# به نام خدا

# جزوه‌ی درس شبکه‌های کامپیوتری پیشرفته

دکتر میرعابدینی

۱۳۹۳-۹۴

رفرنس اصلی: Computer Network / Tanenbaum

فرمت ارسال تمرین‌ها: [نام :Net :Bushr :Subj]

حضور ۵٪ + تمرین و تحقیق ۱۰٪ + میان ترم ۱۵٪ + پایان ترم ۶۵٪

نرم افزارهای مناسب برای پیاده سازی: Omnet ++, Opnet, Ns2, Jsim, glomosim (روی لینوکس نصب می‌شود)

**پروژه**: انتخاب حداقل ۳ تا ۵ مقاله سال ۲۰۱۳ به بعد در یک زمینه خاص از شبکه. ترجمه تحلیل، بررسی و ارزیابی وپیاده سازی یکی از روش‌ها. آماده‌سازی گزارش بر مبنای شیوه ارائه مطالب، doc و سمینار .ppt

**موضوع‌های پیشنهادی:** کاربردهای اتصال گرا و غیراتصال گرا، الگوریتم تشخیص زود هنگام RED، روش‌های رمز گذاری، سیاه چاله‌ها، SDN، مسیریابی امن، چند مسیره (Multi pass) و...

(از Survey هم می‌توان استفاده کرد).

**تمرین ۱**

تفاوت بین SDN (Software Defined Network) و NOC (Network On Chip) چیست؟

**تمرین ۲**

انواع نرم افزارهای ویروس یاب را بررسی و ارزیابی نمایید.

## «جلسه اول»

**شبکه Adhoc:** شبکه‌ای که در هیچ یک از دسته‌بندی‌های Star Bus، رینگ و ... نباشد. تعداد گره‌ها نامعلوم است. گره‌ها بصورت تصادفی حرکت می‌کنند. (اگر متحرک باشند این شبکه را Adhoc Mobile گویند)

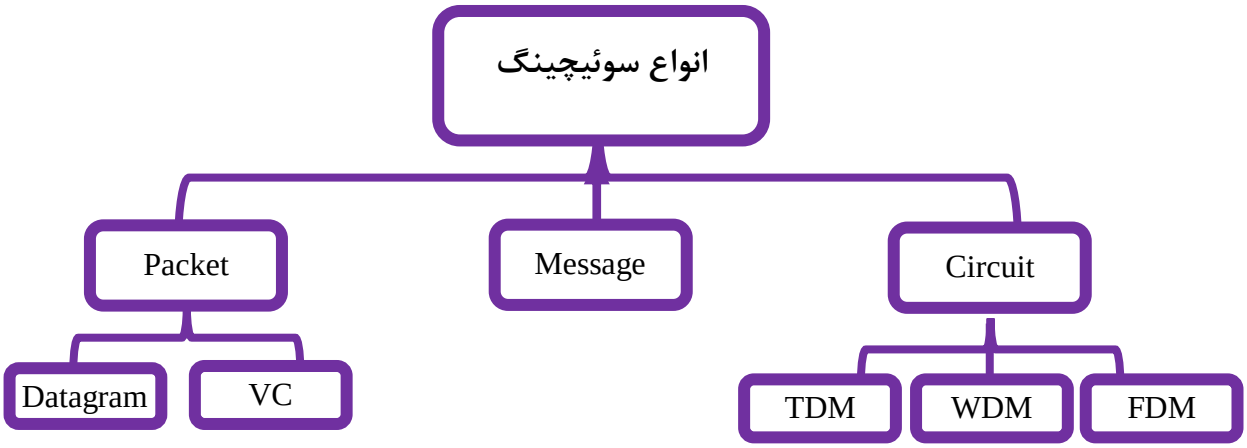
### لایه بندی OSI (*Open System Interconnection*)

- این پروتکل توسط IEEE ایجاد و مدیریت شد.
- بین هر لایه یک اینترفیس وجود دارد.
- لایه چهارم روی QOS کار می‌کند. در واقع این لایه که بر روی کیفیت خدمت متمرکز است توسط لایه انتقال مدیریت می‌شود.
- جدول مسیریابی توسط لایه ۳ ارائه می‌شود.

- لایه دوم آنلاین است. فریم بندی، کنترل خط، ایجاد داده‌های کنترلی و تضمین ارسال داده‌ها در این لایه است. تفاوت آن با لایه چهارم در سطح بررسی است. کنترل خطا می‌تواند به وسیله Parity, CRC یا همینگ انجام شود. کنترل جریان هم در لایه دوم انجام می‌شود. کنترل جریان کار هماهنگی سرعت انتقال بین فرستنده پرقدرت و گیرنده کم قدرت انجام می‌شود.

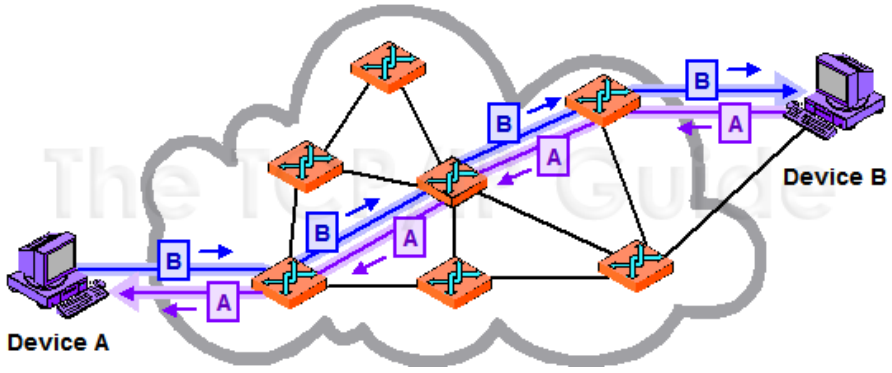
- ۸۰٪ عملکرد اینترنت در لایه دو و سه است. (علی الخصوص ۲)

- در لایه سوم آدرس دهی انجام می‌شود، اما تضمینی در مورد رسیدن بسته، طول عمر و ... ندارد اما لایه چهارم این تضمین را انجام می‌دهد.




## *سوئیچینگ مداری*

کانال اختصاصی بین ایستگاه‌ها وجود دارد. (از طریق اتصال فیزیکی) سه مرحله‌ی برقراری

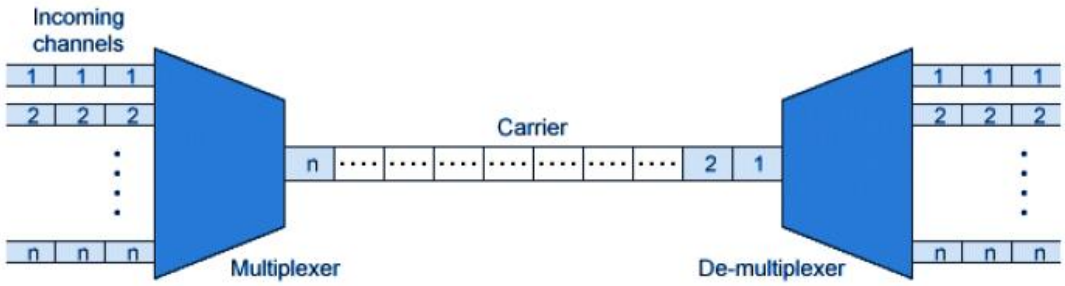


ارتباط، انتقال داده و قطع ارتباط دارد. این روش دارای سه مرحله است: برقراری ارتباط – انتقال داده- قطع ارتباط. از نمونه‌های بارز این شبکه: شبکه تلفن شهری.

**مزایا:** عدم نیاز به بسته بندی و خارج از کردن از بسته بندی (سربار پردازشی ندارد)

**معایب:** زمانبر بودن فاز برقراری ارتباط- اشغال شدن خط

***سوئیچینگ تسهیم زمانی:*** در هر زمان یک سیگنال عبور می‌کند که بین یکسری استفاده کننده تقسیم شده است.



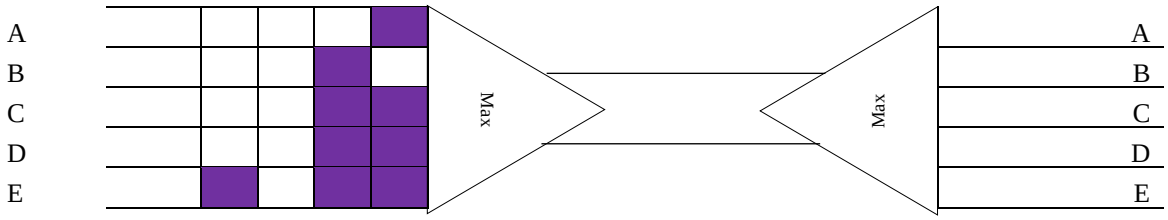

***سوئیچینگ تقسیم فرکانسی:*** پهنای باند به بازه‌های مشخص تقسیم می‌شود.

***سوئیچینگ تقسیم طول موج:*** طول موج و دامنه‌ی آن یکی کم و یکی زباد است تا تداخل ایجاد نشود.

**مثال:** فرض کنید روش استفاده شده برای ارسال فریم‌ها بر اساس TDM ثابت باشد. در اینصورت با توجه به شکل زیر اگر اندازه فریم‌ها پنج بخشی باشد با روش TDM معمولی بازدهی کدام است؟ ثانیاً اگر از روش TDM, STDM آماری استفاده شود و اندازه فریم و پنج بخشی باشد بازدهی کدام است؟ ثالثاً اگر اندازه فریم ۴ بخشی و ۵ بخشی و زمان انتظار تا ۲ سیکل مجاز باشد بازدهی کدام است؟ ضمن اینکه در TDM معمولی هر فریم فقط یک سیکل زمان دارد تا فریم را پرکند و بیش از یک سیکل نمی‌تواند منتظر بماند.

نکته: در روش TDM آماری یا STDM بسته به فضای مسئله ما می‌توانیم بخش‌های یک فریم و تعداد سیکل‌های مورد انتظار را تغییر دهیم.

حل:




| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| F1 | E | D1 | C1 | X | A | F2 | E2 | D2 | C2 | B2 | X | F3 | E4 | X | X | X | X |

با روش TDM معمولی : $\frac{9}{15} \times 100 = 60\%$

حل ب: (STDM)

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| فریم اول | B2 | E1 | D1 | C1 | A1 | فریم سوم | X | X | X | X | E |
| فریم دوم | X | X | E2 | D2 | C2 | | | | | | |

بازدهی (U) : $\frac{9}{15} \times 100 = 60\%$

همین‌طور که میبینیم زمان انتظار TDM کمتر از STDM شد ولی در کل SDTM بسیار سریع‌تر عمل می‌کند. (معمولاً)

حالت ۴ بخشی با زمان انتظار حداکثر ۲ سیکل

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| F1 | E1 | D1 | C1 | A1 | F2 | E2 | D2 | C2 | P2 | F3 | X | X | X | E2 |

بازدهی (U) : $\frac{9}{12} \times 100 = 75\%$

مشاهده شد که برخلاف تصور با اینکه فریم به بخش‌های کوچکتر تقسیم شد ولی بازدهی بیشتر شد.

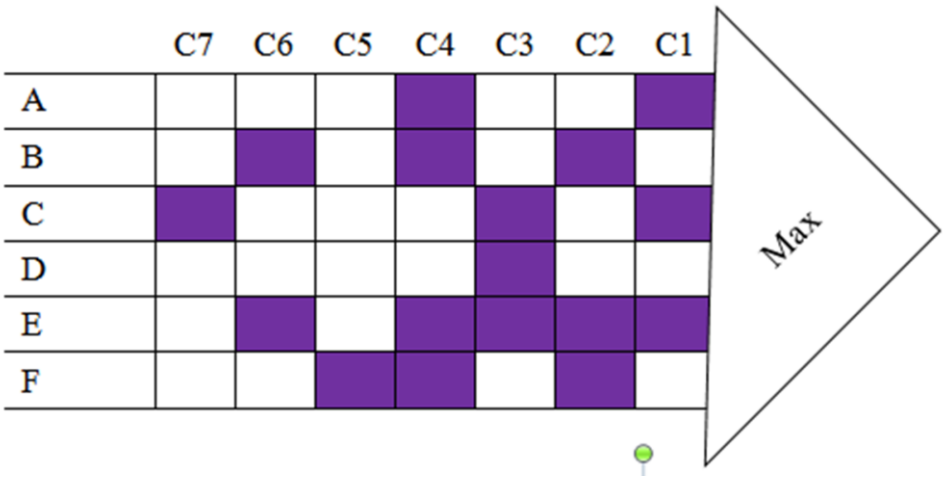
## تمرین ۳

تمرین۳: با توجه به شکل زیر مطلوبست:

الف) TDM با ۴ بخش (TDM Asy ncron غیر همزمان)

ب) STDM با ۴ بخش و نهایتاً ۲ سیکل انتظار

ج) STDM با پنج بخش و ۲ سیکل انتظار



## «جلسه دوم»

**رابطه شانون:** نسبت سیگنال به نویر نشان می‌دهد در یک کانال فیزیکی به ازای عبور چند سیگنال یکی از آنها دچار نویز می‌شود که به آن نسبت توان سیگنال به توان نویز یا S/N گویند.

مثال: در یک کانال فیزیکی به ازای هر ۱۰۰۰۰ سیگنال ارسالی یک سیگنال دچار خطا می‌شود. نسبت سیگنال به نویز آن چند است؟ جواب: S/n =۱۰۰۰۰

رابطه‌ی سیگنال به نویز بصورت دسیبل نیز نشان داده می‌شود:

$$db = 10 \qquad 10g_{10}^{S/N}$$

مثال: در یک کانال داده نسبت S/N برابر ۱۰۰۰۰۰ است. این نسبت معادل چند دسیبل است؟

$$db = 10 \qquad 10g_{10}^{100000} = 10 \times 5 = 50db$$

$\log 10_{10}^{k} = k \qquad *\log_a^{ab} = b \qquad *\log_c^{ab} = \log_c^a + \log_c^b$ : *هک می‌ناد یم

$\log_b^a = c \Rightarrow a = b^c$

مثال: در یک کانال نسبت سیگنال به نویز برابر ۴۰ وسیبل است. در این صورت S/N کدام است؟

$$40 = 10\log_{10}^{S/N} \Rightarrow \log_{10}^{S/N} = 4 \Rightarrow \frac{S}{N} = 10^4 = 10000$$

در فیبر نوری نسبت S/N برابر با ۱۰ دسیبل است، یعنی به ازای هر ۱۰۰۰۰۰۰۰ یک سیگنال خطا داریم.

**واحدهای حافظه**

$$1.Byte = 8bit \Rightarrow 1KByte = 1024 \qquad or \qquad 2^{10} = 8196bit$$

$$Nibel = \frac{1}{2}byts = 4bit$$

$1Mb = 1024Kb = 1024*1024*8 = 2^{23}$

$1Gb = 1024Mb = 2^{33}$

$1TB = 1024Gb = 2^{43}$

**واحدهای انتقال داده Transmission Rate**

$R = 1Bit / S\,\text{ec}$

$R = 1KBit / S\,\text{ec} = 1024Bit / S\,\text{ec}$

$R = 1Mb / \text{sec} = 1024KB / \text{Sec} = 1024*1024 = 2^{20} bit / \text{sec}$

$R = 1Gb / \text{Sec} = 1024MB / \text{Sec} = 2^{32} bit / \text{sec}$

**واحدهای سیگنالی**

$BW = 1KHz = 1000HZ$

$BW = 1MHZ = 1000KH\,2 = 10^6 HZ$

$BW = 1GHZ = 1000MHZ = 10^9 HZ$

**رابطه شانون**

این رابطه نشان می‌دهد که با توجه به نسبت سیگنال به نویز و پهنای باند کانال فیزیکی داده نرخ ارسال داده چند بیت در ثانیه می‌باشد و بصورت زیر است:

$$R = BW.L\,\text{og}_2\left(1+\frac{S}{N}\right)$$

$$\downarrow \quad \downarrow$$

$$\frac{bit}{\text{sec}} \quad H_2$$

مثال: یک شرکت ISP به مشتریان خود پهنای باند $512kb/\text{sec}$ تخصیص می‌دهد. در این صورت یک مشترک 1MB داده را در چند ثایه می‌تواند بفرستد؟ ثانیاً اگر کانال فیزیکی استفاده شده دارای نسبت سیگنال به نویز ۱۰۰۰ می‌باشد پهنای باند کانال فیزیکی چند هرتز است؟

حل:

$$R = 512Kb/\sec = 512*1024bit/\sec$$
$$M = 1Mbyte = 1024KB = 1024*8bit$$
$$\Rightarrow \frac{1\sec}{y} = \frac{Rbit}{M} \Rightarrow \frac{1}{y} = \frac{512*1024}{1024*1024*8} \Rightarrow y = \frac{1\times 16}{1}$$

پس ۱۶ ثانیه طول می‌کشد تا 1MB داده Send یا Receive شود.

قسمت ب: گفته شده که S/N برابر ۱۰۰۰ است.

$$R.BW.Log_2^{(1+S/N)} \Rightarrow 512*1024 = BW.Log_2^{(1+1000)}$$

برای حل می‌گوییم ۱۰۰۱ بین کدام دو عدد توان دو است (چون لگاریتم در پایه ۲ است) می-بینیم بین $2^9, 2^{10}$ است، $512\langle 1001\langle 1024$ پس به 1024 نزدیک‌تر است و حدوداً آن را $2^{10}$ می‌گیریم پس در نهایت $\log_2^{2^{10}}$ 10 و خواهیم داشت:

$$512*1024 = BW*10$$
$$\Rightarrow BW \approx 3120/1000 = 5/1$$

مثال: در یک کانال داده پهنای باند برابر 2MHZ و نسبت سیگنال به نویز 50db است. در اینصورت 100Mbyte داده در این کانال چند ثانیه برای ارسال نیاز دارد؟

حل: اول باید S/Nرا بدست بیاوریم.

$$xdb = 10\log_{10}^{S/N} \Rightarrow 50 = 10\log_{10}^{S/N} \Rightarrow 100000$$
$$2^{17} \approx 2^{16/5} = 7$$ تقریبا

$$R = 2\times 10^6 \times \log(1+100.000) \Rightarrow R = 2\times 10^6 \times 17 = 34\times 10^6 bit/\sec$$
$$(2^{16})65534\langle 100001\langle 131000(2^{17})$$

۳۴ میلیون بیت در ثانیه ارسال می‌شود.

اگر در صورت سوال Mb خواست $= \frac{34\times10^6}{1024\times1024} \Leftarrow 34Mbit/\sec$

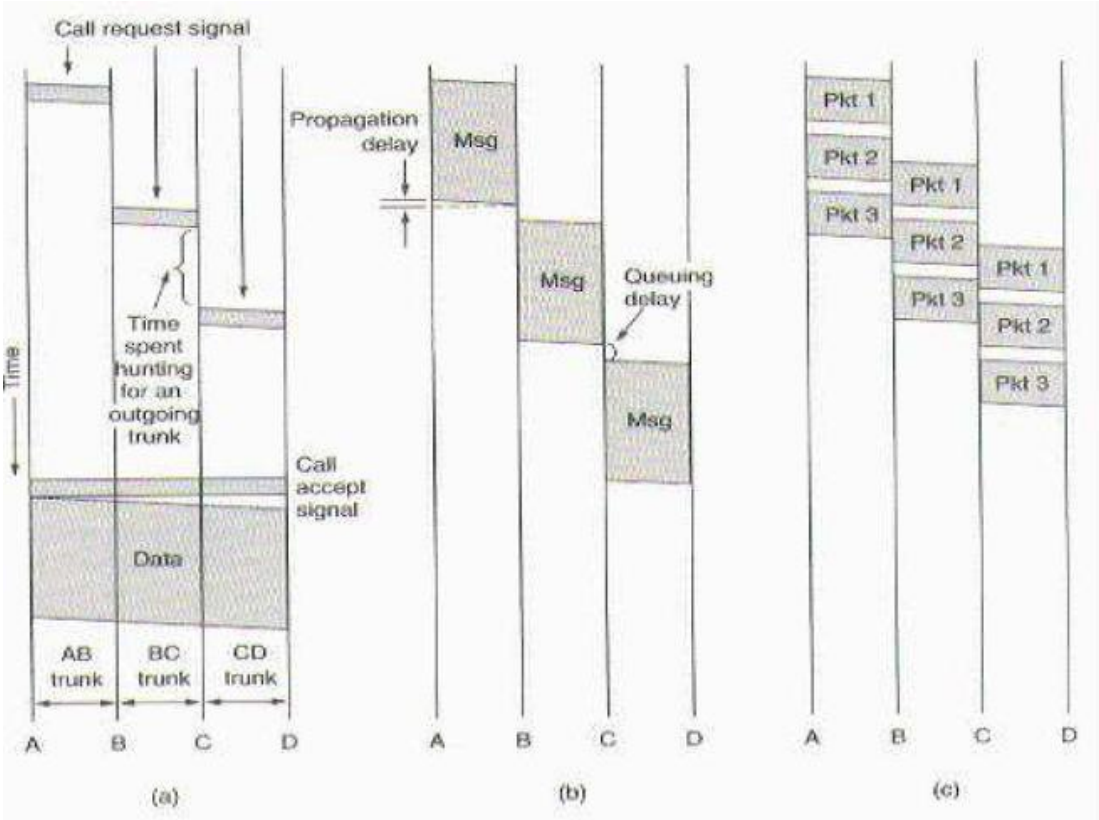
W ارسال ۱۰۰ مگابایت داده چقدر زمان نیاز دارد؟ (حل)

$$\frac{1\sec}{y} = \frac{34\times10^6}{100\times1024\times1024\times8} = \frac{17}{400} \Rightarrow y = \frac{400}{17} = 23.5\sec$$

---

**سوئیچینگ پیام**: حتما مبدا و مقصد باید آنلاین باشند و کل اطلاعات را در قالب یک پیام ارسال می‌کند. این روش هنگامی خوب عمل می‌کند که بین تمام گره‌های شبکه ارتباط وجود داشته باشد (بدون واسطه) و نویز هم وجود نداشته باشد.

**سوئیچینگ بسته:** قویترین روش موجود بوده و در اینترنت استفاده می‌شود. در این روش داده به پکت‌هایی با اندازه یکسان تقسیم می‌شود.




در این شکل A مبدا، D مقصد و C, B بافر هستند.

در حالت b، در هر بافر کل پیام باید خوانده شود و سپس به بافر بعدی فرستاده شود. در حالت C همزمان با ارسال پکت ۱، پکت ۲ گرفته می‌شود و در واقع عمل Overlap باعث افزایش سرعت می-شود.

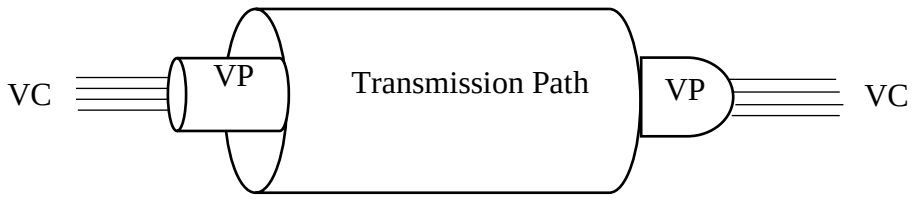
سوئیچینگ سلول در شبکه‌های تلفن همراه نیز استفاده می‌شود.

۵ بایت از بسته ها سرآیند است.

**تعریف TP**

مسیر انتقال است که شامل مجموعه‌ای از کابل و کانال‌های فیزیکی جهت انتقال داده می‌باشد. هر مسیر انتقال شامل چندین مسیر مجازی است (VP). همچنین هر مسیر مجازی متعلق به یک شرکت یا سازمان است که از طریق اتصالات سوئیچی کانال‌های فیزیکی بین مبدا و مقصد این مسیر ایجاد می‌شود و با پایان یافتن آبونمان مسیر مجازی از بین می‌رود. همچنین هر مسیر مجازی دارای چندین مدار مجازی (Virtual Circuit) می‌باشد. هر مدار مجازی یک زیرمجموعه از یک شرکت است که در واقع یک مسیر مداری یعنی اتصال الکتریکی با کمک سوئیچ را بین مبدأ و مقصد پدید آورده است. علت گفتن مدار مجازی این است که این مدار از طریق سوئیچ‌های میانی و با تنظیم آنها برقرار گردیده است و با OFF نمودن سوئیچ‌های میانی مدار ایجاد شده از بین می‌رود به این جهت به آن مدار مجازی می‌گویند.

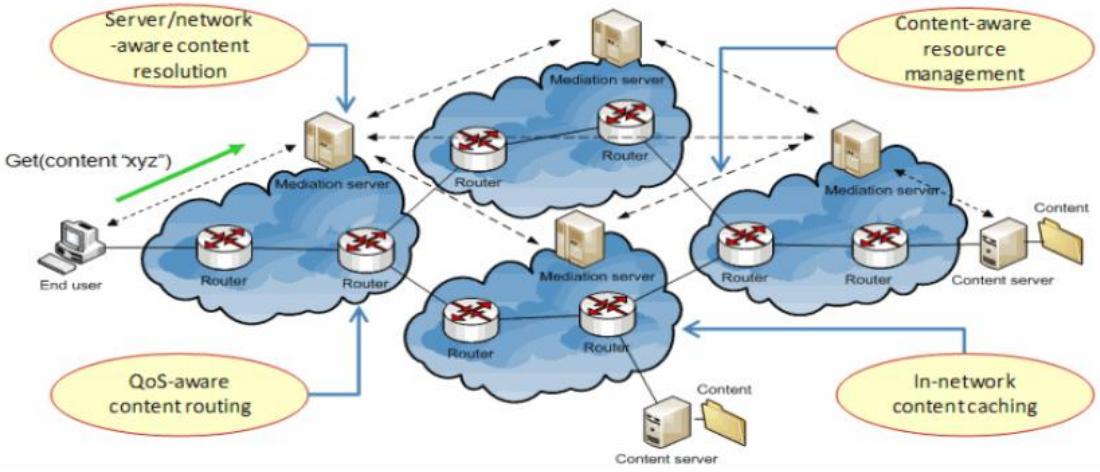
در مقابل این مفهوم، مدار واقعی یا Actual Circuit است که دارای یک مدار یا کانال فیزیکی و اتصال دائمی بین مبدأ و مقصد است.



## «جلسه سوم»

### شبکه‌های اتصال‌گرا

در این شبکه‌ها تمامی بسته‌ها باید به موقع، مرتب و سالم به مقصد برسند (تضمین شده). در شبکه‌های غیراتصال‌گرا هدف ارسال بسته‌هاست و اصراری بر روی کیفیت و سلامت و مرتب بودن بسته‌ها نیست. بسته‌ها بر روی شبکه پخش می‌شوند و ممکن است از روترهای مختلف عبور کنند. بعنوان مثال TCP مشابه پست پیشتار و UDP مشابه پست معمولی است و یا TCP مشابه وایبر و UDP مشابه ایمیل. در شبکه‌های اتصال‌گرا یک مسیر ایجاد و رزرو می‌شود و پس از ارسال آخرین پکت این مسیر نابود می‌شود.

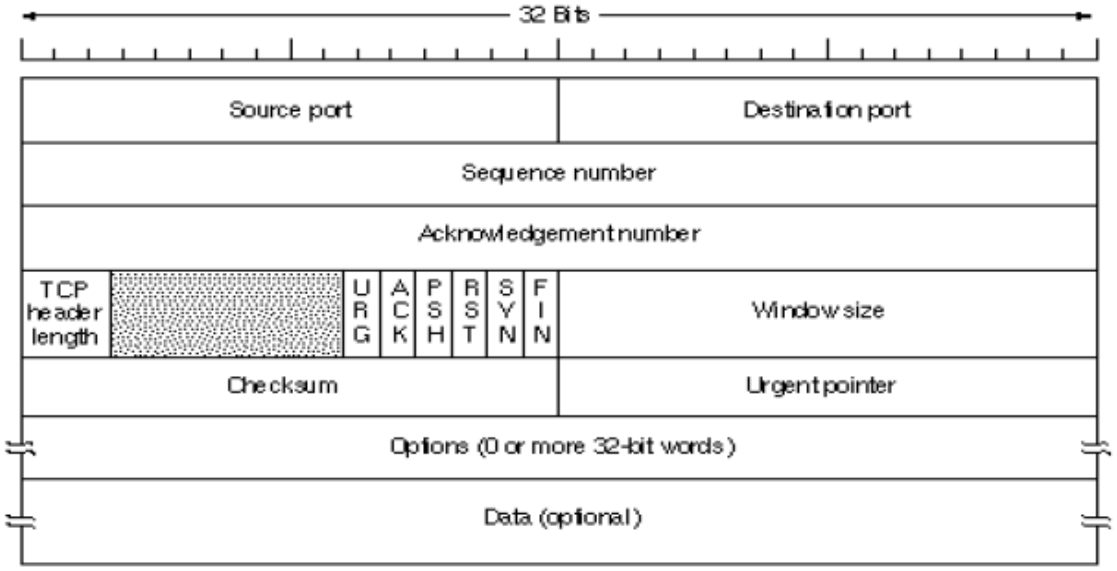



در شبکه‌های غیراتصال‌گرا پکت‌ها از مسیریاب‌های مختلف مسیردهی می‌شوند و دیگر مسیر رزرو شده‌ای برای بسته‌ها وجود ندارد. اما مزایای دیگری دارد ازجمله هزینه کمتر و ترافیک کمتر (ترافیک به دلیل نبود یک مسیر رزرو شده بین نودهای مختلف پخش می‌شود). چون ACK نداریم ترافیک کمتر است، در کاربردهای Real Time مثل موسیقی زنده، ویدئوی زنده و ... استفاده می‌شود.

**پروتکل RTP (Real Time Protocol) (زمان واقعی)**

- ✓ برای پخش موسیقی، فیلم و ... بصورت زنده بکار می‌رود.
- ✓ به سه لایه اول اصطلاحاً backbone می‌گویند که با سخت افزار رابطه‌ی قوی دارند. لایه‌ی ۴ میانی و لایه‌های بعد از آن ارتباط با کاربر دارند.
- ✓ در سه لایه آخر واحد خاصی نداریم، داده‌ها بصورت جریانی از داده‌ها عبور می‌کنند (سه لایه آخر منظور سه لایه‌ی بالاست)

در شکل زیر ساختار یک قطعه‌ی TCP را میبینیم. اندازه هر سطر ۳۲ بیت است. (اندازه هر سطر ۴ بایت). ۵ سطر اول اندازه مشخص دارند ولی دو سطر آخر اندازه‌شان نامعلوم است.




۵ سطر اول جز ثابت هر قطعه در پروتکل TCP است.

Destination Address, Source Port هر یک ۱۶ بیت هستند.

---

سوال: بطور کلی چند نوع آدرس دهی در شبکه داریم؟ با مثال بنویسید.

جواب:

۱) آدرس دهی فیزیکی (مک آدرس) : Mac Address مخفف Media Access Control که یک آدرس ۴۸ بیتی مانند آدرس زیر است.

$$FC-A8-CD-FF-10$$
$$\downarrow\downarrow \qquad\qquad \downarrow \quad \downarrow$$
$$1111,1100 \qquad\qquad 0001,0000$$

و بر روی کارت‌های شبکه (NIC) توسط شرکت سخت افزاری تولید کننده کارت شبکه تخصص می-یابد.

$$\text{NIC} \rightarrow \text{۶ بایت} \rightarrow \text{Hex ۱۲ عدد}$$

**تمرین ۴**

در مورد موضوع پرکاربرد Internet Of Things تحقیق نمائید و موارد کاربرد آن را بنویسید.

۸ بیت از این آدرس متعلق به شرکت و ۴۰ بیت سریال کارت است.

**۲) آدرس دهی منطقی:**

همان آدرس دهی IP در لایه شبکه می‌باشد که یک آدرس شامل ۴ عدد بصورت *x.y.z.w* است که هر عدد بین 0 تا 255 می‌باشد. هر دستگاهی که بخواهد به اینترنت متصل شود باید یک آدرس IP داشته باشد و آدرس IP شامل ۴ کلاس عمده‌ی D, C, B, A می‌باشد و هر کلاس شامل تعدادی زیر شبکه و تعدادی میزبان می‌باشد.

**تمرین ۵**

مشخصات انواع کلاس‌های IP را بطور کامل نوشته و برای هر یک مثال ذکر کنید. (مثلاً کلاس A چند زیر شبکه دارد هر یک چند میزبان دارد و .... )

**۳) آدرس دهی پورت یا پردازشی**

مربوط به آدرس دهی لایه‌ی انتقال می‌باشد. بدین معنا که هر می‌تواند به اندازه ۱۶ بیت پورت پردازشی داشته باشد (معادل ۶۵۵۳۵ پورت) که بطور همزمان قابل استفاده می‌باشد. مثلاً یک سرور می‌تواند از طریق پورت 8080 برای اینترنت و همزمان با پورت 215 از پروتکل FTP و از 80 با پروتکل http صفحات وب را دریافت کند.

فرض کنید قطعه TCP شامل ۲۱ بیت و checksum آن 8 بیت باشد. مقدار checksum کدام است؟

8 بیت 8 بیت جدا می‌شود: 00011000,01010101,01000011

سپس Xor می‌شود:

$$\begin{array}{r} 01000011 \\ 01010101 \\ \hline 00011000 \end{array} \rightarrow checksum$$

ارف یـلد

**تمرین ۶**

روش‌های دیگر چک کردن داده مثل CRC، همینگ (hamming) و ماتریس parity را بررسی کرده و مثال‌هایی برای هر یک ذکر کنید.

حداکثر اندازه قطعه می‌تواند به اندازه بافر باشد که حداکثر $۲^{۱۶} \Leftarrow$ ۶۵۵۳۵ بایت معادل ۶۴ کیلوبایت است.

حداکثر ۶۵۵۱۵=۲۰-۶۵۵۳۵ بایت دیتا در یک قطعه TCP است. (حالتی که حداقل میزان هدر (۲۰) را داریم) اگر مثلاً سرآیند ۶۰ باشد آنگاه ۶۰-۶۵۵۳۵ حالت داریم. پس:

$$0 \le Data \le 65535 \rightarrow TCP = 5$$

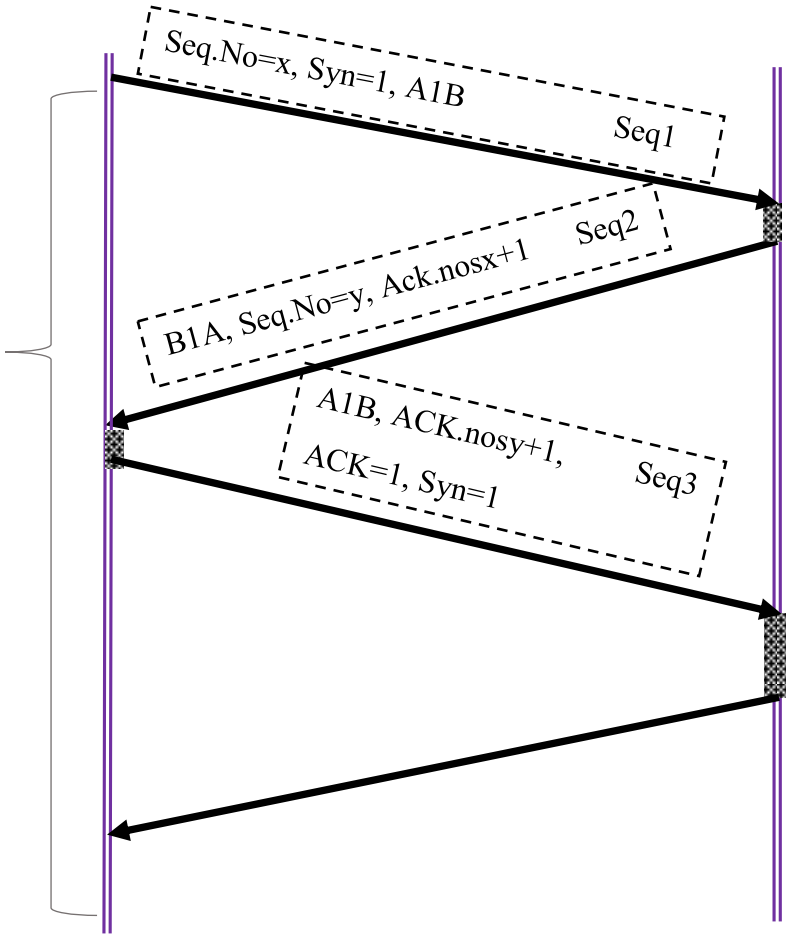
**روش دست تکانی سه مرحله‌ای 3Steps Hand Shaking**

این روش برای فاز تنظیم ارتباط استفاده می‌شود، یعنی زمانی که ماشین مبدا می‌خواهد با ماشین مقصد ارتباط برقرار کند (مثلاً Client با سرور)

نکته: بطور کلی برای تبادل داده‌ها ۳ مرحله داریم: مرحله ۱: فاز برقراری ارتباط (Setup Phase) ۲: فاز تبادل داده Transmission Phase ۳: فاز قطع ارتباط Disconnect Phase

**فاز برقراری ارتباط:**

عملکرد روش دست تکانی سه مرحله‌ای بصورت زیر است:




**SetUp**

**Phase**

**Seq. No*

*یک عدد تصادفی است.*

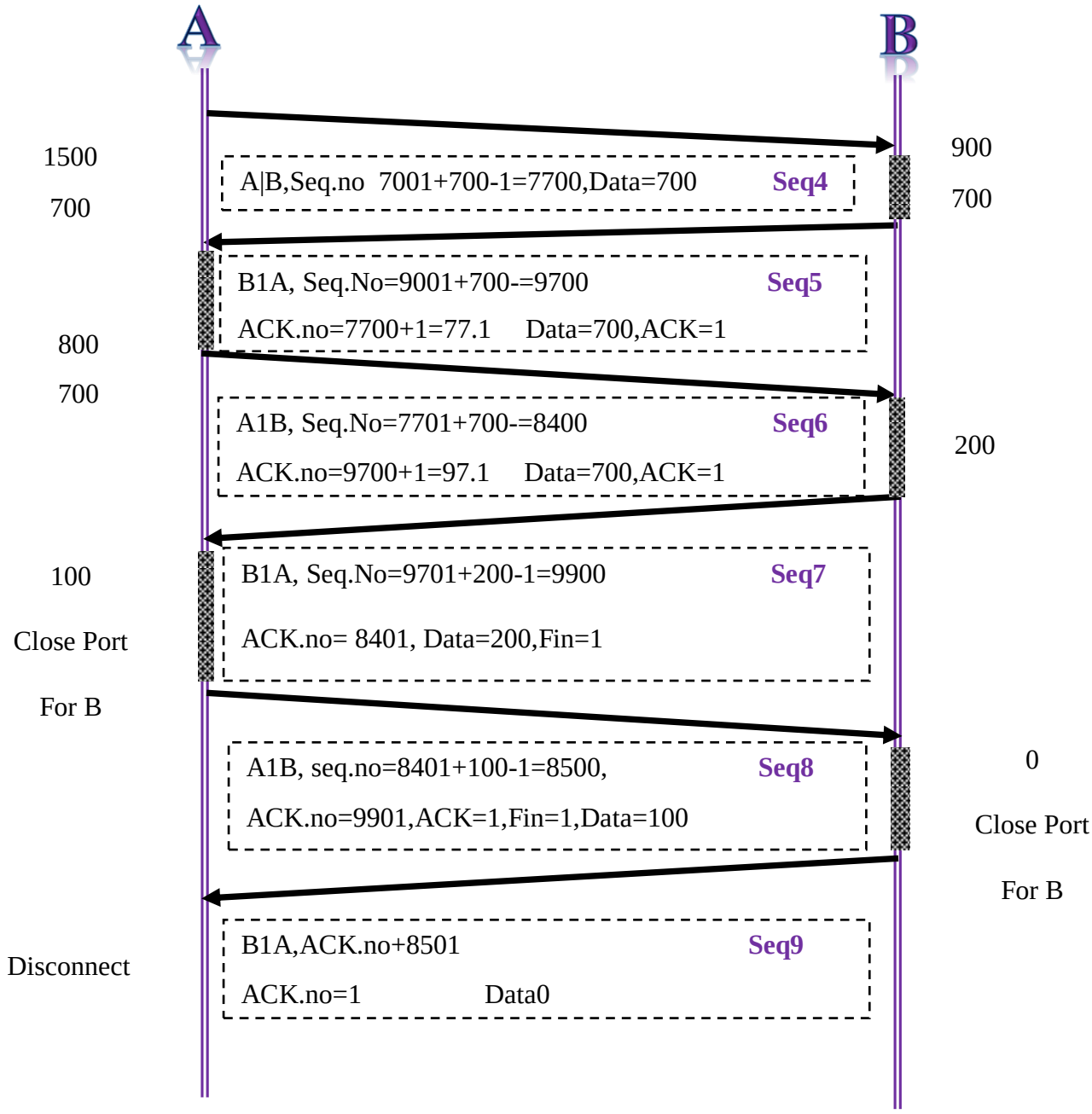
**A|B*

*یعنی مبدأ A و مقصد B است.*

*Y عددی تصادفی بین ۰ تا ۳۲ بیت است.*

مثال: فرض کنید بین دو ماشین B, A تبادل داده انجام هیم. حداکثر اندازه داده در قطعه TCP برای هر دو طرف ۷۰۰ بایت درنظر گرفته شده است. (WinSize=700). ماشین A، ۱۵۰۰ بایت و ماشین B، ۹۰۰ بایت داده برای ارسال دارند. در این صورت اگر عدد مصادفی ماشین A برابر ۷۰۰۰ و عدد تصادفی ماشین B برابر ۹۰۰۰ باشد کلیه عملیات در سه فاز Setup، انتقال و خاتمه را ترسیم نمائید.

پاسخ: فاز اول مشابه شکل بالاست با این تفاوت که x اول برابر ۷۰۰۰، y دوم ۹۰۰۰ و ACK سوم که در شکل بالا Y+1 است برابر ۹۰۰۰+۱ خواهد بود. ادامه‌ی شکل:




**تمرین ۷**

فرض کنید ماشین A, B دارای Winsize=920 باشد (۲۰ بایت سرآیند) اگر ماشین A، ۱۹۰۰ بایت و ماشین B، ۲۸۰۰ بایت داده ارسال کنند، کلیه فازهای Setupرا ترسیم نمایید. (فرض کنید x=۸۰۰۰ و y=۴۰۰۰ باشد)

## «جلسه چهارم»

**توضیح فیلدهای سرآیند قطعه‌ی TCP/IP (TCP Segment Forment)**

۱) Source Port: پورت مبدأ (پورتی که از آنجا قطعه ارسال می‌شود) ۱۶ بیت است.

۲) Destination Port: پورت مقصد ۱۶ بیت است.

۳) Sequence Number: شماره ترتیب ۳۲ بیت است یعنی $۲^{۳۲} \Leftarrow$ ۴ میلیارد حالت دارد. تصادفی است.

۴) Acknowledgment Number: شماره وصول

۵) STCP Header Lenght: چهار بیت است. عدد داخل این فیلد تعداد دابل بورد یا ۴ بایت را نشان می‌دهد، یعنی عدد آن باید در ۴ ضرب شود تا مقدار بایتی بدست آید چون طول پکت حداقل ۲۰ بایت است پس این فیلد مقدار ۵ را دارد و چون چهار بیتی است حداکثر ۱۵×۴=۶۰ حالت دارد.

۶) قسمت خالی: ۶ بیت خالی است که برای اهداف استراتژیک استفاده می‌شود. (۶۴ حالت)

۷) فلگ URG: اگر یک باشد یعنی فیلد Urgent Pointer مقدار معتبری دارد و باید خوانده شود و بالعکس.

۸) فلگ ACK: اگر این فیلد یک باشد یعنی فیلد Acknumber مقدار معتبری دارد و باید خوانده شود.

۹) فلگ PSH (Push): اگر این فیلد یک باشد یعنی مسیریاب این بسته را به سرعت در بافر خروجی بگذارد و در صف قرار ندهد.

۱۰) فلگ RST: این فلگ اگر یک باشد یعنی کامپیوتر مبدأ یکطرفه می‌تواند ارتباط را قطع کند (به دلایل امنیتی)

۱۱) فلگ SYN: اگر این بیت یک باشد یعنی کامپیوتر مبدأ می‌خواهد با کامپیوتر مقصد ارتباط برقرار کند (یعنی در فاز Setup قرار می‌گیرد)

۱۲) فلگ Fin: اگر این فلگ یک باشد یعنی قطعه ارسالی این کامپیوتر آخرین قطعه است و دیگر قطعه‌ای وجود ندارد.

۱۳) Window Size: اندازه پنجره یا میزان بافر آزاد این کامپیوتر را نشان می‌دهد. عدد آن بین ۰ تا ۶۵۵۳۵ است. (این فیلد موجب کنترل جریان داده می‌شود)

۱۴) Checksum: این فیلد استفاده می‌شود تا سلامت قطعه را مشخص کند، یعنی محتوای سرآیند قطعه TCP بصورت دسته‌های ۱۶ بایتی زیر هم قرارگرفته و Xor می‌شود. نتیجه به عنوان Checksum ذخیره شده و قطعه ارسال می‌گردد. در مقصد این مقدار مجدداً محاسبه می‌شود و با عدد Checksum ذخیره شده مقایسه می‌گردد. اگر یکسان بود یعنی خطایی رخ نداده است.

۱۵) Urgent pointer (اشاره گر ضروری): اگر پرچم Urg یک باشد یعنی این فیلد محتوای معتبر دارد و به نقطه‌ای از متن داده‌های قطعه اشاره می‌کند که داده‌های مهمی در آنجا هستند.

۱۶) Option: (داده‌های اختیاری): اگر در هدر ۴ باشد این مقدار صفر و اگر ۶۰ باشد آنگاه ۴۰=۲۰-۶۰ است. پس این مقدار این فیلد بین صفر تا ۴۰ است $(0 \le option \le 40)$

۱۷) Data: محلی که داده‌ها از آن شروع می‌شود.

بطور کلی در سرآیند قطعه ۷ سطر داریم که ۵ سطر اول ضروری هستند. هر سطر ۴ بایت است.

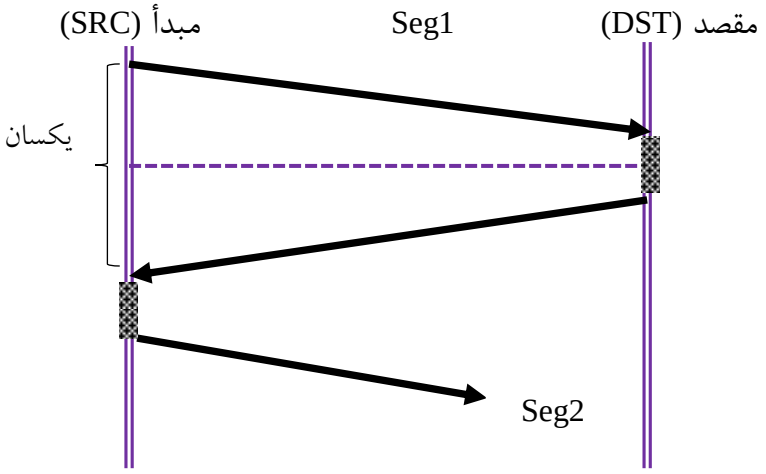
این قطعه روی لایه Transport قرار دارد و از پروتکل TCP که اتصال‌گرا است، استفاده می‌کند.

## «جلسه پنجم»

ساده‌ترین روش ارسال ACK در پروتکل TCP آن است که هر قطعه که ارسال شده فرستنده به انتظار ACK می‌ماند و سپس قطعه بعدی را ارسال می‌کند. مشکل این روش یکی تاخیر در ارسال داده‌ها به علت انتظار برای ACK و دگیری تحمیل ترافیک مضاعف به شبکه به دلیل تولید یک ACK به ازای هر ارسال است.

وضعیت‌های مختلف ارسال قطعه و دریافت ACK:

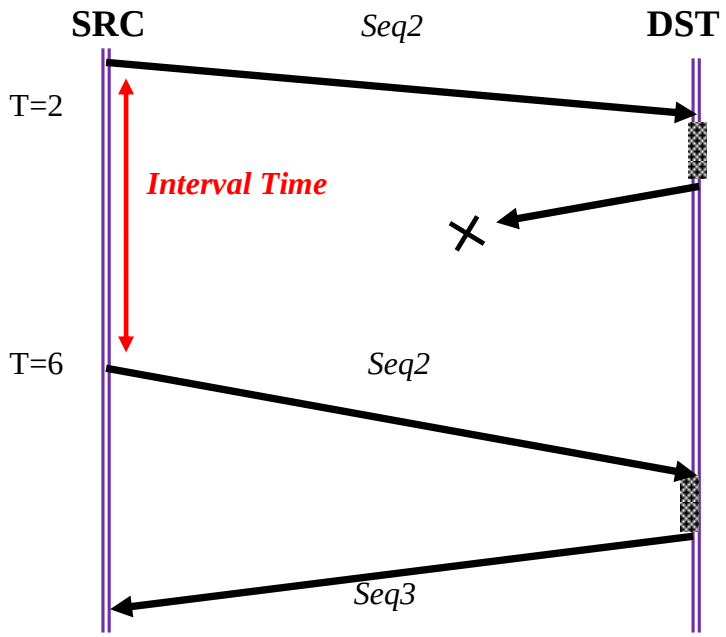
روش نرمال: در این روش فرستنده قطعه‌ی یک را می‌فرستد، سالم به مقصد می‌رسد، مقصد ACK آن را تولید می‌کند و برای ماشین مبدا می‌فرستد و مبدأ ACK1 را دریافت کرده و سپس قطعه‌ی بعدی را ارسال می‌کند. نکته‌ای که وجود دارد آنست که زمان رفت و برگشت تقریباً برابر است.



نکته: معمولاً فرستنده زمان رفت و برگشت را بطور تخمینی می‌داند، مثلاً اینکه از لحظه ارسال تا زمان دریافت ACK قطعه ۴ ثانیه طول می‌کشد. (۲ ثانیه رفت و ۲ ثانیه برگشت) بنابراین هنگام ارسال یک قطعه زمان سنج یا کرنومتر خود را فعال کرده (با شروع از ۴) تا اینکه به صفر برسد. در این مدت ACK باید برسد. اگر ACK نرسید به وضعیت‌های زیر می‌رسیم.

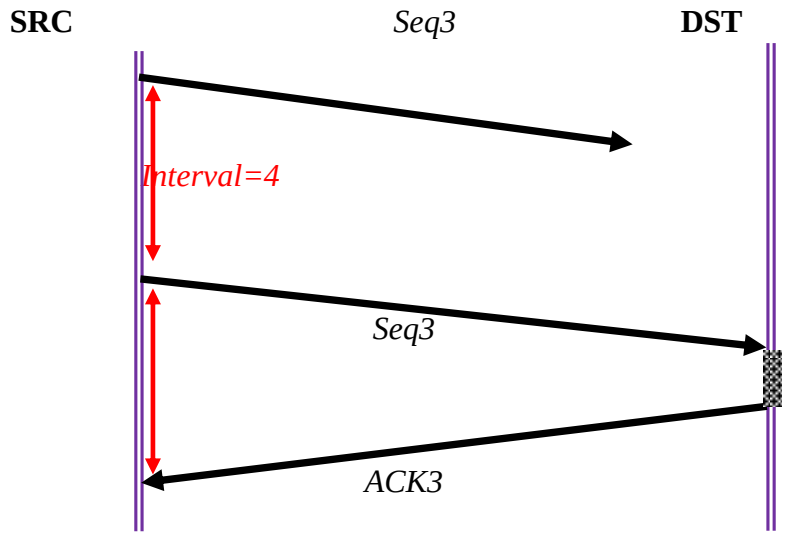
۲) در این وضعیت فرستنده قطعه ۲ را فرستاده و به مقصد می‌رسد. مقصد قطعه را دریافت و ACK آن را تولید می‌کند اما ACK در بین راه از بین می‌رود. در این حالت فرستنده چون ACK را در بازه زمانی ۴ ثانیه نمی‌گیرد تصور می‌کند قطعه به مقصد نرسیده است لذا دوباره آن را ارسال می‌کند. مقصد با دیدن مجدد قطعه ۲ آن را از بین می‌برد و فقط ACK2 را مجدداً ارسال می‌کند.

اگر انتظار برای دریافت ACK بیش از حد طولانی درنظر گرفته شود، کاهش بازدهی و عدم استفاده از بهینه از منابع شبکه را خواهیم داشت:




در این وضعیت قطعه‌ی ۳ ارسالی در بین راه گم می‌شود و به مقصد نمی‌رسد. در این حالت مقصد هیچ اطلاعی از موضوع ندارد و بنابراین ماشین مبدأ پس از انقضای زمان interval و اینکه ACK آن نمی‌آید دوباره قطعه ۳ را می‌فرستد و گیرنده با دریافت آن ACK3 بازپس می‌فرستد.

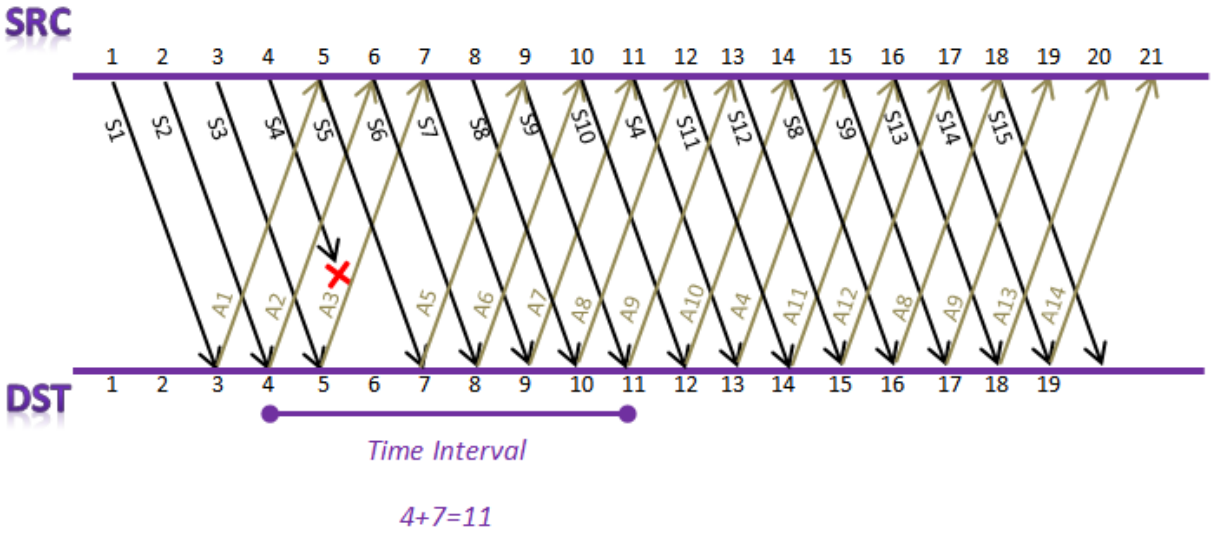
اگر بازه زمانی انتظار یا Interval کمتر از حد نرمال گرفته شود هدر رفت منابع، افزایش ترافیک زائد و ایجاد ازدحام خواهد شد.




مثال: فرض کنید که زمان انتقال مبدأ برابر ۱۰ میلی ثانیه می‌باشد. در اینصورت اگر زمان رفت و برگشت ۴ ثانیه باشد، فرستنده بدون عجله برای دریافت ACK چند قطعه راست سرهم می‌فرستد زیرا امیدوار است آنها بعداً برسند به این ترتیب بازدهی افزایش می‌یابد.

این روش را Selective repeat گویند.

***Internal = 7***

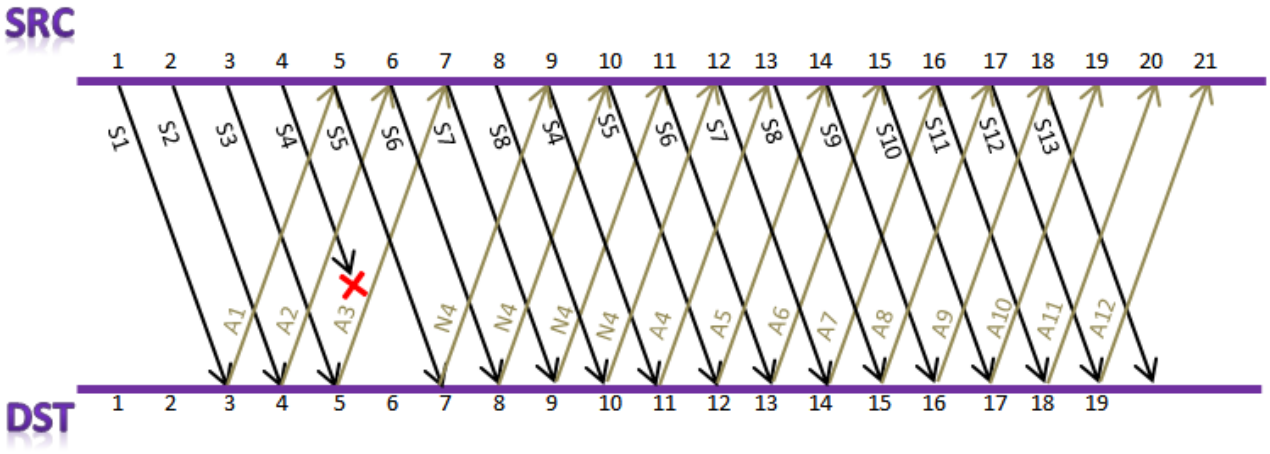


### روش Go back n

در این روش برخلاف روش قبلی در صورتی که از سوی فرستنده بطور مرتب قطعه‌ها دریافت نشود. مثلاً قطعه‌ی ۴ گم شود و قطعات ۵ و ۶ و بعدی بیاید، تمام قطعات بعدی بزرگتر از ۴ را نابود کرده (DROP) تا اینکه فرستنده متوجه شود و قطعه ۴ بفرستد. از طرفی فرستنده مجبور است قطعات ۵ و ۶ غیره را نیز که قبلاً فرستاده بود، بعد از ارسال ۴ مجدداً ارسال کند. علت این امر آن است که در این روش، گیرنده بافر با حجم زیاد ندارد و هر بار یک قطعه‌ی مرتب را انتظار دارد.

### روش Selective Repeat

در این روش فرض این است که هر دو ماشین مبدأ و مقصد بافر زیاد دارند و می‌توانند قطعات را حتی در صورت نامرتب بودن در خود نگهداری نموده و پس از دریافت همه قطعه‌ها آنها را مرتب کند. در این روش فرستنده ابتدا بدون توجه به ACK ؟؟؟ قطعات ارسالی آنها را می‌فرستد و برای هر قطعه یک Time Internal درنظر می‌گیرد. اگر پس از انقضای آن مدت ACK قطعه نیامد، در اینصورت مجدداً همان قطعه ارسال می‌شود و به ارسال بقیه قطعات نیز ادامه می‌دهد زیرا مقصد قطعات بعدی آن را در خود ذخیره و بافر نموده است و با دریافت قطعه گم شده ACK آن را می‌فرستد و آن را در بافر مرتب می‌کند.



## تمرین ۸

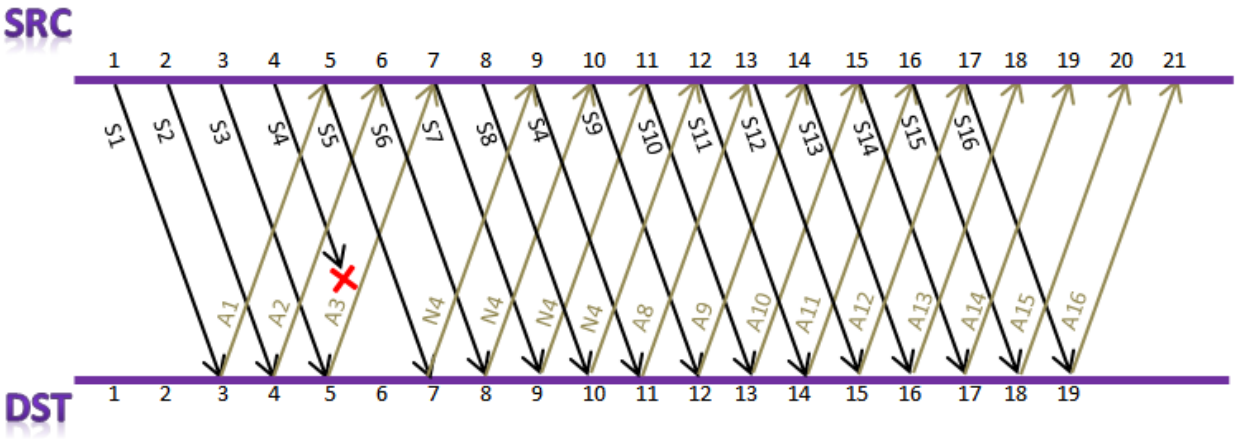
می‌خواهیم با استفاده از روش Selective Repeat و با Time Internal تعداد ۱۸ قطعه S1…….S18 را بفرستیم. زمان رفت ۲ ثانیه و زمان برگشت ۲ ثانیه محاسبه شده است. در اینصورت اگر در بین راه قطعات 9, 8, 5 گم شوند چگونه این روش این خطاها را مدیریت می‌کند؟ اگر در مسئله فوق از روش Go Back n استفاده شود چگونه خطاها برای قطعات گم شده مدیریت می‌شوند؟ زمان را از ۱ تا ۳۵ درنظر بگیرید.

-این مسئله را با روش Selective Repeat بهبود یافته حل کنید.

### روش Selective Repeat اصلاح یافته

بهتر است به منظور جلوگیری از تأخیر ارسال قطعه ۴ گم شده گیرنده با دریافت S5 در مثال قبلی، سریعاً Nak4 را بفرستد. همچنین با دریافت Nak4, S8, S7, S6 را دوباره تأکید کند. فرستنده با اولین دریافت S4, Nak4 را بفرستد اما به ادامه روال طبیعی خود بپردازد.

---

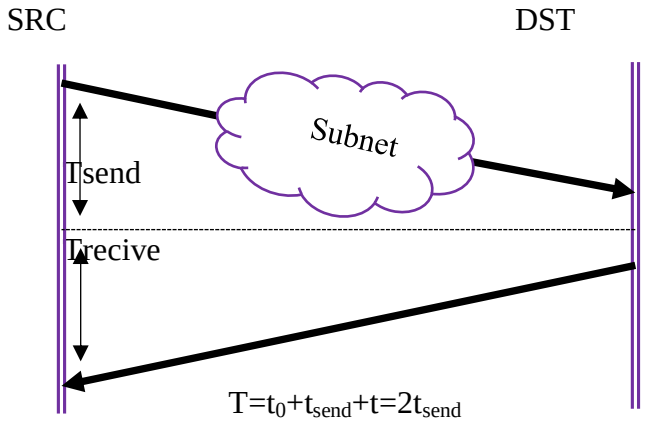


در این مثال گیرنده با دریافت S4 چون S8, S7, S6, S5 را دریافت کرده است به مبداء فقط ACK8 را می‌فرستد و این موضوع مؤید آن است که ماشین مبدأ نگران نباشد و بداند که قطعات S8, S7, S6, S5, S4 همه با هم ACK شده‌اند.

## «جلسه ششم»

مکانیزم ارسال مجدد (RTO ⇐ *Retransmission Time Out*)

در مکانیزم ارسال مجدد تلاش بر این است که بتوانیم در شرایط مختلف ترافیکی، زمان رفت و برگشت نسبتاً دقیقی را تخمین بزنیم بطوری که اگر ACK بسته‌ی ارسالی بعد از انقضای آن مدت زمان نیامد، یعنی اینکه بسته در بین راه گم شده است و لذا ماشین مبدأ می‌بایست مجدداً آن بسته‌ی داده را بفرستد. اما مسئله‌ی مهم تخمین دقیق RTO یا همان زمان ارسال مجدد است که سه حالت را می‌توان برشمرد.




الف) اگر RTO بسیار کوچکتر از زمان رفت بسته و برگشت ACK آن باشد (مثلاً زمان RTO ی تخمینی ۶ ثانیه و زمان رفت و برگشت بسته و ACK آن ۱۰ ثانیه باشد) نتیجه بوجود آمدن ترافیک زائده شبکه، اشغال پهنای باند و کاهش بهره‌وری موثر از منابع شبکه و حتی گاهی ازدحام گلوگاه و از کار افتادن سرور می‌شود.

ب) اگر RTOی تخمینی بسیار بیشتر از زمان رفت و برگشت بسته و ACK آن درنظر گرفته شود مشکلات زیر را خواهیم داشت: نارضایتی مشتری که با تأخیر بسیار زیاد بسته را دریافت می‌کند.

همچنین عدم استفاد بهینه از پهنای باند شبکه و همچنین عدم استفاده بهینه از منابع شبکه. زیرا تعداد کلیه بسته‌ها بسیار کمتر از ظرفیت واقعی شبکه ارسال می‌شود.

به منظور تخمین دقیق RTO، پارامترهای زیر را درنظر می‌گیریم:

۱) Sample RTT: یعنی زمان رفت و برگشت بسته و ACK آن در لحظه فعلی.

۲) Estimated RTT: زمان رفت و برگشت تخمینی است و زمان متوسط رفت و برگشت را نشان می‌دهد.

۳) Deviation: به معنای انحراف معیار است.

۴) Alpha: آلفا را فاکتور فراموشی نیز می‌گویند و معمولا برابر ۲ درنظر گرفته می‌شود. این ضریب نشان می‌دهد که مثلاً میانگین RTT چقدر به عدد میانگین قبلی و چقدر به عدد بدست آمده‌ی RTT فعلی وابستگی یا تأثیر پذیری دارد.

نکته: آلفا قابل تنظیم است.

برای محاسبه‌ی RTT روابط زیر را داریم:

1) $\text{Estim RTT}_{new}=(1-\alpha)\ \text{Estim RTT}_{old}+\alpha_0 \text{Sample RTT}$
2) $\text{deviation}_{new}=(1-\alpha)\ \text{deviate}_{old}+\alpha_0\ (\text{Sample RTT-EstimRTT})_{new}$
3) $\text{RTO=Estim RTT}_{new}+\text{k.deviate}_{new} \qquad 3\langle k\langle 4$

---

مثال) فرض کنید در یک شبکه $\text{Estim RTT}_{old}=10$, $\alpha=0/2$ می‌باشد و $\text{Deviate}_{old}$ برابر ۰/۱ تنظیم شده است. مطلوبست:

الف) اگر Sample RTT برابر ۲۰ شود RTO کدام است؟ (K=4)

ب) اگر $\alpha=0/8$ فرض شود و RTO کدام است؟

حل:

$$EstimRTT_{new}(1-0.2)10+0.2.20 \Rightarrow 0.8\times10+0.2\times20$$
$$formula2: Devi_{new} = 0.2\times0.1+0.2\times8 = 1.68$$
$$formula3: RTO = 12+4\times1.68 = 18.72$$

پس RTO برابر 18.72 درنظر گرفته می‌شود. بعبارتی زمان انتظار برای دریافت ACK برابر 18.72 خواهد بود.

حل قسمت ب:

$$EstimRTT = (1-0.8)10+0.8.20 = 0.2\times10+0.8.20$$
$$\Rightarrow 16+2 = 18$$
$$Devi_{new} = 0.2\times0.1+0.8\times2 = 1.62$$
$$RTO = 18+4\times1.62 = 24.48$$

**تمرین ۹**

در صورتی که در یک شبکه Sample RTT=70, $EstimRTT_{old}$=40 باشد و K=4 مطلوبست:

الف) اگر ۰/۱ = $\alpha$ باشد RTO چند است؟

ب) اگر ۰/۵ = $\alpha$ باشد RTO چند است؟

ج) اگر ۰/۹ = $\alpha$ باشد RTO چند است؟

ثانیاً کدام آلفا بهینه است و چرا؟ Deviet=0.2

## «جلسه هفتم»

### الگوریتم Jacobson با شروع آهسته

این الگوریتم به منظور کنترل ازدحام و همچنین استفاده حداکثر از پهنای باند بکار می‌رود یعنی در ابتدا پس از برقراری بین ماشین مبدأ و مقصد قطعه‌ی TCP به حجم ۱ KB ارسال می‌شود. سپس اگر ACK آن آمد ۲ KB بعد ۴ KB بعد ۸ KB و ... هر بار در واقع دو برابر اندازه بایت قبلی داده ارسال می‌شود و این موضوع تا زمانی ادامه می‌یابد که مقدار داده‌ی بایتی از آستانه (Threshold) بیشتر نشود. زمانی که اندازه بایت قطعه به آستانه رسید از آن پس هر بار ۱ KB به مقدار قبلی اضافه می‌شود، تا اینکه در نهایت به یک ازدحام واقعی برخورد کنیم؛ یعنی مقدار K بایت ارسال شده و ACK آن نیاید. در این حالت ازدحام رخ داده است و مقدار جدید ازدحام برابر K/2 و مقدار جدید اندازه‌ی قطعه برابر ۱ KB انتخاب می‌شود و از واحد زمانی بعدی مانند ابتدای روش ۱ KB پس ۲ KB و ۴ KB و ... ادامه می‌یابد.

البته این نکته قابل توجه است که اندازه‌ی قطعه‌ی TCP ارسالی نباید از مقدار بافر آزاد مقصد (Winsize) بیشتر شود.

در هر مرحله اندازه‌ی قطعه ارسالی به صورت زیر تعریف می‌شود:

$$t : S : Size - TCP - Seg, threshold = x$$
$$t+1: \qquad , WinSize = w$$
$$if\ (2 \times S \langle thresh\ \&\ 2 \times 5 \langle WinSize)$$
$$S \leftarrow S \times 2$$
$$else \qquad if\ (S \rangle = thresh\ \&\ S \langle WinSize)$$
$$S \leftarrow S + 1$$
$$else \qquad if\ (S = \ = WinSize$$
$$S \leftarrow S$$

اگر S بایت فرستادیم ولی نرسید:

T: send S byte

Congestion Nack

Threshold $_{new}$=S/2;

T+1: S $\leftarrow$ 1

اگر $S(t)\langle thresh, WinSize$

$$if S(t) = size\ of\ TCP\ \text{int}\, ime\ step\ t$$
$$\Rightarrow t+1: S(t+1) = Min(2 \times S(t), threshold, winsize$$

اگر $St \geq threshold$:

S(t+1)=Min (S(t)+1, Winsize)

---

مثال: فرض کنیم یک ماشین مبدأ و مقصد فاز تنظیم ارتباط را انجام داده و با الگوریتم جکوبسون با شروع آهسته داده‌ها ارسال می‌شوند. مقدار آستانه‌ی ازدحام پیش فرض برابر ۸۳ و ۸۶ = winsize مشخص گردیده است. اگر در Step زمانی ۱۸ ازدحام رخ دهد نمودار جکوسون را تا Step زمانی ۳۲ ترسیم نمایید.

حل: اعداد را بصورت جدولی می‌نویسیم:

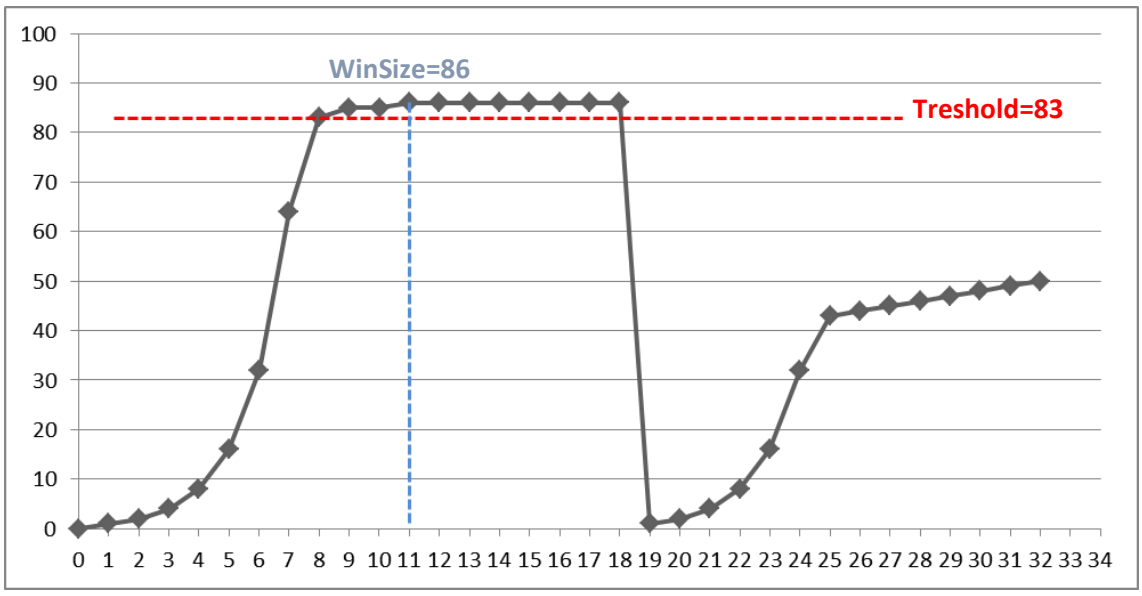
| *t* | *S(t)* |
|---|---|
| 1 | 1 |
| 2 | 2 |
| 3 | 4 |
| 4 | 8 |
| 5 | 16 |
| 6 | 32 |
| 7 | 64 |

| *t* | *S(t)* |
|---|---|
| 8 | 83 |
| 9 | 85 |
| 10 | 85 |
| 11 | 86 |
| 12 | 86 |
| 13 | 86 |
| 14 | 86 |

| *t* | *S(t)* |
|---|---|
| 15 | 86 |
| 16 | 86 |
| 17 | 86 |
| 18 | 86 |
| 19 | 1 |
| 20 | 2 |
| 21 | 4 |

| *t* | *S(t)* |
|---|---|
| 22 | 8 |
| 23 | 16 |
| 24 | 32 |
| 25 | 43 |
| 26 | 44 |
| 27 | 45 |
| 28 | 46 |

| *t* | *S(t)* |
|---|---|
| 29 | 47 |
| 30 | 48 |
| 31 | 49 |
| 32 | 50 |




**تمرین ۱۰**

با استفاده از پروتکل TCP/IP و با الگوریتم جکوبسون با شروع آهسته ماشین مبدأ در ۳۰ است زمانی داده‌ها را برای ماشین مقصد می‌فرستد. در ابتدا استانه ازدحام برابر Window Size=125, 121 است. اگر در واحد زمانی ۱۹ ازدحام رخ دهد نمودار جکوسون آن را رسم کنید.

ثانیاً اگر در مسئله فوق در ثانیه ۲۵ ام اندازه بافر به ۵۷ کاهش یابد نمودار آن چگونه خواهد شد؟

## صف اولویت دار

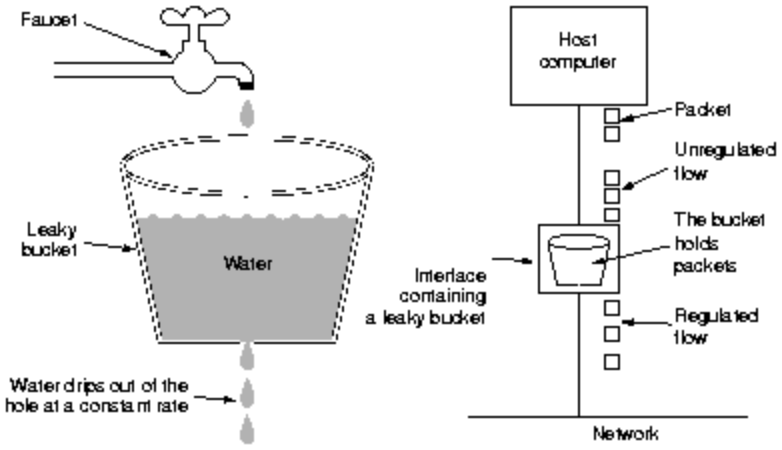
کیفیت خدمات یعنی نرخ قابل قبول و متوازنی در زمینه ترافیک ایجاد کنیم. در صف اولویت دار به بسته‌های مهم الویت بالاتر داده می‌شود.

*مشکل صف الویت دار: قحطی زدگی یا گرسنگی*

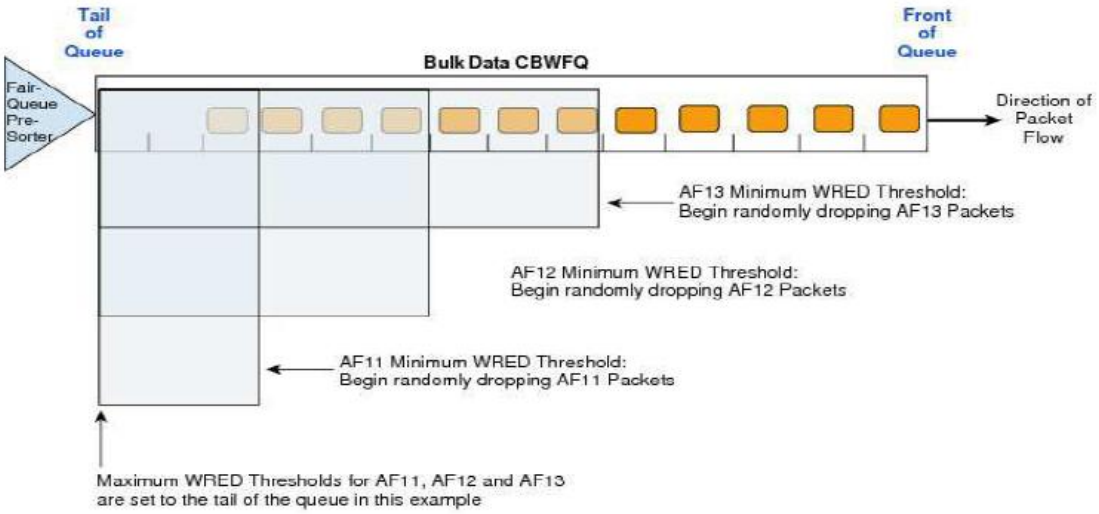
**FIFO**: بسته‌ها طبق الگوریتم FiFo سرویس دهی می‌شوند. این حالت عدالت را رعایت نمی-کند، مگر اینکه بسته‌ها همه هم الویت باشند.

## «جلسه هشتم»

### الگوریتم سطل سوراخ‌دار




این الگوریتم بر کنترل جریان تاکید دارد. اینکه هر چقدر هم ترافیک زیاد باشد و خروجی همواره نرخ مشخص و ثابت داشته باشد. در واقع کار تبدیل ترافیک با نرخ متغیر را به نرخ ثابت تغییر می‌دهد.




**انواع تأخیر در شبکه‌های ارتباطی**

۱) پردازشی ۲) انتظار در صف ۳) انتشار ۴) انتقال ۵) پهنای باند

از بین این موارد تأخیر انتشار اهمیت زیادی دارد.

تأخیر پردازش: اشاره به این موضوع دارد که چه هنگام طول می‌کشد بسته از پورت ورودی به روتر رفته، پردازش شده و به پورت خروجی که بر اساس مسیریابی انتخاب شده خارج شود. معمولاً این زمان خیلی کوچک است (در حد ۰/۰۱ ) بنابراین گاهی نادیده گرفته می‌شود.

**تأخیر انتظار در صف**

زمانی که بسته در صف روتر قرار دارد. فرمول آن: $Tqueue=k*t_{proc}$

**تأخیر انتشار**

زمانی که طول می‌کشد یک سیگنال از نقطه A زمین به B برسد. می‌دانیم نور نهایتا ۳۰ هزار کیلومتر سرعت در هر ثانیه دارد. پس در قطعات یک رایانه سرعت انتقال اطلاعات در حد نانو ثانیه خواهد بود. نحوه محاسبه تأخیر انتشار:

$$T_{prop} = \frac{d^{(m)}}{V^{(m/s)}}$$

مثال: فاصله‌ی یک ماهواره با ایستگاه زمینی ۶۰۰۰۰ km می‌باشد. در اینصورت اگر ایستگاه زمینی سیگنالی ارسال کند پس از چه مدت ACK آن را دریافت می‌کند؟

$V = C = 300000 \qquad km/s$ سرعت نور در خلأ

$$T_{Latency} = T_{send} + T_{proc} + T_{ret}$$

تأخیر پردازش را ۱ms. درنظر می‌گیریم. زمان رفت و برگشت یکسان است.

فرمول $T_{send}$ نیز $\frac{d}{v}$ است پس:

$$T_{Latency} = 2T_{send} + \frac{0.01}{1000}$$
$$\Rightarrow T_{send} = \frac{d}{v} = \frac{60000 \times 10^3}{300000 \times 10^3} \Rightarrow 2\frac{6 \times 10^7}{3 \times 10^8} + 0.1 \times 10^{-4}$$
$$T_{Latency} = 0.4 + 0.00001 = 0.4\sec \times 1000 = 400 m\sec$$

**تاخیر انتقال**

این زمان که گاهی با تأخیر انتشار به اشتباه یکی در نظر گرفته می‌شود، مدت زمانی است که یک فریم بطور کامل از ماشین فرستنده به گیرنده برسد. واحد آن فریم بر ثانیه است.

عوامل تأثیرگذار: تعداد سیگنال‌های یک فریم، زمان ارسال اولین بیت تا دریافت و آخرین بیت در مقصد در زمان تحلیل و پردازش و چک نمودن سلامت بیت‌ها و تشکیل فریم و تشخیص اینکه آن فریم دریافتی سالم آمده است. تأخیر انتقال از فرمول زیر محاسبه می‌شود:

F = طول بیتی فریم

$$T_{Trans} = \frac{F}{R} \qquad \frac{bit}{bit/\sec}$$

مثال: فریمی به طول 1KB را می‌خواهیم از یک کانال با نرخ ارسال ۵۰kb/sec ارسال کنیم. زمان تأخیر انتقال فریم کدام است؟

حل:

$$T_{trans} = \frac{F}{R} \qquad F = 1kbyte = 1024*8$$
$$R = 50\frac{Kbit}{\text{Sec}} = 50 + 1024bit/\sec$$
$$\Rightarrow T_{trans} = \frac{1024*8^4}{50+1024} = 0.16\text{Sec} = 0.16 - 1000 = 16ms$$

**بایت‌های حافظه‌ای**

$1B \rightarrow 8b$
$1KB \rightarrow 1024B = 8192b$
$1MB \rightarrow 1024KB = 1024 + 1024B = 1024 \times 1024 \times 8b$
$1GB \rightarrow 1024MB$
$1TB \rightarrow 1024GB$

**واحدهای نرخ ارسال**

$1bit / \sec$
$1kbit / \sec = 1024bit / \sec$
$1mbit / \sec = 1024 \times 1024bit / \sec$
$1Gbit / \sec = 1024Mbit / \sec$

---

مثال: فاصله بین دو ایستگاه زمینی ۱۲۰۰۰ کیلومتر و سرعت سیگنال در آن km/sec ۲۴۰۰۰۰ است. اگر فریم ما ۸ کیلوبایت باشد در این صورت ارسال KB ۱۰۰ داده چه مدت زمان طول می‌کشد؟ همچنین زمان تأخیر انتشار و تأخیر انتقال کدام است؟ (نرخ ارسال کانال برابر Kb/Sec است).

$d = 12000km$
$V = 240000km / \sec$
$R = 100Kbit / \text{Sec}$
$F = 8KByte \qquad 8*8*1024$
$M = 100KB \qquad \rightarrow Tm = ? \qquad T_{trans} = ?$
$T_{prop} = \frac{d}{V} = \frac{12 + 10^6}{24 + 10^7} = \%5_{\text{Sec}} = 50ms$
$T_{Trans} = \frac{L}{R} = \frac{8*1024 + 8}{100*1024} = 0.64$
$\Rightarrow \frac{8KB}{100m} = \frac{0.64}{x} = 8\text{Sec}$

---

**بازدهی در روش Stop and Wait**

فرض می‌کنیم تعداد n فریم را می‌خواهیم از ماشین مبدا به ماشین مقصد ارسال کنیم. روش stop and wait است. (یعنی ابتدا فریم را می‌فرستیم، صبر می‌کنیم تا ACK آن بیاید و بعد فریم بعدی را می‌فرستیم) زمان‌های تأخیر مورد نظر برای ارسال یک فریم و دریافت ACK آن بصورت زیر است:

$$\frac{\rightarrow Frame \rightarrow}{A \leftarrow ACK \leftarrow B}$$

$$T_{Latency} = (T_{prop} + T_{trans} + T_{Queue} + T_{proc}) + (T_{prop} + T_{trans} + T_{Queue} + T_{proc})$$

اعداد ناچیز را صفر درنظر می‌گیریم.

$$T_{Latency} \Rightarrow 2T_{prop} + T_{trans} \qquad V = \frac{\text{کار مفید}}{\text{کار کل}} \times 100$$

$$V = \frac{nl(T + rans)}{n(2T_{prop} + T_{trans})} \Rightarrow \frac{1}{2\frac{T_{prop}}{T_{trans}} + 1} \times 100 \Rightarrow V = \frac{1}{1 + 2a}$$

$$a = \frac{T_{prop}}{T_{trans}}$$

مثال: در مثال فوق اگر تعداد ۱۰۰ فریم ارسال بشود بازدهی کانال فوق کدام است؟ (از روش Step & wait استفاده می‌شود)

$$a = \frac{T_{prop}}{T_{trans}} \qquad V = \frac{1}{1 + 2*0.072} * 100 = 86\%$$

اگر سرعت انتقال فریم دو برابر شود بازدهی چه تغییری می‌کند؟ (وقتی سرعت دو برابر شود زمان $T_{trans}$ کم می‌شود (نصف))

در مثال قبلی $a = \frac{\frac{0.05}{0.64}}{2} = \frac{0.1}{0.64} = 0.156, V = \frac{1}{1.31} * 100 = 76\%$

می‌بینیم با افزایش سرعت انتقال بازدهی کم می‌شود.

**الگوریتم RED (Random Early Detection)**

استراتژی این الگوریتم آن است که در هر شرایطی روتر باید فعال باشد یعنی اگر بافر روتر ۱۰ تایی است، همواره باید کمتر از ۱۰ بسته در آن باشد. (بافر هرگز فول نشود) در این حالت هرگاه صف پر شد با الگوریتم خاصی مثلا تصادفی ، تعدادی از بسته‌ها دراپ می‌شود.

فرمول:

$$Avg_{new} = (1-\alpha)Avg_{old} + (\alpha.q_t)$$
$$y = Mp(Avgnew - Min_{th}) / (Max_{th} - Min_{th})$$
$$y = Mp + (Avgnew - Min_{th}) + Mp(Max_{th} - Min_{th})$$

مثال: فرض کنیم در یک روتر مدیریت صف بر مبنای مکانیزم RED صورت می‌پذیرد. اگر $\alpha = 0.3$ و متوسط طول صف برابر ۷۰ و طول صف لحظه‌ای ثبت شده ۱۰۰ باشد و احتمال Mark Probability برابر ۰/۵ باشد و به ترتیب $Min_{th}$ , $Max_{th}$ برابر ۳۰ و ۸۰ باشد احتمال ورود بسته کدام است؟

حل:

$$\alpha = 0.3, Min_{th} = 30, Max_{th} = 80, q(t) = 100$$
$$(1-0.3) \times 20 + +0.3 \times 120 = 14 + 36 = 50$$

میانگین طول صف، از ۲ فرمول ستاره‌دار بالا استفاده می‌کنیم.

با فرمول اول و فرمول دوم:

$$3\times 70+0.3\times 100=79$$
$$0.5\frac{(79-30)}{80-30}=0.49$$
$$\left.\begin{array}{l} drop \leftarrow Rnd \le y \\ pass \leftarrow Rnd \rangle y \end{array}\right\} \leftarrow 0 \le y \le Mp$$
$$Drop \leftarrow y \rangle Mp$$

در این مثال فرض کنید عدد تصادفی $Rnd = 0.49$ باشد. اگر مثلاً عدد تصادفی ۰/۷ باشد بسته عبور می‌کند و اگر ۰/۳ باشد دراپ می‌شود.

---

نکته ۱: بعد از اینکه احتمال دور ریز بدست آمد، آنگاه روتر عددی تصادفی بین صفر تا یک تولید می‌کند (به اسم مثلا Rnd) اگر این Rnd کوچک‌تر از احتمال دورریز باشد آنگاه بسته دورانداخته می‌شود وگرنه بسته در صف می‌ماند و سرویس دریافت خواهد کرد.

نکته ۲: ممکن است در بافر ورودی مسیر یاب، در انتهای صف تعداد زیادی بسته با احتمال دورریز Y باشد لذا مسیریاب برای هر یک عدد تصادفی Rnd را تولید می‌کند و عمل مقایسه و دورریز و نگهداری را بر مبنای روش فوق انجام می‌دهد.

نکته ۳: اگر عدد احتمال دورریز y بیشتر از $Man_{th}$ باشد بسته حتماً دورانداخته می‌شود.

## «جلسه نهم»

### شدت ترافیک (Traffic Intensity)

اگر شدت ورود ترافیک از نرخ خروج بیشتر باشد به حالت ناپایدار می‌رسیم. (پکت‌ها در صف مانده و پس از مدتی drop می‌شوند)

فرمول شدت ترافیک:

L = طول پکت - R = نرخ ارسال- a = نرخ وورد پکت در ثانیه

$$Traf_{\text{int}} = \frac{L.a \ \rightarrow P/\text{Sec}}{R \ \rightarrow Bit/\text{Sec}}$$

عددی که بدست می‌آید به این شکل آنالیز می‌شود:

- ✓ ۱ <<اگر خیلی بزرگتر از ۱ باشد، ازدحام و ریزش بسته در بافر مسیر یاب
- ✓ ۱ <اگر بزرگتر از ۱: ازدحام و ترافیک سنگین و احتمالاً ریزش
- ✓ ۱=اگر برابر ۱: ترافیک سنگین، عدم ریزش بسته
- ✓ ۱>اگر کمتر از ۱: ترافیک نرمال و عدم ازدحام
- ✓ ۱>>اگر خیلی کمتر از ۱، عدم ورود ترافیک داده و هدر رفت منبع

در هنگام ارسال بسته، روتر به روترهای همجوار خود درخواست می‌فرستد که شدت ترافیک خود را اعلام کنند و در نهایت بسته را از مسیر خلوت‌تر می‌فرستد (با ازدحام کمتر) .

---

مثال: در یک روتر نرخ بسته‌های وارده عدد ۱۰۰ و نرخ ارسال داده برابر kb/sec ۵۰۰ است. اگر اندازه هر بسته KB ۱ باشد وضعیت ترافیک کدام است؟

ب: اگر نرخ ورود بسته‌ها ۳ برابر شود شدت ترافیک چه تغییری می‌کند؟

ج: اگر اندازه بسته‌ها نصف شود چه تغییری در شدت ترافیک خواهیم داشت؟

حل:

$$R = 500Kbit/\sec \qquad ,a = 100p/s, L = 1KB$$
$$Traf_{\text{int}} = \frac{[1\times1024\times8]\times[100]}{500\times1024} = \frac{8}{5} - 1/6$$

یعنی به ازای هر ۱۶ بسته که وارد می‌شوند تنها ۱۰ بسته خارج شده و ۶ بسته در صف می‌مانند.

ب) $1.6\times3 = 4.8$

ج) $\frac{1}{6} \div 2 = 0.8$

در حالت ج بهترین شرایط را خواهیم داشت و به حالت پایداری می‌رسیم.

---

مثال: فرض کنید در مثال قبل طول بسته‌ها اندازه‌ای شده است که شدت ترافیک ۱ است. در این صورت اندازه بسته چند بیت می‌باشد؟

R ثابت است و a هم تغییری نکرده، شدت ۱ یعنی ترافیک ورود و خروج برابر شده است.

$$1 = \frac{La}{R} \Rightarrow R = La \Rightarrow L = \frac{R}{a}$$
$$\Rightarrow \frac{500\times1024}{100} = 5120/8 = 640Byte$$

پس اندازه قطعه TCP 640 بایت شده است.

اگر فیلد Option صفر باشد، می‌دانیم هدر ۲۰ بایت است پس ۶۲۰ بایت از این قطعه داده است.

نکته: اگر بخواهیم ماشین‌ها مجبور کنیم همه بسته‌های مبادله شده‌شان این اندازه‌ای باشد، فیلد WinSize در بسته TCP را برابر ۶۴۰ قرار می‌دهیم.

---

## تمرین ۱۱

فرض کنید در یک روتر مدیریت صف بر مبنای مکانیزم RED صورت می‌پذیرد. بسته‌ها در مدت ۱۰ ثانیه با طول صف ۹۰ و در هر ثانیه ۲ بسته ارسال می‌شود. اگر $\alpha=0.2$ و RND تصادفی باشد نمودار مربوط به دورریز و یا عبور بسته‌ها را بر اساس محاسبات فوق رسم کنید. (Min=40 و Max=90 و MP=0.6 و Avg=70)

## «جلسه دهم»

نکته: در مورد حل مسائل Red:

$$Avg_{new}(t) < Min_{th} \rightarrow pass$$
$$Min_{th} \le Avg_{new}(t) \le Max_{h} \rightarrow y \qquad Rnd()$$
$$Rnd() < y \rightarrow Drop$$
$$Avg > Max_{th} \rightarrow Drop$$

شبکه‌های موردی (Ad hoc): توپولوژی مشخصی ندارند و در هر لحظه توپولوژی آنها تغییر می-کند.

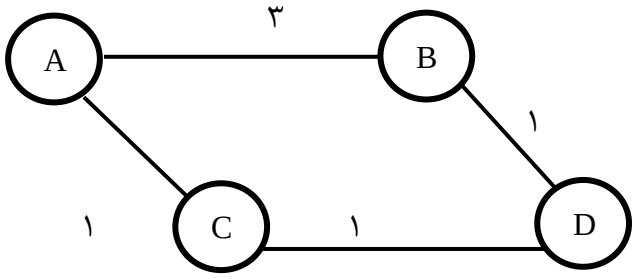
$$Mobile \quad Adhoc \quad Network(MANET)$$
$$Dis\tan ce \quad Network \quad Routing(DSDV)$$

---

**مسیریابی بردار فاصله Distance Vector Routing (DVR)**

**مسیریابی حالت لینک Link State Routing (LSK)**

این تکنولوژی‌ها در لایه سوم قرار داشته و در دسته Routing قرار می‌گیرند. مثلا در گراف زیر سعی داریم کوتاهترین مسیر بین A و D را بیابیم.




در روش Link State هر گره نقشه کاملی از شبکه گره‌ها و لینک‌ها را در خود دارد و برای به روز رسانی هر ۲۷ ثانیه به گره‌های همسایه بسته‌ی Hello Packet ارسال و پاسخ دریافت می‌کنند و آنها

اطلاعات همسایگان خود را در اختیار وی قرار داده و بدین ترتیب هر بار با احتساب هزینه رفت و برگشت، هزینه‌های زمانی رسیدن به آن گره را محاسبه می‌کند.

$$Delay = \frac{t_{send} + t_{Rec}}{2}$$

در مرحله بعد با الگوریتم دایجسترا کوتاهترین مسیر بین خود وگره مقصد D را می‌یابد.

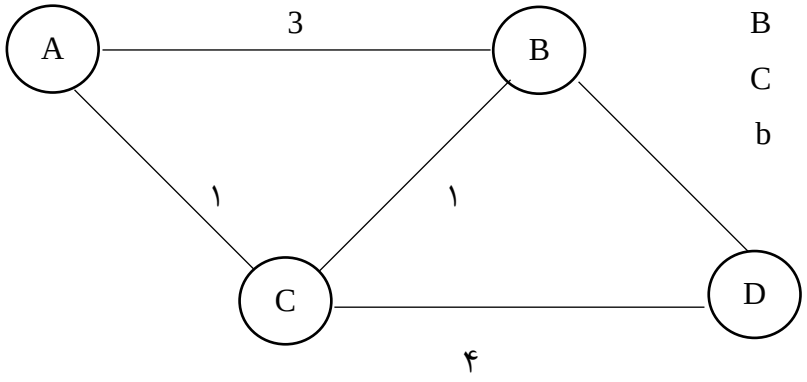
**تمرین ۱۲**

الگوریتم دایجسترا را بررسی کرده و الگوریتم آن را برای شبکه ذکر شده در صفحه‌ی قبل جهت یافتن کوتاهترین مسیر از A به D و C به D و B به A مشخص نمایید

## روش Distance Vector

در روش بردار فاصله، هر گره در صورتی که بخواهد کوتاهترین مسیر را از خود به گره‌ی مقصد (مثلا D) بدست آورد، از همسایه‌ها درخواست می‌کند که آنها بردار فاصله‌ی خود را تا همسایگان و گره‌های مقصدشان برای او بفرستند. سپس گره مبدأ A ، فاصله‌ی خود تا گره‌های همسایه را نیز دارد و این فاصله را نیز به مقادیر بردار آمده از گره‌های همسایه اضافه می‌کند و فاصله نهایی را با محاسبه کوتاهترین مسیر بدست می‌آورد. در مثال قبل هر گره جدولی به شکل زیر ایجاد خواهد کرد:

| | |
|---|---|
| A | ۳ |
| B | – |
| C | ۱ |
| b | ۱ |




| | |
|---|---|
| A | – |
| B | ۳ |
| C | ۱ |
| D | ؟ |

| | |
|---|---|
| A | ۱ |
| B | – |
| C | – |
| D | 4 |

$$DV_B = A - B + B - D = 3 + 1 = 4$$
$$DV_C = A - C + 4 = 5$$

Min دو DR بالا درنظر گرفته می‌شود (در اینجا ۴)

(پس B برای ارسال پکت انتخاب می‌شود): $A \to B \to D$

حالا فرض کنیم بعد از تبادل بردارهای فاصله، مسیر جدید B به C با فاصله ۱ بوجود آمده است.

جدول C اصلاح می‌گردد.

| |
|---|
| ۱ |
| ۱ |
| – |
| ۴ |

$$\left.\begin{array}{l} DV_C = C.....D:4 \\ DV_B = C.....\overset{0}{D}....D:B = C - B + B - D = 2 \end{array}\right\} = 2Min$$

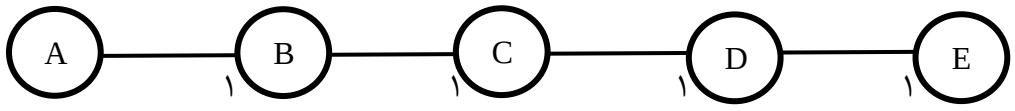
## مشکلات روش بردار فاصله

ممکن است که گره همسایه به غلط فاصله‌ی خود را گره D بیان کند که این گره گره‌ی مخرب یا نفوذگر است. در این صورت، بسته‌ها به غلط به مسیر نادرست فرستاده می‌شوند.

## مزیت روش بردار فاصله

در این روش هزینه محاسباتی برای گره A تنها از طریق بردارهای فاصله دریافتی از همسایه‌ها و یک مقایسه ساده و محاسبه بدست می‌آید، پس سرعت محاسبه بالاست.

نکته: این روش مشکل شمارش تا بی‌نهایت را دارد که بصورت زیر است:




| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| A | _ | A | ۱ | A | ۲ | A | ۳ | A | ۲ |
| B | ۱ | B | _ | B | ۱ | B | ۲ | B | ۳ |
| C | ∞ | C | ۱ | C | _ | C | ۱ | C | ۲ |
| D | ∞ | D | ∞ | D | ۱ | D | _ | D | ۱ |
| E | ∞ | E | ∞ | E | ∞ | E | ۱ | E | _ |
| زمان : | T=1 | | T=2 | | T=3 | | T=4 | | T=5 |

اگر در ثانیه‌ی T=20 لینک D به E قطع شود، گره‌ی D تنها شاهراه ارتباطی به E بوده است و دیگران جدول مسیریابی یا بردار فاصله خود را براساس اطلاعات D به روز نموده‌اند اما خبر ندارند که اتصال قطع شده است، لذا خود D نیز از این موضوع بی‌اطلاع است. پس با قطع لینک D تلاش می‌کند مسیر جدید و فاصله جدید E را از گره‌های همسایه بدست آورد، بنابراین از گره‌های C مسیر رسیدن به E را می‌پرسد. این گره اعلام می‌کند فاصله تا E برابر۲ است. با اضافه کردن فاصله‌ی خود تا C (۱) فاصله تا E عدد ۳ لحاظ می‌شود و پس این بردار جدید را برای همسایگان خود می‌فرستد و همسایگان بصورت انتشاری جداول خود برای رسیدن به E را افزایش می‌دهند و این شمارش تا بی‌نهایت ادامه می‌یابد. این در حالی است که اصلاً هیچ مسیری به E از سوی این گره‌ها وجود ندارد:

$$T = 20 \qquad \underset{\leftarrow 6}{A}\,1\,\underset{\leftarrow 5}{B}\,1\,\underset{\leftarrow 4}{C}\,1\,\underset{\leftarrow 3}{D}\,1E$$

$T = 28$ $\underset{\leftarrow 8}{A} - \underset{\leftarrow 7}{B} - \underset{\leftarrow 6}{C} - \underset{\leftarrow 5}{D} - E$

$T = 36$ $\underset{\leftarrow 10}{A} - \underset{\leftarrow 9}{B} - \underset{\leftarrow 8}{C} - \underset{\leftarrow 7}{D} - E$

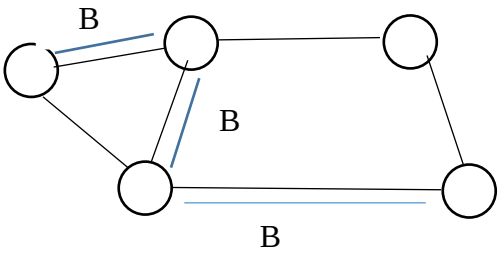
**مقایسه Link State با Distance vector**

| پارامتر | Link-State | Dis-Vector |
|---|---|---|
| دقت – قابلیت اعتماد Reliability | عالی | ضعیف |
| سرعت به روز رسانی Response Time | کم | زیاد |
| احتمال شمارش تا بی‌نهایت Can’t – to- infinity | ندارد | دارد |
| مصرف حافظه هر گره Memory | زیاد | کم |
| مصرف انرژی energy | زیاد | کم |
| مصرف پهنای باند. ترافیک- بسته‌های کنترلی Band Midth | زیاد | کم |

## «جلسه یازدهم»

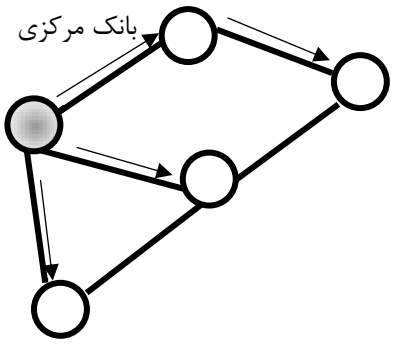
انواع روش‌های ارسال داده ۱) Unicast ۲) MultiCast

۱) در این روش یک بسته داده دارای یک مبدأ، یک مقصد و همچنین دارای داده‌های کنترلی به همراه داده اصلی (payload) می‌باشد.



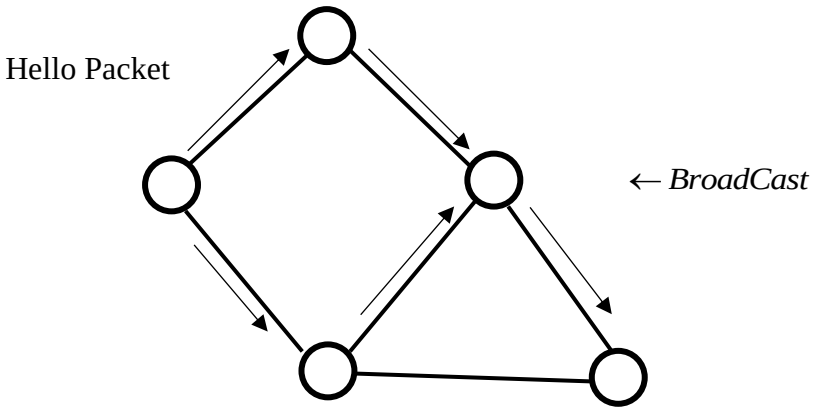

←*Unicast*

۲) در روش MultiCast یا چند بخشی، یک بسته داده باید به چند مقصد معین برسد. در واقع گره‌های مقصد اعضای یک گروه می‌باشند. مثلاً ارسال بخشنامه‌های بانک مرکزی به کلیه بانک‌های کشور مبنی بر کاهش سود سپرده‌ها بانک




←*MultiCast*

۳) در روش همه پخشی یا BroadCast یک بسته‌ی داده به تمامی گره‌های شبکه بایستی ارسال شود؛ مثلاً بسته‌های کنترلی در شبکه اینترنت هر ۲۷ ثانیه یکبار به تمامی روترها و سوئیچ-های فعال ارسال می‌شوند که به Hello Packet معروف است.




---

**انواع پروتکل‌های مسیریابی در حالت تک بخشی Unicast**

سه نوع پروتکل مختلف داریم:

۱) پروتکل‌های Reactive (واکنشی): در این پروتکل‌ها تنها زمانی که بسته‌ای به روتر برسد و مقصد خاصی را بخواهد، آن مسیریاب اقدام به عملیات مسیریابی و یافتن گره مقصد می‌نماید.

۲) پروتکل‌های Proactive (از پیش تدارک دیدن برای عملیات مسیریابی): یعنی گره مسیر یاب از قبل تمامی مسیرهای گره‌های مقصد را بدست آورده است و زمانی که گره جدید می‌رسد، بدون تأمل به مقصد تعیین شده آن را ارسال می‌دارد.

۳) پروتکل‌هایی Hybrid (ترکیبی): به مقتضای حال ممکن است از روش واکنشی یا پیش تدارکی و غیره استفاده کند.

در شبکه‌های موردی دو نوع بسته‌های سیگنال داریم. این بسته‌ها تلاش می‌کند که از تصادم (Collision) اجتناب نمایند:

۱) بسته RTS (Request to send): درخواست برای ارسال بسته است. پیش از ارسال داده فرستاده می‌شود.

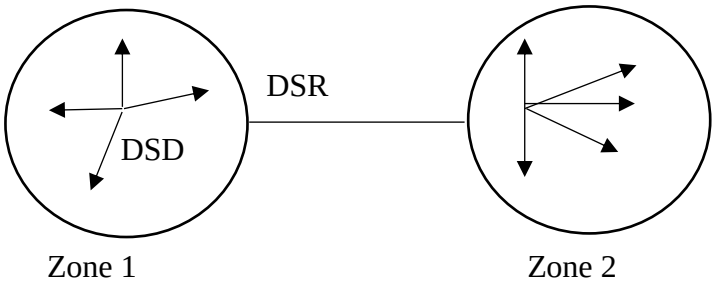
۲) بسته CTS (Clear to send) : یعنی گره مقصد پس از آماده شدن برای دریافت داده آن را می‌فرستد.

پروتکل‌های Proactive: همواره به دنبال کوتاه‌ترین مسیر هستند. مثل DSDV

پروتکل‌های Reactive: هر کجا لازم بود مسیر یابی را انجام می‌دهند. مثل DSR

پروتکل‌های Hybrid: مثل پروتکل‌های ZRP (Zone Routing Protocol)

مثلا تعدادی ناحیه داریم که در درون هر یک نیز تعدادی نود است. مسیریابی بین نودها از طریق DSR و مسیریابی درون نودها از طریق DSDV انجام می‌شود.




در روش ZRP اگر تعداد Zoneها زیاد باشد کار پروتکل DSR سنگین شده و هزینه مسیریابی بالا می‌رود و در واقع به جدول زیر می‌رسیم:

| Zone | DSDV | DSR |
|---|---|---|
| کوچک | کوچک و سریع، فضای کوچک، تعداد لینک کم | کمد، فضای بزرگ، تعداد لینک زیاد |
| بزرگ | بزرگ می‌کند، فضای بزرگ، تعداد لینک زیاد | سریع، فضای کوچک، تعداد لینک کم |

DSR (Dynamic Source Routing)

مسیریابی توسط مبدأ بصورت پویا انجام می‌شود. (هر وقت لازم بود). در این الگوریتم هرگاه بسته‌ای با مقصد D به گره مبدأ S برسد بطور معمول گره S هیچ گونه مسیری از پیش برای رسیدن به D آماده نکرده است، لذا عملیات Route Discovery (کشف مسیر) را آغاز می‌کند.

عملیات کشف مسیر به قرار زیر است:

الف) گره مبدأ S ، بسته‌ی درخواست مسیر (Route Request) یا Rreq را بصورت سیل آسا به تمامی همسایگان خود را ارسال می‌کند (همه پخشی یا Broad Cast).

ب) تمامی گره‌های همسایه دریافت کننده‌ی بسته‌ی RReq شناسه‌ی خود را به این بسته اضافه نموده و آن را به همسایه‌های مجاور خود Forward می‌کنند.

ج) این عملیات برای تمامی گره‌های شبکه تکرار می‌شود تا اینکه گره مقصد D بسته‌ی RREQ را دریافت کند. چون این گره مقصد است آنگاه بسته‌ی پاسخ Route Reply یا RREP را آماده نموده و از مسیری که اولین Route Request آمده و گره‌های بین مسیر در آن قید شده بسته Route Reply به طور معکوس آن مسیر را پیمایش می‌کند. بنابراین بسته‌ی RREP حاوی مسیر مناسب و کوتاه از S به D می‌باشد.

د) با دریافت بسته‌ی RREP توسط S مسیر ذخیره شده در RREP در گره S، کش (ذخیره) می‌شود. لذا از این پس اگر S بخواهد بسته وی به D ارسال کند، کافیست در بسته‌ی خود (دور سرآیند آن) تمامی گره‌های میانی تا رسیدن به D را قرارد هد.

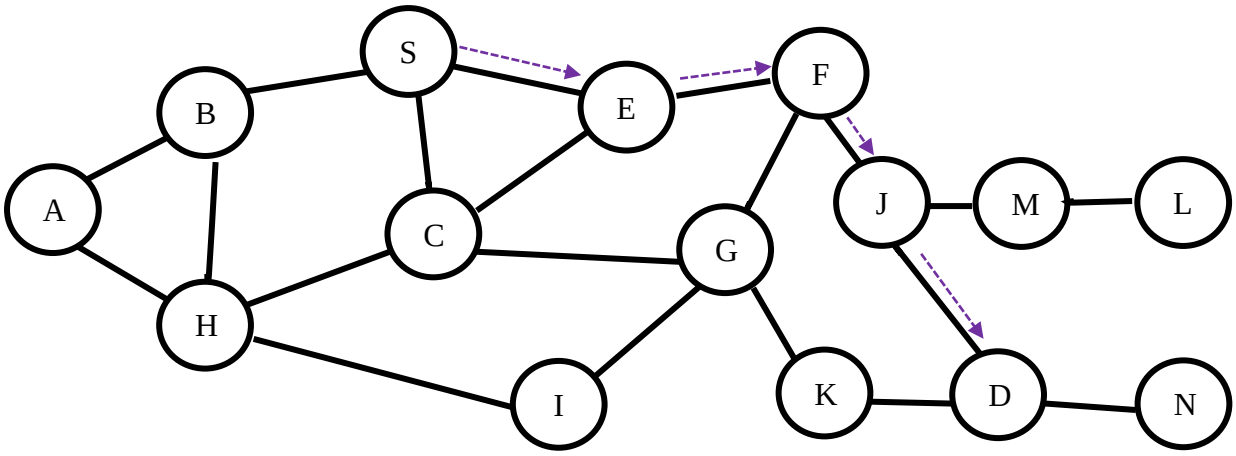
بسته Route Request به اندازه n (تعداد نودهای شبکه) حافظه برای کشف مسیر دارد.

برخی بسته‌های RREQ ممکن است چند بار از طریق گره‌های مختلف به یک نود ارسال شوند و ایجاد Overhead نمایند. نود مقصد می‌تواند این بسته‌ها را دور بریزد.

**نکاتی در مورد بهینه سازی DSR**

هرگره میانی به سادگی می‌تواند مسیرهای ذخیره شده در بسته‌های RREQ را در کش خود ذخیره کند و در واقع از آخرین تغییرات و مسیرهای موجود مطلع شود. تنها مشکل آن است که در هر بار که بسته‌های Rout Request از آنها عبور کنند بایستی آنها را کش کنند و مسیرهای خود را به روز نمایند که اینکار سربار محاسباتی دارد.

به منظور افزایش قابلیت اطمینان گره مقصد D و همچنین گره مبدأ S می‌توانند دو مسیر بدست آمده: ۱) SEFJD و ۲) SCGKD (شکل زیر) را نگهداری کنند. در مواقع قطع لینک از مسیر اول یا از بین رفتن گره میانی بدون نیاز به فرآیند کشف مسیر، از مسیر دوم بسته‌ها را هدایت نمایند.




نکته: در روش DDSR بایستی مراقب باشیم که گره‌های میانی مشترک بین دو مسیر اصلی و رزرو شده وجود نداشته باشد. مثلاً در مثال بالا اگر مسیر دوم را SCEFJD قرار دهیم عملاً چندان بکار نمی‌آید.

هرگاه دو مسیر کامل بدست آمد با توجه به پارامترها (تاخیر کلی مسیر، تعداد چاپ‌های میانی مسیر، پهنای باند در مسیر، هزینه ارسال داده از دو مسیر و کیفیت یا میزان نویز دو مسیر) تصمیم گیری بهینه بر مبنای فازی محقق می‌گردد.

**الگوریتم LAR یا مسیریابی بر مبنای مکان**

در این روش مسیریابی، عملیات مسیریابی مانند الگوریتم DSR شکل گرفته و بسته‌های RREQ ، عملیات مسیریابی مانند الگوریتم DSR شکل گرفته بسته‌های RREQ بصورت سیل آسا ارسال می-شود تنها تفاوت LAR با DSR آنست که در LAR ناحیه‌ها یا Zone ها موظف به ارسال بسته‌های

RREQ هستند، تا اینکه به ناحیه‌ای برسیم که گره مقصد D در آن ناحیه است و سپس عملیات ارسال بسته RREQ شکل می‌گیرد.

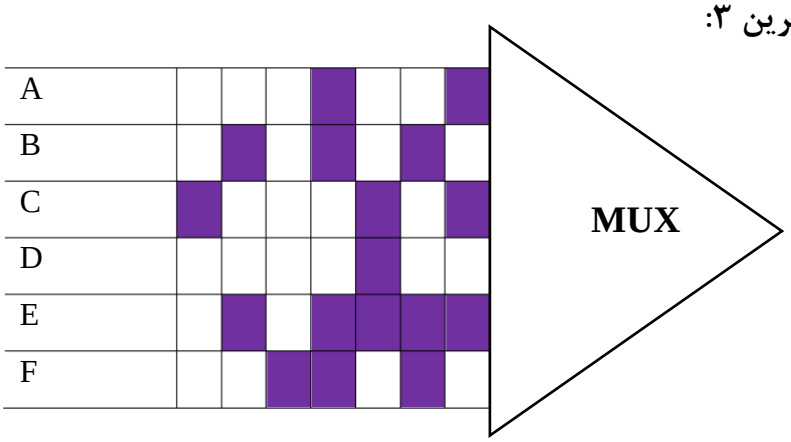
مزیت LAR: ۱) فضای انتشار سیل آسای بسته RREQ کاهش می‌یابد. ۲) هزینه کشف مسیر کاهش می‌یابد.

معایب LAR: ۱) گره‌ها بایستی مکان‌های فیزیکی خودشان را بدانند (نیاز به GPS) ۲) این روش احتمال وجود موانع در بین گره‌های از لحاظ جغرافیایی نزدیک را درنظر نمی‌گیرد.

# پیوست:

# حل برخی از تمرین‌ها

**حل تمرین ۳:**




**الف) TDM با ۶ بخش:**

| | | | | | |
|---|---|---|---|---|---|
| × | E | × | C | × | A |

| | | | | | |
|---|---|---|---|---|---|
| F | E | × | × | B | × |

| | | | | | |
|---|---|---|---|---|---|
| × | E | D | C | × | × |

| | | | | | |
|---|---|---|---|---|---|
| F | E | × | × | B | A |

| | | | | | |
|---|---|---|---|---|---|
| F | × | × | × | × | × |

| | | | | | |
|---|---|---|---|---|---|
| × | E | × | × | B | × |

| | | | | | |
|---|---|---|---|---|---|
| × | × | × | C | × | × |

$$TDM = \frac{17}{42} \times 100 = 40\%$$

**ب) STDM با ۴ بخش و نهایتا ۲ سیکل انتظار**

| | | | |
|---|---|---|---|
| B2 | E1 | C1 | A1 |

| | | | |
|---|---|---|---|
| D1 | C2 | F1 | E2 |

| | | | |
|---|---|---|---|
| E4 | B2 | A2 | E3 |

| | | | |
|---|---|---|---|
| E5 | B3 | F3 | F2 |

| | | | |
|---|---|---|---|
| × | × | × | C3 |

$$STDM = \frac{17}{20} \times 100 = 85\%$$

**ج) STDM با ۵ بخش و ۲ سیکل انتظار**

| | | | | |
|---|---|---|---|---|
| E2 | B1 | E1 | C1 | A1 |

| | | | | |
|---|---|---|---|---|
| A2 | E3 | D1 | C2 | F1 |

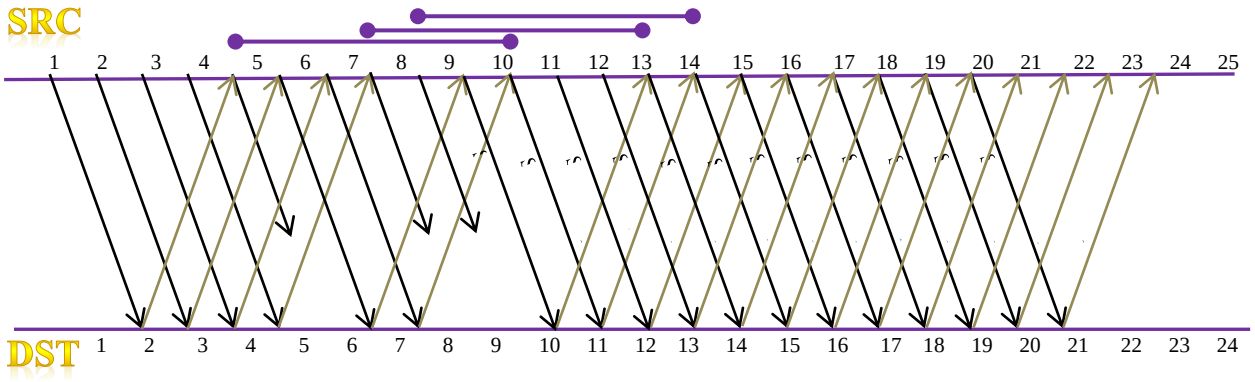
| | | | | |
|---|---|---|---|---|
| B3 | F3 | F2 | E4 | B2 |

| | | | | |
|---|---|---|---|---|
| × | × | × | C3 | E5 |

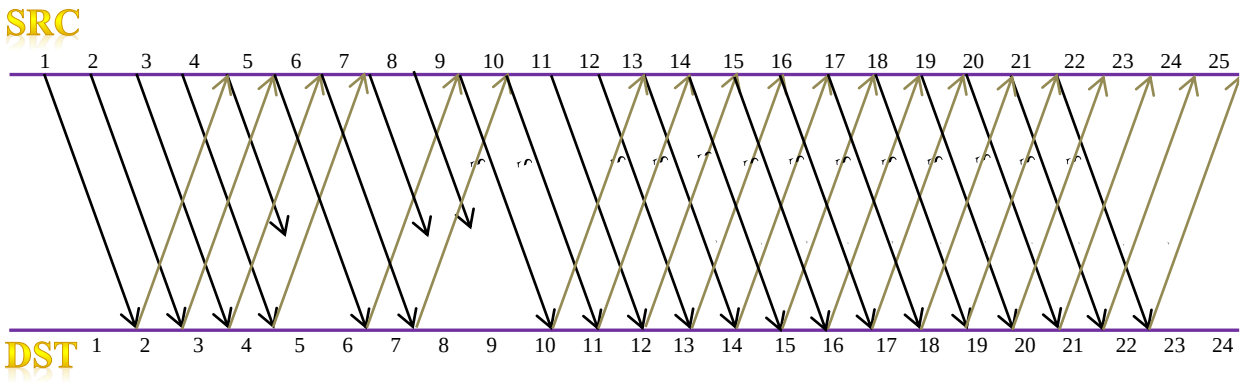
$$STDM = \frac{17}{20} \times 100 = 85\%$$

## حل تمرین ۸:

**Selective Repeat, interval = 6**




**Go back n, Time= 1-35**



حل تمرین ۹:

۱- اگر α=0.1 باشد RTO چند است؟

$EstimRTT_{new} = (1-0.1)*40 + 0.1 * 70 = 36 + 7 = 43$

$Deviation_{new} = (1-0.1)*0.2 + 0.1*(70-43) = 0.18 + 2.7 = 2.88$

$RTO = 43 + (4*2.88) = 54.52$

۲- اگر α=0.5 باشد RTO چند است؟

$EstimRTT_{new} = (1-0.5)*40 + 0.5 * 70 = 20 + 35 = 55$

$Deviation_{new} = (1-0.5)*0.2 + 0.5*(70-55) = 0.1 + 7.5 = 7.6$

$RTO = 55 + (4*7.6) = 85.4$

۳- اگر α=0.9 باشد RTO چند است؟

$EstimRTT_{new} = (1-0.9)*40 + 0.9 * 70 = 4 + 63 = 67$

$Deviation_{new} = (1-0.9)*0.2 + 0.9*(70-67) = 0.02 + 2.7 = 2.72$

$RTO = 67 + (4*2.72) = 77.8$

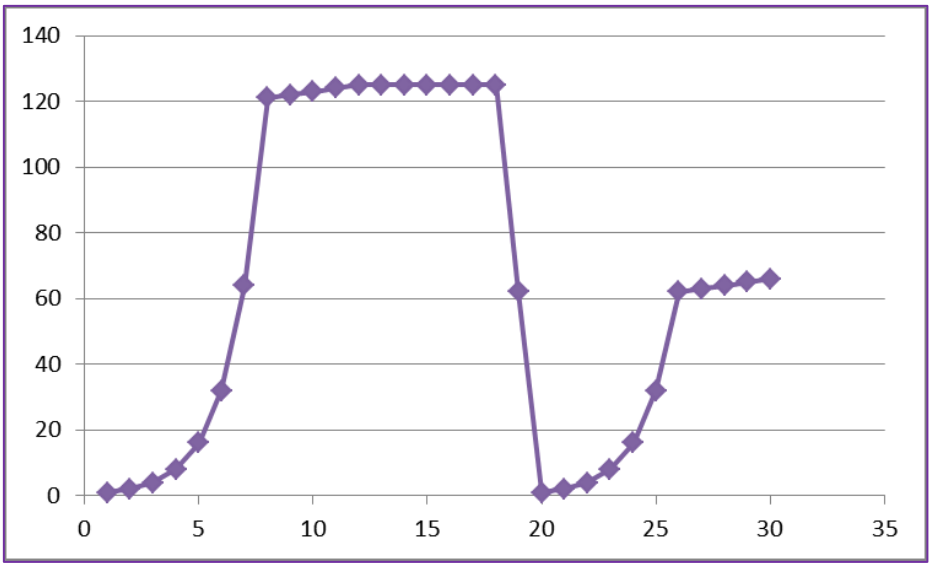
◄ ***همانطور که می‌بینیم در قسمت ۲ و ۳ مقدار RTO زیاد شده است که عقلانی به‌نظر نمیرسد و نتیجه‌ی آن کاهش بهره‌وری از منابع خواهد بود. بنابراین حالت ۱ بهترین زمان RTO را به ما می‌دهد.***

حل تمرین ۱۰:

- با استفاده از پروتکل TCP/IP و با الگوریتم Jacobson با شروع آهسته، ماشین مبدا در ۳۰ Step زمانی داده ها را برای ماشین مقصد میفرستد. در ابتدا آستانه ازدحام برابر ۱۲۱ و WindowSize برابر ۱۲۵ است. اگر در واحد زمانی ۱۹ ازدحام رخ دهد نمودار Jacobson آن را رسم کنید.
- اگر در این مسئله در ثانیه ۲۵ ام اندازه بافر به ۵۷ کاهش یابد، نمودار به چه صورت خواهد شد؟

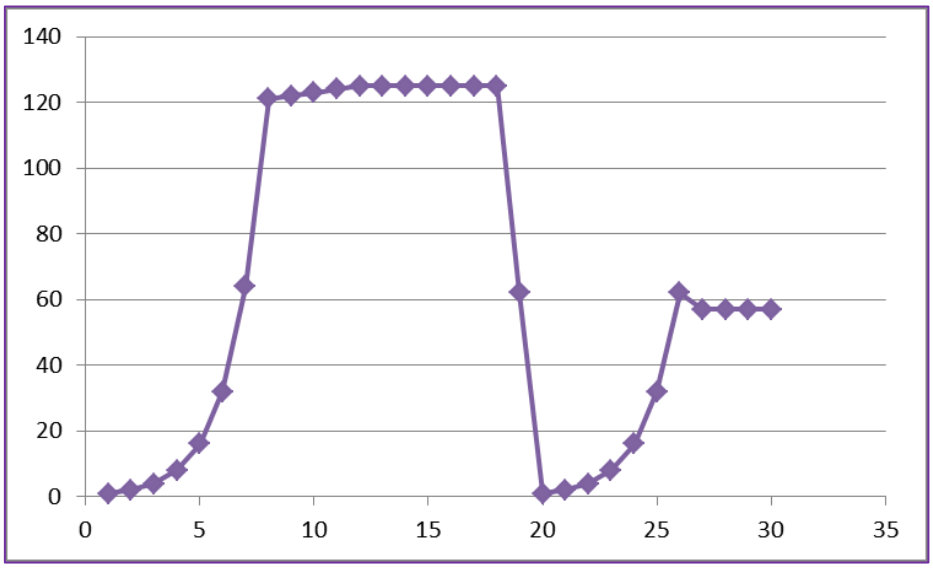
| t | S(t) |
|---|---|
| 1 | 1 |
| 2 | 2 |
| 3 | 4 |
| 4 | 8 |
| 5 | 16 |
| 6 | 32 |
| 7 | 64 |
| 8 | Min (2*64, 121, 125) = |
| 9 | Min (122, 125) = 122 |
| 10 | Min (123, 125) = 123 |
| 11 | Min (124, 125) = 124 |
| 12 | Min (125, 125) = 125 |
| 13 | Min (126, 125) = 125 |
| 14 | Min (126, 125) = 125 |
| 15 | Min (126, 125) = 125 |

| t | S(t) |
|---|---|
| 16 | Min (126, 125) = 125 |
| 17 | Min (126, 125) = 125 |
| 18 | Min (126, 125) = 125 |
| 19 | 125, threshold=62 |
| 20 | 1 |
| 21 | 2 |
| 22 | 4 |
| 23 | 8 |
| 24 | 16 |
| 25 | 32 |
| 26 | Min (64, 62, 125) = 62 |
| 27 | Min (63, 125) = 63 |
| 28 | Min (64, 125) = 64 |
| 29 | Min (65, 125) = 65 |
| 30 | Min (66, 125) = 66 |




اگر در این مسئله در ثانیه ۲۵ ام اندازه بافر به ۵۷ کاهش یابد:

| *t* | *S(t)* |
|---|---|
| *25* | *32* |
| *26* | *Min (64, 62, 57) = 57* |
| *27* | *57* |
| *28* | *57* |
| *29* | *57* |
| *30* | *57* |



حل تمرین ۱۲:

۱- کوتاهترین مسیر از A به D

| S | B | C | D |
|---|---|---|---|
| {A} | 3 | 1 | ∞ |
| {A, C} | 2 | 1 | 5 |
| {A, C, B} | 2 | 1 | 3 |
| {A ,C, B, D} | 2 | 1 | 3 |

**A> C> B> D**

۲- کوتاهترین مسیر از C به D

| S | B | A | D |
|---|---|---|---|
| {C} | 1 | 1 | 4 |
| {C, A} | 1 | 1 | 4 |
| {C, A, B} | 1 | 1 | 2 |

**C> B> D**

۳- کوتاهترین مسیر از B به A

| S | C | A | D |
|---|---|---|---|
| {B} | 1 | 3 | 1 |
| {B, C} | 1 | 1 | 1 |
| {B, C, A} | 1 | 1 | 2 |

**B> C> A**