

Relatório de Andamento: Notificação Explícita de Congestionamento em Protocolos Multicast

André Detsch

PIPCA- Programa de Pós-Graduação em Computação Aplicada
Centro de Ciências Exatas e Tecnológicas
Unisinos - Universidade do Vale do Rio dos Sinos
Av. Unisinos, 950 - São Leopoldo, RS - CEP93022-000
detsch@exatas.unisinos.br

1 Introdução

O uso de algoritmos para controle de congestionamento é fundamental para que os protocolos de comunicação possam coexistir em um ambiente de rede como a Internet. Por mais que se aumente a capacidade dos buffers, o poder de processamento e a velocidade dos links de comunicação, o problema do congestionamento continuará tendo que ser tratado ([12]).

Entretanto, sem suporte adequado da rede, um congestionamento só pode ser detectado pelo protocolo através do aumento no atraso dos pacotes (devido ao atraso de enfileiramento) ou pela detecção de perdas sistemáticas de pacotes [4]. Dentro deste contexto, a tecnologia de Explicit Congestion Notification (ECN) [6] aparece recentemente como uma forma de permitir aos hosts envolvidos na comunicação a detecção de um possível congestionamento na rede, antes mesmo que este congestionamento gere perdas de pacotes.

Este trabalho tem por objetivo investigar o uso de ECN para controle de congestionamento em protocolos multicast. Para tanto, será realizado um estudo bibliográfico sobre o assunto, juntamente com a avaliação do comportamento das abordagens propostas até o momento, culminando com simulações a serem realizadas no simulador ns-2.

O presente relato apresenta uma síntese dos assuntos estudados (seção 2) até o momento, bem como a descrição do andamento das etapas (seção 3) e expectativas para o andamento do trabalho (seção 4).

2 Revisão Bibliográfica

2.1 Controle de Congestionamento

Com a evolução das tecnologias de rede, a maior causa de perda de pacotes na Internet se deve ao congestionamento que ocorre em roteadores sobrecarregados ([13]). Quando o somatório dos fluxos que chegam a um roteador e devem ser redirecionados a um determinado link de saída excede a capacidade do mesmo, a fila de saída associada a esse link no roteador tende a encher e, mais cedo ou mais tarde, causar o descarte de pacotes por estouro de buffer.

Controle de congestionamento é um desafio na Internet, devido ao limitado grau de observação e controle da rede (particularmente, nas "pontas"). A principal forma de controle de congestionamento possível na Internet é o controle fim-a-fim do tráfego do usuário na camada de transporte. Este controle deve ser exercido utilizando apenas a observação limitada da rede que pode ser feita localmente, baseado na sua própria performance ([9]).

Perdas de pacotes são geralmente resultado da sobrecarga dos buffers dos roteadores quando a rede torna-se congestionada. Em protocolos com controle de erro baseado em ARQ (Automatic Repeat reQuest) ([11]), como é o caso da maioria dos protocolos atuais (incluindo-se aí o TCP), perdas são recuperadas através da retransmissão de pacotes. A retransmissão trata um sintoma do congestionamento da rede, mas não trata sua causa: um conjunto de remetentes tentando enviar dados a uma taxa muito alta. Para tratar a causa, são necessários mecanismos para diminuir a taxa de envio desses remetentes, de forma igualitária, quando ocorre congestionamento na rede ([13]).

Controle de congestionamento está relacionado à alocação de recursos na rede de forma que a mesma possa operar com nível de desempenho aceitável quando a demanda exceder ou estiver próxima de superar os recursos da rede. Estes recursos incluem a largura de banda dos links, tamanho dos buffers (memória) e a capacidade de processamento dos nodos intermediários. Sem mecanismos de controle, o throughput pode ser reduzido consideravelmente quando a rede estiver sobrecarregada ([12]).

A seguir são apresentados alguns conceitos importantes para o estudo do controle de congestionamento.

TCP Friendliness

Conexões TCP utilizam um algoritmo de aumento aditivo e diminuição multiplicativa (denominado AIMD, ou Additive Increase and Multiplicative Decrease) para regular o fluxo de dados sendo transmitido ([8]).

Em [7], fluxos não TCP são definidos como sendo amigáveis ao TCP quando “seu *throughput* a longo prazo não ultrapassa o *throughput* de uma conexão TCP nas mesmas condições”. Outra definição possível, apresentada em [18], está baseada no efeito de um fluxo não TCP sobre o fluxo TCP concorrente, e não na banda ocupada por este fluxo não TCP. Mais especificamente, um fluxo unicast é considerado amigável ao TCP se este não reduz o *throughput* de uma conexão TCP concorrente mais de que um outro fluxo TCP o faria. Para o caso de um fluxo multicast, estende-se a definição caracterizando-o como amigável ao TCP se, para cada par transmissor-receptor, a regra acima for obedecida.

É importante ressaltar que existe uma grande discussão sobre a definição de como o conceito de TCP *friendliness* se aplica ao caso de protocolos multicast. Uma alternativa à definição dada acima é permitir aos fluxos multicast utilizarem uma maior largura de banda que fluxos unicast, uma vez que fluxos multicast atendem a vários receptores ([18]).

Controle de Congestionamento Baseado em Janela x Baseado em Taxa

Um possível critério de classificação para esquemas TCP-*friendly* é se estes adaptam a largura de banda utilizada pelo seus fluxos de dados baseando-se em uma janela de congestionamento (congestion-window) ou na sua taxa de transmissão.

Algoritmos que pertencem a categoria baseada em janela, usam uma janela de congestionamento no transmissor ou no(s) receptor(es) para assegurar o TCP *friendliness*. Similar ao TCP, cada pacote transmitido consome um *slot* na janela de congestionamento, enquanto cada pacote recebido ou com recebimento confirmado libera um *slot*. O transmissor pode enviar um pacote apenas quando pelo menos um *slot* está livre. O tamanho da janela de congestionamento é aumentado na ausência de indicativos de congestionamento, e reduzido quando um congestionamento é detectado.

Controle de congestionamento baseado em taxa alcança TCP *friendliness* ao, dinamicamente, adaptar a taxa de transmissão de acordo com algum mecanismo que indique congestionamento na rede. Os algoritmos desta categoria podem ser subdivididos em esquemas AIMD simples e baseados em modelo (*model-based*). Esquemas AIMD simples imitam o comportamento do controle de congestionamento TCP. Isto resulta em um comportamento instável da taxa de envio de dados, o que torna este grupo de algoritmos desaconselhável para a transmissão de *streams* contínuas de mídia. Algoritmos baseados em modelo usam um modelo de TCP, como o apresentado em [14], ao invés de utilizar diretamente o mecanismo empregado pelo TCP. Sua característica básica é regular a taxa de transmissão do fluxo tendo um prazo mais longo como referência, alcançando resultados menos instáveis e, considerando a transmissão como um todo, atendendo aos requisitos estabelecidos para ser considerado um fluxo amigável ao TCP. Isto torna este tipo de protocolo mais adequado ao tipo de aplicação acima citado.

Controle de Congestionamento *End-to-end* x *Router-supported*

Muitos dos esquemas amigáveis ao TCP propostos são desenvolvidos para redes IP “*best effort*”, que não possuem nenhum mecanismo adicional de suporte ao protocolo por parte dos roteadores, de forma que as únicas referências para detecção de congestionamento são a perda de pacotes e o aumento no atraso (devido ao atraso de enfileiramento). Estes esquemas são denominados controle de congestionamento *end-to-end*, e podem ser empregados diretamente na Internet atual.

Entretanto, o desenvolvimento de protocolos de controle de congestionamento e, em especial, o compartilhamento justo de recursos pode ser consideravelmente facilitado pela colocação de “inteligência” na rede, tipicamente, nos roteadores ([18]). Esquemas de controle de congestionamento que dependem de funcionalidade adicional por parte da rede são denominados *router-supported*.

2.2 Notificação Explícita de Congestionamento

A dependência de perdas de pacotes para detecção de congestionamento (como no caso de TCP) é apropriada para o caso de redes cujos roteadores apenas direcionam pacotes para a porta de saída apropriada. Neste cenário, o congestionamento leva a perdas devido a overflow nas filas, perdas estas que, uma vez detectadas, levam a uma redução da geração de fluxo por parte do transmissor dos dados. O mecanismo de *random-early-detection* (RED) ([17]) visa evitar que as filas fiquem completamente cheias antes de os transmissores reduzirem sua taxa de transmissão. Para isso, o roteador deve possuir um mecanismo para detectar congestionamento em formação (por exemplo, verificando que a ocupação de suas filas ultrapassou um determinado limite). Uma vez detectado este tipo de situação, o roteador passa a, aleatoriamente, descartar pacotes. Desta forma, perdas ocorrem e são detectadas pelos transmissores antes que as filas fiquem completamente cheias, possibilitando que sejam tomadas medidas contra o congestionamento com maior antecedência.

Roteadores com mecanismos para detectar congestionamento em formação não estão limitados a descartar pacotes como forma de informar aos transmissores esta situação. Mensagens de *Source Quench* ([15]) podem ser geradas como forma de informar rapidamente ao transmissor do pacote que este se encontra em um roteador em processo de congestionamento. Este mecanismo permite que a sinalização de congestionamento seja rapidamente informada ao transmissor, mas a geração de mensagens extras por parte do roteador pode ser prejudicial em uma rede já congestionada. Outra forma de notificar aos participantes da transmissão a ocorrência de congestionamento é através da marcação do bit de ECN (*Explicit Congestion Notification*, notificação explícita de congestionamento) presente no cabeçalho dos pacotes IP ([15]). Isto evita a geração de pacotes adicionais, mas pode retardar a reação dos protocolos de comunicação envolvidos, pois o pacote marcado chegará ao receptor, e não diretamente ao transmissor do fluxo de dados.

De forma geral, mecanismos de ECN previnem descartes desnecessários de pacotes, permitindo um melhor aproveitamento da rede. Além disso, possibilita aos protocolos a determinação não ambígua de que há um congestionamento em algum ponto da rede, diferentemente do que ocorre quando uma perda é detectada, uma vez que esta pode ter ocorrido por erro em algum link. Mecanismos de controle de congestionamento que fazem uso de ECN muitas vezes são classificados como prevenção de congestionamento (*congestion avoidance*), como em [5], já que não é preciso ocorrer perdas de pacotes para que o fluxo de dados seja ajustado.

[6] estuda o impacto do uso de ECN em comunicações TCP. Observou-se que sua principal utilidade é em aplicações sensíveis ao atraso de pacotes individuais, como no caso de telnet, onde o processo retransmissão de um pacote acarreta em perda na interatividade. Já no caso de aplicações de transmissão maciça de dados (*bulk-data*), o atraso adicional de pacotes durante a transmissão não é tão prejudicial. Entretanto, mesmo nestes casos, o uso de ECN representa um ganho em termos de eficiência da utilização da rede, principalmente em cenários com alto índice de congestionamento.

2.3 Controle de Congestionamento Multicast

Assim como no caso de protocolos unicast, é desejável que protocolos multicast sejam amigáveis ao TCP. Entretanto, o design de um esquema de controle de congestionamento multicast envolve uma série de questões não presentes no caso de protocolos unicast. Esquemas de controle de congestionamento multicast

devem ser escaláveis para um grande número de receptores, devendo também, ser capaz de lidar com a heterogeneidade de condições de rede dos mesmos. Por exemplo, se a taxa de transmissão for a mesma para todos os receptores, deve-se cuidar como a taxa será reduzida no caso de congestionamento da rede. Se o transmissor reage a cada perda diminuindo a janela, a taxa de transmissão acabará sendo reduzida de maneira demasiada. Este problema é conhecido como *loss path multiplicity problem* [18]. Se o ajuste da taxa de transmissão for baseado, não em apenas um receptor, mas em toda uma árvore de distribuição, o desempenho pode cair consideravelmente caso o protocolo não tenha sido projetado corretamente ([18]).

Em [10] são analisados diversos aspectos de controle de congestionamento multicast de forma geral. Uma conclusão importante deste trabalho, é a de que enquanto protocolos que empregam mecanismos de controle de congestionamento baseados em janela podem ser amigáveis ao TCP sem precisar de estimativas de RTT, protocolos que utilizam controle de congestionamento baseados na taxa de transmissão precisam ter esta estimativa para alcançar este objetivo. Este fato tem importância especial para o caso de protocolos multicast, pois em muitos casos, o processo de estimação destes valores podem levar a uma perda na escalabilidade ([18]).

Existe uma grande diversidade de características e restrições nas aplicações multicast, o que faz com que um simples esquema de controle de congestionamento não seja capaz de atender a todas elas ([10]). Como reflexo desta necessidade, diversos esquemas de controle de congestionamento multicast foram propostos, os quais podemos, inicialmente, classificar em dois grupos: *single-rate* (onde todos os receptores recebem os dados a uma mesma taxa) e *multirate* (onde os receptores podem receber os dados a taxas diferentes entre si).

Controle de Congestionamento *Single-rate*

Este tipo de controle de congestionamento multicast tem como característica básica a uniformidade na taxa de envio de dados aos receptores, ou seja, o fluxo de dados sendo enviado para todos eles é o mesmo. Isto faz com que a taxa de recebimento dos dados de todos os receptores esteja limitada a taxa que atenda os requisitos de TCP *friendliness* para o receptor gargalo.

Uma tática comum para a implementação deste tipo de controle, é a escolha do receptor mais lento como uma espécie de “guia” (*acker*), cujo comportamento (em termos de perdas de pacotes) irá reger o ajuste da taxa de transmissão ([16]). É importante notar que esta escolha não é simples, até porque as condições da rede podem variar sensivelmente durante a transmissão.

Controle de Congestionamento *Multirate*

Controle de congestionamento *multirate* permite uma alocação mais flexível aos diferentes caminhos da rede. Este esquemas são mais escaláveis, pois se adaptam melhor a transmissões onde existe um número de receptores, o que tende a levar a uma maior heterogeneidade ([18]). Uma abordagem típica para controle de congestionamento *multirate* é o uso de multicast em camadas: o transmissor divide os dados várias camadas e transmite-as a diferentes grupos multicast. Cada receptor pode, individualmente, assinar quantos grupos forem possíveis, de forma a não exceder a capacidade do gargalo entre o transmissor e o receptor. Quanto mais grupos um determinado receptor assinar, melhor a qualidade da recepção. No caso de transmissões de vídeo, um grande número de camadas levará a uma melhor qualidade do vídeo, enquanto no caso de transmissões confiáveis de arquivos, mais camadas irão reduzir o tempo de transmissão.

Esta flexibilidade extra apresentada por este tipo de controle de congestionamento é compensada pela maior complexidade necessária nos esquemas que o seguem, além de um maior *overhead* em termos de largura de banda utilizada. Um exemplo de dificuldade enfrentada neste tipo de controle, é a sincronização dos receptores abaixo de um gargalo comum: de nada adianta alguns receptores deixarem de assinar um determinado grupo multicast, se os dados referentes a este grupo ainda precisam trafegar pelo gargalo para chegar a um receptor que não deixou de assinar o grupo.

Segundo [2], controle de congestionamento *multirate* é necessário para transmissões escaláveis para grandes audiências, por evitar o problema de estabelecer uma única taxa, ajustada de acordo com a demanda do receptor de menor taxa.

2.4 Controle de Congestionamento Multicast com uso de ECN

A utilização de ECN em protocolos multicast é um tópico de pesquisa recente. Além dos aspectos envolvidos no controle de congestionamento multicast *end-to-end*, este tópico trata com a busca de métodos eficientes e escaláveis para tratamento do feedback de congestionamento gerado pela rede, tanto no uso do *Source Quench*, quanto, principalmente, no uso do bit de sinalização no cabeçalho IP dos pacotes. Neste último caso, surgem questões como por exemplo, a maneira como os receptores enviarão esta informação ao transmissor, para que este tome as medidas necessárias.

[5] propõem um mecanismo de prevenção de congestionamento que utiliza tanto mensagens de *Source Quench* quanto o bit de congestionamento no cabeçalho IP. Com isso, o trabalho visa tanto minimizar o atraso na reação quando do início do processo de congestionamento quanto minimizar o overhead de controle. Uma característica singular deste trabalho, é a utilização do *router RTT* (RTT entre o transmissor e o roteador em processo de congestionamento). Este valor é obtido a partir dos pacotes de *Source Quench* gerador pelo roteador em questão. O esquema descrito é aplicável para protocolos unicast e multicast (*single-rate*). Foram realizadas simulações no ns-2 que atestaram a eficiência do modelo proposto para cenários de pequena escala.

[3] apresenta um algoritmo de controle de congestionamento *multirate* descentralizado baseado em taxa. Como forma de evitar congestionamento, é utilizada a marcação do bit de congestionamento no cabeçalho IP. O estudo utiliza análise numérica para analisar as características de *friendliness* e estabilidade do modelo.

3 Andamento das etapas

Revisão bibliográfica

Durante o semestre, foi realizada a leitura de artigos sobre controle de congestionamento multicast, controle de congestionamento utilizando ECN e trabalhos envolvendo os dois tópicos. O trabalho neste sentido segue, visando focar em artigos relacionados a este último item. Estes estudos vêm a complementar os conhecimentos já adquiridos ao longo do trabalho como bolsista de iniciação científica na área de multicast confiável.

Determinação dos modelos para simulação

Até o presente momento, a escolha dos experimentos a serem realizados ainda está em aberto. Entretanto, através da revisão bibliográfica e da experiência anterior, sabe-se que estes serão realizados no simulador ns-2 e espera-se que os envolvam:

- Adição de ECN em modelo(s) (realístico(s)) de protocolo(s) multicast existente(s), ou
- Adição de CC e ECN em modelos (abstratos) de protocolos multicast confiável.

Implementação e Simulação

O processo de implementação (e posterior simulação) dos modelos para este trabalho ainda não foi iniciado. Entretanto, ao longo do semestre (e do anterior), acumulou-se uma considerável experiência tanto na implementação de protocolos para simulação no ns-2 (implementações estas, que poderão servir de base para este trabalho) quanto na realização das simulações e aspectos teóricos (escolha de topologias, análise dos dados gerados) e práticos (execução de simulações em larga escala) relacionados.

4 Estimativas de andamento do trabalho

Em termos de pesquisa, o semestre que passou esteve focado na evolução do trabalho apresentado no SBRC 2002 ([1]). Desta forma, à realização das tarefas envolvidas exclusivamente com o trabalho de conclusão

foi dada menor prioridade. Entretanto é planejado para o semestre que se inicia, dar atenção quase integral ao trabalho de conclusão, posto que será cursada nenhuma outra disciplina neste período.

Referências

- [1] A. M. Barcellos, A. Detsch, "Avaliação de Desempenho de Protocolos para Multicast com Conhecimento de Grupo baseados em Polling" In: SBRC2002 - XX Simpósio Brasileiro de Redes de Computadores, 2002, Búzios. Porto Alegre: SBC, 2002. v.1. p.1 - 16.
- [2] J. Byers, M. Frumin, G. Horn, M. Luby, M. Mitzenmacher, A. Roetter, W. Shaver. "FLID-DL: Congestion Control for Layered Multicast". Proceedings of NGC2000, pages 71-81, November 2000.
- [3] S. Deb and R. Srikant, "Congestion Control for Fair Resource Allocation in Networks with Multicast Flows." IEEE Conference on decision and control. December 2001.
- [4] D. DeLucia, K. Obraczka, "Congestion Control Performance of a Reliable Multicast Protocol", Proceedings of the IEEE ICNP'98, August 1998.
- [5] A. Dracinschi, S. Fdida, "Congestion Avoidance for Unicast and Multicast Traffic", IEEE European Conference on Universal Multiservice Networks, ECUMN'2000, October 2000.
- [6] S. Floyd, "TCP and Explicit Congestion Notification", ACM Computer Communication Review, V. 24 N. 5, October 1994, p. 10-23. URL "ftp://ftp.ee.lbl.gov/papers/tcp_ecn.4.ps.Z". <http://citeseer.nj.nec.com/floyd94tcp.html>
- [7] S. Floyd and K. Fall, "Promoting the Use of End-to-end Congestion Control in the Internet", IEEE/ACM Trans. Net., vol. 7, no. 4, Aug. 1999, pp. 458-72.
- [8] S. Floyd, Mark Handley, Jitendra Padhye, and Joerg Widmer, "Equation-Based Congestion Control for Unicast Applications", ACM SIGCOMM 2000, Stockholm, Aug 2000.
- [9] S.J. Golestani and S. Bhattacharyya, "A Class of End-to-End Congestion Control Algorithms for the Internet," in Proceedings of ICNP, 1998.
- [10] S. J. Golestani and K. K. Sabnani, "Fundamental Observation on Multicast Congestion Control in the Internet", Proc. IEEE INFOCOM, New York, NY, Mar. 1999, vol. 2, pp. 856-63.
- [11] G. J. Holzmann, "Design and Validation of Computer Protocols", Prentice Hall Software Series, 1991.
- [12] R. Jain, "Congestion Control in Computer Networks: Issues and Trends", IEEE Network Magazine, v. 4, n. 3, May 1990, pp. 24-30.
- [13] J. F. Kurose and K. W. Ross, "Computer Networking - A top-down approach featuring the internet", Addison-Wesley.
- [14] J. Padhye et al., "Modeling TCP Reno Performance: A Simple Model and Its Empirical Validation", IEEE/ACM Trans. Net., vol. 8, no. 2, Apr. 2000, pp. 133-45.
- [15] K. Ramakrishnan, S. Floyd. A proposal to add Explicit Congestion Notification (ECN) to IP. <ftp://ftp.isi.edu/in-notes/rfc2481.txt>.
- [16] L. Rizzo, "pgmcc: A TCP-friendly Single-Rate Multicast Congestion Control Scheme," Proceedings of Sigcomm'2000, Stockholm, August 2000.
- [17] K. Ramakrishnan and R. Jain, "A Binary Feedback Scheme for Congestion Avoidance in Computer Networks", ACM Transactions on Computer Systems, Vol. 8 No. 2, pp. 158-181, 1990.
- [18] J. Widmer et al., "A Survey on TCP-Friendly Congestion Control", IEEE Network, May 2001, pp. 28-35.