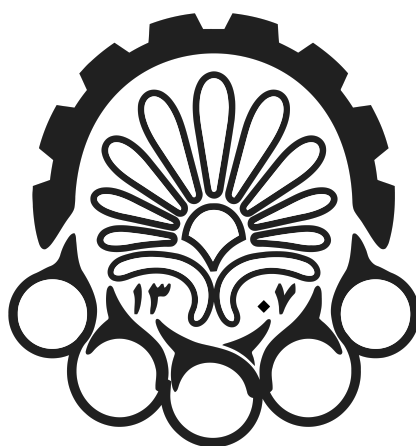


معماری افزاره‌های شبکه دکتر صبا ئی



دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)
دانشکده مهندسی کامپیوتر

رضا آدینه پور ۴۰۲۱۳۱۰۵۵

تمرین سری چهارم

۱۹ آذر ۱۴۰۳

سوال اول

فرض کنید یک شبکه دارای دو نوع ترافیک با اولویت بالا و پایین است. نرخ ورود بسته‌های اولویت بالا ۶ بسته در ثانیه و نرخ ورود بسته‌های اولویت پایین ۴ بسته در ثانیه است. حداکثر ظرفیت صف برابر با ۱۰ بسته بوده و از روش FIFO برای مدیریت صف‌ها استفاده می‌شود.

۱. اگر صف به حداکثر ظرفیت خود برسد تعداد بسته‌های هر دسته که در صف باقی می‌مانند را محاسبه کنید. همچنین فرض کنید ورود بسته‌ها به مدت ۵ ثانیه ادامه داشته باشد. نشان دهید که در این سناریو استفاده از FIFO ممکن است باعث افزایش زمان انتظار برای بسته‌های اولویت بالا شود.

پاسخ

• ورود بسته‌ها به شبکه:

- نرخ ورود بسته‌های High Priority: ۶ بسته در ثانیه
 - نرخ ورود بسته‌های Low Priority: ۴ بسته در ثانیه
 - کل نرخ ورود: $6 + 4 = 10$ بسته در ثانیه
- با توجه به نرخ ورود برابر با ظرفیت صف، صف به حداکثر ظرفیت خود می‌رسد.

• وضعیت صف در پایان ۵ ثانیه:

- تعداد بسته‌های High Priority: $6 \times 5 = 30$ بسته
 - تعداد بسته‌های Low Priority: $4 \times 5 = 20$ بسته
 - مجموع: $30 + 20 = 50$ بسته
- با توجه به ظرفیت صف که برابر ۱۰ بسته است، بسته‌های زیادی کنار گذاشته می‌شوند.

• اثر روش FIFO:

- بسته‌های High Priority که پس از پر شدن صف می‌رسند، حذف می‌شوند.
- بسته‌های Low Priority که زودتر وارد شده‌اند، ممکن است پردازش شوند.
- این امر باعث افزایش زمان انتظار و احتمال حذف بسته‌های High Priority می‌شود.

۲. روش HOL Priority Queueing را به عنوان جایگزین پیشنهاد دهید و تحلیل کنید که چگونه استفاده از این روش می‌تواند زمان انتظار برای بسته‌های اولویت بالا را کاهش دهد و تأثیر آن بر بسته‌های اولویت پایین را ارزیابی کنید.

پاسخ

در روش HOL Priority Queueing:

- بسته‌های High Priority همیشه اولویت پردازش دارند.
 - بسته‌های Low Priority تنها زمانی پردازش می‌شوند که صف High Priority خالی باشد.
- مزایای بسته‌های High Priority:

- کاهش زمان انتظار: بسته‌های High Priority به محض ورود پردازش می‌شوند.
- کاهش احتمال حذف: این بسته‌ها کمتر احتمال دارد که به دلیل پر شدن صف حذف شوند.

تأثیر بر بسته‌های Low Priority:

- افزایش زمان انتظار: بسته‌های Low Priority تنها زمانی پردازش می‌شوند که هیچ بسته High Priority در صف نباشد.
- احتمال حذف: به دلیل حجم بالای ترافیک High Priority، احتمال حذف بسته‌های Low Priority افزایش می‌یابد.

می‌توان جدولی به صورت زیر از مقایسه این دو روش ارائه نمود:

HOL Priority Queueing	FIFO	ویژگی
به حداقل می‌رسد	ممکن است افزایش یابد	زمان انتظار بسته‌های High Priority
افزایش می‌یابد	معمولاً کمتر است	زمان انتظار بسته‌های Low Priority
بیشتر بسته‌های Low Priority حذف می‌شوند	هر دو نوع بسته ممکن است حذف شوند	احتمال حذف بسته‌ها
پیچیده‌تر به دلیل مدیریت اولویت	ساده‌تر	پیچیدگی اجرایی

پاسخ

بنابر این:

- اگر هدف کاهش تأخیر برای بسته‌های High Priority باشد، روش HOL Priority Queueing پیشنهاد می‌شود.
- اگر هدف پردازش عادلانه باشد، روش FIFO مناسب‌تر است.

سوال دوم

فرض کنید که در یک سیستم صف عادلانه وزن دار (Weighted Fair-Queuing system)، یک بسته با برچسب اتمام F (finish tag) در زمان t وارد خدمت می‌شود. آیا ممکن است بسته‌ای بعد از زمان t به سیستم برسد و برچسب اتمام آن کمتر از F باشد؟ اگر بله، مثالی بزنید و اگر خیر، توضیح دهید.

پاسخ

بله، ممکن است.

در WFQ، مقدار (F) به عوامل زیر بستگی دارد:

- وزن جریان (W) : جریان‌هایی که وزن بیشتری دارند، ممکن است برچسب اتمام کمتری دریافت کنند.
- اندازه بسته (L) : بسته‌های کوچکتر معمولاً زودتر تمام می‌شوند و برچسب اتمام کمتری دارند.
- زمان ورود (t) : بسته‌ای که بعد از زمان t وارد شود، ممکن است به دلیل وزن بیشتر یا اندازه کوچکتر، برچسب اتمام کمتری دریافت کند.

برای مثال فرض شود دو جریان به صورت زیر داریم:

• جریان A با وزن $W_A = 1$

• جریان B با وزن $W_B = 2$

سناریو:

۱. در زمان $t = 10$ ، بسته P_1 از جریان A وارد سیستم می‌شود:

• اندازه بسته $L_1 = 100$

• برچسب شروع (S_1) برابر با t است: $S_1 = 10$

• برچسب اتمام (F_1) : $F_1 = S_1 + \frac{L_1}{W_A} = 10 + \frac{100}{1} = 110$

۲. در زمان $t = 15$ ، بسته P_2 از جریان B وارد سیستم می‌شود:

• اندازه بسته $L_2 = 50$

• برچسب شروع (S_2) برابر با حداکثر زمان جاری (15) یا برچسب اتمام جریان قبلی است: $S_2 = 15$

• برچسب اتمام (F_2) : $F_2 = S_2 + \frac{L_2}{W_B} = 15 + \frac{50}{2} = 40$

در نتیجه، بسته P_2 که در زمان $t = 15$ وارد سیستم شده است، برچسب اتمام $F_2 = 40$ دارد که کمتر از $F_1 = 110$ است، با اینکه P_2 بعد از P_1 وارد سیستم شده است.

سوال سوم

با توجه به مدیریت صف با استفاده از تشخیص زودهنگام تصادفی (RED - Random Early Detection):

۱. توضیح دهید که چرا RED به جلوگیری از شناسایی ترافیک TCP از طریق فرستنده‌ها و کاهش همزمان نرخ انتقال آن‌ها کمک می‌کند.

پاسخ

الگوریتم RED به گونه‌ای طراحی شده است که با اعمال سیاست افت تصادفی بسته‌ها، از شناسایی الگوهای خاص توسط فرستنده‌های TCP جلوگیری می‌کند:

- در RED، بسته‌ها به صورت تصادفی و بر اساس طول متوسط صف (average queue length) حذف می‌شوند.
- این حذف تصادفی به فرستنده‌های TCP امکان نمی‌دهد که الگوهای ثابتی از افت بسته‌ها را شناسایی کرده و نرخ انتقال خود را همزمان کاهش دهند.
- این امر باعث می‌شود که جریان‌های TCP به تدریج و نه ناگهانی، نرخ انتقال خود را تنظیم کنند و از کاهش شدید Throughput شبکه جلوگیری شود.

۲. تأثیر RED بر روی توان شبکه (Throughput) را بررسی کنید.

پاسخ

- افزایش پایداری: RED با مدیریت فعالانه صف، از پر شدن کامل بافر و افت دسته‌جمعی بسته‌ها (Global Synchronization) جلوگیری می‌کند.
- بهبود Throughput: افت تصادفی بسته‌ها، منابع شبکه را به صورت عادلانه بین جریان‌های مختلف تقسیم می‌کند و Throughput کلی شبکه را افزایش می‌دهد.
- تأخیر کمتر: کاهش احتمال پر شدن بافر باعث کاهش تأخیر در شبکه می‌شود، که به بهبود عملکرد جریان‌های حساس به تأخیر کمک می‌کند.

۳. پیچیدگی پیاده‌سازی الگوریتم RED را بررسی کنید.

پاسخ

پیاده‌سازی RED دارای پیچیدگی نسبی است:

- نیاز به محاسبه مداوم طول متوسط صف (average queue length) با استفاده از فیلتر نمایی (Exponential Weighted Moving Average - EWMA).
 - تنظیم مقادیر پارامترهای کلیدی نظیر Max_{th} ، Min_{th} و احتمال افت بسته.
 - نظارت پیوسته بر وضعیت صف و تصمیم‌گیری در مورد افت تصادفی بسته‌ها.
- این پیچیدگی در مقایسه با روش‌های ساده مانند Tail Drop بیشتر است، اما مزایای عملکردی آن در بسیاری از شبکه‌ها بر این پیچیدگی غلبه می‌کند.

۴. توضیح دهید که اگر به جای استفاده از طول متوسط صف (average queue length) از طول لحظه‌ای صف (in-stantaneous queue length) استفاده شود، چه پیامدهایی خواهد داشت.

پاسخ

اگر از طول لحظه‌ای صف به جای طول متوسط صف استفاده شود:

- رفتار ناپایدار: تغییرات ناگهانی در طول صف باعث می‌شود افت بسته‌ها به صورت ناپایدار انجام شود.
- افزایش احتمال حذف دسته‌جمعی بسته‌ها: نوسانات لحظه‌ای طول صف می‌تواند منجر به حذف ناگهانی تعداد زیادی از بسته‌ها شود که عملکرد TCP را مختل می‌کند.
- کاهش پایداری شبکه: استفاده از طول لحظه‌ای باعث کاهش پایداری و افزایش احتمال وقوع Global Synchronization می‌شود.

۵. راه‌هایی برای پیدا کردن مقادیر معقول برای پارامترهای RED (یعنی Min_{th} و Max_{th} و احتمال افت بسته زمانی که طول متوسط صف به Max_{th} می‌رسد) را بررسی کنید.

پاسخ

انتخاب مناسب پارامترهای RED برای عملکرد بهینه شبکه حیاتی است:

• مقدار Min_{th} :

- باید کمتر از ۵۰٪ ظرفیت صف باشد.
- انتخاب خیلی کوچک باعث افت بیش از حد بسته‌ها می‌شود.
- انتخاب خیلی بزرگ باعث تأخیر بالا و پر شدن صف می‌شود.

• مقدار Max_{th} :

- معمولاً بین $2 \times Min_{th}$ و $3 \times Min_{th}$ تنظیم می‌شود.
- باید به گونه‌ای باشد که از پر شدن کامل صف جلوگیری کند.

• احتمال افت بسته (P_{max}):

- مقدار بین 0.01 تا 0.1 معمولاً مناسب است.
- مقدار خیلی بزرگ منجر به افت زیاد بسته‌ها و کاهش Throughput می‌شود.
- مقدار خیلی کوچک باعث تأخیر و پر شدن صف می‌شود.

- تنظیم پویا: استفاده از الگوریتم‌های تطبیقی برای تنظیم پویا مقادیر بر اساس ترافیک شبکه می‌تواند به بهبود عملکرد کمک کند.

سوال چهارم

یک شبکه از الگوریتم RED (Random Early Detection) برای مدیریت ازدحام استفاده می‌کند. ظرفیت صف بین آستانه‌های حداقل (min-threshold) و حداکثر (max-threshold) تنظیم شده است. در این شبکه: آستانه حداقل برابر ۲۰ بسته و آستانه حداکثر برابر ۵۰ بسته است. اگر طول صف از آستانه حداقل عبور کند، احتمال حذف بسته‌ها به تدریج افزایش می‌یابد و با رسیدن به آستانه حداکثر، این احتمال به ۱۰۰٪ می‌رسد.

۱. اگر طول صف در لحظه‌ای به ۴۰ بسته برسد، با توجه به مقادیر حداقل و حداکثر، نرخ حذف بسته‌ها را محاسبه کنید.

پاسخ

الگوریتم RED از یک سیاست احتمالی برای حذف بسته‌ها استفاده می‌کند. احتمال حذف بسته (P_{drop}) با توجه به طول صف (q) و مقادیر آستانه حداقل (Min_{th}) و حداکثر (Max_{th}) به صورت زیر محاسبه می‌شود:

$$P_{drop} = \frac{q - Min_{th}}{Max_{th} - Min_{th}}$$

در اینجا داریم:

$$Min_{th} = 20$$

$$Max_{th} = 50$$

$$q = 40$$

بنابراین می‌توان نوشت:

$$P_{drop} = \frac{40 - 20}{50 - 20} = \frac{20}{30} = 0.6667$$

بنابراین، احتمال حذف بسته در طول صف $q = 40$ برابر با ۶۶.۶۷٪ است.

۲. یک سناریو شبیه‌سازی کنید که در آن طول صف به طور پیوسته افزایش می‌یابد و تأثیر الگوریتم RED بر ترافیک شبکه را تحلیل کنید. مشخص کنید که چگونه RED می‌تواند به کاهش ازدحام و جلوگیری از پر شدن کامل صف کمک کند و چه تأثیری بر تأخیر و نرخ ازدحام در شبکه دارد.

پاسخ

فرض کنید طول صف به تدریج افزایش می‌یابد. رفتار RED را در سه مرحله زیر بررسی می‌کنیم:

(آ) مرحله اول: طول صف کمتر از Min_{th}

- در این مرحله، هیچ بسته‌ای حذف نمی‌شود زیرا طول صف کمتر از Min_{th} است.
- RED به بسته‌ها اجازه می‌دهد بدون هیچ مداخله‌ای وارد صف شوند.

تأثیر بر شبکه:

- تأخیر کم
- نرخ ارسال ترافیک بالا
- بدون افت بسته

پاسخ

(ب) مرحله دوم: طول صف بین Max_{th} و Min_{th}

- در این مرحله، احتمال حذف بسته‌ها به صورت خطی با افزایش طول صف، افزایش می‌یابد.
- هدف اصلی در این مرحله جلوگیری از پر شدن کامل صف با حذف تدریجی بسته‌ها است.

تأثیر بر شبکه:

- احتمال حذف بسته‌ها افزایش می‌یابد، اما به صورت کنترل‌شده.
- تأخیر اندکی افزایش پیدا می‌کند.
- نرخ افت بسته‌ها متوسط است.

(ج) مرحله سوم: طول صف برابر یا بیشتر از Max_{th}

- در این مرحله، احتمال حذف بسته‌ها به 100% می‌رسد.
- بسته‌های ورودی جدید به صورت کامل حذف می‌شوند تا فضای بیشتری برای بسته‌های موجود فراهم شود.

تأثیر بر شبکه:

- جلوگیری از پر شدن کامل صف و کاهش پدیده Global Synchronization.
- تأخیر در این مرحله بسیار زیاد است.
- نرخ افت بسته‌ها بسیار بالا است.

نقش RED در کاهش ازدحام و تأخیر

- کاهش ازدحام: RED با حذف تدریجی بسته‌ها، از پر شدن کامل صف جلوگیری می‌کند.
- کاهش تأخیر: با مدیریت طول صف و جلوگیری از پر شدن کامل آن، تأخیر صف‌بندی (Queuing Delay) کاهش می‌یابد.
- بهبود توزیع منابع: RED به بسته‌ها اجازه می‌دهد به صورت عادلانه وارد صف شوند و جریان‌های مختلف شبکه فرصت‌های برابری برای استفاده از منابع داشته باشند.
- جلوگیری از پدیده Global Synchronization: حذف تصادفی بسته‌ها مانع از کاهش هم‌زمان نرخ ارسال جریان‌های TCP می‌شود.

سوال پنجم

فرض کنید یک منبع با حجم نامحدودی از اطلاعات برای ارسال، از یک کنترل حلقه بسته (closed-loop control) استفاده می‌کند تا نرخ ارسال خود را براساس اطلاعات بازخورد (feedback) تنظیم کند. در صورتی که اطلاعات بازخورد نشان دهد هیچ ترافیکی (traffic) در مسیر وجود ندارد، منبع به صورت پیوسته نرخ ارسال خود را به صورت خطی (linear) افزایش می‌دهد. اما اگر اطلاعات بازخورد حاکی از وجود ترافیک در مسیر باشد، منبع نرخ ارسال را به صفر کاهش می‌دهد و سپس این چرخه را با افزایش تدریجی نرخ ارسال ادامه می‌دهد تا بار دیگر ترافیک شناسایی شود. حال فرض کنید که مدت زمانی معادل T ثانیه طول می‌کشد تا اطلاعات بازخورد پس از وقوع ترافیک به منبع برسد. نمودار نرخ ارسال منبع را نسبت به زمان برای مقادیر کوچک و بزرگ T ترسیم کنید و توضیح دهید که تأخیر انتشار T (Propagation Delay) چه نقشی در این کنترل حلقه بسته ایفا می‌کند.

پاسخ

در این مسئله، یک منبع با استفاده از کنترل حلقه بسته نرخ ارسال خود را بر اساس اطلاعات بازخورد تنظیم می‌کند. رفتار منبع به صورت زیر توصیف می‌شود:

- افزایش نرخ خطی در نبود ترافیک: در صورتی که هیچ ترافیکی شناسایی نشود، نرخ ارسال منبع به صورت خطی افزایش می‌یابد:

$$R(t) = k \cdot t \quad \text{برای } t \text{ پس از بازنشانی نرخ}$$

که در آن k شیب افزایش نرخ ارسال است.

- بازخورد ترافیک: وقتی ترافیک شناسایی شود (بر اساس اطلاعات بازخورد که بعد از تأخیر T به منبع می‌رسد)، نرخ ارسال ناگهان به صفر کاهش می‌یابد. پس از این، منبع دوباره نرخ را از صفر شروع به افزایش می‌کند.

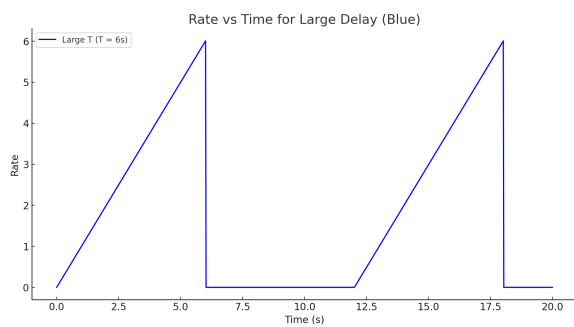
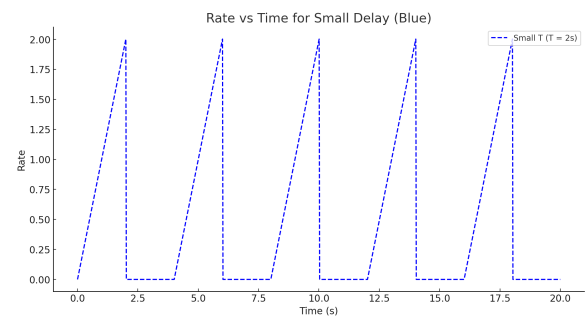
۱. نقش تأخیر T

- مقادیر کوچک T : در صورتی که تأخیر بازخورد T کوچک باشد، منبع سریعاً از وجود ترافیک مطلع شده و نرخ ارسال را متناسب با بازخورد به صفر کاهش می‌دهد. این باعث می‌شود که نوسانات در نرخ ارسال کمتر باشد و شبکه پایداری داشته باشیم.
- مقادیر بزرگ T : اگر T بزرگ باشد، منبع به مدت بیشتری نرخ ارسال را افزایش می‌دهد حتی پس از بروز ترافیک. در نتیجه، نرخ ارسال به مقادیر بالاتری می‌رسد پیش از اینکه بازخورد ترافیک دریافت شود و نرخ ناگهان به صفر کاهش یابد. این می‌تواند به نوسانات شدید در نرخ ارسال و احتمالاً بارگذاری بیش‌ازحد شبکه منجر شود.

- ۲. نمودار نرخ ارسال $(R(t))$ نسبت به زمان (t) در شکل «۱»، نمودار نرخ ارسال منبع برای مقادیر کوچک و بزرگ T رسم شده است.

نمودارها نشان می‌دهند که تأخیر بازخورد T تأثیر زیادی بر رفتار کنترل حلقه بسته دارد:

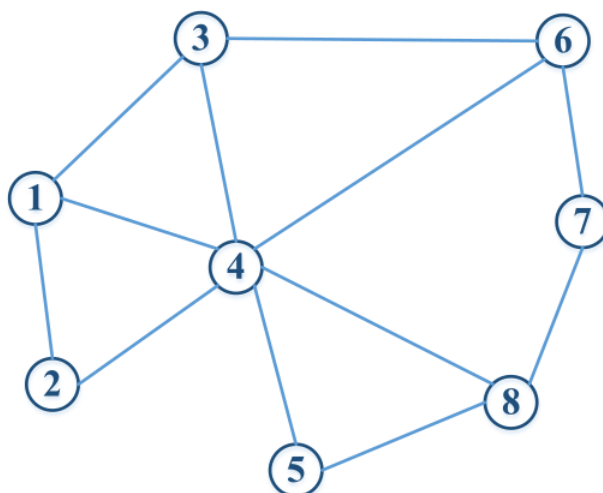
- تأخیر کوچک T منجر به نوسانات کوچک‌تر در نرخ ارسال و شبکه‌ای پایدارتر می‌شود.
- تأخیر بزرگ T باعث افزایش نوسانات و احتمال تراکم بیش از حد در شبکه می‌گردد.

(ب) نمودار نرخ ارسال برای تأخیر بزرگ ($T = 6$)(آ) نمودار نرخ ارسال برای تأخیر کوچک ($T = 2$)

شکل ۱: مقایسه نرخ ارسال منبع برای مقادیر مختلف تأخیر T

سوال ششم

شبکه زیر را در نظر بگیرید:



شکل ۲: شبکه سوال ۶

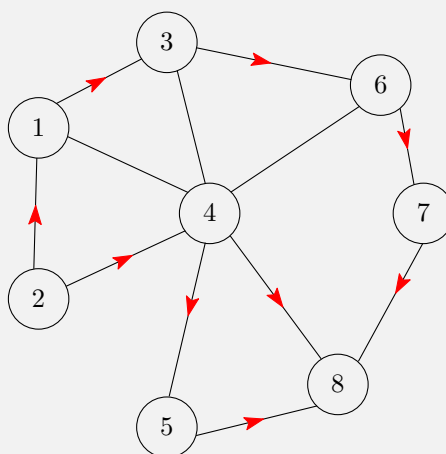
فرض کنید که ارتباط‌های زیر به ترتیب (چپ به راست) باید ایجاد شوند:

$5 \rightarrow 8$, $1 \rightarrow 8$, $2 \rightarrow 4$, $3 \rightarrow 8$, $3 \rightarrow 5$, $2 \rightarrow 1$, $1 \rightarrow 3$, $3 \rightarrow 6$, $6 \rightarrow 7$, $7 \rightarrow 8$

۱. با استفاده از الگوریتم مسیریابی کوتاه‌ترین مسیر (Shortest-Path) بیشترین ارتباطی که می‌توانید را برقرار کنید. ارتباط‌های قطع شده را نیز مشخص کنید.

پاسخ

با فرض اینکه ظرفیت هر لینک ۱ واحد و ظرفیتی که هر ارتباط به خود اختصاص می‌دهد نیز ۱ واحد باشد، می‌توان گفت که هر ارتباط فقط یک لینک را اشغال می‌کند و هر لینک صرفاً برای یک ارتباط استفاده می‌شود. ابتدا ارتباط تک گام را رسم می‌کنیم سپس بررسی می‌کنیم که کدام ارتباط دو گام را اگر بررسی کنیم بهتر است و ارتباطات به بیشترین تعداد خود می‌رسد.



شکل ۳: بیشترین ارتباطات قابل رسم

پاسخ

همه ارتباطات به جز ارتباط $8 \rightarrow 3$ برقرار هستند.

۲. با استفاده از الگوریتم مسیریابی Constraint Shortest-Path بیشترین ارتباطی که می‌توانید را برقرار کنید. ارتباط‌های قطع شده را نیز مشخص کنید.

پاسخ

اگر در این قسمت نیز همان فرض قسمت قبل را در نظر بگیریم، الگوریتم CSPF همانند الگوریتم Shortest Path عمل می‌کند و نتیجه با قسمت قبل یکسان می‌شود.

۳. آیا می‌توانید الگوریتم Constraint Shortest-Path را بهبود دهید؟

پاسخ

اگر ظرفیت لینک‌ها را تقسیم کنیم، الگوریتم CSPF می‌تواند لینک $8 \rightarrow 3$ را نیز برقرار کند، برای افزایش سرعت الگوریتم نیز می‌توان آن را Pipeline کرد.

سوال هفتم

فرض کنید در یک شبکه، مسیریابی به صورت خودکار از کوتاه‌ترین مسیر برای هر جریان استفاده می‌کند. این امر باعث شده است که یک لینک مشخص به ظرفیت حداکثری خود برسد و دچار ازدحام شود، در حالی که سایر لینک‌ها کمتر از ظرفیت خود استفاده می‌شوند.

۱. با در نظر گرفتن ظرفیت هر لینک و نیازمندی‌های پهنای باند برای هر جریان، یک طرح توزیع بهینه برای جریان‌ها ارائه دهید که بار را در شبکه به طور یکنواخت توزیع کند.

پاسخ

برای توزیع یکنواخت بار در شبکه، می‌توان از رویکردهای زیر استفاده کرد:

(آ) استفاده از الگوریتم‌های مسیریابی چندمسیره (Multipath Routing): به جای ارسال تمام جریان‌ها از یک مسیر، می‌توان ترافیک را به چند مسیر تقسیم کرد. به عنوان مثال، استفاده از پروتکل‌هایی مانند ECMP (Equal-Cost Multi-Path) پیشنهاد می‌شود.

(ب) مدل‌سازی ریاضی: با تعریف مدل بهینه‌سازی ریاضی، توزیع بار شبکه را می‌توان به شکل زیر بهینه کرد:

- متغیر تصمیم: x_{ij} مقدار ترافیک ارسال شده از جریان j روی لینک i .
- محدودیت:

$$\sum_j x_{ij} \leq C_i \quad \forall i$$

که C_i ظرفیت لینک i است.

- هدف: کمینه‌سازی ازدحام یا توزیع یکنواخت بار:

$$\max_i \frac{\sum_j x_{ij}}{C_i}.$$

(ج) مراحل طراحی:

- تحلیل شبکه: شناسایی لینک‌های پرتراфик و مسیرهای جایگزین.
- تقسیم ترافیک: انتقال بخشی از جریان‌ها به مسیرهای کم‌مصرف‌تر.
- بازتخصیص منابع: بر اساس ظرفیت لینک‌ها و نیازمندی‌های جریان‌ها، توزیع بهینه انجام می‌شود.

۲. نشان دهید که این بازطراحی چگونه می‌تواند تأخیر ناشی از ازدحام را کاهش دهد. برای این منظور، فرض کنید ظرفیت لینک ازدحام‌کرده ۱۰۰ مگابیت در ثانیه است و میزان ترافیک جاری روی آن به ۱۲۰ مگابیت در ثانیه رسیده است. توزیع جدید را طوری طراحی کنید که استفاده از لینک به کمتر از ۸۰ درصد ظرفیت برسد و میزان تأخیر را محاسبه و با وضعیت اولیه مقایسه کنید.

پاسخ

وضعیت اولیه:

• ظرفیت لینک: $C = 100 \text{ Mbps}$.• ترافیک جاری: $T = 120 \text{ Mbps}$.

• درصد استفاده:

$$\frac{T}{C} \times 100 = \frac{120}{100} \times 100 = 120\%.$$

• مدل تأخیر: فرض کنیم تأخیر بر اساس رابطه $D = \frac{1}{C-T}$ محاسبه شود. در این حالت:

$$D_{\text{initial}} = \frac{1}{100 - 120} = \infty.$$

ازدحام شدید باعث تأخیر بی‌نهایت می‌شود.

وضعیت بازطراحی شده:

• هدف: کاهش استفاده لینک به 80% ظرفیت:

$$\text{Traffic Target} = 0.8 \times 100 = 80 \text{ Mbps}.$$

• بازتوزیع جریان‌ها: با انتقال 40 Mbps ترافیک به مسیرهای دیگر، ترافیک لینک به $T_{\text{new}} = 80 \text{ Mbps}$ کاهش می‌یابد.محاسبه تأخیر در وضعیت جدید: با استفاده از رابطه $D = \frac{1}{C-T}$ در وضعیت جدید:

$$D_{\text{new}} = \frac{1}{100 - 80} = \frac{1}{20} = 0.05 \text{ ms}.$$

مقایسه تأخیر:

• وضعیت اولیه: تأخیر بی‌نهایت به دلیل ازدحام شدید.

• وضعیت بازطراحی شده: کاهش تأخیر به 0.05 ms.