



دانشگاه علم و صنعت ایران  
دانشکده مهندسی کامپیوتر

عنوان درس:

## ارزیابی کارایی سیستم‌های کامپیوتری Performance Evaluation of Computer Systems (PECS)

جلسه ۱۷: تحلیل شبکه‌های پتری

مدرس:

محمد عبداللهی ازگمی  
(Mohammad Abdollahi Azgomi)

[azgomi@iust.ac.ir](mailto:azgomi@iust.ac.ir)

### فهرست مطالب

- مقدمه
- خصوصیت‌های رفتاری سیستم‌های همروند:
  - تعاریف خصوصیت‌های رفتاری
  - دسته‌بندی خصوصیت‌های رفتاری
- فضای حالت شبکه‌های پتری
- تحلیل شبکه‌های پتری
- درخت دسترس پذیری
- تولید گراف دسترس پذیری
- شبیه‌سازی شبکه‌های پتری

## مقدمه

- در این جلسه ابتدا **خصوصیت‌های رفتاری (behavioral properties)** مدل‌های سیستم‌های همروند که با شبکه‌های پتری ایجاد می‌شوند را تعریف می‌کنیم.
- سپس در مورد تحلیل شبکه‌های پتری (**analysis of Petri nets**) با استفاده از **درخت دسترس‌پذیری (reachability tree)** و تبدیل آن به **گراف دسترس‌پذیری (reachability graph)** صحبت خواهیم نمود.
- در خاتمه هر اشاره‌ای به **شبیه‌سازی شبکه‌های پتری** خواهیم نمود.

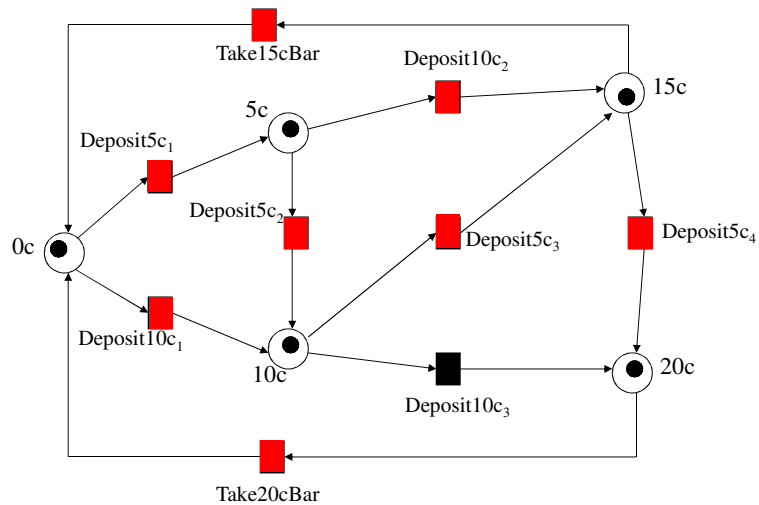
## خصوصیت‌های رفتاری

- در این بخش برخی از **مهمترین خصوصیت‌های رفتاری (behavioral properties)** مدل‌های سیستم‌های همروند، ایجاد شده با شبکه‌های پتری، را معرفی می‌کنیم. این خصوصیت‌ها عبارتند از:
  - دسترس‌پذیری (reachability)**: به این معنی است که آیا می‌توان از یک حالت خاص به حالت مورد نظر دیگری رسید؟
  - کران‌دار بودن (boundedness)**: آیا تعداد نشانه‌های موجود در یک مکان محدود به تعداد مشخصی است یا کرانی برای آن وجود ندارد؟
  - ایمنی (safety)**: آیا یک حالت ممنوع‌شده در رفتار سیستم بروز می‌کند؟
  - زنده‌بودن (liveness)**: آیا سیستم در یک حالت خاص متوقف می‌شود (یا اصطلاحاً می‌میرد)؟
  - فقدان بن‌بست (deadlock-freeness)**: آیا در رفتار سیستم، حالت بن‌بست بروز می‌کند؟
  - برگشت‌پذیری (reversibility)**: آیا امکان بازگشت به حالت اولیه وجود دارد؟
  - ماندگاری (persistence)**: آیا وقتی یک گذر توانا می‌شود در آن حالت باقی می‌ماند تا شلیک کند؟
  - انصاف (fairness)**: آیا انصاف در دفعات شلیک کردن یک گذر نسبت به سایر گذرها رعایت می‌شود؟

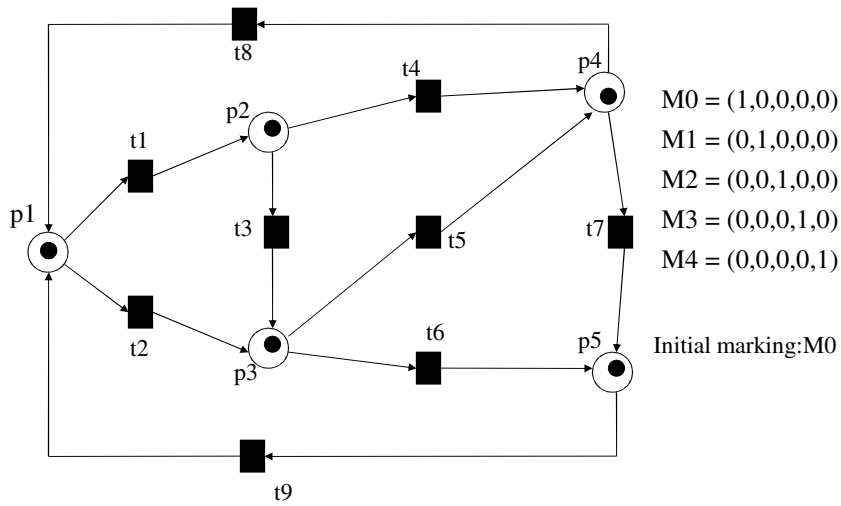
## دسترس پذیری

- دسترس پذیری در مورد حالت‌ها یا نشانه‌گذارهای خاص یک مدل شبکه پتری مطرح است و به این معنی است که:
  - آیا در اجرا یا رفتار پویای یک شبکه پتری از یک حالت خاص می‌توان به حالت مورد نظر دیگری رسید؟
- برای تشریح این خصوصیت رفتاری از مثال ماشین خودکار فروش خوراکی استفاده می‌کنیم و انیمیشن مدل را دوباره می‌بینیم...

## مرور مجدد مدل ماشین خودکار فروش خوراکی



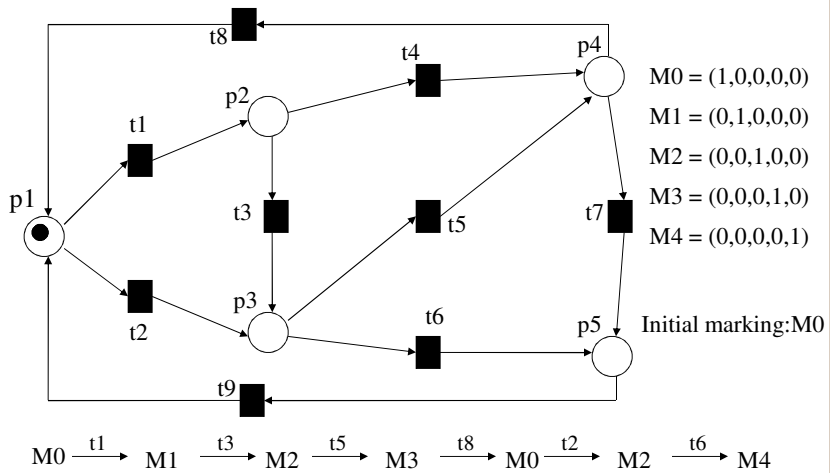
## مرور مجدد مدل ماشین خودکار فروش خوراکی



PECS#17 - Analysis of Petri Nets - By: M. Abdollahi Azgomi - IUST-CE



## دسترس پذیری



PECS#17 - Analysis of Petri Nets - By: M. Abdollahi Azgomi - IUST-CE



## دسترس پذیری

■ دنباله‌ای از شلیک گذرها به صورت زیر را در نظر بگیرید:

$$M_0 \xrightarrow{t_1} M_1 \xrightarrow{t_3} M_2 \xrightarrow{t_5} M_3 \xrightarrow{t_8} M_0 \xrightarrow{t_2} M_2 \xrightarrow{t_6} M_4$$

■ در دنباله فوق:

□  $M_2$  از  $M_1$  دسترس پذیر است.

□  $M_4$  هم از  $M_0$  دسترس پذیر است.

■ در حقیقت در مدل ماشین خودکار فروش خوراکی هر حالت یا نشانه گذاری از هر حالت یا نشانه گذاری دیگری دسترس پذیر است.

## مجموعه دسترس پذیری

■ **تعریف:** مجموعه دسترس پذیری (reachability set) یک سیستم شبکه پتری با نشانه گذاری اولیه  $M_0$  با  $RS(M_0)$  نشان داده شده و به عنوان کوچکترین مجموعه نشانه گذاری تعریف می شود، به نحوی که:

□  $M_0 \in RS(M_0)$

□  $M_1 \in RS(M_0) \wedge \exists t \in T: M_1[t] M_2 \Rightarrow M_2 \in RS(M_0)$

■ وقتی که امکان اشتباه نباشد، از  $RS$  برای مشخص کردن مجموعه  $RS(M_0)$  استفاده می کنیم.

■ همچنین از  $RS(M)$  برای مشخص نمودن مجموعه نشانه گذاریهای دسترس پذیر از یک حالت مشخصی مثل  $M$  استفاده می کنیم.

## کران دار بودن

- به یک شبکه پتری کران دار-K ( $K$ -bounded) یا به طور ساده کران دار ( $bounded$ ) گفته می شود اگر تعداد نشانه های موجود در هر مکان برای هر نشانه گذاری دسترس پذیر از  $M_0$  از تعداد متناهی  $K$  بیشتر نشود.
- برای مثال شبکه پتری ماشین خودکار فروش خوراکی کران دار-1 ( $1$ -bounded) است.

## ایمنی و زنده بودن

- طبق نظر ال. لامپورت (Leslie Lamport)، که از پیشگامان سیستم های همروند است، دو خصوصیت ایمنی ( $safety$ ) و زنده بودن ( $liveness$ )، نیازمندی های مهمی هستند که در طراحی سیستم های همروند باید اعمال شوند.
- در ادامه این دو خصوصیت را تعریف می کنیم.

## ایمنی

- طبق تعریف لامپورت، ایمنی (safety) بیان می‌کند که "هرگز اتفاق بدی نخواهد افتاد!"
  - این خصوصیت در مورد سیستمی برقرار است که یک حالت ممنوع شده در رفتار آن بروز نکند.
  - برای نمونه، در مورد یک ماشین خودکار فروش خوراکی رخداد بد و ممنوع شده آن است که پس از پرداخت پول کافی، خوراکی فراهم نشود. یعنی اگر این طور باشد ایمن نخواهد بود.
- اصطلاحاً به یک شبکه پتری که کران دار-۱ باشد، ایمن (safe) گفته می‌شود.
- یعنی یک راه تشخیص ایمن بودن یک شبکه پتری آن است که ببینیم کران دار-۱ است یا نه؟
  - اگر کران دار-۱ بود حتماً ایمن است، ولی اگر کران دار-۱ نبود نمی‌توان نتیجه گرفت که ایمن نیست. یعنی مدلهایی وجود دارند که علی‌رغم کران دار-۱ نبودن، ایمن هستند.

## زنده بودن

- طبق تعریف لامپورت، خصوصیت زنده بودن (liveness) بیان می‌کند که "سرانجام اتفاق خوبی خواهد افتاد!"
  - این خصوصیت در مورد سیستمی برقرار است که رفتارش مطابق کارکرد مورد نظر باشد.
- یک مثال از خصوصیت زنده بودن در مورد ماشین خودکار فروش خوراکی آن است که پس از پرداخت پول کافی، خوراکی مورد نظر فراهم شود.
- به بیان دیگر، همیشه مدتی پس از وارد نمودن پول کافی، ماشین خوراکی را فراهم می‌کند.

## زنده بودن

- یک شبکه پتری، **زنده (live)** است اگر هر گذری سرانجام بتواند شلیک کند.
  - به این نوع شبکه پتری، **شبکه پتری زنده (live Petri net)** گفته می‌شود.
  - از میان مثالهایی که در جلسه قبل زدیم، هم مدل ماشین خودکار فروش خوراکی و هم مدل تولیدکننده-مصرف‌کننده زنده بودند.
- یک گذر، **مرده (dead)** است اگر هرگز و در هیچ دنباله‌ای از شلیک کردن‌ها نتواند شلیک کند و در دنباله نشانه‌گذاریها به عنوان برجسب هیچ کمائی ظاهر نشود.

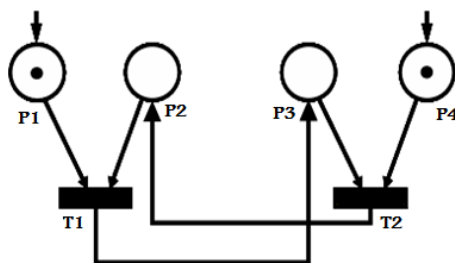
## فقدان بن بست

- یک سیستم شبکه پتری دارای یک **بن بست (deadlock)** است اگر بتواند به حالتی برسد که در آن هیچ گذری توانا نبوده و نتواند شلیک کند.
- **فقدان بن بست (deadlock-freeness)** برای یک سیستم شبکه پتری دارای نشانه‌گذاری اولیه  $M_0$  به طور صوری این گونه بیان می‌شود که:
$$\forall M \in RS(M_0): E(M) \neq \emptyset$$
که در آن  $E(M)$  مجموعه همه گذرهای توانا در نشانه‌گذاری  $M$  است.
- **فقدان بن بست** برای مدل‌های شبکه‌های پتری تعریف مستقلی نداشته و بر اساس سیستم‌های شبکه‌های پتری مربوطه به صورت زیر تعریف می‌شود:
  - یک مدل شبکه پتری فاقد بن بست است اگر همه سیستم‌های شبکه پتری مشتق از آن (یعنی با نشانه‌گذاری‌های اولیه متفاوت و پارامتری)، **فاقد بن بست (deadlock-free)** باشند.



## فقدان بن بست

- یک شبکه پتری زنده، عملیات فاقد بن بست را تضمین می کند، بدون توجه به اینکه چه دنباله ای از شلیک کردن های گذرها انتخاب می شود.
- با توجه به ساختار گرافی شبکه پتری امکان وجود بن بست بالقوه در آن را می توان تعیین نمود. برای مثال مدل زیر دارای بن بست است:



## برگشت پذیری

- به یک سیستم شبکه پتری برگشت پذیر (reversible) گفته می شود، اگر و فقط اگر از هر حالت دسترس پذیر از  $M_0$  امکان بازگشت به  $M_0$  وجود داشته باشد.
- به بیان صوری، یک سیستم شبکه پتری با نشانه گذاری اولیه  $M_0$  برگشت پذیر است اگر و فقط اگر داشته باشیم:

$$\forall M \in RS(M_0): M_0 \in RS(M)$$

- برگشت پذیری تصریح می کند که امکان بازگشت به نشانه گذاری اولیه بی نهایت بار وجود دارد.
- ممکن است حالتی که همیشه سیستم قادر به بازگشت به آن است، نشانه گذاری اولیه سیستم شبکه پتری نباشد. به چنین حالتی، حالت اقامت (home state) گفته می شود.
- به طور صوری می گوییم که  $M \in RS(M_0)$  یک حالت اقامت برای یک سیستم شبکه پتری است اگر و فقط اگر داشته باشیم:

$$\forall M' \in RS(M_0): M \in RS(M')$$

## ماندگاری

- یک سیستم شبکه پتری ماندگار (persistent) یا مانا است اگر، برای هر دو گذر توانا شده، شلیک کردن یکی از آن دو منجر به ناتوان شدن دیگری نشود.
- به عبارت دیگر، هر گاه یک گذر توانا شد، توانا بماند تا وقتی که شلیک کند.

## انصاف

- برخی اوقات ممکن است که فرضیاتی در مورد رعایت انصاف (fairness) در یک سیستم همروند جزء نیازمندیهای آن باشد.
- برای مثال ممکن است که بخواهیم یک زمانبند (scheduler) هرگز برخی از پردازنده‌های اجرایی را برای همیشه در تخصیص پردازنده از قلم نیاندازد.
- این بدان معنی است که چنین زمانبندی تضمین می‌کند که سرانجام پردازنده به همه پردازنده‌های اجرایی تخصیص داده خواهد شد و هیچ پردازنده‌ای الی الابد بدون پردازنده نخواهد ماند.
- این بدان معنی خواهد بود که چنین زمانبندی منصف (fair) است و انصاف را مابین پردازنده‌ها رعایت می‌کند.

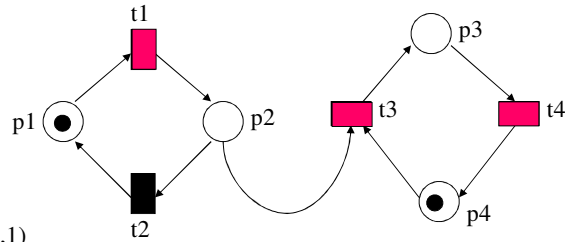
## انصاف

- انصاف در شبکه‌های پتری با توجه به تعداد دفعات شلیک کردن گذرها نسبت به همدیگر تعریف می‌شود و دو نوع است:
  - انصاف کران‌دار (bounded fairness): حداکثر بعد از تعدادی دفعات معین که یک گذر مثل  $T_1$  شلیک می‌کند، گذر دیگری مثل  $T_2$  هم شلیک نماید.
  - انصاف بدون شرط (unconditional fairness) یا انصاف عمومی (global fairness): هر گذری سرانجام شلیک کند.
- کران خاصی برای تعداد دفعات شلیک نمودن یک گذر نسبت به دیگری معین نمی‌شود.

## دسته‌بندی خصوصیت‌های رفتاری

- خصوصیت‌های رفتاری تعریف شده قابل دسته‌بندی به دو رده اصلی هستند:
  - خصوصیت‌هایی که باید برای همه حالتها در فضای حالت برقرار باشند که در ادبیات سیستم‌های همروند به آنها خصوصیت‌های ایمنی یا نامتغیر (invariant) گفته می‌شود.
  - خصوصیت‌هایی که باید برای برخی حالتها برقرار باشند که به آنها خصوصیت‌های امکان (eventuality) یا پیشرفت (progress) گفته می‌شوند.
- مثالهایی از خصوصیت‌های نامتغیر عبارتند از:
  - فقدان بن بست
  - کران‌دار بودن
  - مانع‌الجمع (یا دو به دو انحصاری)
- مثالهایی از خصوصیت‌های امکان عبارتند از:
  - دسترس‌پذیری: یک نشانه‌گذاری داده شده سرانجام ظاهر خواهد شد.
  - زنده‌بودن: یک گذر سرانجام توانا خواهد شد.

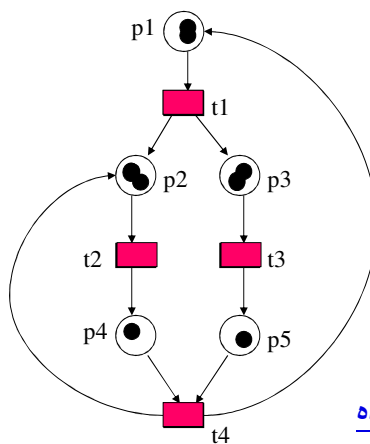
## یک مثال



$M_0 = (1, 0, 0, 1)$   
 $M_1 = (0, 1, 0, 1)$   
 $M_2 = (0, 0, 1, 0)$   
 $M_3 = (0, 0, 0, 1)$

یک شبکه پتری کران دار و غیرزنده

## مثال دیگر



$M_0 = (1, 0, 0, 0, 0)$   
 $M_1 = (0, 1, 1, 0, 0)$   
 $M_2 = (0, 0, 0, 1, 1)$   
 $M_3 = (1, 1, 0, 0, 0)$   
 $M_4 = (0, 2, 1, 0, 0)$

یک شبکه پتری بی کران اما زنده

## فضای حالت شبکه‌های پتری

- حالت (state) در شبکه‌های پتری با نشانه‌گذاری مشخص می‌شود.
- تحلیل شبکه‌های پتری بر اساس فضای حالت (state space) یا مجموعه نشانه‌گذاریهای دسترس‌پذیر مدل انجام می‌شود.
- شلیک یک گذر در شبکه پتری باعث تغییر نشانه‌گذاری آن شده و نشان دهنده یک تغییر حالت است.
- به مجموعه نشانه‌گذاریهای حاصل از اجرای یک شبکه پتری حالت‌های دسترس‌پذیر (reachable states) یا مجموعه دسترس‌پذیری (reachability set) یا فضای حالت (state space) گفته می‌شود.
- اغلب فضای حالت را با درخت دسترس‌پذیری (reachability tree) نمایش می‌دهیم که نشانه‌دهنده دنباله‌ای از حالتها در فضای حالت یا دنباله‌ای از شلیک گذرهای شبکه پتری است.
- برای بهینه‌سازی ذخیره‌سازی درخت و حذف گره‌های تکراری، درخت دسترس‌پذیری را تبدیل به گراف دسترس‌پذیری (reachability graph) می‌کنند.

## تحلیل شبکه‌های پتری

- روش اصلی تحلیل شبکه‌های پتری، تحلیل فضای حالت (state space analysis) است.
- تحلیل شبکه پتری ممکن است به دو منظور انجام می‌شود:
  - تحلیل جنبه‌های کارکردی (functional) یا منطقی، که به آن درستی‌یابی (verification) یا بررسی مدل (model checking) گفته می‌شود:
  - تحلیل جنبه‌های کارکردی عبارت است از بررسی درستی یا درستی‌یابی خصوصیت‌های رفتاری مدل، نظیر زنده‌بودن، انصاف و غیره.
  - برای این منظور ممکن است از منطق زمانی (temporal logic) برای بیان خصوصیت‌های مورد نظر استفاده شود.
- ارزیابی معیارهای عملیاتی (operational measures) یا خصوصیت‌های وابسته به زمان (time-dependent)، نظیر کارایی، اتکاء‌پذیری و انجام‌پذیری:
  - ارزیابی در مورد بسطهایی از شبکه‌های پتری که امکان مدل‌سازی جنبه‌های وابسته به زمان را می‌دهند امکان‌پذیر است.
  - این بسطها در جلسه بعد معرفی خواهند شد.

## درخت دسترس پذیری

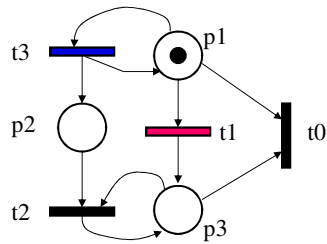
- درخت دسترس پذیری یک نمایش از همه نشانه گذاریهای امکان پذیر است به صورت یک درخت است، به نحوی که:
  - ریشه (root) درخت: نشانه گذاری اولیه ( $M_0$ ) است.
  - گره ها (nodes): نشانه گذاریهای دسترس پذیر از  $M_0$  هستند.
  - کمانها (arcs): شلیک کردن گذرها را مشخص می کنند. گذر شلیک کننده، به عنوان برچسب کمان مشخص می شود.

## گره های قابل تولید مجدد بی نهایت

- گره های قابل تولید مجدد بی نهایت (infinitely reproducible nodes) گره های خاصی در درخت دسترس پذیری هستند که در صورتی که شبکه بی کران باشد، اندازه درخت دسترس پذیری با معرفی نماد  $\infty$ ، محدود و متناهی نگهداشته شده و مابقی حالت های امکان پذیر نمایش داده نمی شوند.
- با توجه به نماد  $\infty$ ، خصوصیت های زیر از درخت دسترس پذیری قابل نتیجه گیری است:
  - یک شبکه پتری کراندار است اگر و فقط اگر  $\infty$  در هیچ گره درخت دسترس پذیری ظاهر نشده باشد.
  - یک شبکه پتری ایمن است اگر و فقط اگر تنها 0 و 1 در گره های درخت دسترس پذیری ظاهر شده باشد.
  - یک گذر مرده است اگر و فقط اگر در هیچ کمانی ظاهر نشده باشد.
- همچنین و در حالت کلی، اگر حالت  $M$  از  $M_0$  دسترس پذیر باشد، یک گره  $M$  باید در درخت دسترس پذیری وجود داشته باشد.
- علاوه بر خصوصیت های فوق الذکر، سایر خصوصیت ها رفتاری نظیر برگشت پذیری، فقدان بن بست و غیره را نیز می توان با تحلیل فضای حالت درستی یابی نمود.

## مثالی از تولید درخت دسترس پذیری

### مدل شبکه پتری



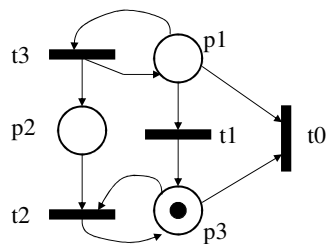
### درخت دسترس پذیری

$M_0=(100)$

■ در این حالت، گذرهای t1 و t3 توانا هستند، فرض می‌کنیم که t1 شلیک می‌کند...

## مثالی از تولید درخت دسترس پذیری

### مدل شبکه پتری



### درخت دسترس پذیری

$M_0=(100)$

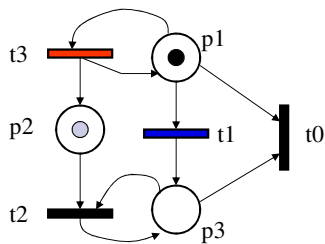
t1

$M_1=(001)$   
"dead end"

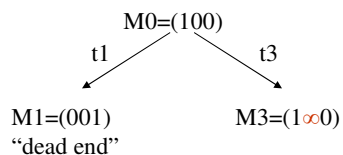
■ در این حالت، هیچ گذری توانا نیست.

## مثالی از تولید درخت دسترس پذیری

مدل شبکه پتری



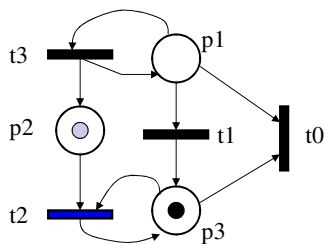
درخت دسترس پذیری



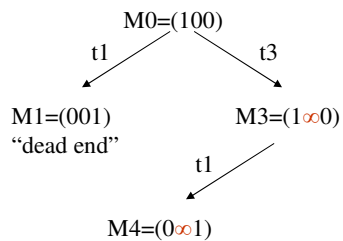
■ از طرفی در همان حالت  $M_0$ ، گذر  $t_3$  هم توانا بود. حال فرض می کنیم که  $t_3$  شلیک کند. مشخص است که در این شرایط  $t_3$  می تواند برای هر تعداد دفعه ( $\infty$ ) شلیک کند. در این صورت  $p_2$  بی کران است که در گره  $M_3$  نشان داده شده است. در این حالت اگر  $t_1$  شلیک کند...

## مثالی از تولید درخت دسترس پذیری

مدل شبکه پتری



درخت دسترس پذیری

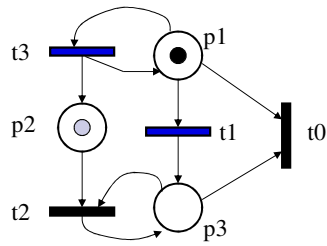


■ در این حالت  $t_2$  توانا است و می تواند برای هر تعداد دفعه ( $\infty$ ) شلیک کند. در این صورت چون  $p_2$  بی کران است، به حالت  $M_4$  می رسیم.

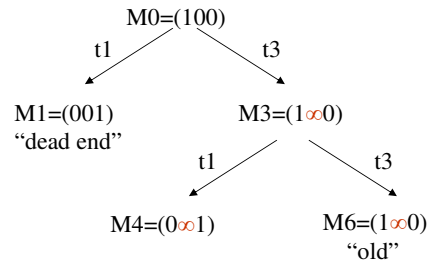


## مثالی از تولید درخت دسترس پذیری

مدل شبکه پتری



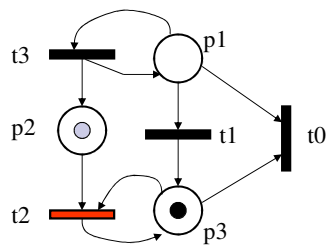
درخت دسترس پذیری



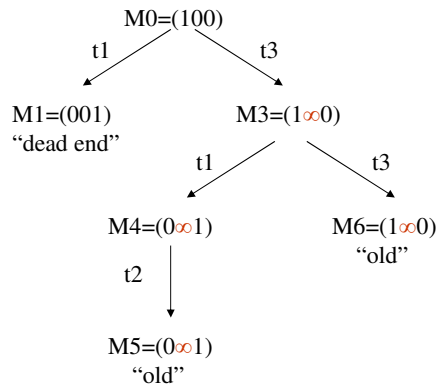
■ اما اگر حالت M3 دوباره t3 شلیک کند، حالت M6 ایجاد می شود که جدید نبوده و همان حالت قدیمی M3 است.

## مثالی از تولید درخت دسترس پذیری

مدل شبکه پتری



درخت دسترس پذیری

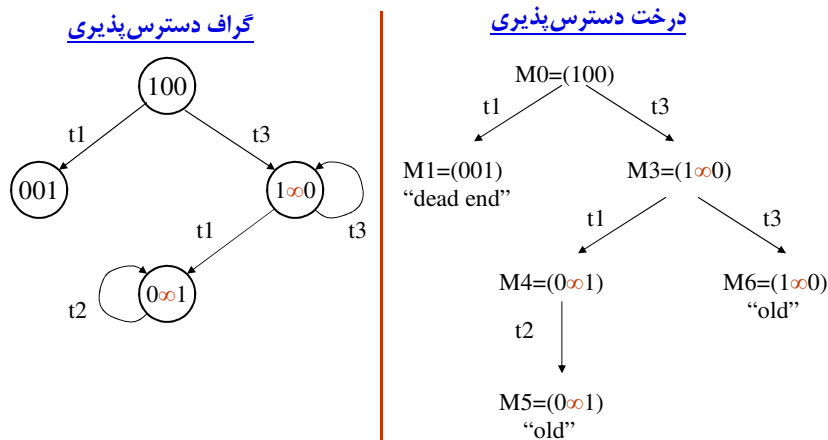


■ در حالت M4 گذر t2 توانا است که اگر شلیک کند، حالت M5 ایجاد می شود که جدید نبوده و همان حالت قدیمی M4 است.

## مثالی از تولید درخت دسترس پذیری

- در این صورت همه گره‌های درخت دسترس‌پذیری، گره‌های مرزی (frontier nodes) بوده و تولید درخت به پایان رسیده است.
- گامهای ذکر شده برای تولید درخت دسترس‌پذیری را می‌توان به صورت یک الگوریتم تولید درخت دسترس‌پذیری (reachability tree generation) که به آن الگوریتم تولید فضای حالت (state space generation) هم گفته می‌شود، ارائه نمود.
- کارهای تحقیقاتی فراوانی در این خصوص انجام شده است تا الگوریتم‌هایی ایجاد شود که هم از نظر زمان و هم حافظه مورد نیاز، کارا باشند. این مساله با توجه به بزرگ شدن سریع فضای حالت در اغلب مدلها امر مهمی است.
- روشهایی نظیر نمودارهای تصمیم دودویی (BDDs: Binary Decision Diagrams) در همین چارچوب مطرح شده‌اند که هدف آنها فشردگی فضای حالت است.

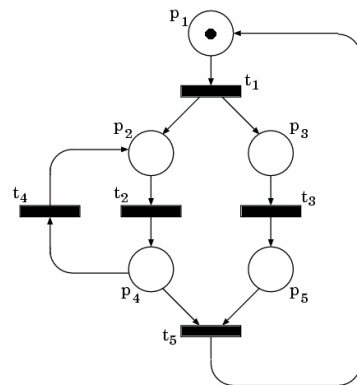
## تبدیل درخت دسترس‌پذیری به گراف دسترس‌پذیری



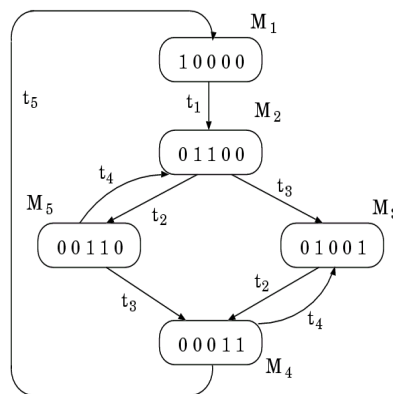
## تولید گراف دسترس پذیری

- به وسیله شناسایی صحیح گره‌های مرزی، تولید گراف دسترس پذیری شامل گام‌های محدودی خواهد بود، حتی اگر شبکه پتری بی کران باشد.
- سه نوع گره مرزی وجود دارد:
  - گره‌های پایانی یا مرده (terminal or dead nodes): در نشانه‌گذاریها مرتبط به آنها، هیچ گذری توانا نیست.
  - گره‌های تکراری (duplicate nodes): این گره‌ها قبلاً تولید شده‌اند.
  - گره‌های قابل تولید مجدد بی‌نهایت.

## مثالی از تولید گراف دسترس پذیری

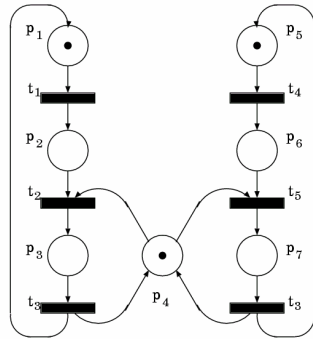


یک مدل شبکه پتری نشانه‌گذاری شده

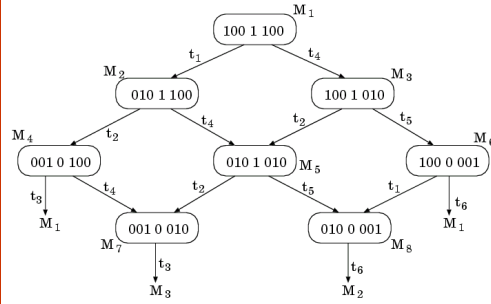


گراف دسترس پذیری متناظر با مدل مقابل

## مثال دیگری از تولید گراف دسترس پذیری



یک مدل شبکه پتری نشانه گذاری شده



گراف دسترس پذیری متناظر با مدل مقابل

## شبیه سازی شبکه های پتری

- شبیه سازی گسسته-رخداد در حالت کلی و برای تحلیل شبکه های پتری اعم از کران دار یا بی کران قابل استفاده است.
- شبیه سازی گسسته-رخداد شبکه پتری عبارت است از اجرای گام به گام مدل بر اساس قواعد اجرای شبکه های پتری:
  - (1) مجموعه گذرهای توانا در هر نشانه گذاری تعیین می شود.
  - (2) یکی از گذرهای توانا به صورت غیرقطعی انتخاب شده و شلیک می کند.
  - (3) نشانه گذاری جدید ایجاد می شود.
  - (4) در نشانه گذاری جدید برای خصوصیت های مورد نظر بررسی انجام می شود.
  - (5) در صورت عدم برقراری قواعد خاتمه شبیه سازی به مرحله (1) برگشت انجام می شود.