



به نام او و به یاد او  
 ”و إِنَّ رَاحِلَ إلیک قَرِیبُ الْمَسَافَةِ ...“

تمرین‌های فصل سوم (لایه حمل) درس شبکه‌های کامپیوتری

پاییز ۱۳۸۶

دانشکده مهندسی برق و کامپیوتر دانشگاه تهران

○ مفاهیم مهم مورد استفاده در حل مسایل این فصل:

- ❖ بحث پروتکل‌های rdt مختلف ارائه شده با توجه به فرض‌های صورت گرفته در هر مورد و FSM‌های آنها
- ❖ انواع ARQ: Stop-and-Wait, Selective Repeat و Go-Back-N
- ❖ فرمت بسته‌های TCP و UDP، اندازه و معنای هر فیلد:

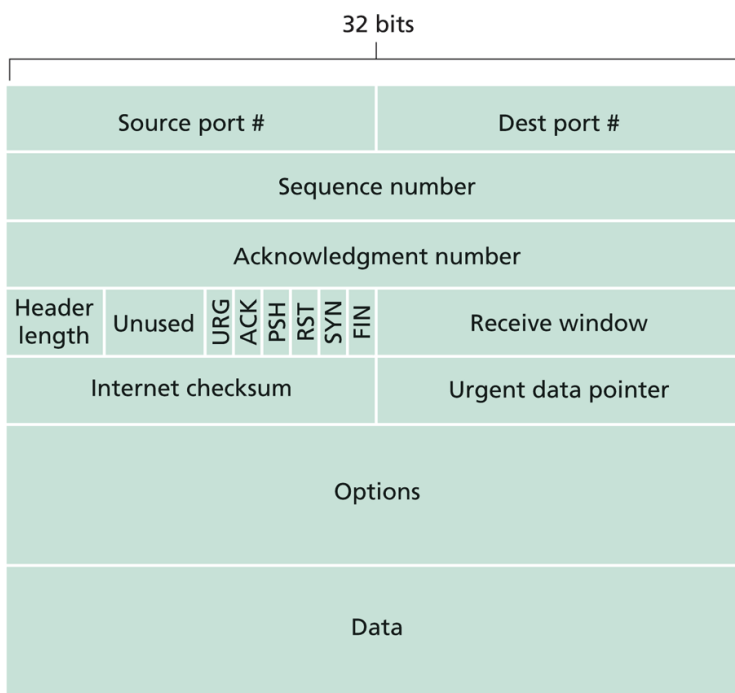


Figure 3.29 ♦ TCP segment structure

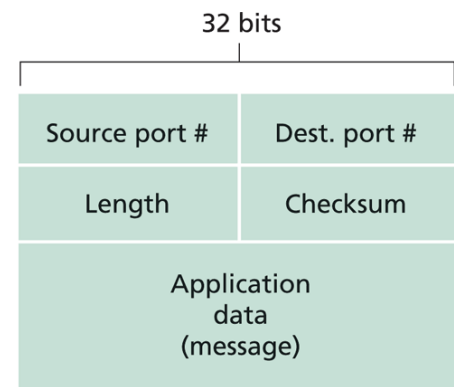


Figure 3.7 ♦ UDP segment structure

❖ روش محاسبه RTO در TCP:

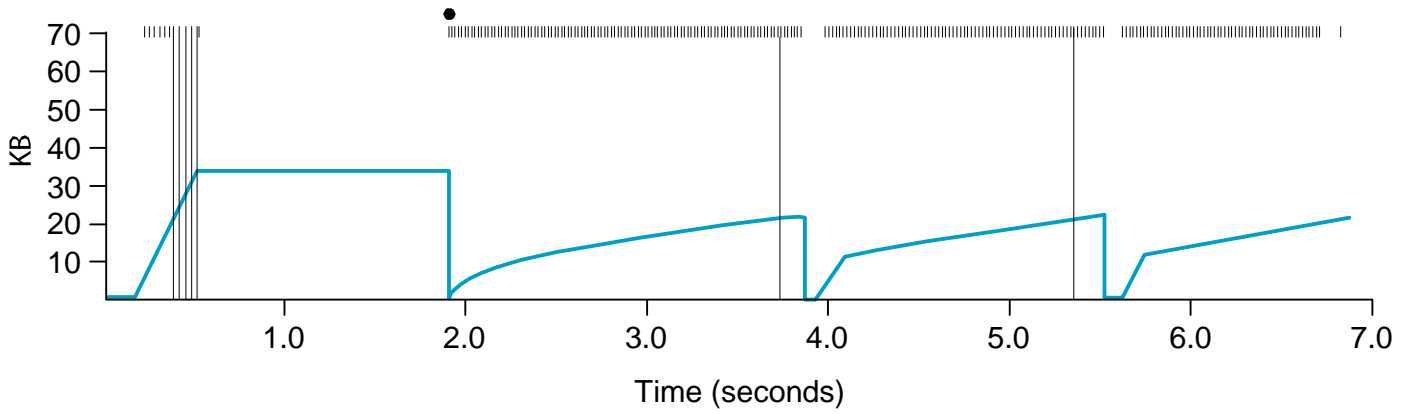
$$EstimatedRTT = (1 - \alpha).EstimatedRTT + \alpha.SampleRTT$$

$$DevRTT = (1 - \beta)DevRTT + \beta|SampleRTT - EstimatedRTT|$$

$$TimeoutInterval = EstimatedRTT + 4DevRTT$$

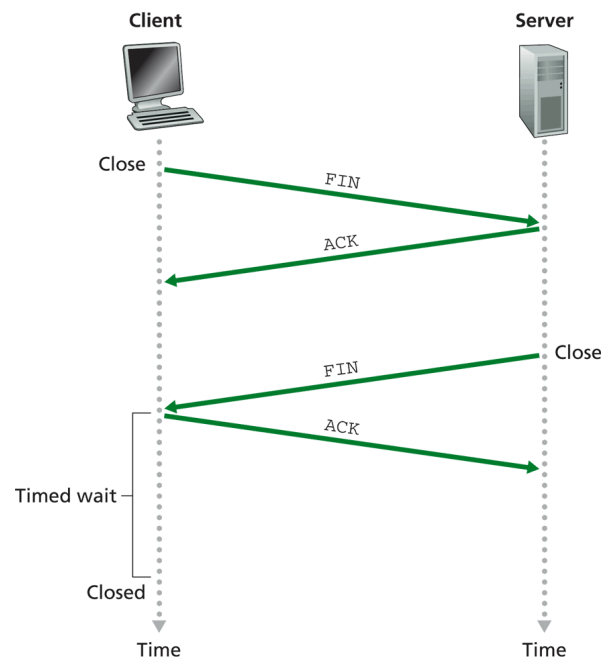
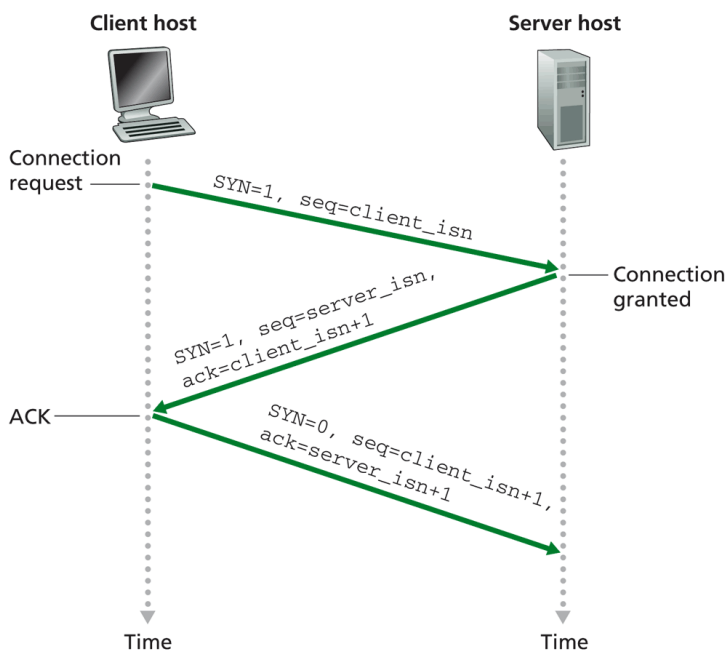
مقدار  $\alpha$  در RFC 2988 معمولاً ۰/۱۲۵ (یک هشتم) لحاظ می‌شود و مقدار نوعی  $\beta$ ، ۰/۲۵ می‌باشد.

- ❖ مراحل مختلف الگوریتم کنترل تراکم TCP: Slow Start و Adaptive Increase-Multiplicative Decrease (توجه شود که لازم است که جزئیات این الگوریتم‌ها را بدانید تا بتوانید به مسایل ریاضی مربوطه پاسخ دهید. ضمناً این جزئیات و کلاً ویژگی‌های گوناگون TCP، ممکن است در مسایل صحیح-غلط نیز مورد سؤال قرار گیرند).



توضیح شکل (مربوط به کتاب Peterson): ♦ نشان دهنده وقوع Time-out و در نتیجه بازارسال، خط‌های عمودی کوچک (هاشورها) نشان دهنده لحظه‌های ارسال بسته‌ها و خط‌های عمودی بزرگ که در طول محور عمودی امتداد یافته‌اند نماینده اولین دفعه ارسال بسته‌ای است که دوباره در آینده ارسال شده است.

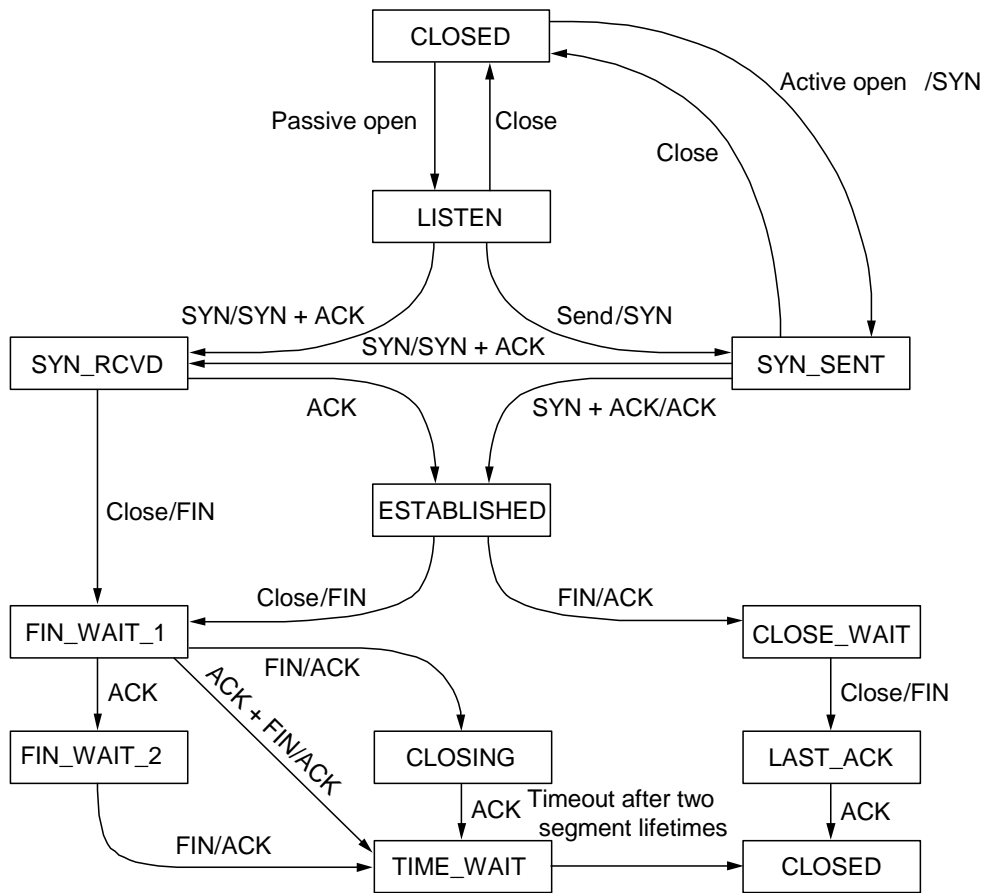
♦ مکانیزم شروع اتصال TCP (دست تکانی سه طرفه) و مکانیزم پایان اتصال:



**Figure 3.38** ♦ TCP three-way handshake: segment exchange

**Figure 3.39** ♦ Closing a TCP connection

❖ TCP State Transition Diagram: (برای توضیحات مراجعه شود به کتاب Peterson و یا RFC 793)



❖ مفهوم Duplicate ACK و الگوریتم‌های Fast Recovery و Fast Retransmit

❖ TCP Flow Control: توجه شود که Flow Control جهت جلوگیری از غوطه ور شدن دریافت کننده می‌باشد، در حالی که Congestion Control جهت جلوگیری از ارسال بیش از حد فرستنده در مواقع ایجاد تراکم در شبکه به جهت منفعت عمومی تمام جریان‌ها می‌باشد.

❖ اندازه‌گیری بازدهی TCP (Throughput) و مدل‌های ریاضی مربوطه

❖ بحث Fairness

❖ مدل‌های ریاضی ارائه شده برای محاسبه تأخیر

## ○ تمرین های این فصل:

۱- Applet های مربوط به فصل سوم در سایت کتاب درس مشاهده شوند.

[http://media.pearsoncmg.com/aw/aw\\_kurose\\_network\\_2/applets/go-back-n/go-back-n.html](http://media.pearsoncmg.com/aw/aw_kurose_network_2/applets/go-back-n/go-back-n.html)

[http://media.pearsoncmg.com/aw/aw\\_kurose\\_network\\_3/applets/SelectRepeat/SR.html](http://media.pearsoncmg.com/aw/aw_kurose_network_3/applets/SelectRepeat/SR.html)

[http://media.pearsoncmg.com/aw/aw\\_kurose\\_network\\_2/applets/flow/flowcontrol.html](http://media.pearsoncmg.com/aw/aw_kurose_network_2/applets/flow/flowcontrol.html)

۲- سؤال های (Problems) زیر از کتاب Kurose:

✓ ۴ (بررسی یک FSM اشتباه برای سمت گیرنده rdt2.1 که موجب بن بست می شود)

✓ ۶ (رسم FSM مربوط به rdt3.0)

✓ ۱۱ (مقایسه پروتکل NAK-only با ACK-only)

✓ ۱۴ (طراحی یک پروتکل reliable خاص و کشیدن FSM برای آن)

✓ ۱۹ (صحیح-غلط در مورد ARQ ها)

✓ ۲۱ (محاسبه EstimatedRTT، میان ترم پاییز ۸۴)

✓ ۲۷ (بحث در یک نمونه نمودار تغییرات پنجره TCP)

۳- در مورد صحت یا عدم صحت جمله های زیر با ذکر علت مختصر و مفید قضاوت نمایید.

❖ Checksum در TCP فقط شامل فیلدهای هد TCP و Payload می باشد. (پایان ترم پاییز ۸۵)

❖ بسته های TCP فقط در صورت پر شدن Queue روترها ممکن است drop شوند. (پایان ترم پاییز ۸۵)

❖ در UDP، دریافت کننده برای بررسی چک سام، تمامی کلمات ۱۶ بیتی بسته UDP را جمع کرده و با مشاهده حاصل جمع باینری ۱۱۱۱۱۱۱۱۱۱۱۱۱۱۱۱ فرض می کند که خطایی در بیت های بسته رخ نداده است.

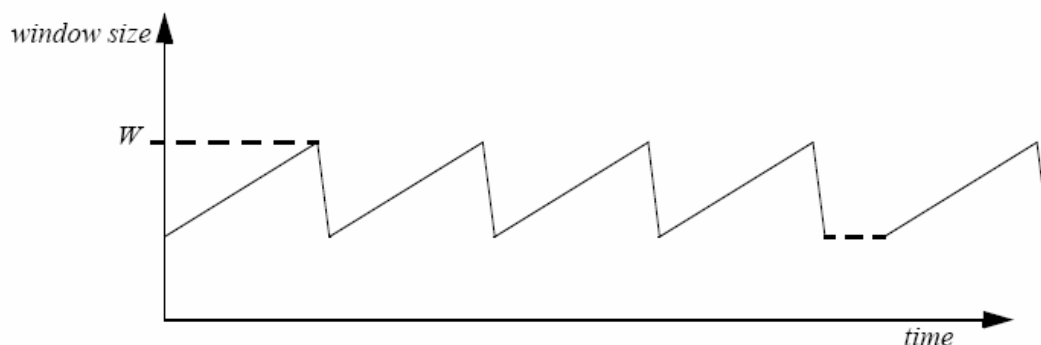
❖ در پدیده TCP, Triple Duplicate ACK، اقدام به ارسال مجدد آن بسته می کند. (Fast Retransmit) (پایان ترم پاییز ۸۵)

❖ بازارسال در TCP، فقط در صورت تمام شدن زمان RTO برای بسته ارسالی، صورت می گیرد. (پایان ترم پاییز ۸۵)

❖ TCP, self-clocking می باشد. (در جواب خود حتماً توضیح دهید که self-clocking به چه معناست) (پایان ترم پاییز ۸۵)

۴- به چه علت محاسبه دقیق مقدار RTT در پروتکل TCP حیاتی است؟ TCP Extension چه کمکی در این راستا می تواند انجام دهد؟ (پایان ترم پاییز ۸۳)

۵- نمودار زیر روند دندان اره ای تغییرات اندازه پنجره فرستنده TCP را بر حسب زمان نشان می دهد.  $W$  ماکزیمم سایز پنجره است. فرض کنید تمامی بسته ها  $P$  بیت باشند و هرگاه سایز پنجره به  $W$  می رسد، دقیقاً یک بسته drop می شود. (پایان ترم بهار ۸۴، مشابه پایان ترم پاییز ۸۵ و مشابه سؤال ۳۱ Kurose)



الف) چرا TCP به جای افزایش multiplicative از افزایش additive استفاده می کند؟

ب) اگر از مرحله slow-start صرف نظر کنیم، نشان دهید میانگین نرخ ارسال فرستنده از رابطه زیر بدست می آید:

$$\bar{R} = \frac{3}{4} \cdot \frac{W}{RTT} \text{ packets per second.}$$

ج) نشان دهید که میزان بسته‌های drop شده از رابطه زیر بدست می‌آید:

$$L = \frac{1}{\frac{3}{4}W\left(1 + \frac{W}{2}\right)}$$

د) با استفاده از بخش‌های ب و ج و فرض این که  $W$  خیلی بزرگ است، نشان دهید که میانگین نرخ ارسال فرستنده از رابطه زیر به دست می‌آید:

$$\bar{R} \approx \frac{\frac{3}{4}\sqrt{\frac{8}{3}}}{\sqrt{L} \cdot RTT} P \approx \frac{1.22P}{\sqrt{L} \cdot RTT} \text{ bits per second.}$$

ه) **goodput** در اتصال TCP تعریف می‌شود به نحوی که در آن داده یکبار فرستاده می‌شود یعنی **goodput** داده‌ای را که دوباره فرستاده شده شامل نمی‌شود؛ آیا میزان **goodput** کمتر از نرخ ارسالی که در قسمت قبل محاسبه کردید می‌باشد؟  
 ز) فرض کنید هر بار که یک بسته drop می‌شود یک پنجره کامل داده دوباره فرستاده می‌شود، با استفاده از رابطه‌ای که در قسمت د بدست آوردید، نرخ **goodput** جریان را بدست آورید.

۶- فرض کنید که هاست A از طریق روترهای R1 و R2 به هاست B می‌رسد (A-R1-R2-B). برای محاسبه زمان تایم اوت از رابطه  $3 \cdot \text{SampleRTT}$  استفاده می‌کند. فرض کنید که لینک‌های A-R1 و R2-B پهنای باند نامحدودی داشته باشند ولی لینک R1-R2 موجب بافر شدن بسته‌ها و ایجاد تأخیر یک ثانیه برای هر بسته حاوی داده (و نه ACKها) شود. اگر A همواره داده برای ارسال داشته باشد، آیا امکان دارد که لینک R1-R2، ۱۰٪ استفاده مفید نشود؟ اگر امکان دارد یک سناریو که موجب این اتفاق می‌شود را بیان کنید و اگر امکان ندارد علت را توضیح دهید.

۷- اگر فریم  $n_f$  بیتی و نرخ خطای بیت  $p$  باشد؛ (میان ترم پاییز ۸۴)

الف) خطای فریم ( $p_f$ ) را حساب کنید.

ب) اگر  $R$  (bps) نرخ ارسال خط،  $n_o$  تعداد بیت‌های سربرار در فریم و  $n_a$  تعداد بیت‌های ACK باشد و  $t_{proc}$  تأخیر پردازش گره‌ها برای محاسبه CRC و ACK و  $t_{prop}$  تأخیر انتشار باشد، نحوه به دست آوردن کارایی را بیان نمایید.

ج) اگر قرار دهیم  $L = 2(t_{prop} + t_{proc}) \frac{R}{n_f}$  و نیز کارایی ارسال روش‌های مختلف ARQ شامل Stop and Wait (SW) و Go-

Back-N (GBN) و Selective Repeat (SR) به ترتیب باشند:

$$\eta_{SR} = \left(1 - \frac{n_o}{n_f}\right)(1 - p_f) \quad \text{و} \quad \eta_{GBN} = \frac{1 - \frac{n_o}{n_f}}{1 + 2(t_{prop} + t_{proc}) \frac{R}{n_f} p_f} (1 - p_f) \quad \text{و} \quad \eta_{SW} = \frac{1 - \frac{n_o}{n_f}}{1 + \frac{n_a}{n_f} + 2(t_{prop} + t_{proc}) \frac{R}{n_f}} (1 - p_f)$$

و بیت‌های سربرار فریم CRC و تعداد بیت‌های ACK، ناچیز باشند؛ بحث کنید که کارایی سه روش چه رابطه‌ای با هم دارند و کارایی SR در چه صورتی بهبود می‌یابد؟ و در چه شبکه‌هایی (از لحاظ لایه فیزیکی) این گونه می‌شود؟

موفق باشید

حمید حاج‌عبدالعلی