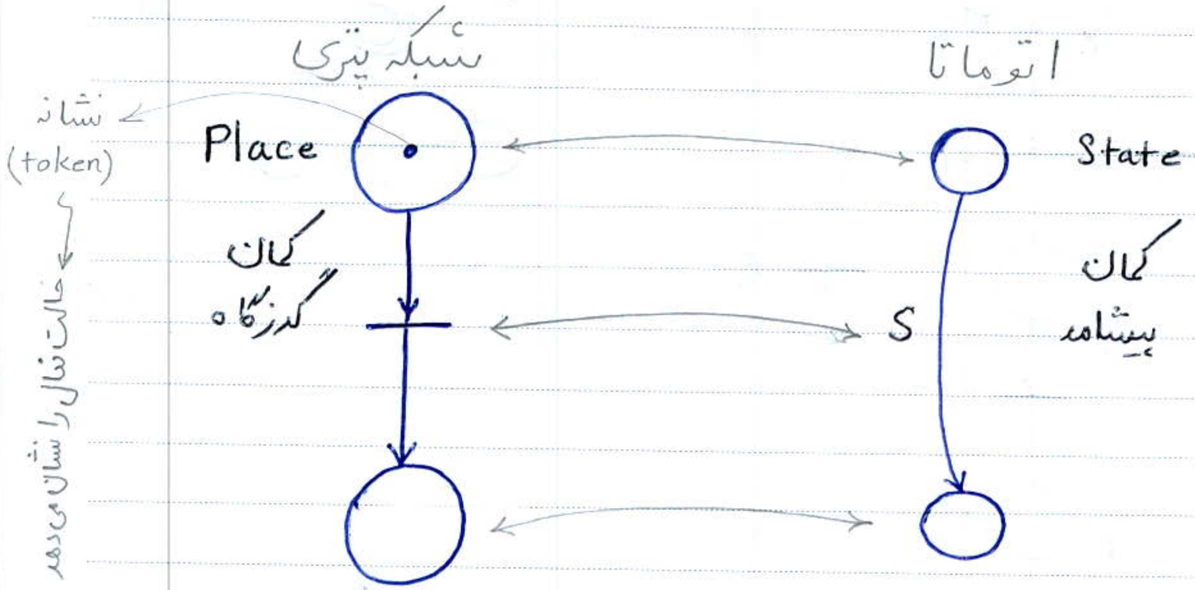
مدلسازی با شبکه‌های پتری



کمان و پیشامد در اتومات به کمان و گذرگاه در شبکه پتری تبدیل می‌شوند.

* چرا از اتومات به پتری می‌رویم؟

- در اتومات تنها یک حالت فعال برای مدل داریم (حالت اولیه در ابتدا و پس از رخداد هر پیشامد، حالت فعلی سیستم، حالت فعال است) اما در شبکه پتری هر زیرسیستم می‌تواند یک نشانه داشته باشد.
- انفجار حالت ناشی از ضرب در اتومات، در شبکه پتری تبدیل به در کنار هم قرار گرفتن می‌شود (جمع حالات)

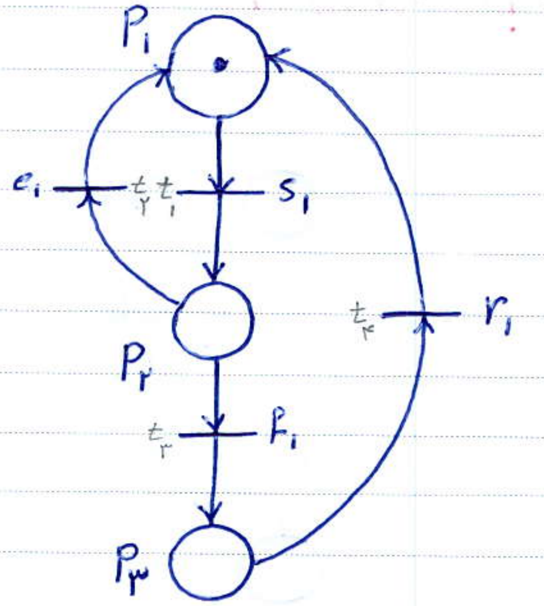
بدین ترتیب شبکه پتری شامل ۳ بخش اصلی است:

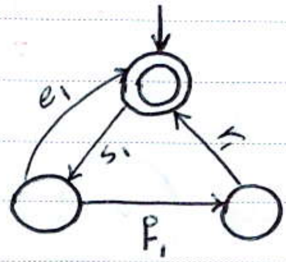
۱- Place (و نشانه)
۲- گذرگاه
۳- کمان

در مدلسازی گاهی تفکیک بین پیشامد و حالت مشکل است. بطور مثال: (باز شدن درب یک سیگنال است اما درب باز شده یک حالت می‌باشد)

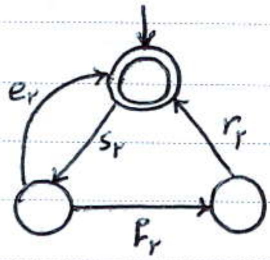
Subject
Date

مثال ـ مدل پتری ماشین زیر (مدل اتومات) را ترسیم نمائید.

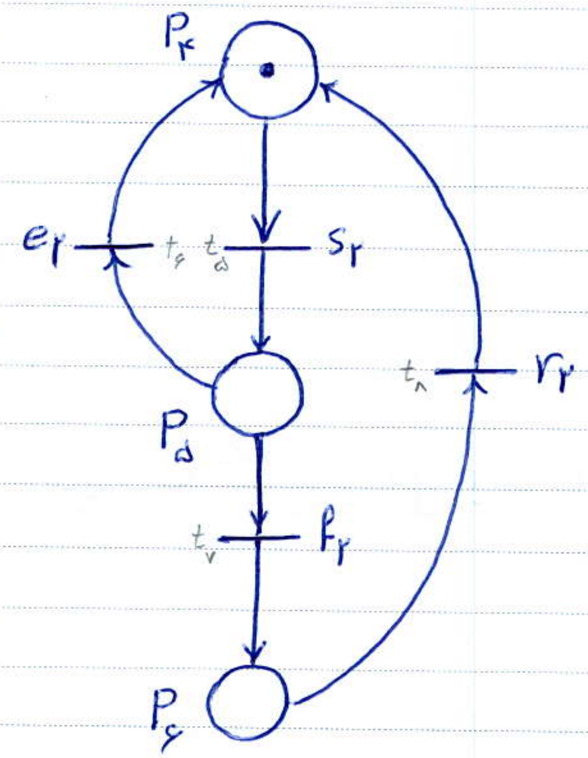




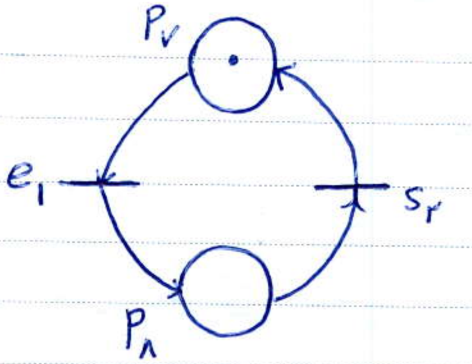
: M_1

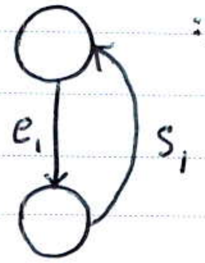


: M_2

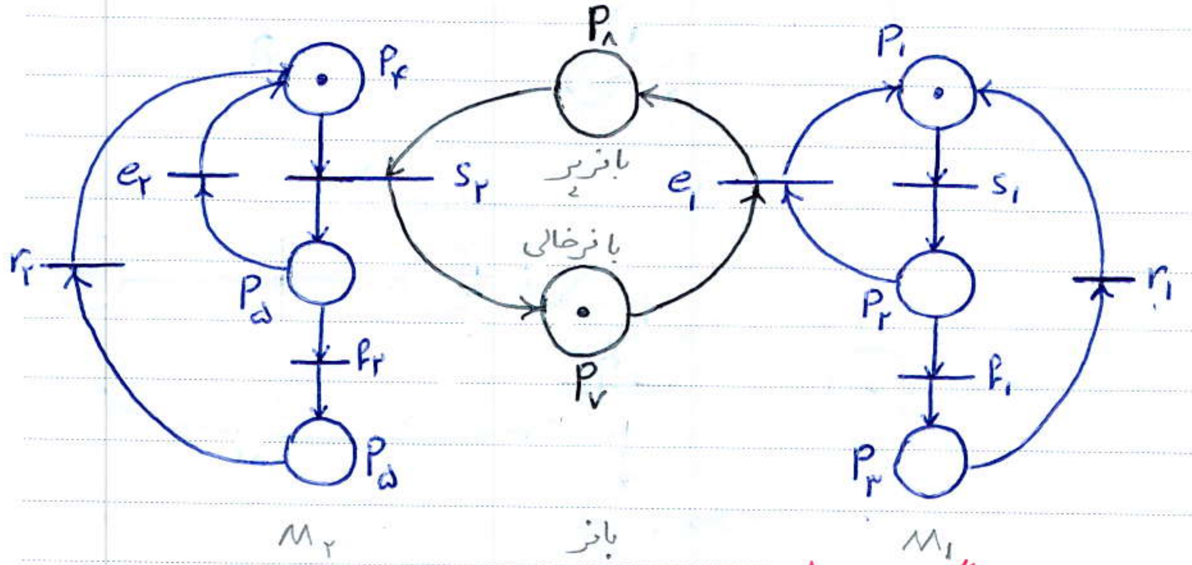


بافر:





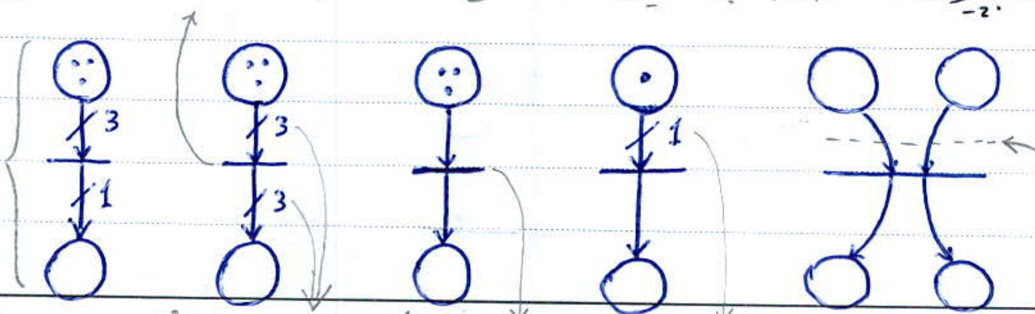
سنکرون کردن مدل های تولید در شبکه پتری بصورت زیر است:



آتش شدن گذرگاه ها:

شرط آتش شدن یک گذرگاه این است که ① تمام مکان های قبل از گذرگاه به اندازه وزن کمان متصل به گذرگاه دارای نشانه بوده و ② پیشامد تخصیص داده شده به گذرگاه نیز به وقوع بپیوندد ← به عبارت دیگر گذرگاه خاصیت and دارد. تا ۳ نشانه در مکان قبل نباشد آتش نمی شود.

وقتی هر دو پر شدن، گذرگاه آتش می شود



۳ نشانه می گیرد و ۱ نشانه می دهد

مشابه بسته بندی در یک بسته

با هر پیشامد ۳ نشانه می گیرد و ۳ نشانه می دهد

با هر پیشامد، یک نشانه به بافر می رود

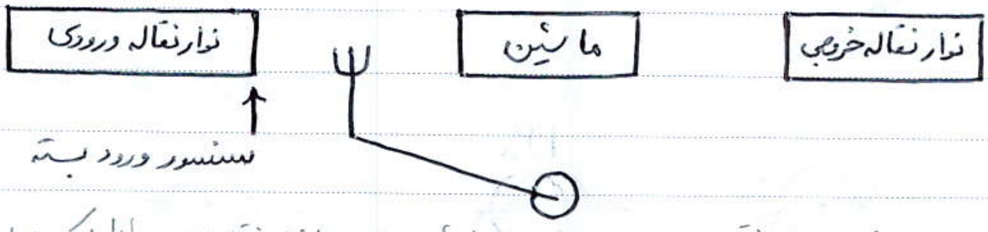
وزن 1

Subject
Date

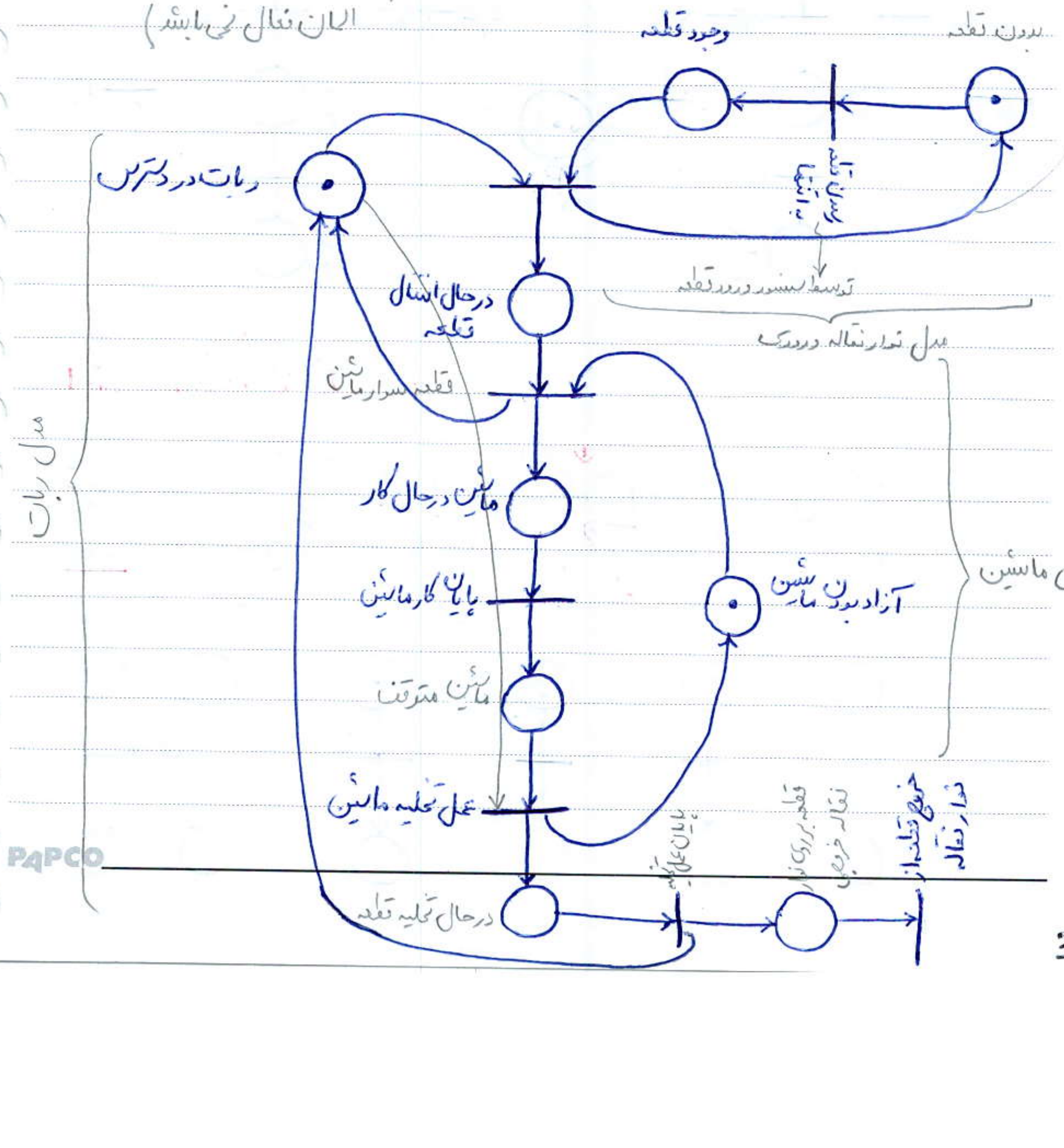
مثال - در سیستم ذیل، رسیدن قطعه به انتهای نوار نقاله ورودی، از طریق سنسور ورود بسته تشخیص داده می‌شود. سپس با انتقال قطعه توسط ربات از نوار نقاله به ماشین، ماشین شروع بکار کرده و پس از اتمام ماشینکاری، ربات قطعه را از ماشین به نوار نقاله خروجی (که همیشه روشن فرض می‌شود) منتقل می‌کند.

مدل شبکه پتری سیستم را ترسیم نمایید.

token < حرکتش در سیکل کردن باید برگردد.



۳ المان فعال، نوار نقاله ورودی، ربات و ماشین داریم (نوار نقاله خروجی بدلیل اینکه همواره روشن است المان فعال نمی‌باشد)



نکته ۱ - در نمایش شبکه پتری نیازی به مدل اولویت نیست ولی در عمل که به SFC تبدیل می‌شود، موردی که در سمت چپ است اولویت دارد.

نکته ۲ - وقتی به سراغ SFC می‌رویم به اجبار درگیر جزئیات (لبه بالارونده و ...) می‌شویم اما در پتری با جزئیات کاری نداریم و در گام اولیه با پتری پیش می‌رویم و در گام نهایی با SFC جزئیات را مشخص می‌کنیم.

نکته ۳ - هرجا بخواهیم اعمال محدودیت ظرفیت کنیم، با place ظرفیت آن را نشان می‌دهیم (با تعداد token ظرفیت).

نکته ۴ - در کنترل، حد الامکان تک نشانه استفاده می‌کنند و چند نشانه‌ها را باید تک نشانه در نظر بگیریم و بعد پیاده سازی کنیم.

نکته ۵ - تهیه جداول توصیف برای مکان ها و گذرگاه‌ها ضروری است:

مکان	توصیف
P_1	
P_2	

گذرگاه	
t_1	
t_2	

نکته ۶ - اگر به هر گذرگاه یک پیشامد تخصیص داده شده باشد، گذرگاه ≡ پیشامد است.

نکته ۷ - پیشامد همیشه یک داریم که حالت گذرا می‌باشد. این پیشامد را با e نمایش می‌دهند که پیشامد همیشه در حال وقوع است.

نکته ۸ - مثالی از پیشامدی که ۲ گذرگاه به آن اختصاص داده شود می‌توان به کلید ریموت تک شاسی اشاره نمود که اگر درب باز باشد سبب بسته شدن درب و اگر بسته باشد سبب باز شدن درب می‌شود.

معرفی رسمی یک شبکه پتری: شبکه پتری بصورت یک مجموعه ۳ تایی (۴ تایی یا بیشتر) بصورت زیر تعریف می شود:

$N(P, T, F)$ ۳ تایی

مجموعه مکان ها — مجموعه گذرگاه ها — تابع تعریف کننده رابطه بین گذرگاه و مکان

$N(P, T, F$ یا $W, M_0)$ ۴ چهارتایی

$W = W^+ - W^-$ ماتریس تلاقی — بردار نشانه اولیه

$N(P, T, W^-, W^+, M_0)$ ۵ پنج تایی ← بیشتر استفاده می شود

W^-: اتصالات Place های قبل از گذرگاه به گذرگاه — W^+: اتصالات Place های بعد از گذرگاه به گذرگاه

- در بیان ۶ تایی شبکه پتری، بردار Σ (تخصیص پیشامدها به گذرگاه) نیز بیان می شود.
- در بیان ۷ تایی شبکه پتری، بردار کنترلی نیز بیان می شود.

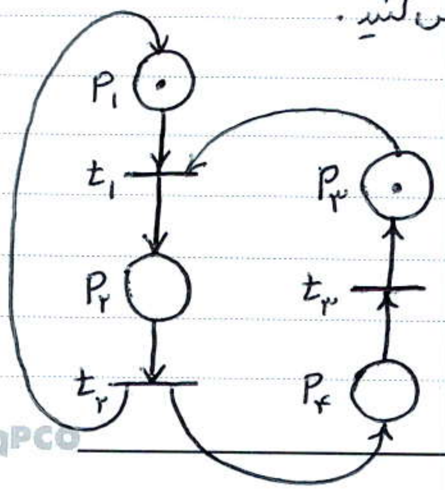
گذرگاه (ستون)

$W^- =$ مکان (سطر) $\begin{bmatrix} \quad \end{bmatrix}$

برای W^- (W^+) اگر مکان قبل (بعد) از گذرگاه بود 1 قرار می دهیم در غیر اینصورت 0 می گذاریم.

$$W^+ = \begin{array}{c|c} & t_1 \;\; t_2 \;\; \dots \\ \hline \begin{array}{c} P_1 \\ P_2 \\ \vdots \end{array} & \begin{bmatrix} \quad \end{bmatrix} \end{array}$$

مثال - ماتریس تلاقی را در شبکه پتری زیر مشخص کنید.



$$W^- = \begin{array}{c} \\ P_1 \\ P_2 \\ P_3 \\ P_4 \end{array} \begin{array}{c} \begin{array}{ccc} t_1 & t_2 & t_3 \end{array} \\ \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{array}$$

بصورت ستونی پر می شود اگر مکان قبل گذرگاه باشد ۱، در غیر اینصورت ۰.

$$W^+ = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

به همین دلیل در تحلیل از W استفاده نمی شود و از W^+ و W^- استفاده می شود

$$W = W^+ - W^- = \begin{bmatrix} -1 & +1 & 0 \\ +1 & -1 & 0 \\ -1 & 0 & +1 \\ 0 & +1 & -1 \end{bmatrix}$$

از آنجا که ممکن است یک Place قبل و بعد آن قرار گیرد ضریب آن صفر می شود

safe PN شبکه پتری تک نشانه
non-safe PN شبکه پتری چند نشانه

نشانه گذاری (marking) : مشابه State در اتوماتا می باشد و وضعیت سیستم را نشان می دهد

* تابع $M(p)$ به عنوان تابع نشانه گذاری تعریف می شود اگر تعداد نشان های هر یک از مکان ها را نشان دهد.

$$M : P \longrightarrow N \qquad M(p) = \begin{bmatrix} m(p_1) \\ m(p_2) \\ \vdots \\ m(p_n) \end{bmatrix}$$

تعداد نشانه های مکان p_1

$$p = \begin{bmatrix} p_1 \\ p_2 \\ \vdots \\ p_n \end{bmatrix}$$

بردار مکان ها

مثال - در مثال قبل، $M_0 = \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \end{bmatrix}$ می باشد که نشان دهنده حالت اولیه سیستم است. اگر گذرگاه t_1 آتش شود حالت سیستم به $M_1 = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}$ تبدیل می شود (تغییر می کند)

* برای آتش شدن هر گذرگاه، لازم است کلیه مکان های قبل از آن دارای نشانه باشند. این مکان ها را با علامت $\bullet t_i$ و با عنوان preset گذرگاه t_i معرفی می شوند:

$$\bullet t_i = \{ p_j \mid w^-_{ji} > 0 \} =$$ مجموعه کلیه مکان های قبل از گذرگاه

عنصر به عنصر بزرگتر باشد

شرط آتش شدن گذرگاه t_i در حالت M_j عبارت است از :

$$M_j \geq W^-_{t_i} \longrightarrow$$ ستون مربوط به t_i در ماتریس W^- ← با W^-_{*i} نیز نشان می دهند

* پس از آتش شدن هر گذرگاه، کلیه مکان های بعد از آن گذرگاه نشان دار می شوند. این مکان ها را با علامت $t_i \bullet$ و با عنوان postset شناخته می شوند:

$$t_i \bullet = \{ p_j \mid w^+_{ji} > 0 \} =$$ مجموعه کلیه مکان های بعد از گذرگاه

تغییرات انجام شده در نشانه مکان ها در ۴ دسته زیر قرار می گیرد:

$m'(p_j) = m(p_j) - w^-_{ji} + w^+_{ji}$ اگر $(p_j \in \bullet t) \wedge (p_j \in t\bullet)$

$m'(p_j) = m(p_j) - w^-_{ji}$ اگر $(p_j \in \bullet t) \wedge (p_j \notin t\bullet)$

$m'(p_j) = m(p_j) + w^+_{ji}$ اگر $(p_j \notin \bullet t) \wedge (p_j \in t\bullet)$

$m'(p_j) = m(p_j)$ اگر $(p_j \notin \bullet t) \wedge (p_j \notin t\bullet)$

$\times w_{ji}$

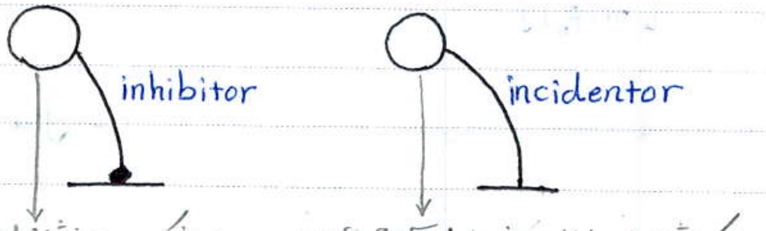
$$M(p) \xrightarrow{t_i} M'(p)$$

می توان محاسبات فوق را با فرمول زیر بعد از آتش شدن تکراری گذرگاه ها داشته باشیم.

$$M' = M + W \cdot X \qquad X = \begin{bmatrix} t_1 \text{ تعداد آتش شدن} \\ t_2 \;\;\text{" " "} \\ \vdots \\ t_n \;\;\text{" " "} \end{bmatrix}$$

کمان های فعال ساز (incidentor) و غیرفعال ساز (inhibitor) :

زمانیکه "فعال بودن" یا "فعال نبودن" یک حالت در آتش شدن یک گذرگاه اهمیت دارد اما نمی خواهیم با آتش شدن آن تغییری در حالت ایجاد بشود، از این کمان ها استفاده می شود.
این کمان ها، بدون فلش هستند و بصورت زیر استفاده می شوند:



زمانیکه بدون نشانه است فعال می شود.

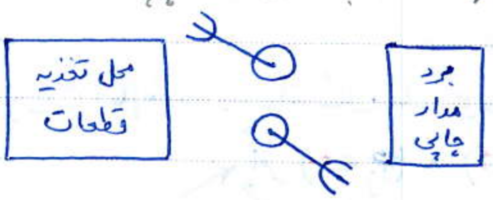
زمانیکه نشانه دارد فعال می شود اما با آتش شدن گذرگاه، نشانه را از دست نمی دهد.

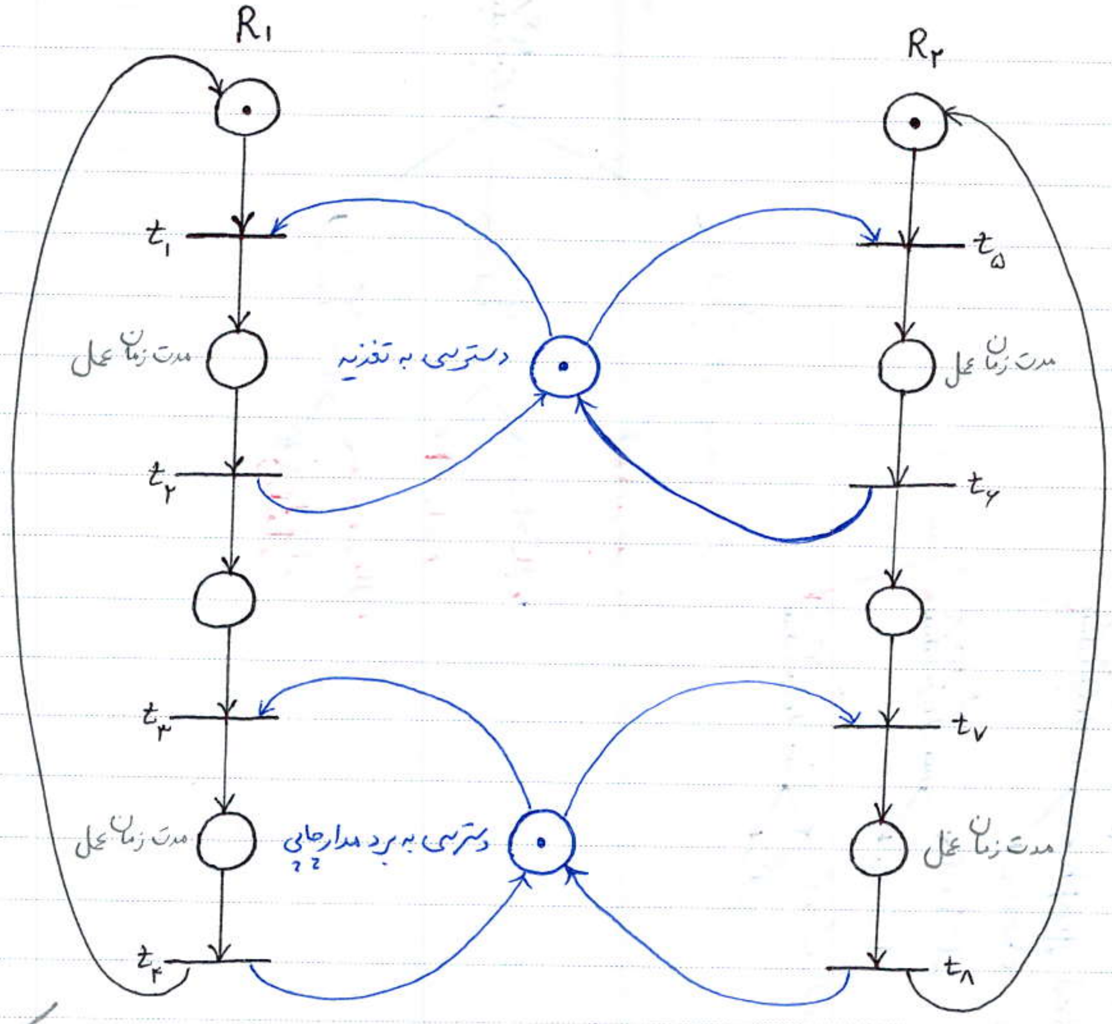
باید توجه داشت این کمان ها در ماتریس W وارد نمی شوند.

براین اساس شرط آتش شدن گذرگاه t_i عبارت است از:

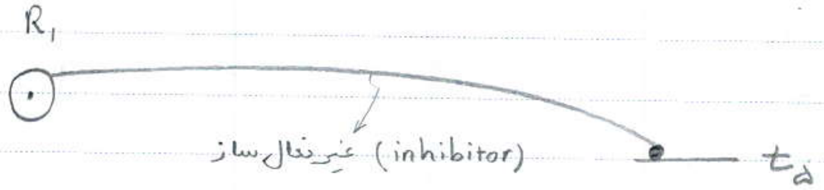
- $M \geq W^-_{t_i}$ ⟵ کمان های فلش دار، در W ظاهر می شود و باید برای آتش شدن گذرگاه برقرار باشد
- $M \geq C_{t_i}$ ⟵ incidentor ⟵ باید حالتهای آن دارای نشانه باشند ⟵ در W ظاهر نمی شوند
- $M < H_{t_i}$ ⟵ inhibitor ⟵ باید حالتهای آن بدون نشانه باشند ⟵ در W ظاهر نمی شوند

مثال - (کتاب ۷-۱۱ و ۷-۱۲) دو ربات به صورت شکل زیر قطعات را از یک مخزن مشترک برداشته و بر روی مدارچاپی قرار می دهند. در هر زمان تنها یکی از آنها می تواند قطعه را بردارد یا در محدوده مدارچاپی باشد (یعنی ۲ منبع (resource) تغذیه و مدارچاپی مشترک با محدودیت)
مطلوبست مدل شبکه پتری این مجموعه.





به منظور اولویت دهی به R_1 در ابتدای کار یا در شرایط مساوی به شکل زیر عمل می‌کنیم :



انواع مدل‌های شبکه‌ی پتری

- شبکه‌های پتری
  - گسسته
    - شبکه‌های پتری گسسته
      - State-machine : $\forall t \in T,\ |\cdot t| = |t \cdot| = 1$ ← تعداد مکان
      - marked graph : $\forall p \in P,\ |\cdot p| = |p \cdot| = 1$
      - free choice : $\forall p \in P,\ |p \cdot| \leqslant 1 \equiv \cdot(p \cdot) = \{p\}$
    - شبکه‌های پتری گسسته
      - تک نشانه :
      - چند نشانه :
    - شبکه‌های پتری گسسته
      - نشانه تک رنگی :
      - رنگی :
  - پیوسته : بطور مثال، در پایش فلو، از این مدل استفاده می‌شود. به جای نشانه، مقدار عددی استفاده می‌شود
  - هایبرید :
- شبکه‌های پتری
  - غیر احتمالی : عمدتاً در پتری گسسته بحث می‌شود و در هایبرید نیز قابل بحث است.
  - احتمالی : هدف آنالیز است (کنترل مورد نظر نیست)
- شبکه‌های پتری
  - بدون زمان :
  - زماندار :

state machine : اگر هر گذرگاه فقط یک مکان قبل و یک مکان بعد داشته باشد.

marked graph : اگر هر مکان فقط یک گذرگاه قبل و بعد آن باشد.

Free choice : هر مکان تنها یک کمان خروجی داشته باشد (گذرگاه بعد از مکان فقط یک کمان ورودی داشته باشد)

Subject
Date

مدل PN قابل تفسیر جهت کنترل :

کسی که بخواهد چنین مدلی را مطرح کند به سراغ SFC می‌رود مگر اینکه در فاز میانی و قبل از SFC مورد بررسی قرار گیرد .

Subject

Date

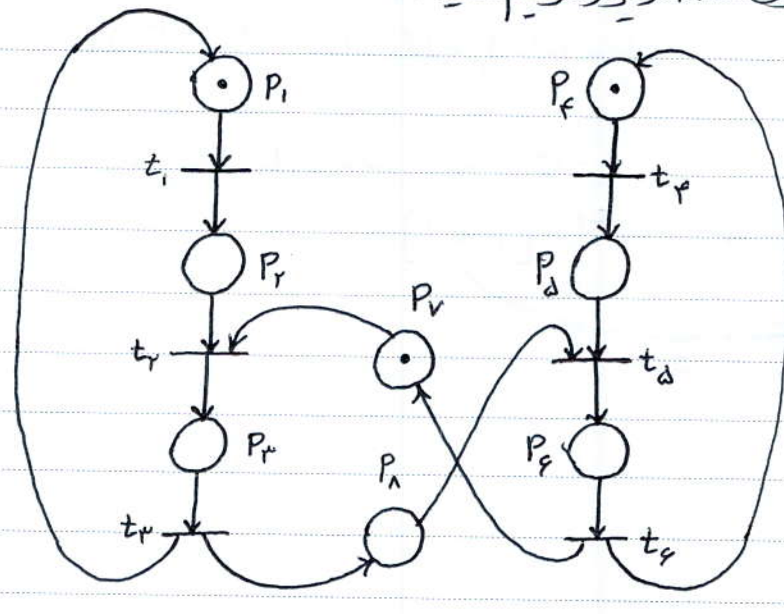
گراف قابل دسترس (reachability graph):

با استفاده از reachability graph، مدل اتوماتا سیستم را می‌توان از دل شبکه پتری استخراج نمود.

برای ترسیم گراف قابل دسترس به صورت زیر عمل می‌کنیم:

۱- بردار نشانه گراف را مشخص می‌کنیم.
۲- فرض می‌کنیم گذرگاهی در شرایط فعلی آتش شده است.
۳- در صورتی که حالت جدیدی برای شبکه ایجاد شده است به مرحله ۱ برمی‌گردیم و اگر گراف قابل دسترس بسته شده است متوقف می‌شویم.

مثال - گراف قابل دسترس PN (شبکه پتری) زیر را ترسیم کنید.



$\left.\begin{array}{r}\text{ماشین } M_1\\ \text{ماشین } M_2\\ \text{ربات}\end{array}\right\}$

$$M(P) = \begin{bmatrix} m(P_1) \\ m(P_2) \\ m(P_3) \\ m(P_4) \\ m(P_5) \\ m(P_6) \\ m(P_7) \end{bmatrix}$$

$$\begin{array}{ccccc}
\overset{m_0}{[1\,0\,0\,|\,0\,0\,|\,0]^T} & \xrightarrow{t_1} & \overset{m_1}{[0\,1\,0\,|\,0\,0\,|\,0]^T} & \xrightarrow{t_2} & \overset{m_4}{[0\,0\,1\,|\,0\,0\,0\,0]^T} \longrightarrow \dots \\
\downarrow t_4 & & \downarrow t_4 & & \downarrow t_4 \quad \dots \\
\underset{m_2}{[1\,0\,0\,0\,|\,0\,|\,0]^T} & \xrightarrow{t_1} & \underset{m_3}{[0\,1\,0\,0\,|\,0\,|\,0]^T} & \xrightarrow{t_2} & \underset{m_5}{[0\,0\,|\,0\,|\,0\,0\,0]^T} \longrightarrow \dots \\
\downarrow t_5 & & & & \\
\times & & & &
\end{array}$$

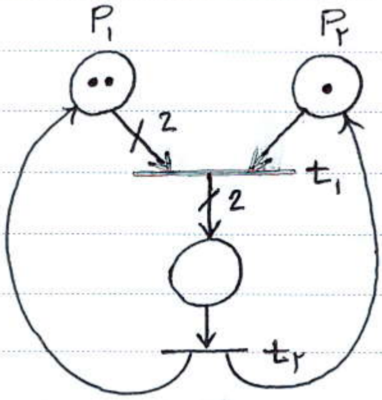
(زیرا P_8 نشانه ندارد)

در نهایت گراف بسته می‌شود و بدین ترتیب مدل اتوماتا سیستم پدید می‌آید.

گراف همپوشانی (Coverability Graph):

در برخی شبکه‌های پتری، مکان‌هایی وجود دارند که نشانه آنها دائماً تغییر نموده و نمی‌توان برای آنها گراف قابل دسترس ترسیم نمود (گراف قابل دسترس، بسته نمی‌شود). در این شرایط گراف همپوشانی ترسیم می‌شود.

مثال - گراف همپوشانی PN زیر را بدست آورید.



در ترسیم گراف قابل دسترس داریم:

$$\begin{array}{ccccc} M_0 & & M_1 & & M_2 \\ [2\ 1\ 0] & \xrightarrow{t_1} & [0\ 0\ 2] & \xrightarrow{t_2} & [1\ 1\ 1] \xrightarrow{t_2} \\ [2\ 2\ 0] & \xrightarrow{t_1} & [0\ 1\ 2] & \xrightarrow{t_2} & [1\ 2\ 1] \xrightarrow{t_2} \\ [2\ 3\ 0] & \longrightarrow & & & \end{array}$$

$\xrightarrow{t_1}$ ✗ ← P_1 باید ۲ نشانه داشته باشد

ملاحظه می‌شود که نشانه مکان P_2 دائماً در حال افزایش است و به شرایط تکراری نمی‌رسد.
در این وضعیت به جای گراف قابل دسترس، گراف همپوشانی ترسیم می‌شود.
گراف همپوشانی به صورت زیر است:

$$[2\ W\ 0] \xrightarrow{t_1} [0\ W\ 2] \xrightarrow{t_2} [1\ W\ 1]$$

$[1\ W\ 1] \xrightarrow{t_2} [2\ W\ 0]$

در شرایط زیر گراف همپوشانی ترسیم می‌شود:

۱ـ نتوان گراف قابل دسترس ترسیم نمود
۲ـ تعدادی حالت نامشخص داشته باشیم
۳ـ $\underbrace{\text{تعداد W ها محدود باشد}}$

همه، W نباشند.

Subject
Date

تمرین: برنامه‌ای بنویسید که ماتریس W^+ ، W^- و بردار M_0 را دریافت کرده و گراف قابل دسترس را محاسبه نماید.

ساده‌ترین روش: لیست حالتها با برچسب حفظ شوند و مشخص گردد، با چه گذرگاهی به چه حالتی می‌رود.

با یک Data structure، که حالتها در آن ذخیره شوند و مشخص گردد با چه گذرگاهی به چه حالتی منتقل می‌شود

$$M_0 \xrightarrow{t_1} M_1 \qquad M_0 \xrightarrow{t_2} M_3$$

PAPCO

Subject

Date

خواص شبکه های پتری :

① محدود بودن (Boundedness) :

اگر به ازای هر حالت اولیه از یک شبکه پتری، تعداد نشانه های تمام مکان های آن محدود باشند، آن شبکه دارای خاصیت "محدود بودن" می باشد.

- در صورتی که PN محدود باشد، می توان گراف قابل دسترس ترسیم نمود.
- در برخی موارد، نامحدود بودن یک اشکال است و با شبکه پتری می توان این عیب را آشکار نمود (مثلاً در بافر)
- "به ازای هر حالت اولیه" به معنی درنظر گرفتن یک شبکه بدون نشانه می باشد که نشانه را در هر مکانی درنظر گرفته ایم.

② زنده بودن (liveness) :

زنده بودن در چند سطح تعریف می شود:

- شبکه کاملاً زنده : است اگر تمام گذرگاههای آن زنده باشد
- گذرگاه زنده : است اگر به ازای هر حالت شبکه، امکان آتش شدن آن گذرگاه وجود داشته باشد.
- گذرگاه درجه k زنده : است اگر امکان آتش شدن k بار آن وجود داشته باشد.
- گذرگاه مرده : است (زنده نیست) اگر امکان آتش شدن آن وجود نداشته باشد.
- شبکه کاملاً مرده : اگر امکان آتش شدن هیچ گذرگاه آن وجود نداشته باشد.

در بررسی زنده و مرده بودن گذرگاه، بررسی می شود که حالتهایی که به گذرگاه وصل هستند می توانند نشانه بگیرند یا نه. اگر هیچگاه نتوانند نشانه بگیرند، گذرگاه مرده است.

③ ناوردائی (Conservativeness) :

یک شبکه ناوردا است اگر بتوان رابطه زیر را برای نشانه های مکان های آن شبکه نوشت :

$$\sum_{i=1}^{n} k_i \overbrace{m(p_i)}^{\text{نشانه های مکان ها}} = C \xrightarrow{} \text{مقدار معین}$$

یعنی بتوان رابطه ای بین نشانه ها و مکان ها پیدا نمود.

بطور مثال، در یک ماشین با ۳ حالت خاموش، روشن و در حال کار، با توجه به اینکه وضعیت ماشین از این ۳ حالت خارج نیست، نشانه بین این حالتها در چرخش است و بدین ترتیب می توان معادله ای بین نشانه این مکان ها نوشت.

Subject
Date

مکان ها و گذرگاه‌های وابسته (P_invariant و T_invariant) :

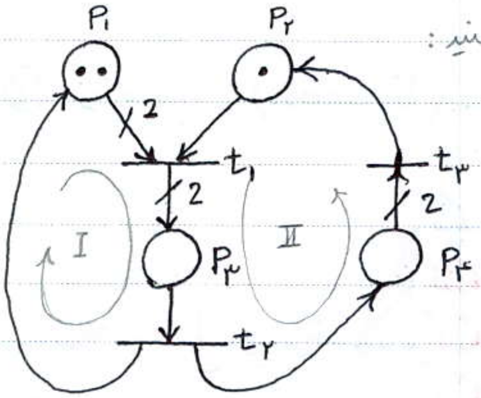
مکان

P_invariant : مجموعه مکان هایی که بتوان یک رابطه بین مقدار نشانه آنها نوشت و با استفاده از معادله زیر قابل استحصال می باشد :

$W^T i_p = 0$ $\quad \sum k_i m_i = C$ بردار k در معادله ← به عبارت دیگر ضرایب m_i می باشد

مجموعه جواب های غیر صفر معادله فوق P_invariant های یک شبکه را مشخص می کنند.

مثال - P_invariant های شبکه پتری زیر را مشخص کنید :



روش اول - ۲ معادله ناوردائی در ۲ حلقه‌ای که نشانه در آن می چرخد می نویسیم :

$$\text{I حلقه}: \underbrace{\frac{m(P_1)}{2}+\frac{m(P_3)}{2}}_{\text{نشانه مکان ها را نرمالیزه کردیم}} = 1 \Rightarrow \underset{m(P_1)}{m_1} + \underset{m(P_3)}{m_3} = 2$$

$$\text{II حلقه}: m(P_2)+\frac{m(P_3)}{2}+\frac{m(P_4)}{2} = 1 \Rightarrow 2m_2+m_3+m_4 = 2$$

minimal P-invariant

با ترکیب معادلات فوق، معادله P_invariant سیستم عبارت است از : $m_1+2m_2+2m_3+m_4=4$

روش دوم - از دستگاه معادلات بصورت زیر استفاده می کنیم :

$$W^T i_p = 0 \Longrightarrow \begin{array}{c c} & \begin{array}{cccc} P_1 & P_2 & P_3 & P_4 \end{array} \\ \begin{array}{c} t_1 \\ t_2 \\ t_3 \end{array} & \begin{bmatrix} -2 & -1 & +2 & 0 \\ +1 & 0 & -1 & +1 \\ 0 & +1 & 0 & -2 \end{bmatrix} \end{array} \begin{bmatrix} i_{P_1} \\ i_{P_2} \\ i_{P_3} \\ i_{P_4} \end{bmatrix} = 0$$

$i_p^T = [1 \ 0 \ 1 \ 0]$

$i_p^T = [0 \ 2 \ 1 \ 1]$

ضرایب m_i

با توجه به اینکه $rank(W^T)=2$ است ۲ جواب مستقل غیر صفر دیگر ندارد.

$-t_3 = t_1 + 2t_2$

* در کتاب " Discrete, Continuse and Hybrid PN " الگوریتم محاسبه P_invariant وجود دارد.

* نرم افزار PN tool از دانشگاه رومانی، معادلات P_invariant را می دهد.

PAPCO

گذرگاه

T-invariant : مجموعه گذرگاهی که سبب می شوند به حالت اولیه برگردیم . به عبارت دیگر، مجموعه گذرگاهی که با آتش شدن آنها تغییری در نشانه ایجاد نمی شود .

و از معادله زیر قابل استحصال است :

$$W i_T = 0$$

باید توجه داشت با توجه به معادله انتقال حالت ، به منظور برگشت به حالت اولیه لازم است WX برابر صفر باشد :

$$M' = M + \overbrace{WX}^{0} \Rightarrow M' = M$$

مثال - در مثال صفحه قبل T-invariant سیستم را بدست آورید .

روش اول

$$W i_T = 0 \Longrightarrow \begin{array}{c c} & \begin{array}{ccc} t_1 & t_2 & t_3 \end{array} \\ \begin{array}{c} P_1 \\ P_2 \\ P_3 \\ P_4 \end{array} & \begin{bmatrix} -2 & +1 & 0 \\ -1 & 0 & +1 \\ +2 & -1 & 0 \\ 0 & +1 & -2 \end{bmatrix} \end{array} \begin{bmatrix} i_{T_1} \\ i_{T_2} \\ i_{T_3} \end{bmatrix} = 0 \rightarrow i_T = [1\ 2\ 1]^T$$

$$X : \operatorname{rank}(W) = 2 \longrightarrow 3 - 2 = 1$$

تعداد متغیرها

rank

تعداد پاسخ مستقل (درجه آزادی)

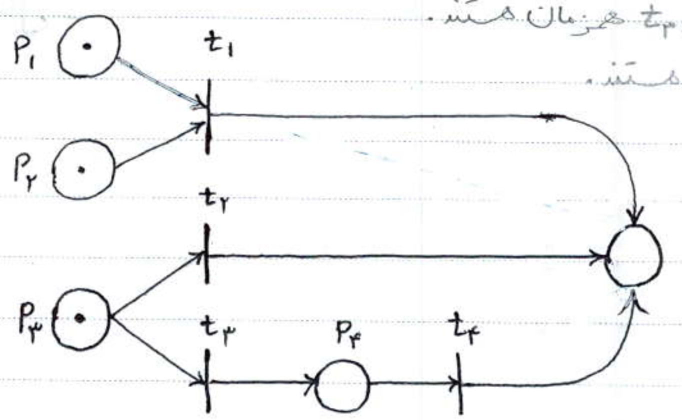
روش دوم : با توجه به شبکه پتری ، ترتیبی مشابه $t_1 t_2 t_2 t_3$ سبب برگشت به حالت اولیه می شود .

همزمانی و تداخل (Concurrency & Conflict) :

اگر تعدادی گذرگاه امکان آتش شدن همزمان آنها وجود داشته باشد آنها را همزمان یا Concurrent می‌گویند و این پدیده را همزمانی یا Concurrency می‌نامند.

اگر ۲ یا چند گذرگاه بالقوه فعال باشند اما امکان آتش شدن تمام آنها وجود نداشته باشد می‌گوییم این گذرگاه‌ها متداخل هستند و این پدیده را تداخل یا Conflict می‌نامند.

مثال - گذرگاه‌های t_1 و t_2 ، t_1 و t_3 همزمان هستند.
گذرگاه‌های t_2 و t_3 متداخل هستند.



در شبکه‌های پتری خیلی با این موارد مواجه نمی‌شویم. در این شبکه‌ها، فرض بر این است که پیشامدها مستقل هستند.
در plc ، امکان همزمانی وجود دارد و در Graph set اولیه که غیر استاندارد بود فرض می‌کرد که هر دو همزمان آتش می‌شوند که با پتری تناقض داشت.
در SFC (که پتری استاندارد شده است) اینطور نیست و اولویت چپ به راست است.

سیفون و تراپ (Siphons & Traps):

اگر مجموعه گذرگاه‌های قبل از تعدادی مکان زیرمجموعه گذرگاه‌های بعد از آن‌ها باشد سیفون داریم.

اگر مجموعه گذرگاه‌های بعد از تعدادی مکان زیرمجموعه گذرگاه‌های قبل از آن‌ها باشد ترپ داریم.

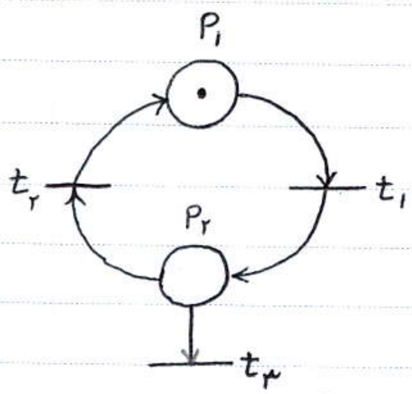
Preset — مجموعه مکان‌های مورد نظر.

بیان ریاضی مفاهیم فوق به شکل زیر است:

$^{\circ}p = {}^{\circ}p_1 \cup {}^{\circ}p_2 \cup \dots$ $\qquad$ $^{\circ}p \subseteq p^{\circ}$ سیفون

$p^{\circ} = p_1^{\circ} \cup p_2^{\circ} \cup \dots$ $\qquad$ $p^{\circ} \subseteq {}^{\circ}p$ ترپ

Postset

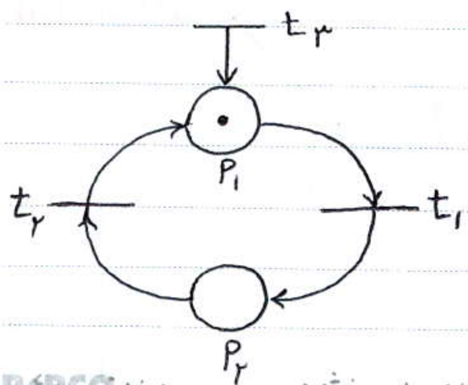


مثال - $^{\circ}p = {}^{\circ}p_1 \cup {}^{\circ}p_2 = \{t_2\} \cup \{t_1\} = \{t_2, t_1\}$

$p^{\circ} = p_1^{\circ} \cup p_2^{\circ} = \{t_1\} \cup \{t_2, t_3\} = \{t_1, t_2, t_3\}$

$^{\circ}p \subseteq p^{\circ} \Rightarrow$ سیفون

مشکل سیفون آن است که همیشه باید پر باشد. در اینجا اگر t_3 آتش شود، سیفون خالی از نشانه می‌شود و معلق می‌گردد.



$^{\circ}p = {}^{\circ}p_1 \cup {}^{\circ}p_2 = \{t_2, t_3\} \cup \{t_1\} = \{t_1, t_2, t_3\}$

$p^{\circ} = p_1^{\circ} \cup p_2^{\circ} = \{t_1\} \cup \{t_2\} = \{t_1, t_2\}$

$p^{\circ} \subseteq {}^{\circ}p \Rightarrow$ ترپ

در ترپ انباشتگی ایجاد می‌شود. در اینجا با هر بار آتش شدن t_3، نشانه به مجموعه اضافه می‌شود و از آن خارج نمی‌گردد.

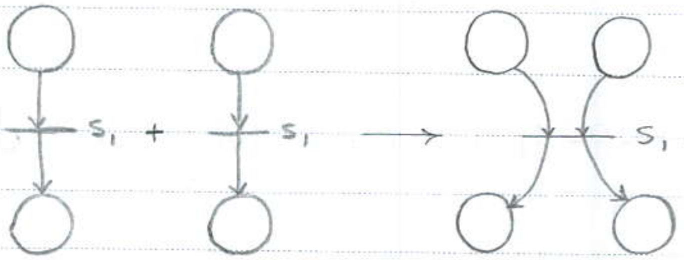
ترکیب اجزای یک سیستم در شبکه پتری :

در اتوماتا وقتی ترکیب اجزاء اتفاق می افتد، حالتها در هم ضرب می شوند اما در پتری، ترکیب اجزاء معادل جمع شدن تعداد حالتها است و این مهمترین مزیت پتری نسبت به اتوماتا است.

در ترکیب دو مدل پتری، با توجه به مشترک بودن یا نبودن پیشامد گذرگاه ها، به یکی از دو شکل زیر اقدام می شود :

۱- اگر پیشامد گذرگاه مشترک نباشد، گذرگاه و مکان های قبل و بعد از آن در مدل ترکیب شده قرار می گیرند.

۲- اگر پیشامد گذرگاه مشترک باشد، در مدل ترکیب شده یک گذرگاه قرار داده شده و مکان های قبل و بعد از گذرگاه در هر دو مدل، به عنوان مکان های قبل و بعد این گذرگاه در نظر گرفته می شوند.



به منظور بیان مفهوم دقیقتری از مشترک بودن پیشامدها، انواع پیشامدها را بصورت زیر خواهیم داشت :

انواع پیشامدها

- پیشامدهای کاملاً مجزا : این نوع پیشامدها دارای اسم متفاوت و مفهومی بی ربط هستند
- پیشامدهای مشترک
  - دقیقاً یکسان : بطور مثال تخلیه ماشین با یک ربات
  - پیشامدهایی که از نظر یک جزء یکسان اما از نظر اجزای دیگر متفاوت هستند
  - مقایسه ای : بطور مثال، دمای بالاتر از ۲۰° در دو بخش متفاوت مدل، بطور جداگانه استفاده می شوند

* تنها پیشامدهای دقیقاً یکسان بعنوان پیشامد مشترک در ترکیب پیشامدها در نظر گرفته می شوند *

تخلیه یک ماشین با دو ربات، از نظر هر ربات پیشامد متفاوتی است و در این حالت باید پیشامدها به تفکیک و جداگانه در نظر گرفته شوند

اجرای کنترل نظارتی با استفاده از شبکه‌های پتری :

ایده کنترل نظارتی نسبتاً ساده است و کافیست مدل اجرای کنترل نظارتی با استفاده از شبکه‌های پتری را داشته باشیم.

اگر بتوانیم ضوابط را با شبکه‌های پتری مدل کنیم ، با ترکیب مدل ضوابط با مدل اجرای سیستم ، مدل حلقه بسته بدست خواهد آمد.

منظور از حلقه بسته ، مشاهده خروجی و صدور فرمان مجاز یا غیر مجاز بودن رخداد یک پیشامد بر اساس وضعیت ورودی و خروجی است.

باید توجه داشت هر مدلی را با شبکه پتری نمی‌توان شبیه‌سازی کرد. همچنین ضابطه را نباید با مدل مخلوط کرد. در طراحی کنترل کننده برای فرآیندهای صنعتی ، تعدادی المان داریم و علاوه بر آن یک خواسته از طرف کارفرما. در مدلسازی ، خواسته کارفرما به ضابطه تعبیر می‌شود و باید تا هر سطح دسترسی به المان که داریم پیش برویم و در مدل اجرای سیستم قرار دهیم ؛ سپس خواسته و آنچه باقی می‌ماند را در مدل ضابطه لحاظ کنیم.

در اکثر مواقع ، زمانیکه به سطح دسترسی به موتور می‌رسیم ، فرامین on و off می‌شود. همچنین ، اگر یک سیستم دمایی را بصورت آستانه در نظر بگیریم ، فرامین on و off را خواهیم داشت و ضابطه در قالب اجازه on یا off مطرح می‌شود.

ضابطه در ۴ قالب
- شبکه پتری استاندارد
- شبکه پتری با کمان‌های فعالساز و غیرفعالساز.
- بیان دسته نامعادلات
- لیست حالات ممنوع

مطرح می‌شود.

در ترکیب مدل ضابطه با مدل اجرای سیستم ، مشابه ترکیب اجزای یک سیستم در شبکه پتری عمل می‌شود و ضابطه در گذرگاه مشترک (بین مدل ضابطه و مدل اجرای سیستم) شرط را تحمیل می‌کند.

اگر گذرگاههای مشترک ، قابل کنترل باشند مدل حلقه بسته بدست آمده همان کنترلر سیستم خواهد بود.

اگر ضوابط و اجزاء در گذرگاه غیرقابل کنترل با هم ترکیب شوند می‌تواند مشکل‌ساز باشد و باید شرایط آتش شدن مورد بررسی قرار گیرد. در گذرگاه با پیشامد غیرقابل کنترل ، می‌خواهیم جلوی آتش شدن را هم بگیریم

کنترل کننده می‌تواند کمان کنترل یا نشانه یک مکان باشد. امکان ورود کمان کنترل بر گذرگاه غیرقابل کنترل باید بررسی شود. در برخی موارد نمی‌توان اینکار را انجام داد و لازم است از نشانه کنترل استفاده نمود.

Subject
Date

مقایسه تئوری کنترل نظارتی در اتوماتا و شبکه‌های پتری:

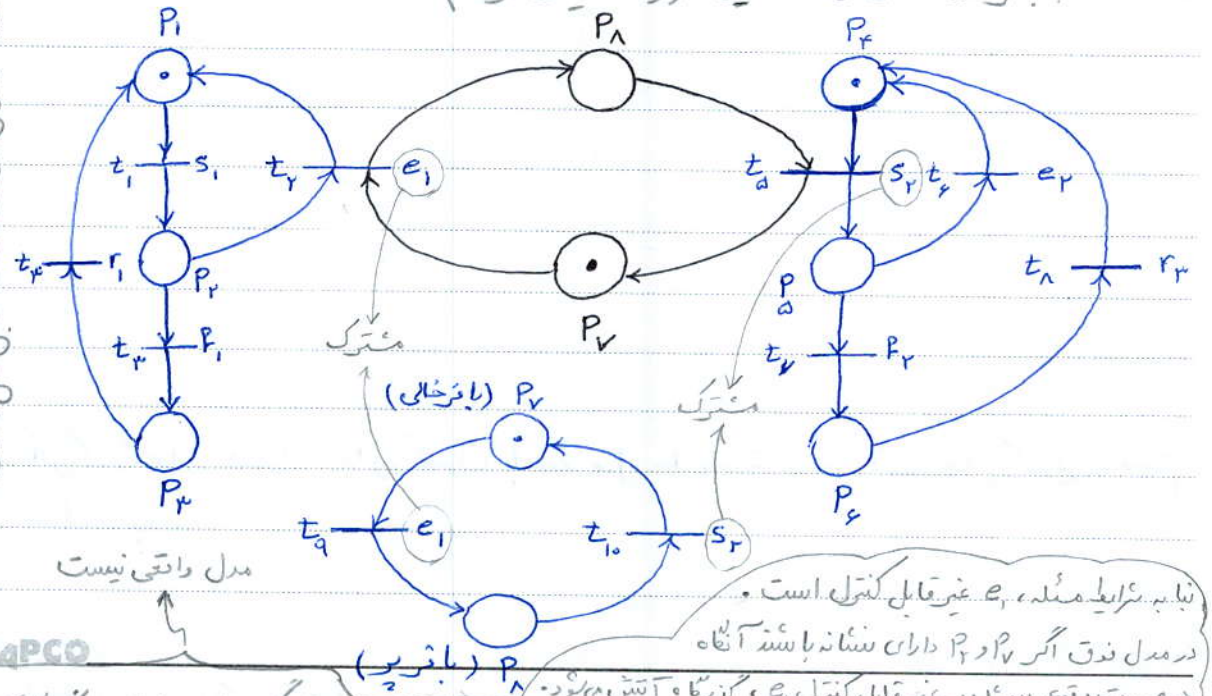
مزیت
- مدل شبکه‌های پتری، فشرده‌تر است.
- مدل پتری در کارهای sequential خیلی کمک کننده و گویا می‌شود و اگر بتوانیم مدل پتری را بنویسیم طراحی کنترل کننده
- امکان استفاده از خواص شبکه‌های پتری برای بیان برخی مسائل و مشکلات وجود دارد.

معایب
- در برخی مواقع (مثلاً در حضور پیشامدهای غیرقابل کنترل) نیاز به استفاده از گراف قابل دسترس است که در این شرایط، مشابه اتومات خواهد بود. (اتومات، مدل بزرگ است و در پتری ۶۰ تا ۷۰٪)
- امکان مدلسازی ضوابط با شبکه‌های پتری، همیشه به راحتی قابل انجام نیست.

بیان راحتتر از ladder خواهد بود. مسائل حل می‌شود و تعدادی نیز با reachability graph که همان اتومات است قابل حل است.

نکته - هرگاه ضوابط و اجزاء در گذرگاه غیرقابل کنترل با هم ترکیب می‌شوند می‌تواند مشکل‌ساز باشد و باید بررسی شود که شرط آتش شدن تامین می‌شود یا نه.

مثال - دو ماشین M_1 و M_2 را درنظر بگیرید که فرآیند کاری آنها شامل پیشامدهای s، e، f، r مشابه مثال قبل می‌باشد. بافر بین دو ماشین دارای یک ظرفیت است و پیشامد اتمام کار ماشین M_1 غیرقابل کنترل است (پس از اتمام کار ماشین M_1، قطعه توسط فرد به بافر انداخته می‌شود و ماشین سریعاً تخلیه می‌گردد)



بنا به شرایط مسئله، e_1 غیرقابل کنترل است.
در مدل فوق اگر P_7 و P_2 دارای نشانه باشند آنگاه در صورت وقوع پیشامد غیرقابل کنترل e_1، گذرگاه آتش می‌شود.
این در حالیست که در واقعیت آتش شدن گذرگاه e_1 مستقل از وضعیت P_8 است و حتی اگر هم بافر پر باشد این گذرگاه آتش می‌شود

۹۶/۹/۱۳ ، ۱۲ ، ۱۸ و ۱۹

Subject

Date

روش‌های مختلف بیان ضوابط :

۱- بیان ضابطه با استفاده از شبکه پتری استاندارد :

استفاده از این روش، نسبت به سایر روش‌ها، ارجحیت دارد و استفاده از آن توصیه می‌شود.
استفاده از این روش سبب جذابیت و بهتر شدن مدل می‌شود اما گاهی امکانپذیر نیست.

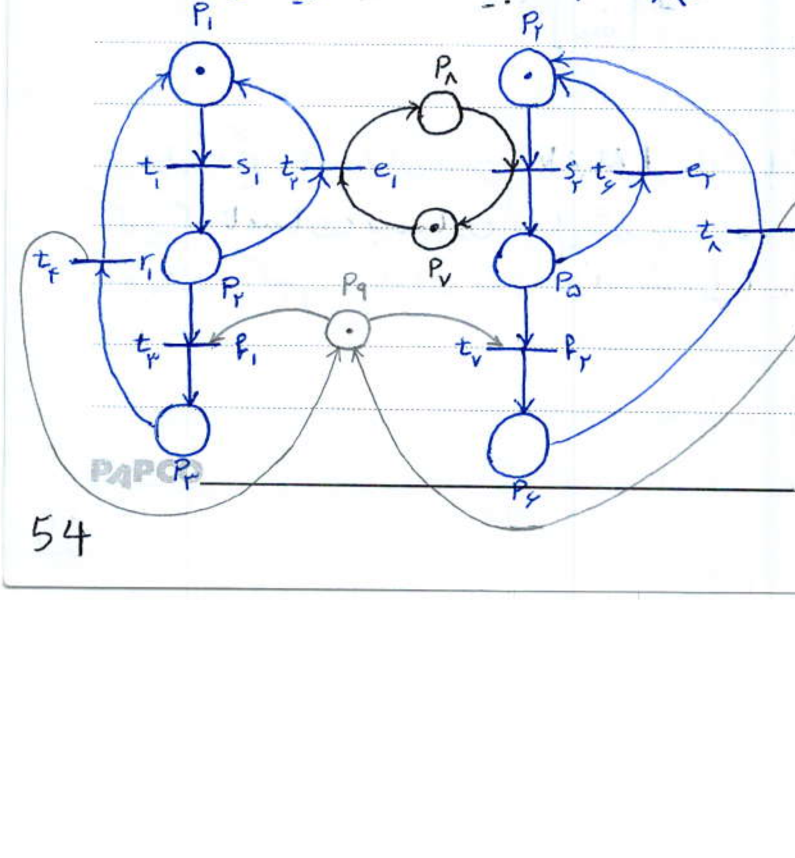
۲- بیان ضابطه با استفاده از مدل شبکه‌های پتری و کمان‌های فعالساز و غیرفعالساز :

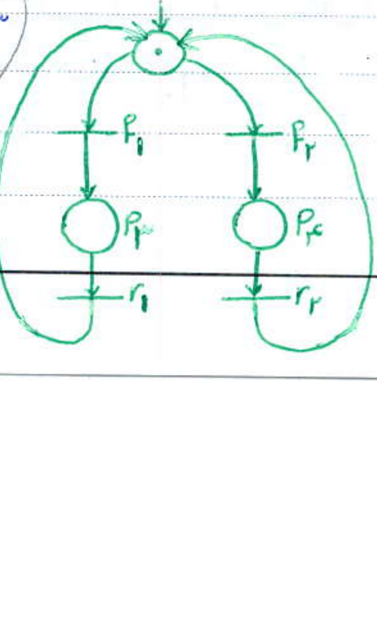
این روش منجر به شبکه پتری می‌شود که به دلیل استفاده از کمان‌های فعالساز و غیرفعالساز، استاندارد نمی‌باشد.
اگر ضابطه بصورت آتش شدن گذرگاهی در صورت حضور در حالتی از اجزای مدل بیان شود، استفاده از این روش کاربرد دارد.
باید توجه داشت گاهی بیان ضابطه بصورت متمم، مدلسازی راحتری دارد. بطور مثال به جای عنوان کردن "اتفاق نیافتن" بتوانیم "اتفاق افتادن" را استفاده کنیم.

۳- بیان ضابطه در قالب دسته نامعادلات خطی :

این روش نیز متداول می‌باشد. در این زمینه ۳ موضوع مطرح می‌شود: ① بیان نامعادلات ② ساده‌سازی نامعادلات ③ اعمال نامعادلات

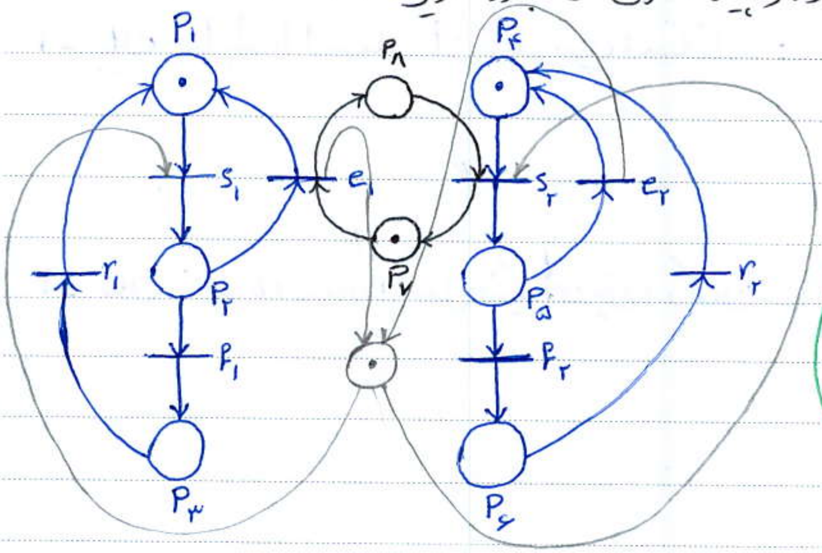
مثال - در مثال ص ، که تمامی پیشامدها، قابل کنترل می‌باشند فرض کنید خواسته‌ای مبنی بر اینکه "دو ماشین همزمان دچار خطا نشوند" (یا اینکه هیچ وقت دو ماشین با هم به تعمیرگاه نروند) با نامعادله $m_3+m_6 \leq 1$ بیان می‌شود. بیان این نامعادله با اضافه کردن یک مکان به شبکه و به صورت زیر امکان پذیر است :

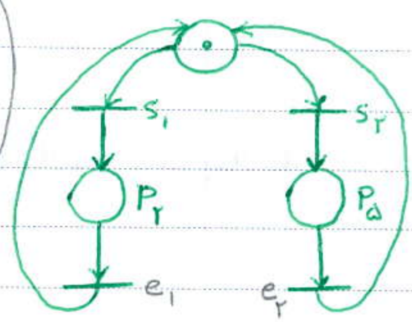




Subject
Date

مثال - در مثال قبل، بیان خواسته اینکه "هیچ وقت دو ماشین با هم کار نکنند" توسط نامعادله $m_2+m_5\leq 1$ انجام می شود و پیاده سازی آن بصورت زیر است .





باید توجه داشت در این مثال و مثال قبل، مسائل مستقل از بافر می باشند (می توان بافر را در نظر نگرفت).

نامعادلات مطرح شده را به صورت زیر نیز می توان نوشت :

$$m_2+m_5\leq 1 \Rightarrow 1\times m(P_2)+1\times m(P_5)\leq 1$$

$$\Rightarrow 0\times m(P_1)+1\times m(P_2)+0\times m(P_3)+0\times m(P_4)+1\times m(P_5)+0\times m(P_6)\leq 1$$

$$\Rightarrow \underbrace{\begin{bmatrix}0 & 1 & 0 & 0 & 1 & 0\end{bmatrix}}_{L}\underbrace{\begin{bmatrix}m_1\\ m_2\\ \vdots\\ m_6\end{bmatrix}}_{M}\leq 1 \Rightarrow LM\leq 1$$

مکان کنترلی

به جای اینکه نامعادله ای به فرم $LM\leq k$ داشته باشیم می توانیم یک مکان مجعول به گونه ای اضافه کنیم که **نامساوی به مساوی** تبدیل شود. به عبارت دیگر، کمبود را در نشانه های مکان اضافه شده جبران می کنیم. نشانه های مکان اضافه شده را با m_c نشان می دهیم. بطور مثال :

$$m_2+m_5\leq 1 \longrightarrow m_2+m_5+m_{c_1}=1$$

باید توجه داشت به ازای هر نامعادله نیاز به یک مکان کنترلی می باشد و تعداد نشانه هر یک از این مکان ها تنها در یکی از نامعادلات وارد شده و آن را به معادله تبدیل می کند.

Subject
Date

با اضافه شدن مکان های کنترلی، بیان معادله انتقال حالت به شکل زیر است:

$$M' = M + W X$$

$$\begin{bmatrix} M'_p \\ M'_c \end{bmatrix} = \begin{bmatrix} M_p \\ M_c \end{bmatrix} + \begin{bmatrix} W_p \\ W_c \end{bmatrix} X$$

در ابتدا، مکان اضافه شده، بصورت یک مکان ایزوله می باشد و می توان از گذرگاه های مختلف به این مکان، کمان داشت. این مکان کنترلی طوری اضافه می شود که P_invariant تشکیل شود. به عبارت دیگر چگونگی اتصال کمان ها به مکان های کنترلی براساس شکل گیری P_invariant ها شکل می گیرد.

داریم:

$$k_1 m_1 + k_2 m_2 + \dots + k_n m_n \leq C \Rightarrow \underbrace{\begin{bmatrix} k_1 & k_2 & \cdots & k_n \end{bmatrix}}_{L} \underbrace{\begin{bmatrix} m(P_1) \\ m(P_2) \\ \vdots \\ m(P_n) \end{bmatrix}}_{M_p} \leq C$$

بیان تمامی نامعادلات به شکل فوق، منجر به فرم ماتریسی به شکل زیر می گردد:

$$L M_p \leq C$$

L → ماتریس ضرائب نامعادلات → $m \times n$

C → بردار مقادیر ثابت → $m \times 1$

M_p → بردار نشانه مکان ها → $n \times 1$

m: تعداد نامعادلات

n: تعداد مکان ها

با اضافه کردن مکان های کنترلی به شبکه، بردار نشانه ها بصورت زیر تغییر می کند:

$$M = \begin{bmatrix} M_p \\ M_c \end{bmatrix} \rightarrow (n+m) \times 1$$

M_c → بردار نشانه های مکان های کنترلی → $m \times 1$

نشانه مکان های کنترلی به نحوی به نامعادلات اضافه می شوند که نامساوی ها به تساوی تبدیل گردند یعنی:

$$L M_p + M_c = C \xrightarrow{\text{وضعیت اولیه نشانه در مکان های کنترلی}} M_c(0) = C - L M_p(0)$$

بیان رابطه فوق برحسب بردار نشانه ها (M) به شکل زیر خواهد بود:

ماتریس همانی $m \times m$

$$\begin{bmatrix} L & I \end{bmatrix} \begin{bmatrix} M_p \\ M_c \end{bmatrix} = C \Rightarrow \begin{bmatrix} L & I \end{bmatrix} M = C$$

از آنجائیکه معادله فوق نشان دهنده P_invariant است $(\sum k_i m_i = c)$ بنابه توضیحات صفحه ۴۷ جزوه، هر یک از بردارهای ضرائب (یعنی هر یک از سطرهای ماتریس $[L \ \ I]$) در معادله اساسی $W^T i_p = 0$ صدق می کند. از آنجائیکه i_p ستونی است (ولی بردار ضرائب سطری هستند) از معادله اساسی ترانهاده گرفته می شود و داریم: $i_p^T W = 0$. همچنین با فرض $X^T = [L \ \ I]$ می توان فرم ماتریسی این معادله برای تمامی بردارهای ضرائب را بصورت $X^T W = 0$ بنویسیم.

$$W = \begin{bmatrix} W_p \\ W_c \end{bmatrix}$$

مشابه بردار نشانه ها (M)، ماتریس تلاقی (W) عبارت است از:

W_c که نشان دهنده نحوه اتصال کمان ها به مکان های کنترلی است از عبارت زیر بدست می آید:

$$X^T W = 0 \Rightarrow \begin{bmatrix} L & I \end{bmatrix} \begin{bmatrix} W_p \\ W_c \end{bmatrix} = 0 \Rightarrow L W_p + W_c = 0 \Rightarrow \underset{m \times j}{W_c} = - \underset{m \times n}{L} \underset{n \times j}{W_p}$$

j: تعداد گذرگاه ها

در W، مقادیر +، ورود به گذرگاه و مقادیر −، خروج است

۴- بیان ضوابط با استفاده از لیست حالات ممنوع (Forbidden States):

در شبکه‌های پتری تک نشانه (safe)، می‌توان حالات ممنوع را به نامعادله تبدیل نمود. در این شرایط تعداد نامعادلات زیاد می‌شود و از آنجا که به ازای هر نامعادله یک مکان کنترلی اضافه می‌شود، تعداد مکانهای کنترلی زیاد خواهد بود.

بیان ضوابط بدین شکل، در شبکه‌های پتری تک نشانه (safe) و چند نشانه (non-safe) متفاوت است که در اینجا تنها به شبکه‌های safe می‌پردازیم.

شبکه‌های پتری تک نشانه (Safe PN):

در این نوع شبکه پتری به هر حالت ممنوع یک نامعادله تخصیص داده می‌شود و از آنجایی که برای هر نامعادله نیاز به یک مکان کنترلی می‌باشد، تعداد مکانهای اضافه شده به سیستم واقعی بسیار زیاد می‌شود و این اشکال تنها از طریق ساده سازی نامعادلات قابل حل است، فرض کنید یک حالت ممنوع در شبکه به فرم زیر باشد:

$$M_F = \begin{bmatrix} m_F(P_1) \\ m_F(P_2) \\ \vdots \\ m_F(P_n) \end{bmatrix}$$

برای جلوگیری از اتفاق افتادن این حالت ممنوع، کافی است نامعادله زیر را به سیستم تحمیل کنیم:

دقت: در شبکه پتری تک نشانه، تنها می‌توانند مقادیر ۰ و 1 داشته باشند

$$\underbrace{m_F(P_1)\,m(P_1) + m_F(P_2)\,m(P_2) + \cdots + m_F(P_n)\,m(P_n)}_{M_F^T M} \leq (\underbrace{m_F(P_1) + \cdots + m_F(P_n)}_{|M_F| \text{ یا } \|M_F\| \text{ یا } \sup(M_F)} - 1)$$

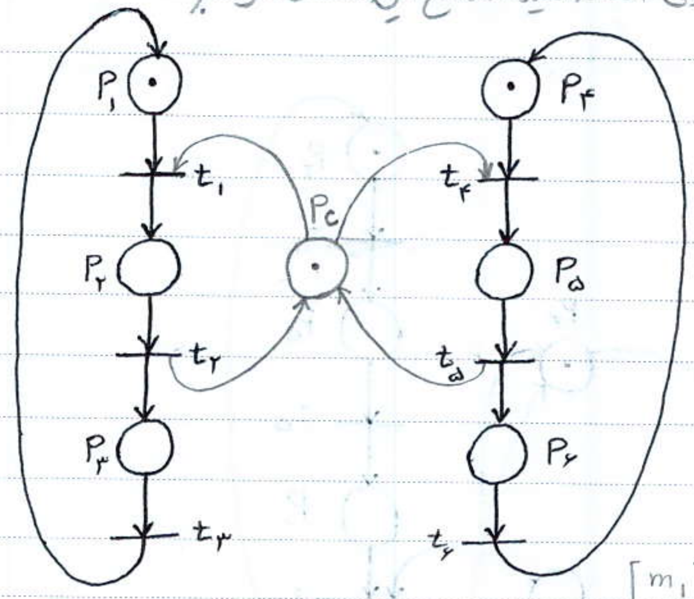
$$\Longrightarrow \quad M_F^T M \leq (\|M_F\|_1 - 1)$$

فرم فوق با شرط تک نشانه و ناورداد بودن شبکه نوشته شده است

باید توجه داشت که این نامعادله باید برای تمام حالتهای مجاز برقرار باشد بعبارت دیگر مانع اتفاق افتادن حالات مجاز نگردد.

مثال - در شبکه پتری تک نشانه زیر، فرض کنید حالت ممنوع $P_2 P_5$ باشد بعبارت دیگر M_F^T بصورت $[0\ 1\ 0\ 0\ 1\ 0]$ قرار گیرد. نامعادله کنترلی که مانع این اتفاق می شود را بدست آورید.

٪ و مکان کنترلی را به نحوی اضافه کنید که مانع این اتفاق گردد٪



$$M_F^T M \leqslant (\|M_F\|_1 - 1) \Rightarrow [0\ 1\ 0\ 0\ 1\ 0]\begin{bmatrix} m_1 \\ m_2 \\ m_3 \\ m_4 \\ m_5 \\ m_6 \end{bmatrix} \leqslant \left((0+1+0+0+1+0)-1\right)$$

$$\Rightarrow m_2 + m_5 \leqslant 1 \longrightarrow$$ این نامعادله برای تمام حالات مجاز باید برقرار باشد

٪

در اضافه کردن مکان کنترلی باید به نکات زیر توجه شود:

① تعداد مکان‌های کنترلی = تعداد نامعادلات

② وضعیت اولیه نشانه مکان‌های کنترلی: $M_c(0) = C - L M_p(0)$

③ اتصال مکان‌های کنترلی به گذرگاه‌ها: $W_c = -L W_p$

بدین ترتیب در این مسئله:

۱ مکان کنترلی P_c داریم.

$$M_c(0) = C - L M_p(0) = 1 - [0\ 1\ 0\ 0\ 1\ 0]\begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} = 1 - 0 = 1$$

$$W_c = -L W_p = -[0\ 1\ 0\ 0\ 1\ 0]\begin{bmatrix} -1 & 0 & +1 & 0 & 0 & 0 \\ +1 & -1 & 0 & 0 & 0 & 0 \\ 0 & +1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & +1 \\ 0 & 0 & 0 & +1 & -1 & 0 \\ 0 & 0 & 0 & 0 & +1 & -1 \end{bmatrix} = -[1\ -1\ 0\ 1\ -1\ 0] = [-1\ 1\ 0\ -1\ 1\ 0]$$

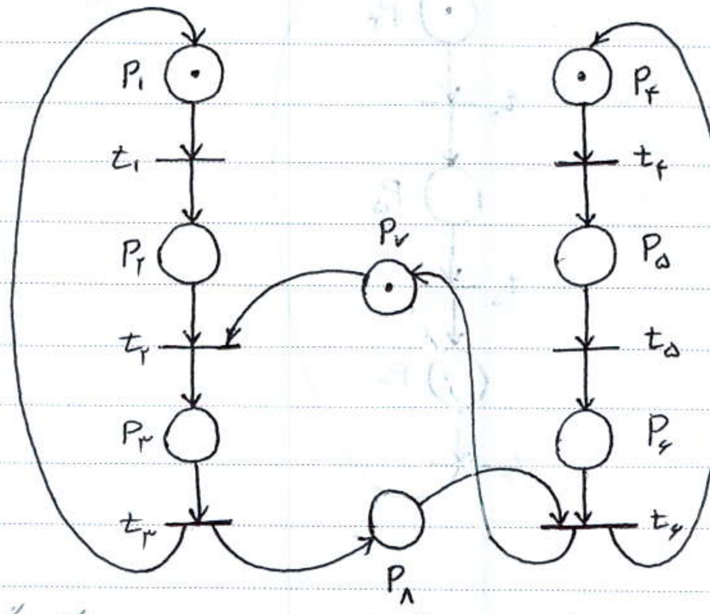
Subject
Date

مثال - در شبکه پتری زیر، فرض کنید گذرگاه t_2 غیرقابل کنترل باشد:

الف - حالات ممنوع را بنویسید

ب - نامعادلات کنترلی را بدست آورید

ج - با ساده‌سازی نامعادلات، حداقل مکان کنترلی را بدست آورید.



الف - غیرقابل کنترل بودن t_2 به معنی آتش شدن گذرگاه t_2 بدون برقراری شرط آتش گذرگاه (یعنی نشانه‌دار بودن همزمان P_2 و P_7) می‌باشد. بدین ترتیب منتقل از حالتی که ماشین M_2 در آن قرار دارد، حالت P_2P_8 حالت ممنوع است بنابراین لیست حالات ممنوع عبارتند از:

سوال: حالت P_1P_7 و P_3P_7 چطور؟

$P_2P_4P_8$
$P_2P_5P_8$
$P_2P_6P_8$

ب - براین اساس نامعادلات کنترلی عبارتند از:

$$m_2+m_4+m_8 \leqslant 2$$
$$m_2+m_5+m_8 \leqslant 2$$
$$m_2+m_6+m_8 \leqslant 2$$

ج - از آنجائیکه مکان‌های P_4، P_5 و P_6 یک P-invariant هستند $(m_4+m_5+m_6=1)$ می‌توان سه نامعادله فوق را به یک نامعادله ساده نمود:

$$m_2+m_8 \leqslant 1$$

اینکار مشابه ساده‌سازی مدار منطقی $x+x'=1$ می‌باشد

نامعادله فوق همان حالت ممنوع P_2P_8 است

طراحی کنترل کننده با استفاده از شبکه‌ی پتری :

هدف از طراحی کنترل کننده می تواند یکی از موارد زیر باشد :

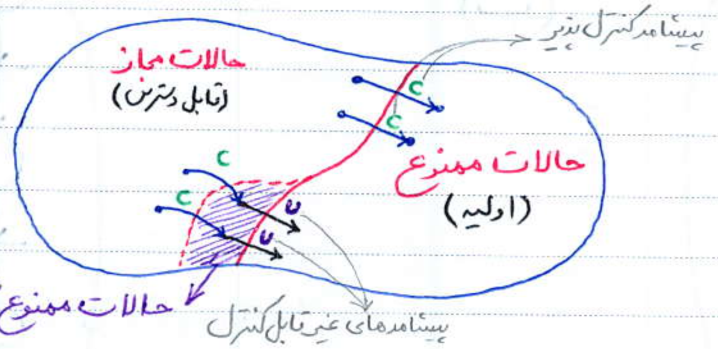
① اجرای ضابطه مورد نظر : مواردی همچون حالت ممنوع و deadlock اهمیت دارد ← در مقالات با این مضمون کمتر کار

② اصلاح کنترل کننده اولیه (یا برطرف کردن اشکالات ناشی از اعمال کنترل کننده اولیه) : می کنند و بیشتر با هدف حل مشکل کار می کنند

در شبکه پتری ، مکان کنترلی به شبکه اضافه می کنیم که این به معنی تغییر برنامه PLC و کاهش درجه آزادی سیستم است.

ورود به حالات ممنوع از حالت‌های مجاز ممکن است به دو شکل اتفاق افتد با
- پیشامدهای کنترل پذیر
- پیشامدهای کنترل ناپذیر

اگر رفتن از حالت مجاز به ممنوع (حالات ممنوع اولیه) توسط پیشامد غیرقابل کنترل اتفاق افتد مرز حالات ممنوع جابه‌جا شده و حالات ممنوع ثانویه ایجاد می شود . بدین ترتیب مرز حالات ممنوع گسترش می یابد . کاری که در هدف ② انجام می شود تعیین این مرز است .



① اصلاح نامعادلات : نامعادلات را طوری اصلاح می کنیم که در حضور پیشامدهای غیرقابل کنترل ، مشکل ایجاد نکنند .

② تئوری ریجیون : برای جلوگیری از حالات ممنوع ، گراف قابل دسترس را رسم می کنند مکان کنترلی به نحوی اضافه می کنند که در آتش شدن گذرگاه‌ها محدودیت ایجاد شود تا به سمت ممنوع نرود و به سمت مجاز برود .

برای دستیابی به هدف ②

③ ایجاد تعدادی نامعادله : برای جلوگیری از حالات ممنوع در حالت کلی .

این نامعادلات باید به ازای تمام حالات مجاز تصدیق شود و به ازای تعدادی یا تمام حالات ممنوع باید نقض گردد .

تبدیل شبکه پتری به زبان Ladder :

• مشابه روش تبدیل اتومات به Ladder ، در تبدیل شبکه پتری نیز برای هر مکان، یک بیت حافظه در نظر می گیریم.

• برنامه برای هر Invariant نوشته می شود.

مراحل کار بدین شکل است : ① برای هر مکان، یک بیت حافظه در نظر می گیریم.

② ایجاد شرایط اولیه : (متفاوت با اتومات است و باید چند حالت اولیه ایجاد شود)

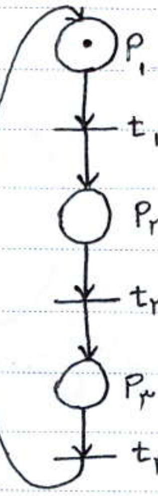
③ نوشتن برنامه برای هر Invariant .

④ مقداردهی خروجی ها براساس حالت ها .

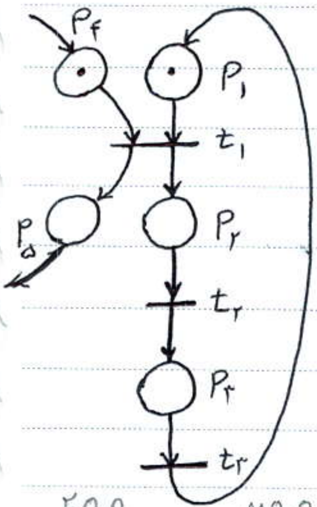
مثال : برنامه LD شبکه های پتری زیر را بنویسید (رسم نمایید) .

یعنی چند نشانه را بصورت تک نشانه در نظر می گیریم

(الف)



(ب)



$P_1 \rightarrow M0.0$

$P_2 \rightarrow M0.1$

$P_3 \rightarrow M0.2$

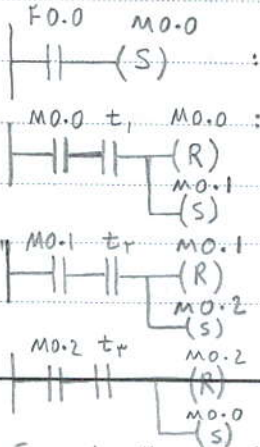
$P_4 \rightarrow M0.3$

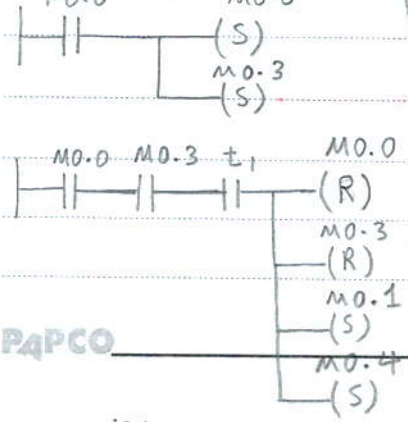
$P_5 \rightarrow M0.4$

اگر مدل چند نشانه باشد (non safe)، مدل کنترلی نیست و مدل ظرفیتی است و به جای بنش چند نشانه، شمارنده در نظر می گیریم و بنش چند نشانه را بعنوان متغیر جدا می بینیم و نشان این متغیر نشان دهنده خالی بودن یا نبودن (پر بودن یا نبودن) است

* برنامه راه انداز :

* برنامه OB100 :





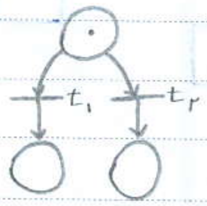
* مقداردهی خروجی براساس خروجی تعریف شده یا نبودن است * بقیه موارد مشابه (الف) می باشد .

زبان برنامه نویسی SFC (Sequential Function Chart)

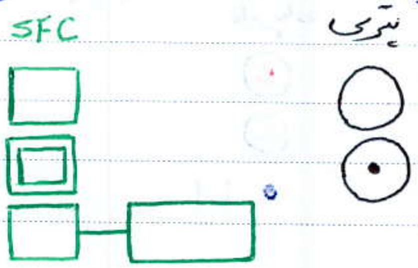
SFC همانند زبان برنامه‌نویسی کامپیوتر است و به صورت سلسله‌وار و گام به گام می‌باشد (برخلاف LD) و در جاهائیکه لازم است یک کاری همواره انجام شود یا بررسی شود (نظیر روشن شدن موتور با یک کلید) مفید نیست و این کار باید در تمام گام‌ها مورد بررسی و انجام قرار گیرد که پیچیده کننده است.

Graphset، همانند SFC است ولی استاندارد نمی‌باشد زیرا در آتش شدن همزمان دو گذرگاه از یک مکان، هر دو مکان بعد از گذرگاه‌ها نشانه‌دار می‌شوند که این مورد با شبکه پتری تناقض دارد و در SFC، با در نظر گرفتن اولویت با چپ، تنها یک مکان نشانه‌دار می‌شوند. SFC همان پتری است

اولویت دهی لازم است. ⟵



پتری از نظر اصولی با SFC یکسان است و تفاوت‌های SFC و پتری عبارتند از:

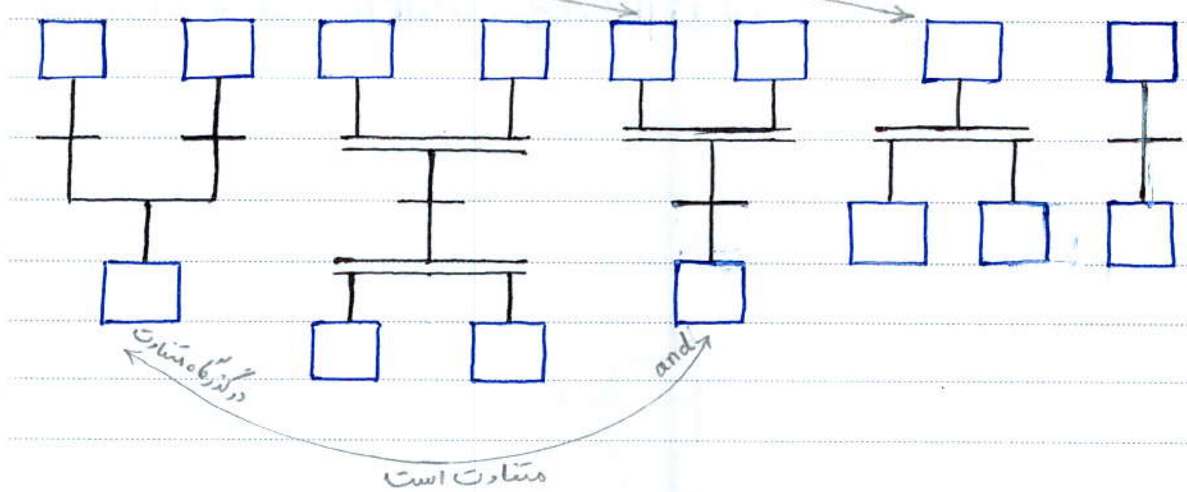


۱) Place
۲) Place اولیه
۳) بلوک‌های عملگر

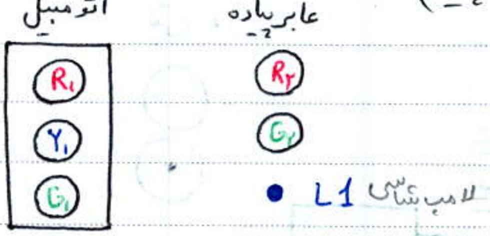
عملگرهایی که در بلوک کنار حالت قرار می‌گیرند عبارتند از:

N	nonstored	—(Q0.0)
S	set	
R	reset	
P	pulse	
D	delay	
SD	stored delay	

گذرگاه در پیاده سازی شبکه پتری با زبان SFC، به شکل های مختلف مدل می شود.
برخی از گذرگاه ها باید با استفاده از ریل مدل شوند اما شرط تمامی گذرگاه ها قابل برنامه نویسی است.
این برنامه از طریق بلوک عملگر در کنار گذرگاه نوشته می شود.
استفاده از ریل در گذرگاه هایی که انشعاب دارند و یا ترکیب می شوند مورد استفاده قرار می گیرند

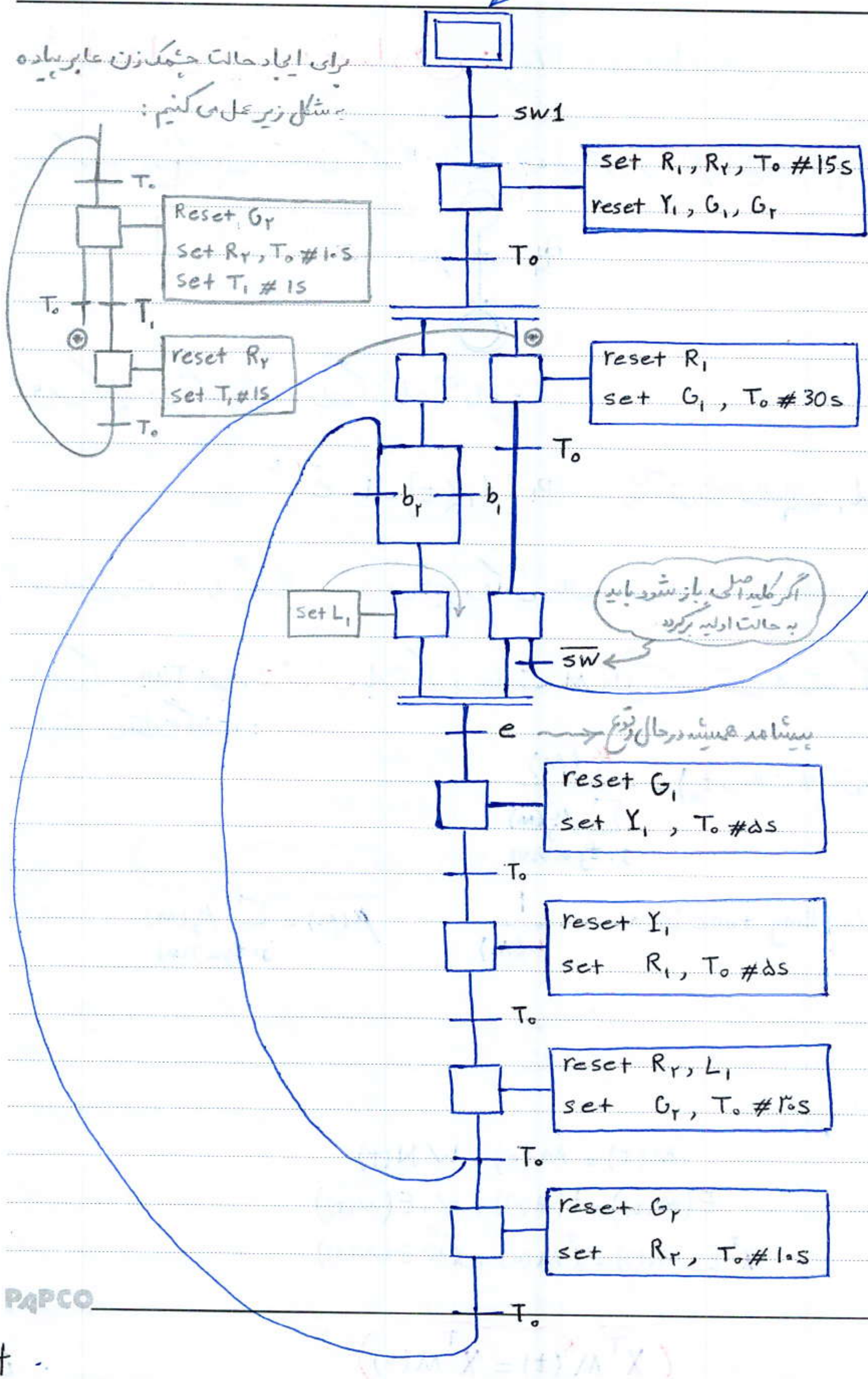


مثال - (چراغ عابر پیاده فرمان پذیر)



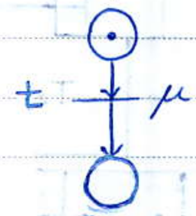
با بسته شدن کلید اصلی، کلیه چراغ ها برای ۱۵ ثانیه در وضعیت قرمز قرار بگیرند و بعد از آن چراغ ماشین ها در حالت سبز و عابر پیاده در وضعیت قرمز بماند. چنانچه هر یک از زمان های B_1 و B_2 (دو شاسی در طرف خیابان) فشرده شوند در صورتی که حداقل ۳۰ ثانیه از سبز بودن چراغ ماشین ها گذشته باشد ابتدا چراغ زرد به مدت ۵ ثانیه و سپس چراغ قرمز برای خودروها روشن شده و بعد از ۵ ثانیه چراغ عابر پیاده سبز شود و چراغ عابر به مدت ۳۰ ثانیه سبز و سپس قرمز می شود و بعد از ۱۰ ثانیه چراغ خودرو ها مجدداً سبز خواهد شد. فشردن کلید ها در وضعیت سبز بودن چراغ عابر تاثیر ندارد اما در حالت قرمز بودن چراغ عابر پیاده ذخیره می شود.

برای ایجاد حالت چشمک‌زن عابر پیاده به شکل زیر عمل می‌کنیم :

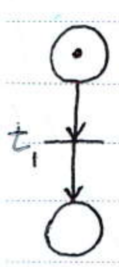


شبکه‌های پتری تصادفی (Stochastic PN):

با رسم گراف قابل دسترس از شبکه پتری تصادفی می‌توان به زنجیره مارکوف رسید. ضریب λ در زنجیره مارکوف در شبکه پتری تصادفی به گذرگاه نسبت داده می‌شود و با μ نمایش می‌دهیم:



فرض کنید در شبکه پتری، برای گذرگاه T_1 داریم:



t_1 $d_1 = t_1$ زمان آتش شدن گذرگاه

$$Pr[d_1 \leqslant t] = 1 - e^{-\mu t}$$

به نوعی ثابت زمانی است.

در اینصورت μ را به گذرگاه نسبت می‌دهیم و عکس زمان حضور در حالت در نظر می‌گیریم.

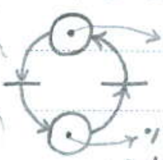
فرض کنید $T(M)$ مجموعه گذرگاه‌هایی باشند که در حالت M قابل آتش شدن هستند. اگر $t_k \in T(M)$ باشد می‌توان گفت:

$$Pr(t_k \text{ در حالت } M \text{ آتش شود}) = \frac{\mu_k(M)}{\sum_{j:\, t_j \in T(M)} \mu_j(M)} = \frac{\mu_k(M)}{\mu(M)}$$

بر این اساس متوسط زمان حضور در حالت M عبارت است از:

$$\text{mean dwelling time in } M = \frac{1}{\mu(M)} \qquad \mu(M) = \sum_{j:\, t_j \in T(M)} \mu_j(M)$$

به دنبال آن هستیم که در یک invariant در شبکه پتری تصادفی، درصد نشانه در یک مکان را بدست آوریم. به عبارت دیگر می‌خواهیم ببینیم چند درصد یک نشانه در یک مکان از invariant حضور دارد:



بدین منظور از معادله زیر استفاده می‌کنیم:

$$M(t) = M(0) + W\,N(t) \quad ?$$

برای بدست آوردن مقادیر مورد انتظار:

$$E(M(t)) = E(M(0)) + W \cdot E(N(t)) \longleftarrow \text{امید ریاضی}$$

با ضرب یک P-invariant:

$$X^T E(M(t)) = X^T M(0) + \cancel{X^T W}^{\,0} E(N(t))$$

بدین ترتیب برای مقدار متوسط نشانه در مکان‌ها می‌توان معادله P-invariant را بصورت زیر استفاده نمود:

$$E(M(t)) \qquad \boxed{X^T M^*(t) = X^T M(0)}$$

درجه آتش شدن یک گذرگاه : (D_j)

حداقل نسبت مقدار نشانه مکان به وزن کمان متصل به مکان گذرگاه را درجه آتش شدن یک گذرگاه می نامند.

$$\text{درجه آتش شدن گذرگاه } j \text{ ام} = D_j = \min\left(\frac{m(P_i)}{W_{ij}} \Big|_{i\in \cdot t_j}\right)$$

براساس درجه آتش شدن یک گذرگاه، μ موثر گذرگاه بصورت زیر تعریف می شود:

$$\text{موثر گذرگاه } j \text{ ام } \mu = \mu_{ej} = D_j \times \mu_j$$

(رابطه متقابل با ساختار infinit می باشد)

متوسط فرکانس آتش شدن یک گذرگاه : (F_j)

متوسط فرکانس آتش شدن یک گذرگاه برابر با امید ریاضی این فرکانس می باشد. در اینجا نیز مشابه بحث زنجیره مارکوف، فرکانس عکس زمان نیست و با μ در ارتباط است:

براساس متوسط فرکانس آتش شدن گذرگاه ها (F^*) می توان مشابه معادله P-invariant در صفحه قبل، در T-invariant نیز معادله ای به فرم زیر داشته باشیم:

$$W.F^* = 0$$

براین اساس متوسط زمان حضور یک نشانه در یک مکان از رابطه زیر بدست می آید:

$$\text{mean dwelling time of a token in place } P_i = D^*(P_i) = \frac{m^*(P_i)}{\underbrace{W_i^+ \cdot F^*}_{\text{یک عدد}}}$$

W_i^+ : سطر i ام ماتریس W^+ (سطری)

F^* (ستونی):

$$F^* = \begin{bmatrix} F^*_{t_1} \\ \vdots \\ F^*_{t_j} \end{bmatrix}$$

مثال ـ سیستمی را در نظر بگیرید که دارای ۲ جزء بافر با نرخ μ_1 و ماشین با نرخ μ_2 به شکل زیر می باشد. ظرفیت بافر حداکثر برابر ۲ و ظرفیت ماشین حداکثر برابر ۱ می باشد و ظرفیت کل سیستم ۲ است.

واحد زمان باید مشخص گردد

(الف) شبکه پتری تصادفی و گراف قابل دسترس آنرا رسم نمایید.

(ب) احتمال حضور در هر مکان را محاسبه کنید. (فرض $\mu_1 = 2$ ، $\mu_2 = 1$)

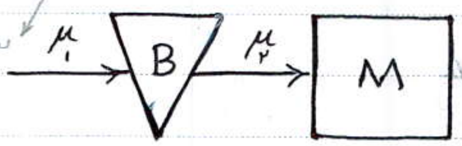
۱ نفر در یک ساعت

بطور مثال: ۲ نفر در یک ساعت

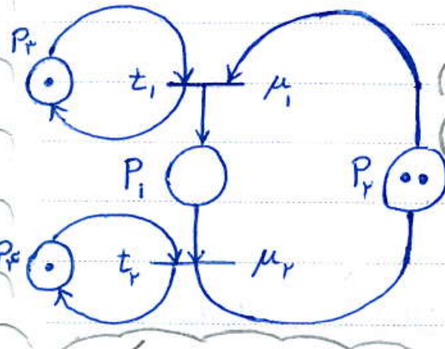
(ج) متوسط نشانه در هر مکان

(د) متوسط نرخ آتش شدن هر گذرگاه

(هـ) متوسط زمان حضور یک نشانه در یک مکان



می توان برای این مسئله، مثال آرایشگاه را زد که دارای ۲ جا رختی است و ۲ صندلی انتظار و ۱ صندلی آرایش می باشد. ورودی آرایشگاه شیشه ای است و افراد از بیرون می بینند، اگر جارختی پر باشد (به معنی پر بودن ظرفیت کل ماشین است) مشتری از بیرون دیده و اصلاً وارد نمی شود به عبارت دیگر در صندلی انتظار (که همان بافر است) قرار نمی گیرد و بدین ترتیب در این شرایط نرخ بافر صفر می شود.



(الف) μ_1 : نرخ ورود است (و متناظر آن زمان انتظار ورود داریم)

μ_2 : نرخ ماشین کاری است (" زمان انتظار ماشین کاری ")

P_1 : ظرفیت اشغال کل سیستم (در نوبت ماشینکاری)

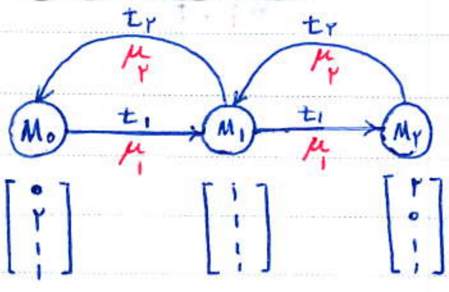
P_2 : ظرفیت آزاد کل سیستم

P_3 : محدود کننده ظرفیت

P_4 : محدود کننده ظرفیت

مدل فوق برای آنالیز است نه کنترل به توضیحات ص ۶۹ توجه کنید

P_3 و P_4 برای ثابت نگهداشتن نرخ مورد استفاده قرار گرفته اند.



$$T_{avg}(M_1) = \frac{1}{\mu_1 + \mu_2}$$

(ب) به منظور محاسبه احتمال حضور در هر یک از حالتها (~~مکانها~~ M_0 ، M_1 و M_2) از روابط زیر استفاده می‌کنیم:

$$\begin{cases} PA = 0 \\ \sum P_i = 1 \end{cases} \Rightarrow [P_0\ P_1\ P_2] \begin{array}{c} \begin{matrix} M_0 & M_1 & M_2 \end{matrix} \\ \begin{bmatrix} -2 & 2 & 0 \\ 1 & -3 & 2 \\ 0 & 1 & -1 \end{bmatrix} \end{array} = 0 \Rightarrow \begin{cases} P_0 = \frac{1}{7} \\ P_1 = \frac{2}{7} \\ P_2 = \frac{4}{7} \end{cases}$$

باید توجه داشت که می‌توانستیم از قانون برابری نسبت دفعات خروج از یک حالت با نسبت دفعات ورود به آن نیز استفاده کنیم (تعداد دفعات خروج از حالت m_i = تعداد دفعات ورود به این حالت)
در واقع فرمول فوق بیانگر همین عبارت است داریم:

احتمال حضور در M_1 — نرخ انتقال

M_0: تعداد دفعات ورود به M_0 = $P_1 \mu_2$ ، تعداد دفعات خروج از M_0 = $P_0 \mu_1$

M_1:

M_2:

$$\left.\begin{aligned} & P_1 \mu_2 = P_0 \mu_1 \\ & P_2 \mu_2 + P_0 \mu_1 = P_1 \mu_1 + P_1 \mu_2 \\ & P_1 \mu_1 = P_2 \mu_2 \\ & P_0 + P_1 + P_2 = 1 \end{aligned}\right\} \begin{cases} P_0 = \frac{1}{7} \\ P_1 = \frac{2}{7} \\ P_2 = \frac{4}{7} \end{cases}$$

(ج) برای محاسبه متوسط نشانه در هر مکان از رابطه $X^T M^*(t) = X^T M(0)$ استفاده می‌کنیم. بطور مشابه مورد (ب)، می‌توان محاسبات را از قانون دیگری نیز بدست آوریم. در اینجا متوسط نشانه در هر مکان از مجموع حاصلضرب تعداد نشانه آن مکان (در حالت) در احتمال حضور در آن حالت برای تمام حالتها بدست می‌آید $\left(\sum_{\text{تمام حالات}} m_j(p_i)\, Pr(M_j) \right)$

$$M^*(P_1) = m_0(P_1)\,Pr(M_0) + m_1(P_1)\,Pr(M_1) + m_2(P_1)\,Pr(M_2) = 0 \times \frac{1}{7} + 1 \times \frac{2}{7} + 2 \times \frac{4}{7} = \frac{10}{7}$$

$$M^*(P_2) = 2 \times \frac{1}{7} + 1 \times \frac{2}{7} + 0 \times \frac{4}{7} = \frac{4}{7}$$

از آنجائیکه P_1 و P_2 یک P-invariant هستند و تعداد نشانه‌ها در این invariant برابر ۲ می‌باشد بنابراین باید مجموع متوسط نشانه‌ها در مکانهای این invariant برابر ۲ باشد.

$$M^*(P_1) + M^*(P_2) = \frac{10}{7} + \frac{4}{7} = 2$$

با توجه به اینکه P_3 و P_4 همواره نشانه دارند بنابراین: $M^*(P_3) = M^*(P_4) = 1$ خواهد بود و اگر از روابط نیز محاسبه کنیم، به این نتیجه می‌رسیم.

Subject
Date

(د) محاسبه متوسط فرکانس آتش شدن هر گذرگاه از رابطه $WF^* = 0$ امکانپذیر است. همچنین می‌توان از قاعده مجموع حاصلضرب‌های احتمال حضور در یک حالت مرتبط با گذرگاه در نرخ آن گذرگاه نیز استفاده نمود $\left(\sum_{\text{حالات مرتبط با } t_j} Pr(M_i)\,\mu_j \right)$

$$F^*_{t_1} = Pr(M_0)\,\mu_1 + Pr(M_1)\,\mu_1 = \frac{1}{7}\times 2 + \frac{2}{7}\times 2 = \frac{6}{7}$$

$$F^*_{t_2} = Pr(M_2)\,\mu_2 + Pr(M_1)\,\mu_2 = \frac{4}{7}\times 1 + \frac{2}{7}\times 1 = \frac{6}{7}$$

(هـ) متوسط زمان حضور یک نشانه در یک مکان، از رابطه زیر بدست می‌آید (ص ۶۶):

$$D^*(P_i) = \frac{m^*(P_i)}{w_i^+ \cdot F^*}$$

بدین ترتیب داریم:

$$D^*(P_1) = \frac{m^*(P_1)}{w_1^+ \cdot F^*} = \frac{\frac{10}{7}}{\begin{bmatrix}1 & 0\end{bmatrix}\begin{bmatrix}\frac{6}{7}\\ \frac{6}{7}\end{bmatrix}} = \frac{\frac{10}{7}}{\frac{6}{7}} = \frac{10}{6} = \frac{5}{3}$$

$$D^*(P_2) = \frac{m^*(P_2)}{w_2^+ \cdot F^*} = \frac{\frac{4}{7}}{\begin{bmatrix}0 & 1\end{bmatrix}\begin{bmatrix}\frac{6}{7}\\ \frac{6}{7}\end{bmatrix}} = \frac{\frac{4}{7}}{\frac{6}{7}} = \frac{4}{6} = \frac{2}{3}$$

بنابراین متوسط حضور نشانه در مکان P_1 بیشتر از P_2 است بنابراین انباشتگی در P_1 بیشتر است و این موضوع به دلیل ۲ برابر بودن نرخ μ_2 (ورود) به نرخ μ_1 (خروج) می‌باشد.
بطور مثال ۴۰ دقیقه طول می‌کشد تا مشتری بیاید اما ۱۰۰ دقیقه طول می‌کشد تا خارج شود.

PAPCO

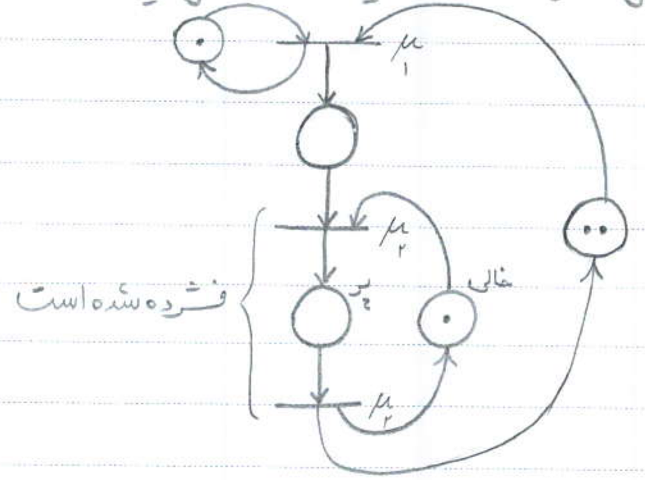
Subject
Date

مدل نشان داده شده در ص ۶۷ برای آنالیز است و نه کنترل.

برای آنالیز، مدل فشرده مورد استفاده قرار می‌گیرد (مثلاً در این مدل، عمل ماشینکاری را نمی‌بینیم).

برای اضافه کردن بخش فشرده شده، نیاز است اطلاعات بیشتری از سیستم داشته باشیم.

بطور مثال، مدلی که عمل ماشینکاری در آن دیده شده به شکل زیر است:



Subject

Date

مباحث با تیمانده :

- برای مکان و گذرگاه می‌توان زمان درنظر گرفت، اما اینکار بیشتر در آنالیز مفید است و نه در کنترل.

- در مدل P-time، برای مکان، زمان درنظر گرفته می‌شود.

و در مدل T-time، برای گذرگاه، زمان درنظر می‌گیرند.

مدل کنترلی باید به صورت P-time باشد.

PAPCO