív szegmensben minden elemi körhöz egy terméktípus tartozik, ezért ezeket **termelési** vagy **folyamatköröknek** nevezzük. A folyamatkörök egy-egy terméktípus ciklikus gyártási folyamatát írják le (*3. ábra*). A termelési útvonalak az (I)-ben megadott műveleti sorrendet követik. A modellezés szabályok a következők:

Minden t_{ij} átmenet megfelel egy gép által végrehajtott műveletnek. Például a t_{11} átmenet jelenti a T_1 termék gyártási folyamatában az első műveletet, amit a G_1 gép végez el. Értelemszerűen az átmenet tüzelési időtartama megfelel a $\tau(t_{11})=1$ műveleti időnek (*1. táblázat*).

A folyamatkörökben a tokenek modellezik a gépek kiszolgálását. A p_{ij} helyeken az elvégzésre váró munkákat fizikailag a tokenek képviselik. A helyek így tároló pufferként is értelmezhetők, ennek megfelelően ezeket **puffer helyeknek** is nevezzük. Azt feltételeztük, hogy tárgoncák mozgatják a raklapokra helyezett terméktípusokat, és a modellben egy token egy tárgoncát vagy raklapot szimbolizál. A tokenek a folyamatkörökben keringenek, és az adott terméktípus ciklikus gyártását generálják. A p_{ij} helyek kezdeti jelölése biztosítja a műveleti sorrendet, vagyis a műveletek közötti elsőbbségi feltételeket.

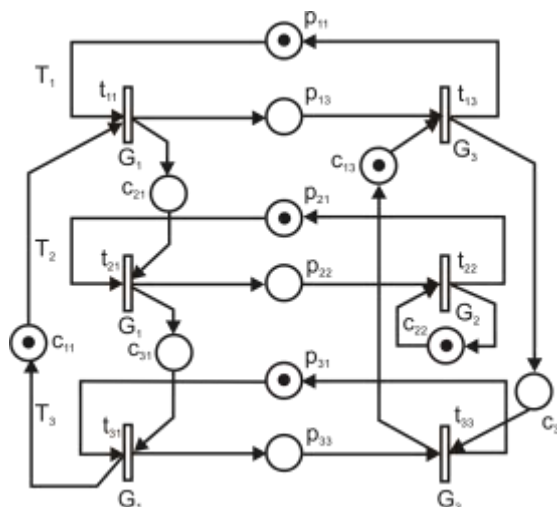


3. ábra: Folyamatkörök a példában (Forrás: saját szerkesztés)

Megjegyezzük, a modell nem követeli meg, hogy a terméktípusokhoz rendelt raklapok száma megfeleljen a termelési arányoknak. A következő szakaszban tárgyalt vezérlőszegmens ugyanis garantálja, hogy a rendszerben a raklapok eloszlásától függetlenül, a termékösszetétel raklapigényei mindig kielégítettek legyenek.

2.3. A vezérlőszegmens

Ahhoz, hogy a gépeken a terméktípusok gyártásának sorrendjét modellezhessük, minden átmenet (mint tudjuk, ezek egy-egy művelet elvégzését képviselik egy adott gépen) egy **vezérlőkörnek** nevezett körhöz is kapcsolódik. Az átmenetek sorrendjét a vezérlőkörben a gép-szekvencia határozza meg, vagyis az, hogy az adott gépen a terméktípusokat milyen sorrendben kell gyártani.



4. ábra: Folyamat- és vezérlőkörök (Forrás: saját szerkesztés)

A vezérlőkörök tehát az adott géphez tartozó gépszekvencia szerint működnek. Például a (II)

előírás szerinti sorrend: $S_1 = \{T_1, T_2, T_3\}$, $S_2 = \{T_2\}$, $S_3 = \{T_1, T_3\}$, és az ezeknek megfelelő vezérlőkörök a folyamatkörökkel együtt a 4. ábrán láthatók.

Hogy megkülönböztessük a vezérlőkörök helyeit folyamatkörök helyeitől, az előbbieket **vezérlőhelyeknek** fogjuk nevezni, és c_{ij} -vel jelöljük. Míg a puffer helyek az operatívsegmens állapotát jellemzik, addig a vezérlőhelyek egy gép lehetséges állapotait írják le.

Fontos felismerni, hogy a tokenek a vezérlőkörökben, bár látszólag nem különböznek a folyamatkörök tokenjeitől, speciális tulajdonsággal bírnak, nevezetesen: minden vezérlőkörben egy és csak is egy token kering (eltérően a folyamatköröktől, amelyek több tokent is tartalmazhatnak), ami azt biztosítja, hogy egyidejűleg egy gép csak egy műveletet végezzen.

Továbbá, egy vezérlőkör egyedüli tokenjének a helyzete egyértelműen meghatározza a gép állapotát, ha a token egy átmenet belsejében helyezkedik el, az azt jelenti, hogy az átmenet tüzelt, és a gép használatban van (a feladat végrehajtása abban a folyamatkörben folyik, amelyhez a tüzelt átmenet tartozik). Ha a gép „tétlen”, akkor a token valamelyik vezérlőhelyen van (a gép a gépszekvencia szerinti következő feladatra várakozik). Szintén fontos megjegyezni, hogy a vezérlőhelyek kezdeti jelölését a munkasorrendben az első feladat határozza meg. Például kezdetben a G_1 gép vezérlőkörében a token a c_{11} helyen tartózkodik (4. ábra), mivel a G_1 gépen az első művelet a T_1 terméktípus megmunkálása.

3. A műhelyrendszerű termelés működésének elemzése

Egyértelmű, hogy a műhelyrendszerű termelés Petri háló modellje (4. ábra) erősen összefügg az időzített esemény gráffal. Így az 1.3. szakaszban közölt eredmények felhasználásával elemezhetjük a rendszer működését.

Az első lépés a hálózat elemi köreinek meghatározása. A folyamat- és a vezérlőkörök a korábbi leírásból már ismertek. Eltekintve ezektől az alapköröktől léteznek további elemi körök, amelyek puffer- és vezérlőhelyeket egyaránt tartalmaznak. Ezeket a köröket **vegyes köröknek** nevezik. A gyártási rendszerünk modelljében (4. ábra) nyolc elemi kör található.

Folyamatkörök:

$$\begin{aligned}\beta_1 &= (p_{11}, t_{11}, p_{13}, t_{13}), & (T_1 \text{ terméktípus}), \\ \beta_2 &= (p_{21}, t_{21}, p_{22}, t_{22}), & (T_2 \text{ terméktípus}), \\ \beta_3 &= (p_{31}, t_{31}, p_{33}, t_{33}), & (T_3 \text{ terméktípus}).\end{aligned}$$

Vezérlőkörök:

$$\begin{aligned}\beta_4 &= (c_{11}, t_{11}, c_{21}, t_{21}, c_{31}, t_{31}), & (G_1 \text{ gép}), \\ \beta_5 &= (c_{22}, t_{22}), & (G_2 \text{ gép}), \\ \beta_6 &= (c_{13}, t_{13}, c_{33}, t_{33}), & (G_3 \text{ gép}).\end{aligned}$$

Vegyes körök:

$$\begin{aligned}\beta_7 &= (c_{11}, t_{11}, p_{13}, t_{13}, c_{33}, t_{33}, p_{31}, t_{31}), \\ \beta_8 &= (c_{31}, t_{31}, p_{33}, t_{33}, c_{13}, t_{13}, p_{11}, t_{21}, c_{21}, t_{21}).\end{aligned}$$

A példában az összes elemi kört az ábrából közvetlen meg lehet határozni, nagyobb hálók esetén azonban ez már nehézségekbe ütközhet. Az elemi körök keresésére viszonylag egyszerű, számítógépes algoritmusokat fejlesztettek, amelyekkel a keresés gyorsan elvégezhető [6].

Miután felsoroltuk az összes kört, alkalmazzuk a 3. és 4. tulajdonságokat annak érdekében, hogy kiértékeljük a rendszer teljesítményét. Megjegyezzük, hogy a kiértékelés függ a kezdeti állapottól, amelyhez meg kell adni a terméktípusok kezdeti eloszlását a folyamatban (azaz a puffer helyek kezdeti jelölését). Emlékezzünk arra, hogy a vezérlőhelyek kezdeti jelölése automatikusan meghatározott az adott géphez tartozó gépszekvenciával.

Tegyük fel, hogy kezdetben minden terméktípus a rendszeren kívül feldolgozásra vár (még

egyetlen művelet sem fejeződött be). Tekintettel arra, hogy a terméktípus a raklappal együtt kering, azt feltételezzük, hogy az MTH-ban felsorolt három terméktípushoz három raklap tartozik. A puffer helyek kezdeti jelölésével leírt megfelelő kezdeti állapot a 4. ábrán látható.

Az átmenetek tüzelési ideje a műveleti időkkal adott (lásd a termelési útvonalakat (I) és az 1. táblázatot):

$$\tau_{11}=1, \tau_{13}=2, \tau_{21}=3, \tau_{22}=9, \tau_{31}=2, \tau_{33}=4 \text{ időegység.}$$

Minden egyes $\beta_k(k=1,\dots,8)$ körre, az (1) egyenlettel most már kiszámíthatjuk a $\tau(\beta_k)$ teljes műveleti időt, a (2) egyenlettel az $M(\beta_k)$ token számot, és a (3) egyenlettel a $C(\beta_k)$ ciklusidőt. A számított értékeket a 2. táblázat tartalmazza.

2. táblázat

Az elemi körök ciklusidői

	Folyamatkör			Vezérlőkör			Vegyes kör	
	β_1	β_2	β_3	β_4	β_5	β_6	β_7	β_8
$\tau(\beta_k)$	5	11	6	5	8	6	9	12
$M(\beta_k)$	1	1	1	1	1	1	2	2
$C(\beta_k)$	5	11	6	5	8	6	4,5	6

A 4. tulajdonság és a (4) egyenlet felhasználásával állandósult állapotban a ciklusidő:

$$C = \text{Max}_{k=1,\dots,8}\{C(\beta_k)\} = C(\beta_2) = 11 \text{ időegység/token.}$$

A kritikus kör β_2 , ami azt jelenti, hogy az átbocsátóképességet a T_2 terméktípus ciklusideje határozza meg. Mivel a termelési ciklus három terméktípust tartalmaz (minden terméktípusból egyet), és a termelési ráta, $\lambda=1/C=0,09$, így a **teljes átbocsátóképesség**:

$$Q=3\cdot\lambda=3\cdot0,09=0,27 \text{ termék/időegység.}$$

4. A műhelyrendszerű termelés optimális vezérlése

4.1. A teljesítményjavítás lehetőségei

A rendszer teljesítményét (Q) adott termékösszetétel esetén csak a ciklusidő (C) csökkentésével tudjuk javítani. Tudjuk azt is, hogy a ciklusidő:

$$C = \text{Max}_{\beta_k}\{C(\beta_k)\} = \text{Max}_{\beta_k}\left\{\frac{\tau(\beta_k)}{M(\beta_k)}\right\} = \frac{\tau(\beta^*)}{M(\beta^*)},$$

a kritikuskör átmeneteinek összes tüzelési időtartamától és a kritikus kör tokenszámától függ. Mivel a tokenek a puffer helyeken a raklapokat reprezentálják, világos, hogy a raklapok számának változtatása hatással van a teljesítményre.

Például növeljük a raklapok számát eggyel a β_2 körben, azaz az új raklapot a T_2 a terméktípushoz rendeljük. Következésképpen, a β_2 körben két token kering (5/a. ábra). A ciklusidő pedig ebben a körben $C(\beta_2)=11/2=5,5$ időegység/token lesz, és az összes többi körben változatlan marad. A raklapszám növelés eredményeként a kritikus kör a G_2 gépéhez tartozó β_5 vezérlőkör lesz, amelyben a $\tau(\beta_5)=C(\beta_5)=8$. A maximális ciklusidő $C(\beta_5)=8$ -ra csökken, és a teljesítmény $Q=3/8=0,375$ termék/időegység-re javul. Ez az eredmény rendkívül fontos következtetésre vezet.

1. tétel: A műhelyrendszerű termelésben, amelyben a gépeket a terméktípusok adott és állandó sorrend szerint keresik fel, mindig létezik egy olyan véges raklapelosztás, amelynél állandósult állapotban a szűk keresztmetszetet jelentő gép teljes mértékben kihasznált [3].

A tétel igazolásához tekintsük ismét a (4) egyenletet, amely az állandósult állapot ciklusidejét

az összes elemi kör ciklusidőinek maximumaként számítja. Jelölje β_v a vezérlőköröket és $\beta \neq \beta_v$ a többi kört (azaz, a folyamat- és a vegyes köröket). A (4) egyenlet az új jelölésekkel a következő alakban írható:

$$(6) \quad C = \text{Max} \left\{ \text{Max}_{\beta_v} (C(\beta_v)), \text{Max}_{\beta \neq \beta_v} (C(\beta)) \right\}.$$

A vezérlőkörök maximális ciklusidejét jelölje C_0 , azaz a

$$(7) \quad C_0 = \text{Max}_{\beta_v} \{C(\beta_v)\}.$$

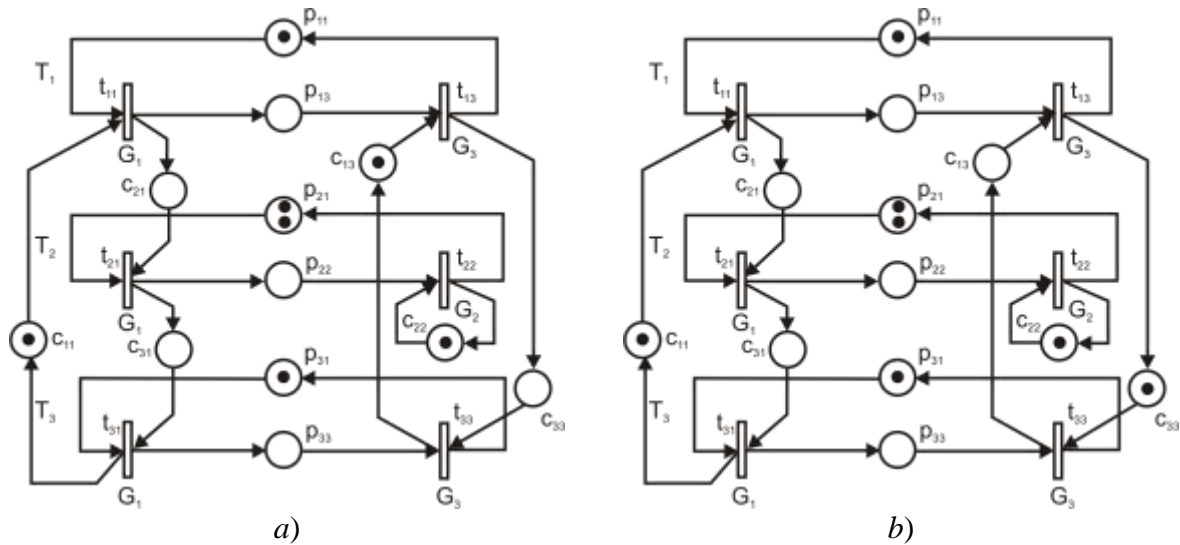
Mivel egy vezérlőkör ciklusideje éppen a körhöz tartozó gép terhelése (azaz az adott gép gépszekvenciájában felsorolt terméktípusok műveleti időinek az összege), ezért C_0 pontosan a szűk keresztmetszetet jelentő gép terhelése. Állandósult állapotban a szűk keresztmetszetet jelentő gép teljes kihasználásának így a szükséges és elégséges feltétele: $C=C_0$, amely szerint a szűk keresztmetszetet jelentő gép vezérlőköre a kritikus kör. Felhasználva a (3) egyenletet, a feltétel kielégítéséhez a

$$(8) \quad C(\beta) = \frac{\tau(\beta)}{M(\beta)} \leq C_0$$

összefüggésnek is teljesülnie kell. Vagyis az

$$(9) \quad M(\beta) \geq \frac{\tau(\beta)}{C_0},$$

minden $\beta \neq \beta_v$ körre.



5. ábra: Műhelyrendszerű termelés 4 raklappal (Forrás: saját szerkesztés)

Használjuk a

$$(10) \quad M_{\min}(\beta) = \left[\frac{\tau(\beta)}{C_0} \right],$$

jelölést, ahol $[y]$ jelöli azt a legkisebb egészszámot, ami nagyobb vagy egyenlő, mint y . Az $M_{\min}(\beta)$ tehát a β kör működéséhez szükséges minimális tokenszám, így a kör ciklusideje kisebb lesz, mint C_0 .

Könnyű belátni azt is, hogy a teljesítményváltozás függ a raklapelosztástól is, és a raklapszám növelés nem feltétlen jár teljesítményjavulással. Tegyük fel, hogy a negyedik raklapot, amely korábban a T_2 terméktípushoz tartozott, most a T_1 terméktípushoz vagy a T_3 terméktípushoz rendeljük. Ebben az esetben világos, hogy a β_2 kör ciklus ideje ismét 11 időegység/token lesz, és így a rendszer teljesítménye változatlan marad (ugyanannyi, mint három raklap esetén).

Végül vizsgáljuk meg a gépszekvenciák hatását az elérhető teljesítményre. Most tegyük fel, hogy a negyedik raklapot ismét a T_2 terméktípushoz rendeljük, és a G_3 gépen a terméktípusok megmunkálási sorrendjét permutáljuk. Az S_3 gépszekvencia (T_1, T_3) helyett legyen (T_3, T_1) . A Petri háló modellben ez azt jelenti, hogy a $\beta_6(G_3)$ vezérlőkör tokenje a c_{13} helyről a c_{33} helyre kerül (5/b. ábra). Könnyen ellenőrizhető, hogy a tokenek száma a két vegyes körben megváltozik, azaz, a β_7 kör egy tokenet nyer ($M(\beta_7)=3$) és a β_8 kör elveszít egyet ($M(\beta_8)=1$). A ciklusidők ennek megfelelően változnak, a $C(\beta_7)=9/3=3$ és a $C(\beta_8)=12/1=12$ időegység lesz. A változás eredményeként most β_8 lesz a kritikus kör, és a ciklusidő $C=12$ időegység. A rendszer teljesítménye pedig romlik:

$$Q=3/12=0,25 \text{ termék/időegység.}$$

Ezek a példák világosan szemléltetik, hogy a rendszer teljesítménye nemcsak a rendelkezésre álló raklapok számától, hanem a raklapok elosztásától, valamint a gépszekvenciától is függ. Természetesen szeretnénk megtalálni azt az optimális vezérlést, amely minimális raklapszám mellett a rendszer lehető legjobb kihasználását biztosítja.

4.2. A raklapok számának minimalizálása

Először koncentráljunk a minimális raklapszámra. Jelölje ismét β_v a vezérlőköreket és $\beta \neq \beta_v$ a többi kört, akkor a szűk keresztmetszetet jelentő gép teljes kihasználásának szükséges és elégséges feltétele:

$$(11) \quad M(\beta) \geq M_{\min}(\beta) \text{ minden } \beta \neq \beta_v \text{ körre,}$$

ahol: $M_{\min}(\beta)$ a (10) képlettel adott, az $M(\beta)$ pedig a tokenek száma a β körben.

Egy puffer helyen egy token egy raklapot, vagy egy terméktípust reprezentál. Ha a modellben s a folyamat- és vegyes körök, n_k pedig a p_1, p_2, \dots, p_{n_k} a puffer helyek száma a k -edik körben, valamint M^0 a kezdeti jelölés, akkor az összes raklap a rendszerben az

$$(12) \quad N = \sum_{k=1}^s \sum_{i=1}^{n_k} M^0(p_{ki})$$

kifejezéssel adott.

Így a maximális termelékenység elérése érdekében a szükséges raklapok számának minimalizálási problémája a következőképpen írható fel:

$$N = \sum_{k=1}^s \sum_{i=1}^{n_k} M^0(p_{ki}) \rightarrow \min$$

$$M(\beta) \geq M_{\min}(\beta) \text{ minden } \beta \neq \beta_v \text{ körre.}$$

A feladat egészértékű lineáris programozási feladatra (ILP) vezethető vissza. Az ILP megoldása adja azt a minimális raklapszámot és raklapelosztást, amelynél a rendszerben a szűk keresztmetszetet jelentő gép állandósult állapotban teljes kihasználással működik. Ez a feltétel csak akkor teljesül, ha a kezdeti állapot megfelel a kapott optimális megoldásnak (\mathbf{x}^*), azaz, a kezdeti jelölés $M^0(p_{ki})=x_j^*$ ($j=1, 2, \dots, n$).

A mintapélda optimális megoldása:

$$x_1^* = 1, x_2^* = 0, x_3^* = 2, x_4^* = 0, x_5^* = 1, x_6^* = 0,$$

amelyből a minimális raklapszám, $N=4$. A megoldásnak megfelelő kezdeti jelölést

$$M^0(p_{11}) = 1, M^0(p_{13}) = 0, M^0(p_{21}) = 2, M^0(p_{22}) = 0, M^0(p_{31}) = 1, M^0(p_{33}) = 0$$

az 5/a. ábra mutatja. Megjegyezzük, hogy más megoldások is léteznek, mint például az

$$x_1^* = 1, x_2^* = 0, x_3^* = 1, x_4^* = 1, x_5^* = 1, x_6^* = 0,$$

$$M^0(p_{11}) = 1, M^0(p_{13}) = 0, M^0(p_{21}) = 1, M^0(p_{22}) = 1, M^0(p_{31}) = 1, M^0(p_{33}) = 0.$$

Ez azt jelenti, hogy a rendszerben a termelési folyamat bármely terméktípus megmunkálásával kezdődhet. A raklapelosztás azonban mindkét megoldásban azonos, mégpedig a T_2 terméktípushoz kettő, a T_1 és T_3 terméktípusokhoz egy-egy raklapot rendelünk.

4.3. Az optimális gépjáratételezés (gépjáratételezés)

Eddig azt feltételeztük, hogy a gépjáratételezés a priori adott, és a megoldandó feladat a szűk keresztmetszetet jelentő gép teljes kihasználásához szükséges minimális raklapszám meghatározása. Már korábban megmutattuk és hangsúlyoztuk, hogy a rendszerteljesítmény nemcsak a raklapok számától és elosztásától, hanem a gépjáratételezéstől is függ. A teljességhez ezért fontos annak az „optimális” gépjáratételezésnek a megkeresése is, amely minimális raklapszám ($N=4$) és elosztás mellett garantálja, hogy a szűk keresztmetszetet jelentő gép műveleti ideje pontosan C_0 legyen.

Az ütemezési algoritmus részletes ismertetésétől most és itt hely hiányában eltekintünk, csupán az algoritmusban követett fontosabb elvekre hívjuk fel a figyelmet.

A műhelyrendszerű termelés összes műveletét olyan π hosszúságú időintervallumokban ütemezzük, amely megfelel a szűk keresztmetszetet jelentő gép (7) egyenlettel számított ciklusidejének, a példában ez $C_0=\pi=8$. Ha a műveleteket π periódusidővel ismételjük, akkor a rendszerben a szűk keresztmetszetet jelentő gép teljes kihasználtsággal fog működni.

Az ütemezéskor két korlátot kell szem előtt tartani: az elsőbbségi és a szétválasztási korlátokat. Az **elsőbbségi korlátok** alatt a termelési folyamatok technológiai korlátait értjük, pontosabban a terméktípusonként előírt műveleti sorrendet.

A **szétválasztási korlátok** egy adott gépen elvégzendő, egymást kölcsönösen kizáró műveletekre vonatkoznak. Egyszerűbben ugyanarra a gépre egyidejűleg nem lehet két terméktípust ütemezni. A Petri háló modellben ez azt jelenti, hogy az azonos vezérlőkörökhöz tartozó átmenetek tüzelés időintervallumai diszjunktak.

Az ütemezés minden z -edik lépésében kiszámítjuk a folyamatkörök ún. megvalósíthatósági fokát:

$$\Delta_k(z) = 1 - \delta_k(z),$$

ahol: $\delta_k(z)$ az ütemezésre váró műveletek folyamatidőinek az összege, és az ütemezésre rendelkezésre álló idő arányát mutatja. A z -edik lépésben azt a folyamatkört választjuk, amelyhez a legkisebb $\Delta_k(z)$ érték tartozik, majd az így kiválasztott körben azt a gépet (átmenetet) ütemezzük, amely az elsőbbségi és a szétválasztási korlátoknak is megfelel.

5. Összefoglalás

Az ismertetett modellben a műhelyrendszerű termelés leírására a Petri hálók egy speciális változatát, az időzített esemény gráfot használtuk. A rendszerben az átmenetek a különböző műveleteket, a tokenek pedig az erőforrásokat jelentik. A modellben alapvetően két erőforrás típus van: a terméktípusok (beleértve a szállító erőforrásokat, mint a targoncák, raklapok, amelyek impliciten a terméktípusokhoz kapcsolódnak), és a gépek. A helyek és a tokenek együtt írják le a rendszer állapotát. A modell további lényeges tulajdonsága, hogy a rendszerben folyamat-, és vezérlőkörök működnek. Az előbbieknél köszönhető a gyártási feladatok előírt műveleti sorrend szerinti ismétlődése, az utóbbiak pedig a feladatok ütemezését biztosítják a gépeken. A modell segítségével kiszámítható az optimális ciklusidő, a bemutatott algoritmusokkal pedig megkereshető az, az optimális vezérlés, amely a minimális raklapszám megfelelő elosztása mellett biztosítja a szűk keresztmetszetet jelentő gép teljes kihasználását, vagyis a rendszer maximális termelékenységét.

Irodalomjegyzék:

1. Commoner, F., Holt, A., Even, S., Pnueli, A. (1971) *Marked Directed Graphs*. Journal of Computer and System Sciences 5(5), pp. 511-523.
2. Hillion, H. P. (1989) *Timed Petri Nets and Application to Multi-stage Production Systems, Advances in Petri Nets*. Springer-Verlag, New York.
3. Hillion, H. P., Proth, J. M. (1989) *Performance Evaluation of Job-Shop Systems Using Timed Event-Graphs*. IEEE Transactions on Automatic Control 34(1), pp. 3-9.
4. Hosszú M. (1974) *Műszaki-gazdasági szélsőérték-feladatok*. Tankönyvkiadó, Budapest.
5. Krekó B. (1972) *Optimumszámítás*. Közgazdasági és Jogi Könyvkiadó, Budapest.
6. Martinez, J., Muro, P., Silva, M. (1987) *Modelling, Validation and Software Implementation of Production Systems Using High-Level Petri Nets*. Proceedings of the IEEE Conference on Robotics and Automation, Raleigh, NC, April 1987, pp. 307-314.
7. Petri, C. A. (1964) *Kommunikation mit Automaten*. Schriften des IIM No. 2, Institut für Instrumentelle Mathematik, Bonn.
8. Ramamoorthy, C. V., Ho, G. S. (1980) *Performance Evaluation of Asynchronous Concurrent Systems Using Petri Nets*. IEEE Transactions on Software Engineering SE-6(5), pp. 440-449.

Cím: Application of Petri nets for modelling of production and logistics processes

Summary

Petri nets were introduced by Carl A. Petri (1964). Since then, they were developed for studying the structure and control of parallel and concurrent systems. In the last decades most studies in connection with the Petri nets were published on the field of communications protocols and computer systems, but they have recently been widespread interest from the field of manufacturing systems. In the study, after a brief introduction to the theory of Petri nets, we present an application which can be used to model and analyze job shop-based and flexible manufacturing systems (FMS). These nets are useful for studying the operation of manufacturing systems, and they make possible to determine important information such as productivity, minimal handling equipment demand, equipment utilization and so on.