

Protocolos em Redes de Dados

Aula 08

Controlo da congestão e QoS

Luís Rodrigues

FCUL

2005-2006



Sumário

- ▶ O problema da congestão.
- ▶ Algoritmos e técnicas de controlo da congestão.
- ▶ Qualidade de serviço.



Problema da congestão

- ▶ Quando são injectados num caminho mais pacotes que este é capaz de suportar.
- ▶ A congestão leva a atrasos.
 - ▶ Que por sua vez pode levar à retransmissão de pacotes e aumentar ainda mais a congestão.



Congestão e controlo de fluxo

- ▶ Controlo de fluxo é feito ponto-a-ponto, entre nós terminais.
- ▶ Controlo da congestão é feito à rede como um todo.
 - ▶ Re-encaminhamento de pacotes, caminhos alternativos, dispersão de carga, etc.



Aproximação ao controlo da congestão

- ▶ Monitorizar o sistema para detectar o problema.
- ▶ Informar as entidades que podem iniciar acções correctivas.
- ▶ Corrigir a configuração do sistema.



Métricas para detectar a congestão

- ▶ Número de pacotes descartados por falta de espaço.
- ▶ Tamanho das filas de espera.
- ▶ Número de re-transmissões.
- ▶ Latência média.
 - ▶ Instabilidade versus reactividade.



Tipos de aproximação

- ▶ Baseado nos encaminhadores ou nos nós terminais.
- ▶ Baseado em reserva ou em re-alimentação.
- ▶ Baseado em janelas ou em taxas de transmissão.



Medidas correctivas

- ▶ Distribuir a carga:
 - ▶ Usar mais linhas ou re-encaminhar os pacotes.
- ▶ Diminuir a carga:
 - ▶ Diminuir a taxa de transmissão, aumentar o tempo entre re-transmissões, etc.



Gestão dos alarmes no TCP

- ▶ Manter uma aproximação da latência tendo por base o tempo que demoram a chegar confirmações.
- ▶ Ir ajustando o tempo dos temporizadores em função deste valor.

Gestão dos alarmes no TCP

- ▶ Problema:
 - ▶ Como emparelhar a confirmação com a transmissão (no caso de re-transmissão).
- ▶ Solução:
 - ▶ Só contabilizar pacotes transmitidos uma vez.

Gestão dos alarmes no TCP

- ▶ Fórmula aplicada (especificação original):
 - ▶ RRTE: “Round-trip-time” estimado.
 - ▶ RRTI: “Round-trip-time” instantâneo.

$$RTTE = \alpha RTTE + (1 - \alpha) RRTI$$

$$\text{Timeout} = \beta RTTE$$

Gestão dos alarmes no TCP

- ▶ Fórmula Jacobson/Karels:

$$\text{Diferença} = RRTI - RTTE$$

$$RTTE = RTTE + \delta \text{ Diferença}$$

$$\text{Desvio} = \text{Desvio} + \delta (|\text{Diferença}| - \text{Desvio})$$

$$\text{Timeout} = \mu RTTE + \phi \text{ Desvio}$$

Gestão dos alarmes no TCP

- ▶ Backoff: Aumentar o timeout quando é necessário retransmitir.
 - ▶ A cada retransmissão, o valor do “timeout” é duplicado.



Controlo da congestão no TCP

- ▶ O TCP mantém uma *janela de congestão* que consiste numa estimativa do número de octetos que a rede consegue encaminhar.
- ▶ Não envia mais octetos do que o mínimo da janela definida pelo receptor e pela janela de congestão.



Controlo da congestão no TCP

$$\text{MaxWindow} = \text{MIN}(\text{CongestionWindow}, \text{RecipientWindow})$$

$$\text{EffectiveWindow} = \text{MaxWindow} - (\text{LastByteSent} - \text{LastByteReceived})$$



Controlo da congestão no TCP

- ▶ Quando se perde um pacote a janela de congestão é dividida por dois.
- ▶ Sempre que chega uma confirmação, a janela é incrementada com uma percentagem do pacote máximo:
 - ▶ Incremento = Segmento * Segmento / Janela.
 - ▶ Janela nunca desce abaixo do segmento.



Controlo da congestão no TCP

- ▶ Arranque a frio:
 - ▶ A janela de congestão é iniciada com o tamanho do segmento.
 - ▶ No entanto incrementa a janela a cada confirmação.
 - ▶ Isto duplica a janela a cada “round trip”.
- ▶ Objectivo:
 - ▶ Chegar rapidamente ao máximo, sem enviar logo de início uma “rajada” de pacotes que pode não ser suportada pela rede.



Controlo da congestão no TCP

- ▶ O “arranque a frio” também é utilizado quando a ligação fica inactiva devido a perda de pacotes e espera por temporizadores.
 - ▶ Neste caso, a janela parte com o valor de 1 e é incrementada a cada confirmação até chegar ao valor que tinha imediatamente após o temporizador expirar.



Controlo da congestão no TCP

- ▶ Retransmissão rápida: método de diminuir os tempos de inactividade.
- ▶ Ao receber um pacote fora de ordem, o recipiente confirma o último pacote recebido pela ordem correcta.
 - ▶ Isto gera várias confirmações para esse último pacote.
- ▶ Receber várias confirmações para um pacote, é um sinal que o próximo se perdeu.
 - ▶ Uma retransmissão é iniciada após N confirmações repetidas (tipicamente três).



Evitar a congestão no TCP

- ▶ Tentar alterar a taxa de transmissão se não se estão a transmitir os pacotes esperados.

$$\text{MelhorDébitoEsperado} = \text{JanelaCongestão} / \text{MenorRTT}$$



Evitar a congestão no TCP

- ▶ Comparar o débito actual (medido num RTT) com o débito esperado.

$$\Delta = \text{MelhorDébitoEsperado} - \text{DébitoActual}$$

- ▶ Tentar manter a diferença dentro de um certo intervalo:

$$\alpha < \Delta < \beta$$

- ▶ Aumentar a janela se menor que α e diminuir se maior que β .



RED (Random Early Detection)

- ▶ Mecanismo usado para evitar a sincronização do controlo de congestão em múltiplas ligações TCP.
 - ▶ Os pacotes são descartados aleatoriamente quando um encaminhador começa a estar saturado.
 - ▶ Evita o backoff simultâneo de todas as ligações.



Notificação explícita de congestão

- ▶ Mecanismo experimental (opcional) que tenta aumentar a reactividade no controlo da congestão.
 - ▶ Os encaminhadores devem usar RED.
 - ▶ Em vez de descartarem os pacotes, encaminham-os com uma flag que indica congestão.
 - ▶ O receptor deve informar o emissor de modo a que este reduza a taxa de transmissão.



Qualidade de Serviço

- ▶ Serviços integrados.
- ▶ Serviços diferenciados.



Serviços Integrados (int-serv)

- ▶ Arquitectura global para a oferta de qualidade de serviço "end-to-end".
 - ▶ Arquitectura com vários componentes .
 - ▶ Definição de classes de serviço.
 - ▶ Protocolo de sinalização complementar (RSVP).
- ▶ Especificação de tráfego (Tspec):
 - ▶ Descreve as características do tráfego.
- ▶ Especificação da QoS pedida (Rspec):
 - ▶ Descreve as características pretendidas da rede.



Componentes da arquitectura

- ▶ Protocolos de sinalização.
- ▶ Policiamento de tráfego.
 - ▶ Verifica se o tráfego cumpre a sua especificação.
- ▶ Controlo de admissão.
 - ▶ Verifica se é possível satisfazer um pedido com os recursos disponíveis.
- ▶ Classificação:
 - ▶ Reconhecer os pacotes com requisitos de QoS.
- ▶ Escalonamento e gestão das filas de espera:
 - ▶ Definição das políticas que escolhem quais os pacotes a encaminhar e quais os pacotes a descartar.



Classes de serviço

- ▶ Débito garantido (guaranteed load):
 - ▶ Taxa garantida.
 - ▶ Atraso máximo garantido.
 - ▶ Ausência de perdas nas filas.
- ▶ Débito controlado (controlled load)
 - ▶ Não garante o total isolamento entre o tráfego mas...
 - ▶ Verifica se existem recursos para satisfazer o débito solicitado.
 - ▶ Usa uma gestão das filas de espera que minimiza a interferência entre fluxos.
- ▶ Melhor esforço (best-effort).

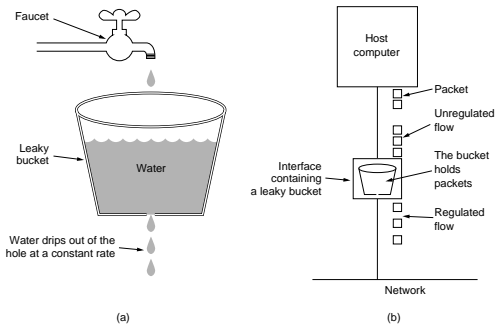


Especificação de fluxo

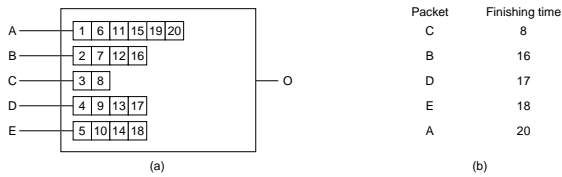
- ▶ Tspec:
 - ▶ Taxa máxima (peak rate).
 - ▶ Tamanho máximo do pacote.
 - ▶ Tamanho da rajada, b .
 - ▶ Taxa de "balde-de-testemunhos" (token-bucket rate), r .
 - ▶ Débito $(t) < rt + b$
- ▶ Rspec
 - ▶ Débito solicitado (service rate).



Aproximação "balde de testemunhos"



Round-robin



(a) A router with five packets queued for line O. (b) Finishing times for the five packets.

Fair queuing

- ▶ Mantêm-se diferentes filas de espera para diferentes fluxos.
- ▶ Independentemente do número de pacotes em cada fila, são processados pacotes de modo equitativo entre todas as filas (por exemplo, envia-se um de cada fila).
 - ▶ Algoritmo real ligeiramente mais complexo, pois os pacotes não possuem todos o mesmo tamanho.

Filas de espera ponderadas

- ▶ Cada fluxo possui um peso w_i (factor de ponderação)
- ▶ Cada fluxo consome uma percentagem da largura de banda disponível:

$$R \frac{w_i}{\sum w_j}$$

- ▶ O atraso máximo para cada fluxo é aproximado por:

$$d_i^{max} = \frac{b_i}{R \frac{w_i}{\sum w_j}}$$

- ▶ Protocolo de sinalização.
- ▶ Permite configurar os encaminhadores para assegurarem uma determinada QoS.
- ▶ Os pacotes devem transportar:
 - ▶ Informação de classificação (que identifica o tráfego).
 - ▶ Especificações de tráfego e QoS pedida.
- ▶ Esta informação deve ser propagada desde o emissor até ao receptor.

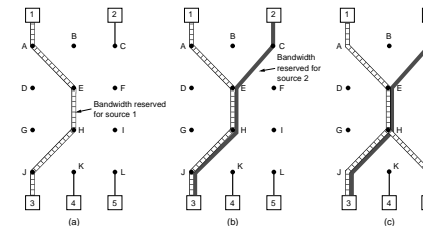
RSVP: mensagens mais importantes

- ▶ RESV:
 - ▶ Enviada no sentido inverso, dos receptores até aos emissores.
 - ▶ Inclui a Rspec.
 - ▶ Inclui uma Filterspec: permite seleccionar um subconjunto do tráfego (por exemplo, pacotes de um emissor específico em grupos de difusão com vários emissores).
- ▶ Também existem mensagens para libertar os recursos (*PathTear* and *ResvTear*).

RSVP: mensagens mais importantes

- ▶ PATH:
 - ▶ São propagadas do emissor até ao receptor.
 - ▶ Inclui a Tspec.
 - ▶ Também suporta difusão (o receptor pode ser um grupo).
 - ▶ No seu percurso o caminho é memorizado (cada encaminhador mantém estado, e a mensagem carrega o identificador do encaminhador a montante).
 - ▶ Nota: no IP multicast, o receptor não necessita de conhecer à partida a identificação dos emissores.

Protocolo de reserva de recursos



(a) Host 3 requests a channel to host 1. (b) Host 3 then requests a second channel, to host 2. (c) Host 5 requests a channel to host 1.

- ▶ Identificação de tráfego:
 - ▶ Concebido originalmente para micro-fluxos.
 - ▶ IP e porto de origem, protocolo, IP e porto de destino.
- ▶ Soft State:
 - ▶ As reservas são mantidas apenas temporariamente.
 - ▶ Devem ser reconfirmadas periodicamente.
 - ▶ Assegura que os recursos são libertados em casos de falha.

RSVP: reservas partilhadas

- ▶ O protocolo foi pensado para suportar aplicações com difusão:
 - ▶ Por exemplo, video-conferência.
- ▶ Vários emissores podem partilhar a mesma reserva (assumindo que transmitem em momentos diferentes).
- ▶ O receptor, na Filterspec, pode indicar que tipo de reservas pretende (partilhadas ou não e que tipo de selecção pretende fazer).

Fiabilidade

- ▶ Em princípio, a necessidade de refrescar as reservas periodicamente resolve eventuais perdas de mensagens durante o protocolo.
 - ▶ Recuperação muito lenta ou refrescamento demasiado frequente.
- ▶ O protocolo foi extendido para permitir a troca de confirmações entre nós consecutivos.
 - ▶ As mensagens possuem um identificador único.
- ▶ "Summary refresh" (SREFRESH):
 - ▶ Para refrescar uma mensagem, pode apenas enviar-se o seu identificador.
 - ▶ Um pacote de dados pode refrescar várias mensagens.

Suporte MPLS para QoS

- ▶ Facilita a identificação de fluxos:
 - ▶ Que passa a ser feita através das etiquetas em vez de se analisar os campos do pacote.
- ▶ O processo de selecção de etiquetas é integrado no RSVP.
 - ▶ As mensagens de RESV transportam a etiqueta reconhecida pelo next-hop.
- ▶ Permite utilizar uma reserva de recursos para diferentes categorias de fluxos (e não apenas uma ligação):
 - ▶ Por exemplo, todo o tráfego entre duas localizações de um cliente.

Serviços diferenciados (diff-serv)

- ▶ Não se fazem reservas para fluxos individuais.
- ▶ Classifica-se o tráfego num pequeno conjunto de categorias e reservam-se recursos para cada uma destas categorias.
- ▶ Utiliza-se o campo "Type of Service" para transportar um código identificador da categoria (6 dígitos):
 - ▶ Differentiated Services Code Point (DSCP).
- ▶ Per-Hop Behavior (PHB).
 - ▶ Suporte dado por cada encaminhador para DS (tipicamente, materializado por filas de espera diferenciadas e políticas de descarte de pacotes).



Comportamento "per-hop"

- ▶ Por omissão : melhor esforço.
- ▶ Encaminhamento expedito (Expedited Forwarding, EF).
 - ▶ O tráfego é controlado na fronteira de modo a não exceder um perfil.
 - ▶ O comportamento "per-hop" deve assegurar que não se verificam atrasos nas filas de espera.



Comportamento nas fronteiras

- ▶ Os encaminhadores na fronteira devem:
 - ▶ Classificar o tráfego.
 - ▶ Marcar o tráfego consoante as categorias.
 - ▶ Policiar e formatar o tráfego ("*meter*", "*shaper*" e "*dropper*").



Comportamento "per-hop"

- ▶ Encaminhamento garantido (Assured Forwarding, AF):
 - ▶ Os utilizadores podem escolher qual a qualidade de serviço de um pequeno conjunto de classes. Cada classe possui um perfil associado.
 - ▶ O cliente ou o fornecedor, marcam cada pacote do perfil com uma preferência.
 - ▶ O comportamento "per-hop" assegura que os pacotes marcados são descartados de acordo com a sua preferência.



Assured Forwarding

- ▶ Classe.
 - ▶ Identifica uma fila de espera.
 - ▶ Três dígitos.
 - ▶ Valores recomendados 1-4
- ▶ Preferência.
 - ▶ Identifica uma preferência no descarte de pacotes.
 - ▶ Três dígitos.
 - ▶ Valores recomendado 1-3

Sumário

- ▶ Controlo da congestão.
- ▶ Qualidade de serviço:
 - ▶ Serviços integrados.
 - ▶ Serviços diferenciados.

Sumário

Controlo da congestão

Controlo da congestão
no TCP
RED

Qualidade de serviço

int-serv
Fair queueing
RSVP
Suporte MPLS

diff-serv

Serviços diferenciados e MPLS

- ▶ A associação entre "fluxos" e etiquetas deve ser feita tendo em conta a categoria do tráfego.
- ▶ A etiqueta identifica implicitamente a classe de serviço.
- ▶ Alguns tipos de encapsulamento MPLS possuem dígitos reservados que permitem a passagem de informação sobre as categorias.
 - ▶ Isto facilita a associação de várias categorias à mesma etiqueta e o seu processamento diferenciado.

Sumário

Controlo da congestão

Controlo da congestão
no TCP
RED

Qualidade de serviço

int-serv
Fair queueing
RSVP
Suporte MPLS

diff-serv