



سوال ۱:

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.

الف) 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.

تعداد بسته‌ها در مدت ۵ ثانیه:

- تعداد بسته‌های اولویت بالا: $6 \times 5 = 30$ بسته
- تعداد بسته‌های اولویت پایین: $4 \times 5 = 20$ بسته
- مجموع بسته‌ها: $30 + 20 = 50$ بسته

زمان انتظار هر بسته در صف FIFO بر اساس ترتیب ورود محاسبه می‌شود. بسته‌های اولویت بالا و پایین نمی‌توانند بر یکدیگر تأثیر بگذارند، زیرا FIFO ترتیب پردازش ثابت و بدون تغییر دارد.

با توجه به اطلاعات داده شده در سوال، در هر ثانیه، مجموعاً ده بسته به صف وارد می‌شود. از آنجا که ظرفیت صف ۱۰ بسته است، در ثانیه اول صف به طور کامل پر می‌شود. در این سناریو، تعداد بسته‌هایی که از هر دسته باقی می‌مانند به نسبت نرخ ورود بسته‌ها محاسبه می‌شود. به این ترتیب، از ۶ بسته اولویت بالا و ۴ بسته اولویت پایین، هر کدام به ترتیب در صف قرار می‌گیرند.

بدترین حالت برای خروج یک بسته اولویت بالا زمانی است که ۴ بسته اولویت پایین در جلوی آن قرار داشته باشند. به دلیل این ترتیب پردازش در روش FIFO، بسته‌های اولویت بالا باید منتظر پردازش تمامی بسته‌های اولویت پایین در صف باشند. این مسئله باعث افزایش زمان انتظار برای بسته‌های اولویت بالا می‌شود.

ب) 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.

در مقایسه با FIFO، استفاده از روش HOL باعث می‌شود بسته‌های اولویت بالا همیشه اولویت پردازش داشته باشند و به این ترتیب زمان انتظار برای این بسته‌ها به حداقل می‌رسد. بسته‌های اولویت بالا هیچ‌گاه منتظر پردازش بسته‌های اولویت پایین نمی‌مانند، حتی اگر بسته‌های اولویت پایین زودتر وارد صف شده باشند. این روش می‌تواند زمان تأخیر بسته‌های اولویت بالا را کاهش دهد.

از طرفی، در این روش ممکن است بسته‌های با اولویت پایین، بسته به نرخ ورود بسته‌های با اولویت بالا، هیچ‌گاه فرصت خروج از صف را پیدا نکنند. این امر می‌تواند باعث بروز مشکلاتی در پردازش بسته‌های با اولویت پایین شود و باعث افزایش زمان انتظار آن‌ها شود، به خصوص در مواقعی که حجم ترافیک با اولویت بالا زیاد باشد.



در نهایت، روش HOL برای سناریوهایی که به تأخیر حساس هستند (مانند بسته‌های اولویت بالا) مناسب‌تر است، اما برای جلوگیری از تأثیرات منفی بر بسته‌های اولویت پایین، نیاز به پیاده‌سازی تدابیر اضافی دارد.

سوال ۲:

فرض کنید که در یک سیستم صف عادلانه وزن دار (weighted fair-queueing 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.

بله، برچسب پایان (Finish Tag) تابعی از زمان ورود بسته به سیستم است، اما همچنین به طول بسته نیز بستگی دارد. بسته‌ای که زودتر وارد می‌شود ممکن است طول بسیار زیادی داشته باشد، در حالی که بسته‌ای که دیرتر می‌آید ممکن است طول بسیار کمتری داشته باشد. بنابراین، بسته‌ای که دیرتر می‌آید، مقدار F کمتری نسبت به بسته‌ای که زودتر وارد شده خواهد داشت.

برای مثال، در یک سیستم صف‌بندی با سه صف منطقی و نرخ سرویس‌دهی یک واحد در ثانیه، بسته‌ها با الگوی زیر وارد می‌شوند:

- بافر ۱: ورود بسته در زمان $t=0$ با طول ۲؛ ورود بسته در زمان $t=4$ با طول ۱.
- بافر ۲: ورود بسته در زمان $t=1$ با طول ۳؛ ورود بسته در زمان $t=2$ با طول ۱.
- بافر ۳: ورود بسته در زمان $t=3$ با طول ۵.

با توجه به این ورودی‌ها، اولین بسته‌ای که به بافر ۲ می‌رسد، در نهایت زمان پایان دیرتری نسبت به دومین بسته‌ای که به بافر ۱ می‌رسد، خواهد داشت. زمانی که بسته دوم به سیستم وارد می‌شود، بسته اول قبلاً وارد سرویس شده است.

سوال ۳:

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

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

RED با حذف تصادفی بسته‌ها پس از عبور از آستانه (Threshold)، باعث می‌شود که تنها بخشی از منابع، نرخ ارسال خود را کاهش دهند. این روش موجب می‌شود که با آغاز ازدحام، فرستنده‌های TCP به شکلی تدریجی و پیوسته نرخ ارسال خود را کاهش دهند، به جای آنکه همه به طور ناگهانی و هم‌زمان این کار را انجام دهند.

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

الگوریتم RED (Random Early Detection) با رهسازی بسته‌ها پیش از وقوع ازدحام، بازده شبکه را در سطحی پایدار حفظ می‌کند و جلوی نوساناتی را می‌گیرد که به دلیل کاهش هم‌زمان نرخ ارسال فرستنده‌ها رخ می‌دهد. عملکرد این الگوریتم و نزدیکی بازده شبکه به ظرفیت واقعی، به طور قابل توجهی به مقادیر تنظیم‌شده برای $minth$ و $maxth$ بستگی دارد.

ج) پیچیدگی پیاده‌سازی الگوریتم RED را بررسی کنید.



در پیاده سازی (RED) (*Random Early Detection*)، هر صف باید دو متغیر min_{th} (حداقل آستانه) و max_{th} (حداکثر آستانه) را نگهداری کند. یک ماژول پردازشی نیز لازم است که به طور مداوم میانگین متحرک (*Running Average*) طول صف را محاسبه کند تا تأثیر انفجارهای موقت ترافیکی (*Traffic Bursts*) را در سیستم مدیریت کند و از تداخل آن‌ها با عملکرد عادی جلوگیری شود. همچنین، یک تولیدکننده اعداد تصادفی (*Random Number Generator*) لازم است تا بسته‌ها را با احتمالی که به میانگین متحرک طول صف بستگی دارد، حذف کند.

(د) توضیح دهید که اگر به جای استفاده از طول متوسط صف (*average queue length*)، از طول لحظه‌ای صف (*instantaneous queue length*) استفاده شود، چه پیامدهایی خواهد داشت.

استفاده از طول میانگین صف (*Average Queue Length*) به RED اجازه می‌دهد تا نوسانات کوچک اطراف سطح آستانه (*Threshold Level*) را تحمل کند. در مقابل، استفاده از طول لحظه‌ای صف باعث می‌شود که نوسانات طبیعی بافر (*Buffer*) به فعال سازی مکرر RED منجر شود و این امر به ناکارآمدی در شبکه منتهی گردد.

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

مقادیر پارامترهای RED به عوامل مختلفی بستگی دارد. انتخاب min_{th} و max_{th} به اندازه‌های بافر موجود وابسته است. تفاوت بین این دو آستانه باید بزرگ‌تر از افزایش معمول طول میانگین در یک زمان رفت و برگشت (*RTT*) باشد. این تفاوت همچنین به اندازه‌های انفجار موقتی انفجار موقتی (*Transient Bursts*) که باید در زمان وقوع ازدحام تحمل شوند، مربوط است. تنظیم پیشنهادی برای max_{th} معمولاً ۲ تا ۳ برابر min_{th} است. به طور کلی، اگر *RTT* بیشتر باشد، min_{th} باید کوچکتر باشد. وابستگی به *RTT* انتخاب آستانه‌ها را به سرعت لینک و تأخیر انتشار (*Propagation Delay*) مرتبط می‌کند. همچنین انتخاب min_{th} بر تأخیر میانگین تأثیر دارد. برای تأخیرهای کوچک‌تر، min_{th} باید کوچک‌تر باشد، اما اگر آستانه خیلی کوچک باشد، بهره‌برداری از لینک ممکن است کاهش یابد. انتخاب احتمال حذف بسته‌ها باید به میزان حذف بسته در حالت پایدار شبکه مرتبط باشد. در *weighted RED*، وزن تعیین‌کننده ثابت زمانی برای میانگین‌گیری اندازه میانگین صف است. اگر وزن خیلی کوچک باشد، اندازه میانگین صف خیلی کند به تغییرات پاسخ خواهد داد. از طرف دیگر، وزن بزرگ‌تر باعث می‌شود که میانگین صف به اندازه لحظه‌ای صف نزدیک‌تر شود و در نتیجه نوسانات بیشتری ایجاد کند.

سوال ۴:

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.

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

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:

$$P_{\text{drop}} = \frac{q - \text{Min}_{th}}{\text{Max}_{th} - \text{Min}_{th}}$$

در اینجا داریم:

- $\text{Min}_{th} = 20$
- $\text{Max}_{th} = 50$
- $q = 40$

پس می توان نوشت:

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

بنابراین، احتمال حذف بسته در طول صف $q = 40$ برابر 66.67% است.

ب) یک سناریو شبیه سازی کنید که در آن طول صف به طور پیوسته افزایش می یابد و تأثیر الگوریتم RED بر ترافیک شبکه را تحلیل کنید. مشخص کنید که چگونه RED می تواند به کاهش ازدحام و جلوگیری از پر شدن کامل صف کمک کند و چه تاثیری بر تأخیر و نرخ ازدحام در شبکه دارد.

در این سناریو، طول صف به طور پیوسته افزایش می یابد. هدف الگوریتم RED این است که ازدحام را قبل از وقوع آن شناسایی کرده و با حذف تصادفی بسته ها، از افزایش بیشتر طول صف و در نتیجه جلوگیری از ازدحام شدید پیشگیری کند.

گام ۱: شروع سناریو و افزایش طول صف

فرض کنید در ابتدا طول صف برابر با ۱۰ بسته است و ترافیک وارد شبکه می شود. با ورود هر بسته، طول صف به تدریج افزایش می یابد. هنگامی که طول صف به آستانه حداقل (۲۰ بسته) می رسد، احتمال حذف بسته ها به تدریج افزایش می یابد.

گام ۲: عبور از آستانه حداقل به آستانه حداکثر

زمانی که طول صف از ۲۰ بسته عبور کرده و به ۳۰ بسته می رسد، احتمال حذف بسته ها به ۳۳.۳۳٪ می رسد. در صورتی که طول صف به ۴۰ بسته برسد، نرخ حذف بسته ها به ۶۶.۶۷٪ خواهد رسید (همانطور که در بخش قبلی محاسبه شد). اگر طول صف به ۵۰ بسته برسد، احتمال حذف بسته ها به ۱۰۰٪ می رسد و بسته ها به طور کامل حذف می شوند.

گام ۳: جلوگیری از ازدحام

با استفاده از الگوریتم RED، بسته ها به طور تصادفی قبل از پر شدن کامل صف حذف می شوند. این اقدام پیشگیرانه به شبکه این امکان را می دهد که از پر شدن کامل صف و ایجاد ازدحام جلوگیری کند. در حقیقت، اگر RED فعال نباشد، شبکه ممکن است با افزایش ناگهانی تعداد بسته ها مواجه شود که منجر به ازدحام شدید و کاهش کیفیت سرویس می شود.

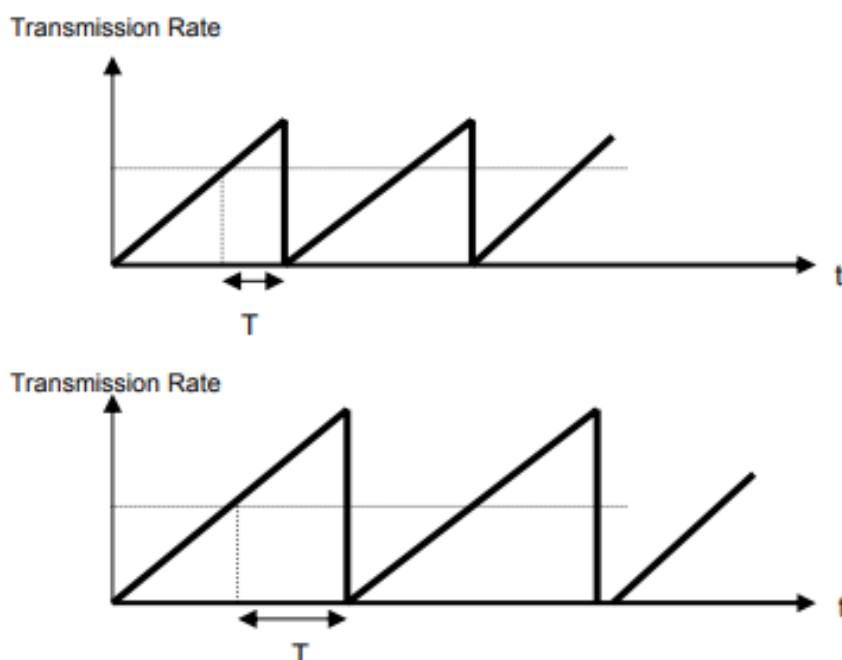


به این ترتیب، الگوریتم RED نه تنها از پر شدن کامل صف جلوگیری می‌کند، بلکه با مدیریت هوشمند و پیشگیرانه ترافیک، ازدحام و تأخیر در شبکه را کاهش می‌دهد.

سوال ۵:

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 شکل 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.

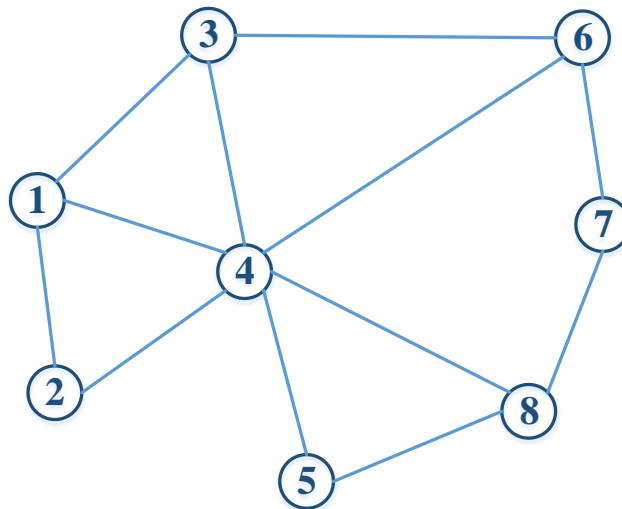
نرخ در منبع، رفتاری شبیه به الگوی دندانه‌اره‌ای (Sawtooth Behavior) از خود نشان می‌دهد؛ به گونه‌ای که ابتدا از مقدار صفر آغاز می‌شود، به صورت خطی افزایش یافته و به نقطه‌ای حداکثری می‌رسد، سپس بار دیگر به صفر بازمی‌گردد. این چرخه به طور مکرر و با فرکانسی وابسته به T تکرار می‌شود، چرا که این مکانیزم توانایی سازگاری با پهنای باند موجود (Available Bandwidth) را ندارد. هر چه مقدار T افزایش یابد، تأخیر در کاهش نرخ پس از شناسایی ازدحام (Congestion) نیز بیشتر می‌شود. از این رو، منبع نمی‌تواند به موقع به وقوع ازدحام واکنش نشان دهد و حتی ممکن است به تشدید آن کمک کند.





سوال ۶:

شبکه‌ی زیر را در نظر بگیرید.



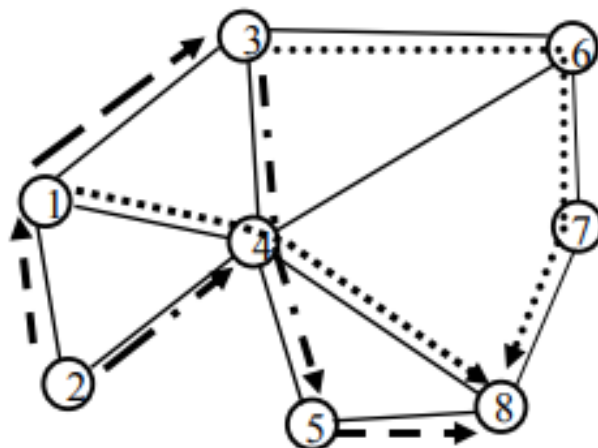
شکل سوال ۶

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

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

همچنین فرض کنید ظرفیت هر لینک ۱ واحد و همچنین ظرفیتی که یک ارتباط به خود اختصاص می‌دهد نیز ۱ واحد باشد.

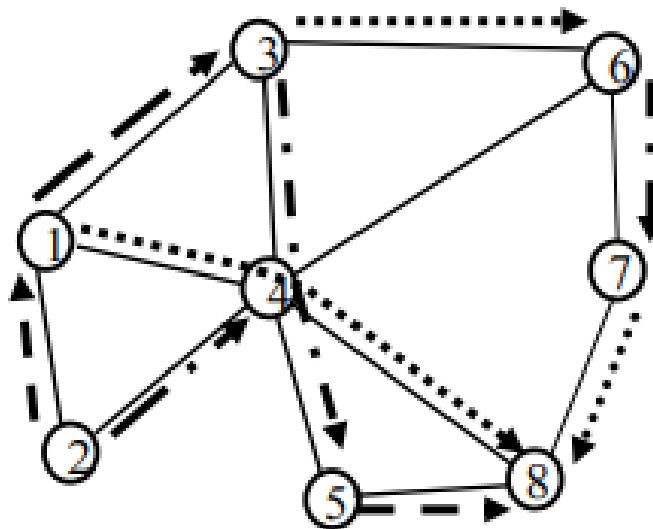
الف) با استفاده از الگوریتم مسیریابی کوتاه‌ترین مسیر (Shortest-Path) بیشترین ارتباطاتی که می‌توانید را برقرار کنید. ارتباط‌های قطع شده را نیز مشخص کنید.





یک اتصال (از ۳ به ۸) مسدود شده است.

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



سه اتصال (از ۳ به ۶، ۶ به ۷، و ۷ به ۸) مسدود شده‌اند.

ج) آیا می‌توانید الگوریتم Constraint Shortest-Path بهبود دهید؟

تعداد ارتباطاتی که توسط هر اتصال مسدود می‌شوند، ردیابی می‌شود و اگر مشخص شود که یک اتصال تعداد زیادی از اتصالات دیگر را مسدود کرده است، آن اتصال به مسیر دیگری هدایت می‌شود.

سوال ۷:

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.

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

برای توزیع بهینه جریان‌ها در شبکه و جلوگیری از ازدحام لینک‌ها، می‌توان از رویکرد مسیریابی مبتنی بر تعادل بار استفاده کرد. در این روش، به جای استفاده از تنها کوتاه‌ترین مسیر، جریان‌ها به صورت هوشمند و بر اساس ظرفیت لینک‌ها و نیازمندی‌های پهنای باند توزیع



می‌شوند. هدف اصلی این است که بار شبکه به‌طور یکنواخت بین لینک‌ها توزیع شود و از استفاده بیش از حد یک لینک خاص جلوگیری گردد.

برای دستیابی به این هدف، ابتدا ظرفیت هر لینک و پهنای باند مورد نیاز هر جریان مشخص می‌شود. سپس، به‌جای مسیریابی تک‌مسیره، از چندین مسیر جایگزین استفاده می‌شود. جریان‌ها به‌صورت دینامیک بین این مسیرها توزیع می‌شوند به‌طوری‌که مجموع بار وارد بر هر لینک از ظرفیت آن تجاوز نکند. این مسئله را می‌توان با یک مدل ریاضی بهینه‌سازی، مانند برنامه‌ریزی خطی، حل کرد که در آن هدف، حداقل کردن بیشترین نسبت استفاده از لینک‌ها به ظرفیت آن‌هاست. در این مدل، قیود شامل محدودیت ظرفیت لینک‌ها و تضمین تأمین پهنای باند مورد نیاز جریان‌ها هستند.

در عمل، این رویکرد می‌تواند با استفاده از الگوریتم‌های مسیریابی چندمسیره مانند *ECMP* یا پروتکل‌های *SDN* اجرا شود. در این حالت، یک کنترل‌کننده مرکزی اطلاعات کاملی از وضعیت شبکه دریافت کرده و مسیرها را بر اساس وضعیت لحظه‌ای بارگذاری لینک‌ها تعیین می‌کند. این روش باعث می‌شود بار شبکه به‌طور یکنواخت توزیع شود و از ازدحام لینک‌های خاص جلوگیری گردد. (پاسخ‌های دیگر نیز مورد تأیید است.)

ب) 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.

فرض کنید:

• $C = 100 \text{ Mbps}$

• $T = 120 \text{ Mbps}$

• $D = \frac{1}{C-T}$

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

• 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 بی نهایت می شود.

وضعیت بازطراحی شده:

- هدف: کاهش استفاده لینک به ۸۰٪ ظرفیت:

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

- بازتوزیع جریان ها: با انتقال 40 Mbps ترافیک به مسیرهای دیگر، ترافیک لینک به $T_{\text{new}} = 80 \text{ Mbps}$ کاهش می یابد.

محاسبه تأخیر در وضعیت جدید:

$$D_{\text{initial}} = \frac{1}{100 - 80} = \frac{1}{20} = 0.005 \text{ s}$$

مقایسه تأخیر :

وضعیت اولیه: تأخیر بی نهایت به دلیل ازدحام شدید.

وضعیت بازطراحی شده: کاهش تأخیر به ۵۰ میلی ثانیه.