دانشکده مهندسی کامپیوتر

عنوان درس:

ارزیابی کارایی سیستم‌های کامپیوتری

Performance Evaluation of Computer Systems (PECS)

<u>جلسه ۱۱: ارزیابی کارآیی با زنجیره‌های مارکوف پیوسته-زمان</u>

مدرس:

محمد عبداللهی ازگمی

(Mohammad Abdollahi Azgomi)

azgomi@iust.ac.ir

فهرست مطالب

- چند مثال از زنجیره‌های مارکوف پیوسته-زمان:
  - مثال ۱: مدل‌سازی یک مرکز تلفن
  - مثال ۲: مدل‌سازی یک سیستم یک‌پردازنده‌ای
  - مثال ۳: مدل‌سازی یک سیستم چندپردازنده‌ای
- استفاده از نرم‌افزارهای جبر خطی برای حل CTMCs
- روشهای محاسبه انواع معیارهای کارآیی:
  - محاسبه معیارهای مبتنی بر حالت
  - محاسبه معیارهای مبتنی بر نرخ
  - محاسبه سایر انواع معیارها

مثال ۱: مدل‌سازی یک مرکز تلفن

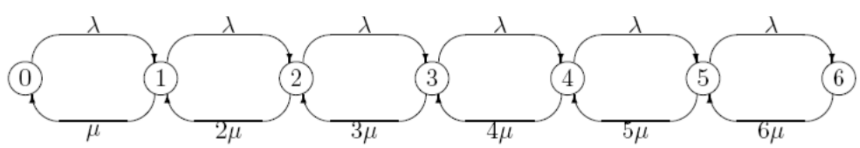
- خطوط تلفنی متصل به یک مرکز تلفن (phone center) را در نظر بگیرید. ساده‌ترین مدل برای چنین سیستمی آن است که تنها حالت مشغول و آزاد بودن خطوط را در نظر بگیریم. بنا بر این اگر این مرکز تلفن دارای ۶ خط باشد، فضای حالت عبارت خواهد بود از:

$$\{0, 1, 2, 3, 4, 5, 6\}$$

- در فضای حالت فوق گذر از حالت صفر تنها به حالت ۱ با یک اتصال جدید امکان‌پذیر است. همچنین گذر به حالت ۶ تنها از حالت ۵ و با آزاد کردن یک خط (گذاشتن گوشی) امکان‌پذیر است. اما برای هر حالت j دیگر، که $1 \le j \le 5$ باشد، دو گذر به حالتهای j+1 و j-1 به ترتیب با برقرار یک اتصال جدید یا آزاد کردن یک خط مشغول امکان‌پذیر است.
- فرض کنید که درخواستهای برقراری اتصال (برداشتن گوشی) طبق یک فرآیند پواسان با پارامتر λ وارد می‌شوند. یعنی گذرهای $j \rightarrow j+1$ دارای نرخ λ هستند. همچنین فرض کنید که میانگین طول مکالمه برابر است با $1/\mu$ (یعنی نرخ یعنی آزاد شدن خط طبق توزیع نمایی با نرخ μ است).
- **مدل زنجیره مارکوف و ماتریس مولد را برای این سیستم بدست آورید.**

مثال ۱: مدل‌سازی یک مرکز تلفن

- **نمودار گذر-حالت این سیستم به صورت زیر خواهد بود:**



- **ماتریس مولد هم به صورت زیر خواهد بود:**

$$Q = \begin{pmatrix} -\lambda & \lambda & 0 & 0 & 0 & 0 & 0 \\ \mu & -(\mu+\lambda) & \lambda & 0 & 0 & 0 & 0 \\ 0 & 2\mu & -(2\mu+\lambda) & \lambda & 0 & 0 & 0 \\ 0 & 0 & 3\mu & -(3\mu+\lambda) & \lambda & 0 & 0 \\ 0 & 0 & 0 & 4\mu & -(4\mu+\lambda) & \lambda & 0 \\ 0 & 0 & 0 & 0 & 5\mu & -(5\mu+\lambda) & \lambda \\ 0 & 0 & 0 & 0 & 0 & 6\mu & -6\mu \end{pmatrix}$$

مثال ۲: مدل‌سازی یک سیستم چندپردازنده‌ای

- یک سیستم چندپردازنده‌ای را که هر پردازنده دارای حافظه اختصاصی (private memory) و یک حافظه مشترک (common memory) است را در نظر بگیرید. حافظه مشترک در هر زمان تنها توسط یک پردازنده قابل دسترسی است.
- هر پردازنده برای یک مدت زمان تصادفی قبل از صدور درخواست دسترسی به حافظه مشترک با حافظه اختصاصی خودش کار می‌کند. فرض کنید که این زمان تصادفی دارای توزیع نمایی با پارامتر λ است. یعنی میانگین زمانی که یک پردازنده با حافظه اختصاصی خود کار می‌کند، قبل از آنکه درخواست دسترسی به حافظه مشترک را صادر کند، $1/\lambda$ است.
- همچنین فرض می‌شود که میانگین زمان دسترسی به حافظه مشترک طبق توزیع نمایی با پارامتر μ است. یعنی میانگین زمان دسترسی به حافظه مشترک، $1/\mu$ است.

مثال ۲: مدل‌سازی یک سیستم تک‌پردازنده‌ای

- **اگر سیستم تنها دارای یک پردازنده باشد، تنها دو حالت وجود خواهد داشت:**
  - **حالت ۱: پردازنده در حافظه اختصاصی‌اش در حال اجرا است.**
  - **حالت ۲: پردازنده در حافظه مشترک در حال اجرا است.**
- **زنجیره مارکوف و ماتریس مولد را برای این سیستم بدست آورده و حل نمایید.**

مثال ۲: زنجیره مارکوف سیستم تک‌پردازنده‌ای

- **نمودار حالت-گذر و ماتریس مولد به صورت زیر خواهد بود:**

$$Q = \begin{pmatrix} -\lambda & \lambda \\ \mu & -\mu \end{pmatrix}$$

مثال ۲: حل زنجیره مارکوف سیستم تک‌پردازنده‌ای

- برای حل حالت پایدار خواهیم داشت:
  - جریان ورودی به حالت ۱ برابر را با جریان خروجی از آن برابر قرار می‌دهیم: $\pi_1 \lambda = \pi_2 \mu$
  - همین طور برای حالت ۲ هم خواهیم داشت: $\pi_2 \mu = \pi_1 \lambda$
  - به علاوه شرط $\pi_1 + \pi_2 = 1$ را هم داریم.
- در نتیجه با حل دستگاه سه معادله و دو مجهول فوق، بردار احتمال حالت پایدار به صورت زیر بدست می‌آید:

$$\pi = \left(\frac{\mu}{\mu+\lambda}, \frac{\lambda}{\mu+\lambda}\right).$$

- بنا بر این می‌توانیم نتیجه‌گیری کنیم که احتمال اینکه پردازنده در حال اجرا در حافظه اختصاصی‌اش باشد برابر است با: $\mu/(\mu + \lambda)$

مثال ۳: مدل‌سازی یک سیستم چندپردازنده‌ای

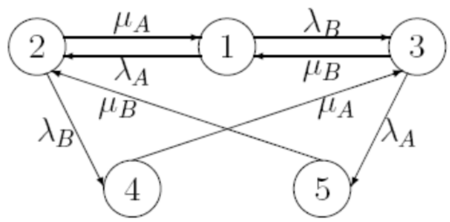
- حال سیستمی را که دارای دو پردازنده است در نظر می‌گیریم. یکی از پردازنده‌ها را با A و دیگری را با B مشخص می‌کنیم.
- فرض می‌کنیم که پردازنده‌ها دارای مشخصات زمانی متفاوتی هستند:
  - زمان دسترسی A به حافظه اختصاصی طبق توزیع نمایی با نرخ λ_A است.
  - زمان دسترسی A به حافظه مشترک طبق توزیع نمایی با نرخ μ_A است.
  - زمان دسترسی B به حافظه اختصاصی طبق توزیع نمایی با نرخ λ_B است.
  - زمان دسترسی B به حافظه مشترک طبق توزیع نمایی با نرخ μ_B است.

مثال ۳: مدل‌سازی یک سیستم چندپردازنده‌ای

- فضای حالت مدل عبارت خواهد بود از:
  - **حالت ۱:** هم A و هم B در حال اجرا در حافظه اختصاصی هستند.
  - **حالت ۲:** A در حال دسترسی به حافظه مشترک و B در حال اجرا در حافظه اختصاصی است.
  - **حالت ۳:** B در حال دسترسی به حافظه مشترک و A در حال اجرا در حافظه اختصاصی است.
  - **حالت ۴:** A در حال دسترسی به حافظه مشترک و B منتظر دسترسی به حافظه مشترک است.
  - **حالت ۵:** B در حال دسترسی به حافظه مشترک و A منتظر دسترسی به حافظه مشترک است.
- **زنجیره مارکوف و ماتریس مولد برای این سیستم را بدست آورده و آنرا حل کنید.**

مثال ۳: مدل‌سازی یک سیستم چندپردازنده‌ای

- نمودار حالت-گذر زنجیره مارکوف و ماتریس مولد این سیستم عبارتند از:



$$Q = \begin{pmatrix} -(\lambda_A+\lambda_B) & \lambda_A & \lambda_B & 0 & 0 \\ \mu_A & -(\mu_A+\lambda_B) & 0 & \lambda_B & 0 \\ \mu_B & 0 & -(\mu_B+\lambda_A) & 0 & \lambda_A \\ 0 & 0 & \mu_A & -\mu_A & 0 \\ 0 & \mu_B & 0 & 0 & -\mu_B \end{pmatrix}$$

استفاده از نرم‌افزارهای جبر خطی برای حل CTMCs

- در صورتی که دستگاه معادلات تنها دارای تعداد معدودی معادله باشد، حل دستی به راحتی امکان‌پذیر است. اما در صورت بیشتر شدن تعداد معادلات مجبور به استفاده از نرم‌افزاهای جبر خطی (نظیر Maple) هستیم.
- برای این منظور می‌توانیم معادلات جریان را به شکل ماتریسی قابل قبول توسط این نرم‌افزارها تبدیل کنیم.
- چنین نرم‌افزارهایی معادلات ماتریسی به شکل $\mathbf{A\ x = b}$ را حل می‌کنند که در آن A یک ماتریس $n \times n$ بوده، x یک بردار ستونی از n مجهول و b یک بردار ستونی از n مقدار است.

استفاده از نرم‌افزارهای جبر خطی برای حل CTMCs

- برای آنکه بتوانیم از چنین نرم‌افزارهایی استفاده کنیم ابتدا باید دو کار را انجام دهیم:

1. معادلات جریان به شکل $\boldsymbol{\pi}\,\mathbf{Q} = 0$ هستند که در آن $\boldsymbol{\pi}$ بردار مجهولها بوده و یک بردار ردیفی است نه ستونی و سمت راست هم یک بردار ردیفی صفر است. بنا بر این اولین کار، بدست آوردن ماتریس ترانهاده (transpose) است. در نتیجه خواهیم داشت: $\mathbf{Q}^{\mathrm{T}}\boldsymbol{\pi} = 0$، که در آن هم $\boldsymbol{\pi}$ و هم بردار صفر سمت راست(که با e نشان می‌دهیم)، بردارهای ستونی هستند.
   - منظور از ماتریس ترانهاده، ماتریسی است که جای سطر و ستون آن با ماتریس اصلی تعویض شده باشد.
2. دومین مساله، حذف یکی از معادلات افزونه معادلات جریان و وارد نموده $\Sigma\pi_i=1$ به جای آن است. برای این منظور یکی از سطرهای ماتریس $\mathbf{Q}^{\mathrm{T}}$ (عموماً سطر آخر) را یک نموده و $\mathbf{Q}^{T}{}_{n}$ را بدست می‌آوریم. همچنین، درآیه سطر متناظر در بردار e را هم یک نموده و $\mathbf{e}_n$ را بدست می‌آوریم.

استفاده از نرم‌افزارهای جبر خطی برای حل CTMCs

- حالا می‌توانیم از نرم‌افزاری مثل Maple برای حل دستگاه معادلات زیر استفاده کنیم:

$$\mathbf{Q}^{T}{}_{n}\boldsymbol{\pi} = \mathbf{e}_n$$

مثال ۳: حل مدل چندپردازنده‌ای با استفاده از Maple

- برای حل حالت پایدار هنوز می‌توانیم از حل دستی استفاده کنیم. منتهی برای تشریح نحوه استفاده از Maple، همین مثال را با استفاده از آن حل می‌کنیم.
- برای این منظور باید فرمان‌های صفحه بعد را که به‌صورت یک برنامه به زبان Maple است در محیط آن وارد نمود...

مثال ۳: حل مدل چندپردازنده‌ای با استفاده از Maple

- برنامه نوشته شده به زبان Maple:

Transcript of maple session

```
> with(linalg):
>
> lambdaA := 0.05;
> muA := 0.02;
> lambdaB := 0.1;
> muB := 0.05;
>
> size := 5;
>
> Q := array(sparse,1..size,1..size);
> e := array(sparse,1..size);
>
> Q[1,2] := lambdaA;
>
> Q[1,3] := lambdaB;
>
> Q[2,1] := muA;
>
> Q[2,4] := lambdaB;
>
> Q[3,1] := muB;
```

ادامه

```
>
> Q[3,5] := lambdaA;
>
> Q[4,3] := muA;
>
> Q[5,2] := muB;
>
> for i to size do
>     s := 0:
>     for j to size do
>         s := s + Q[i,j]
>     od;
>     Q[i,i] := -s
> od;
>
> Qt := transpose(Q);
>
> for i to size do Qt[size,i] := 1 od;
>
> e[size] := 1;
>
> p := linsolve(Qt,e);
```

مثال ۳: حل مدل چندپردازنده‌ای با استفاده از Maple

- پس از اعمال تغییرات مورد نیاز، ماتریس مولد به‌صورت زیر در خواهد آمد:

$$Q_n^T = \begin{pmatrix} -(\lambda_A+\lambda_B) & \mu_A & \mu_B & 0 & 0 \\ \lambda_A & -(\mu_A+\lambda_B) & 0 & 0 & \mu_B \\ \lambda_B & 0 & -(\mu_B+\lambda_A) & \mu_A & 0 \\ 0 & \lambda_B & 0 & -\mu_A & 0 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

- همان طور که در متن برنامه هم مشخص بود، پارامترها به‌صورت زیر انتخاب شده‌اند:

$\lambda_A = 0.05$ $\lambda_B = 0.1$ $\mu_A = 0.02$ $\mu_B = 0.05$

- پس از حل، Maple بردار جواب زیر را تولید خواهد نمود:

$$\boldsymbol{\pi} = (0.0693, 0.0990, 0.1683, 0.4951, 0.1683)$$

روشهای محاسبه انواع معیارهای کارآیی

- **در حالت کلی سه روش برای محاسبه معیارهای کارایی از احتمالات حالت پایدار زنجیره‌های مارکوف وجود دارد. این روشها متناظر با انواع معیارهای زیر هستند:**
  - **معیارهای مبتنی بر حالت** (state-based measures)،
    - نظیر بهره‌وری (utilization)
  - **معیارهای مبتنی بر نرخ** (rate-based measures)،
    - نظیر توان عملیاتی (throughput)
  - **سایر انواع معیارها**، که جزء هیچ‌کدام از دو دسته فوق نیستند،
    - نظیر زمان پاسخ (response time).

محاسبه معیارهای مبتنی بر حالت

- **معیارهای مبتنی بر حالت** معیارهایی هستند که به‌طور واضح متناظر با آن هستند که مدل در یک حالت خاص یا زیر مجموعه‌ای از حالتها که دارای شرایط خاصی باشند قرار داشته باشد.
- برای مثال، بهره‌وری متناظر با آن است که یک منبع (مثلاً سرویس‌دهنده) در حال استفاده یا مشغول باشد.
- در مورد مثال سیستم چندپردازنده‌ای بهره‌وری حافظه مشترک (ρ_{mem}) مجموع احتمالات متناظر با حالت‌هایی است که در آنها حافظه مشترک در حال استفاده است. در نتیجه خواهیم داشت:

$$\rho_{mem} = \pi_2 + \pi_3 + \pi_4 + \pi_5 = 93.07\%$$

محاسبه معیارهای مبتنی بر حالت

- مثال دیگر از این معیارهای مبتنی بر حالت، زمان بیکاری (idle time) یا تعداد کارها در سیستم (**the number of jobs**) است.
- برخی معیارها نظیر تعداد کارها در سیستم، شامل یک جمع وزن‌دار (**weighted sum**) احتمالات حالت پایدار دارای یک وزن مناسب هستند.
- برای مثال، اگر کارهای منتظر برای دسترسی به حافظه مشترک را در نظر بگیریم، میانگین تعداد کارهای موجود در حافظه مشترک، L_{mem}، عبارت خواهد بود از:

$$L_{mem} = 1 \times (\pi_2 + \pi_3) + 2 \times (\pi_4 + \pi_5) = 1.594$$

محاسبه معیارهای مبتنی بر نرخ

- **معیارهای مبتنی بر نرخ** معیارهایی هستند که متناظر با نرخ پیش‌بینی شده رخدادهای مورد نظر هستند.
  - در نتیجه این معیارها حاصل ضرب در نرخ رخداد و احتمال اینکه آن رخداد توانا شده باشد خواهند بود.
  - یعنی احتمال بودن در یکی از حالتهایی که از آنها آن رخداد می‌تواند رخ دهد.
- بنا بر این، برای محاسبه توان عملیاتی حافظه مشترک، میانگین تعداد دسترسی‌ها از هر پردازنده که در واحد زمان پردازش می‌کند، یعنی X_{mem}، خواهیم داشت:

$$X_{mem} = (\mu_A \times (\pi_2 + \pi_4)) + (\mu_B \times (\pi_3 + \pi_5)) = 0.0287$$

  - یا به طور تقریبی، یک دسترسی در هر ۳۵ میلی‌ثانیه انجام خواهد شد:
    - $1/0.0287 \approx 35\ ms$

محاسبه سایر انواع معیارها

- سرانجام آنکه برخی دیگر از معیارها هم وجود دارند که در هیچکدام از دو دسته قبلی قرار نمی‌گیرند.
- در چنین مواردی می‌توانیم از یکی از **قوانین عملیاتی** (operational laws) استفاده کنیم و با استفاده از اطلاعات بدست آمده از مدل، این معیارها را محاسبه کنیم.
- برای مثال با استفاده از **قانون لیتل** (Little's law) که یکی از قوانین عملیاتی است می‌توانیم میانگین زمان صرف‌شده توسط کارها در حافظه مشترک (W_{mem}) را محاسبه کنیم:

$$W_{mem} = L_{mem}/X_{mem} = 1.594/0.0287 = 55.54 \text{ milliseconds}$$

- **این قوانین عملیاتی (نظیر قانون لیتل) در جلسه بعد معرفی خواهند شد.**