

Solução Redes

Caio César Carvalho Dias

4 de Outubro de 2015

Atraso

1. Considere um hospedeiro A enviando um único pacote de tamanho L bytes para o hospedeiro B . A e B encontram-se a uma distância de 25 km e estão conectados por meio de 5 enlaces de 5 km cada, nos quais a velocidade de propagação do sinal é de 2.5×10^8 m/s, totalizando 4 roteadores no caminho. Todos os enlaces tem capacidade de transmissão de 10^6 B/s (≈ 8 MB/s). Todos os roteadores, bem como hospedeiros, demoram $10 \mu\text{s}$ para processar o pacote. Considere que A é o único hospedeiro enviando pacotes pela rede no momento e que a mesma não está congestionada, não havendo atraso de enfileiramento.
 - a) Calcule o atraso, em função de L , para enviar o pacote de A a B usando **comutação de pacotes**, com todos os roteadores utilizando transmissão *store-and-forward* (ou seja, o pacote deve ser recebido por inteiro antes de começar a ser enviado).

Solução: A seguinte fórmula pode ser usada para o cálculo do atraso fim-a-fim numa comutação por pacotes com N roteadores, com atrasos homogêneos e sem congestionamento:

$$d_{fim-a-fim} = (N + 1)(d_{proc} + d_{prop} + d_{trans}).$$

Neste caso temos

$$\begin{aligned} N &= 4 \\ d_{proc} &= 10 \mu\text{s} = 10^{-6} \text{ s} \\ d_{prop} &= \frac{5 \text{ km}}{2.5 \times 10^8 \text{ m/s}} = 20 \times 10^{-6} \text{ s} \\ d_{trans} &= \frac{L \text{ B}}{10^6 \text{ B/s}} = L \times 10^{-6} \text{ s}. \end{aligned} \tag{1}$$

Logo, o atraso em função de L dá-se por

$$\begin{aligned} d_{fim-a-fim}(L) &= (4 + 1)(10^{-6} \text{ s} + 20 \times 10^{-6} \text{ s} + L \times 10^{-6} \text{ s}) \\ &= (21 + L)(5 \times 10^{-6}) \text{ s}. \end{aligned} \tag{2}$$

- b) Calcule o atraso, em função de L , para enviar o pacote de A a B usando **comutação de circuitos**, sem atraso devido ao store-and-forward ou processamento, mas com necessidade de 4.25 ms para estabelecer a conexão e imaginando que todos os recursos são dedicados para essa conexão.

Solução: A seguinte fórmula pode ser usada para o cálculo do atraso fim-a-fim numa comutação por circuitos com N roteadores, com atrasos homogêneos e sem congestionamento:

$$d_{fim-a-fim} = (N + 1)(d_{proc} + d_{prop}) + d_{trans} + d_{con}.$$

Neste caso temos

$$\begin{aligned} N &= 4 \\ d_{proc} &= 0 \text{ s} \\ d_{prop} &= \frac{5 \text{ km}}{2.5 \times 10^8 \text{ m/s}} = 20 \times 10^{-6} \text{ s} \\ d_{trans} &= \frac{LB}{10^6 \text{ B/s}} = L \times 10^{-6} \text{ s} \\ d_{con} &= 4.25 \text{ ms} = 4250 \times 10^{-6} \text{ s}. \end{aligned} \tag{3}$$

Logo, o atraso em função de L dá-se por

$$\begin{aligned} d_{fim-a-fim}(L) &= (4 + 1)(0 \text{ s} + 20 \times 10^{-6} \text{ s}) + L \times 10^{-6} \text{ s} + 4250 \times 10^{-6} \text{ s} \\ &= (4350 + L)10^{-6} \text{ s}. \end{aligned} \tag{4}$$

- c) Analisando os resultados obtidos nos itens anteriores, é possível concluir que, nesse cenário, para pacotes com até um certo tamanho $L1$, o uso da comutação de pacotes é vantajosa, apesar do atraso de store-and-forward, devido ao tempo necessário para estabelecer a conexão, mas para pacotes maiores a comutação de circuitos acaba fornecendo um atraso menor. Calcule o valor de $L1$.

Solução:

$$\begin{aligned} (21 + L1)(5 \times 10^{-6}) \text{ s} &= (4350 + L1)10^{-6} \text{ s} \\ L1 &= 1061.25 \end{aligned} \tag{5}$$

Extras

1. É possível que um protocolo X , da camada A da pilha de protocolos da Internet, realize transferência confiável de dados usando um protocolo não-confiável Y da camada B imediatamente inferior? Justifique.

Solução:

2. Quanto à comutação por circuitos ou por pacotes:

- a) Explique o funcionamento de ambas (não esqueça de falar dos recursos e do atraso de transmissão).

Solução:

- b) Contraste a maior vantagem de cada uma comparada com a outra.

Solução:

- c) Dê um exemplo de aplicação para a qual a comutação por circuitos é mais adequada e um para a qual a por pacotes é preferível.

Solução:

3. Mostre um algoritmo (usando máquina de estados ou pseudo-código) para um remetente realizando entrega confiável sobre IP e utilizando Go-Back-N com uma janela de tamanho fixo de 5 pacotes, um único timer e retransmissão rápida (ou seja, ao receber o terceiro ACK com o mesmo número de sequência o remetente reenvia os pacotes necessários). [Podia ser repetição seletiva ao invés de GBN, e controle de fluxo ao invés de retransmissão rápida.]

Solução:

4. Considere dois hosts A e B que estabelecem uma conexão TCP para troca de dados com $RTT=100ms$. O host A envia 8000 bytes para B, em pacotes de no máximo 2000 bytes a cada 50ms (o primeiro desses pacotes têm número de sequência igual a 100 e número de ACK igual a 50). A e B tem buffers de recepção de 6000 bytes que inicialmente estão vazios ($rwnd = 6000$). A camada de aplicação de B retira no máximo 1500 bytes desse buffer a cada 150ms (os primeiros 1500 bytes são retirado imediatamente quando chegam, antes do envio do primeiro ACK, e os próximos 1500 só 150ms depois). Para eventos que acontecem no mesmo instante, considere que os primeiro os dados são entregues à aplicação, depois os hosts tratam as mensagens recebidas (dados e ACKs) e só depois enviam suas mensagens, os ACKs nem mesmo aparentam ocupar o buffer de recepção de A. Mostre as trocas de mensagem entre A e B para que A possa enviar todos os 8000 bytes, indicando claramente em cada mensagem os números de sequência e de ACK, o tamanho de $rwnd$ e a quantidade de dados que está sendo transmitida. Recomendo indicar também quantos dados estão no buffer de recepção de B, quando são retirados e a quantidade de dados que A enviou e para os quais ainda não recebeu ACK.

Solução: