

UM MECANISMO DE CONTROLE ADAPTATIVO COM REQUISITOS DE JUSTIÇA

Ricardo Ferreira Martins¹, Cristian Koliver², Jean-Marie Farines³

¹Universidade do Estado de Santa Catarina - UDESC
Departamento de Ciência da Computação
Caixa Postal 631 – 89.223-100 Joinville, SC

²Universidade de Caxias do Sul - UCS
Departamento de Informática
Caixa Postal 1352 – 95.001-970 Caxias do Sul, RS

³Universidade Federal de Santa Catarina - UFSC
Departamento de Automação e Sistemas
Caixa Postal 476 – 88.040-900 Florianópolis, SC

rffm@joinville.udesc.br, ckoliver@ucs.br, farines@das.ufsc.br

Abstract. *This paper describes an adaptive control scheme for distributed multimedia applications, dealing with the receiver heterogeneity problematic. The controllers are deployed over the network, where all of them participate actively in the adaptation and are synchronized. They adapt the distributed multimedia flow, according to the end-to-end network conditions as perceived by the receivers. The adaptation is based on simple filtering techniques, and a synchronization between controllers is needed to indicate the maximum flow quality that a network node can receive. We also evaluate the influence of some network resource limitations on the proposed scheme. Our simulation results show the feasibility of our approach for congestion control and allow to obtain information about controller deployment and synchronization.*

Resumo. *Este artigo descreve um esquema de controle adaptativo para aplicações multimídia distribuídas, tratando da problemática da heterogeneidade dos receptores. Os controladores são localizados ao longo da rede, sendo que todos eles participam ativamente da adaptação e são sincronizados. Eles adaptam o fluxo multimídia distribuído de acordo com as condições fim-a-fim da rede, percebidas pelos receptores. A adaptação é baseada em técnicas simples de filtragem, e uma sincronização entre os receptores torna-se necessária para indicar a qualidade máxima do fluxo que uma rede pode receber. Também é feita uma avaliação da influência de algumas limitações de recursos na rede sobre o esquema proposto. Os resultados das simulações mostram a aplicabilidade da proposta para o controle de congestionamento, permitindo ainda obter informações sobre a localização e sincronização dos controladores.*

1. Introdução

A qualidade das Aplicações Multimídia Distribuídas (AMD's) é normalmente representada por parâmetros de Qualidade de Serviço (QoS – *Quality of Service*), sendo que tais parâmetros caracterizam os serviços associados a uma determinada camada (por exemplo, as camadas de rede e de aplicação).

Atualmente, a maioria das AMD's adotam a Internet como plataforma de rede, e estas aplicações são executadas normalmente sobre sistemas operacionais de propósito geral, desenvolvidos para a transmissão e o processamento de dados através do modelo de melhor esforço. Sendo assim, não há garantias de QoS, pois devido a situações de congestionamento na rede, pacotes podem ser descartados ou recebidos com um atraso. Quando o atraso de recepção de pacotes, bem como a variação do atraso (*jitter*), são observados de forma relevante, então os receptores poderão observar perdas (por exemplo, amostras de áudio ou quadros de vídeo). Tais perdas resultam em conversações com longas paradas entre as frases ou sentenças, ou no congelamento da imagem. Portanto, estas situações afetam a qualidade percebida e, conseqüentemente, a satisfação dos usuários.

Dentre as formas existentes para a realização do controle adaptativo de AMD's, duas se destacam por estarem diretamente relacionadas com os mecanismos de codificação. A primeira é baseada no controle de vazão, onde a adaptação ocorre no sentido de alterar as características do fluxo, através da interação (no emissor) entre o controlador adaptativo e a aplicação, ou através de dispositivos localizados entre o emissor e os receptores, podendo operar como filtros ou transcodificadores. A segunda forma em destaque é baseada na codificação do fluxo multimídia em níveis hierárquicos (também conhecidos como camadas), onde o controlador adaptativo seleciona os níveis mais adequados.

Visando o tratamento do problema descrito acima, vários mecanismos de controle adaptativo foram propostos. A maioria destes mecanismos são baseados em um único controlador [4, 5, 10, 11, 16] localizado no emissor e que ajusta a vazão da AMD de acordo com a carga da rede. Este tipo de mecanismo é fácil de ser implementado, mas garante resultados satisfatórios somente quando os receptores são homogêneos. Por um lado, se a vazão da AMD é mantida elevada, então a degradação descontrolada da qualidade, observada nos receptores mais lentos, irá persistir. Por outro lado, se a vazão da AMD é adaptada de acordo com o caminho fim-a-fim do receptor mais lento, então os caminhos fim-a-fim dos receptores mais rápidos poderão ser sub-utilizados. Adicionalmente, este tipo de esquema pode necessitar de um tráfego pesado de mensagens de sinalização/controlado, podendo levar a uma explosão de realimentações.

Devido às limitações do modelo centralizado, pesquisas recentes estão sendo feitas sobre esquemas de controle adaptativo descentralizado ou assistidos pelos roteadores. Vários estudos são baseados na transmissão *multicast* em camadas [3, 8, 18], onde um conjunto de algoritmos de compressão divide os dados em um número de camadas, organizada de uma forma hierárquica, sendo que as informações mais importantes são transmitidas através da camada mais inferior (camada de base), e as camadas superiores aprimoram estas informações.

Em [18], é descrito um esquema descentralizado através de um controle de congestionamento baseado em fluxos *multicast* codificados em camadas. Esta implementação utiliza o conceito de Redes Ativas, onde cápsulas de dados transportam uma indicação do tipo de camada, e cápsulas de inscrição transportam solicitações de inscrição/cancelamento de camadas. No caso de congestionamentos de curta duração, cápsulas de dados são descartadas nos roteadores, de acordo com a importância relativa observada entre as camadas. No caso de congestionamentos de longa duração, os ramos da árvore *multicast* são podados. Esquemas similares são apresentados em [3], [7] e [8], também restritos à transmissões *multicast* codificadas em camadas.

Esquemas mais flexíveis, usados para o controle de congestionamento realizado nos nós intermediários, são propostos em [1], [9], e [12]. Eles consideram vários tipos de operações de filtragem, tais como aqueles descritos em [19] (descarte de quadros, filtragem passa-baixa, redução de cor, transcodificação e requantização). No entanto, o impacto das limitações de recursos da rede sobre os esquemas propostos não é analisado, sendo que a aplicação de tais esquemas sobre toda a rede não é uma alternativa viável, em virtude do *overhead* introduzido.

Neste artigo, propõe-se um modelo descentralizado para controle adaptativo, em um ambiente baseado no modelo de melhor esforço. As discussões feitas foram direcionadas a apresentar como os controladores podem ser posicionados na rede (levando em consideração as limitações de recursos existentes), e descrever o protocolo de troca de mensagens que é utilizado pelos roteadores para resolver o problema de sincronização. Foram utilizados controladores proporcionais simples, pois neste trabalho não se pretende estudar a escolha de um controlador com o melhor desempenho (estudos posteriores poderão ser feitos com este objetivo). O comportamento do modelo sincronizado também é avaliado, com relação à heterogeneidade dos receptores, ou mais especificamente, ao problema da justiça (*fairness*).

O restante deste artigo é estruturado da seguinte forma: a descrição do esquema de controle adaptativo descentralizado é apresentada na próxima seção; na Seção 3, são apresentados os aspectos relacionados ao número e localização dos controladores, relacionando-os com o esquema centralizado; a Seção 4 mostra os resultados obtidos através de um conjunto de simulações; a Seção 5 resume os resultados alcançados e apresenta propostas de trabalhos futuros.

2. Descrição do Esquema de Controle Adaptativo

O esquema de controle proposto neste trabalho está baseado na distribuição dos controladores, responsáveis pela adaptação dos fluxos multimídia distribuídos, nos nós intermediários da rede. A seguir, será descrito como é feita a associação de tais controladores aos roteadores, bem como a forma de atuação (síncrona) dos controladores.

2.1. O Controlador Associado ao Roteador

Visando tratar dos problemas de congestionamento na rede, é proposto o estabelecimento de um ou mais Módulos de Controle Multimídia (*MCM*) em alguns roteadores (mais especificamente, em seus sistemas operacionais). Estes módulos agem

como mecanismos adaptativos para fluxos multimídia distribuídos. Um mecanismo de realimentação é utilizado por este módulo para obter os valores dos parâmetros de QoS apropriados. A Figura 1(a) mostra o modelo geral de um roteador, enquanto que a Figura 1(b) mostra uma possível combinação deste módulo adicional com o módulo de controle do roteador (CM_{ROUTER}).

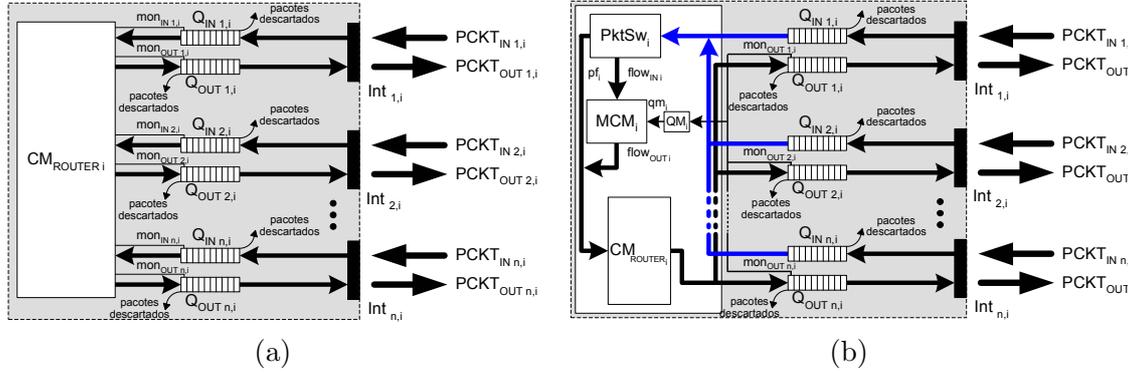


Figura 1: Modelo de um roteador i ((a) Geral e (b) Proposto).

Um roteador i possui n interfaces ($Int_{k,i}$, $k = 1, \dots, n$) que são conectadas a outros dispositivos (roteadores, *switches*, sistemas finais, etc.). Para cada $Int_{k,i}$, existem duas filas (fila de entrada – $Q_{IN_{k,i}}$, e fila de saída – $Q_{OUT_{k,i}}$) responsáveis pelo armazenamento temporário de pacotes ($PCKT_{IN_{k,i}}$ e $PCKT_{OUT_{k,i}}$) antes que eles sejam processados pelo CM_{ROUTER_i} ou transmitidos através das interfaces de saída. CM_{ROUTER_i} é normalmente responsável por diversas tarefas, envolvendo protocolos de roteamento, políticas de fila, etc. Ele também pode monitorar o comportamento de cada interface (por exemplo, a taxa de ocupação da fila de saída – $mon_{OUT_{k,i}}$).

O esquema proposto introduz uma modificação no roteador usual, adicionando um MCM simples que atua sobre todas as interfaces. Neste esquema, a adaptação do fluxo multimídia de saída é a mesma para todas as interfaces de saída. O MCM_i é localizado antes do CM_{ROUTER_i} porque é mais fácil realizar a comutação dos pacotes neste ponto. O módulo seletor ($PktSw_i$) divide os pacotes, antes do módulo MCM_i , em: pacotes de fluxos paralelos (pf_i), representando os pacotes de todos os fluxos que não são controlados, e os pacotes do fluxo multimídia ($flow_{IN_i}$) que deve ser controlado. Um módulo de controle QM_i monitora as filas de saída do roteador i , com o objetivo de informar o seu estado para o módulo MCM_i . A ação de controle será programada para depois que todas as filas tenham sido monitoradas por QM_i .

Este esquema necessita de uma identificação prévia do fluxo multimídia $flow_{IN_i}$, e esta identificação pode ser obtida quando o fluxo multimídia inicia, através do uso de um agente que transporta as informações relevantes do fluxo para os nós que conseguem tratá-la (aqueles que possuem controladores).

Esta abordagem permite o uso de diferentes estratégias de adaptação sobre o fluxo multimídia, tais como: transcodificação, requantização e descarte de quadros. Com relação a última estratégia, cujas técnicas não consomem muito tempo de processamento, um outro esquema com um MCM em cada interface também é possível, mas trata-se de uma abordagem mais complexa, e não será tratada neste artigo.

A Figura 1 mostra um roteador com um módulo *MCM* que monitora apenas as filas do próprio roteador (foco deste trabalho). Todavia, um *MCM* pode, também, monitorar filas remotas de outros roteadores, às custas de um aumento de tráfego de controle.

O diagrama de bloco do *MCM* é mostrado na Figura 2(a). Como foi dito no início do artigo, o controlador utilizado neste artigo é do tipo proporcional, com t valores de ganho ($G_{v,i}$, $v = 1, \dots, t$). Este tipo de controlador foi escolhido pela sua simplicidade. No futuro, o desempenho do controlador deverá ser aprimorado, através de um estudo específico de algoritmos de controle.

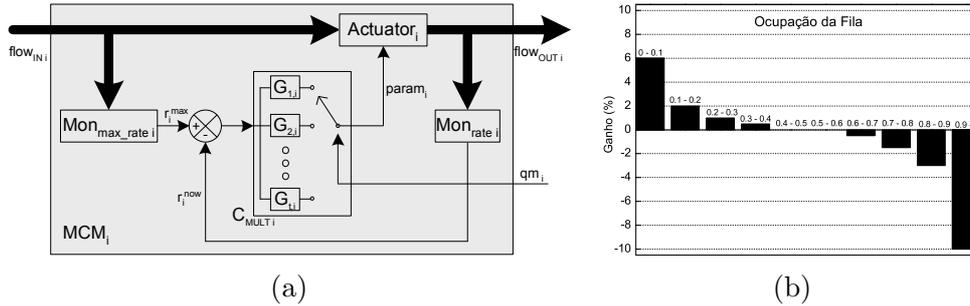


Figura 2: a) Diagrama de blocos do *MCM*; b) Parâmetros do controlador.

Como exemplo de configuração do controlador proporcional, a Figura 2(b) mostra um conjunto de dez parâmetros do controlador, onde os valores de ganho utilizados por um controlador são associados ao percentual de ocupação da fila. É possível observar que o controlador (C_{Multi}) utiliza como referência a taxa de transmissão do fluxo multimídia que chega no roteador ($flow_{IN_i}$). Existem dois monitores ($Mon_{max_rate_i}$ and Mon_{rate_i}) que obtêm as taxas dos fluxos $flow_{IN_i}$ e $flow_{OUT_i}$ (r_i^{max} e r_i^{now}). C_{Multi} também recebe a taxa de ocupação da fila (qm_i), indicando a carga sobre os links (detecção de congestionamentos de curto e longo prazo). A taxa qm_i é usada para comutar C_{Multi} para o valor $G_{v,i}$ apropriado. Além disto, C_{Multi} calcula e envia os m novos valores de parâmetros de QoS $param_i = \langle p_i^1, p_i^2, \dots, p_i^m \rangle$ para o atuador ($Actuator_i$). Então, $Actuator_i$, representado por um conjunto de filtros usados para alcançar o nível de QoS definido, adapta o fluxo de acordo com o estado da rede (estimado através da ocupação das filas de saída envolvidas), gerando o fluxo multimídia de saída ($flow_{OUT_i}$). Os filtros podem somente reduzir a vazão do fluxo de saída.

2.2. Sincronização entre os Controladores

Neste esquema, é necessário sincronizar os controladores, visando garantir que controladores localizados no mesmo caminho fim-a-fim realizem a adaptação ao mesmo tempo, considerando que a unidade de tempo usada na sincronização está associada a cada ciclo de adaptação. Quando um controlador calcula os parâmetros ideais para o atuador, ele tem que garantir que esta ação é possível, e isto envolve a qualidade do fluxo que este controlador está recebendo (do nó localizado acima dele). Para alcançar este requisito, foi utilizada uma abordagem mestre/escravo, onde os módulos de controle MCM_i operam de forma síncrona. Portanto, são definidos dois tipos de controladores:

- controlador mestre: comanda os demais controladores abaixo dele, e possui um período pré-fixado. Após iniciar, ele sempre programa a próxima adaptação para depois de um período P . Em cada período (ciclo de adaptação), ele aciona o processo de adaptação, enviando uma mensagem *adapt* para os controladores seguintes. O controlador mestre é sempre o último de sua sub-árvore a modificar a taxa do fluxo multimídia.
- controlador escravo: um controlador escravo aciona a adaptação após receber uma mensagem *adapt*, que é encaminhada para os controladores seguintes, localizados na sub-árvore *multicast* na qual ele é identificado como raiz. Após receber todas as mensagens de resposta *done* dos controladores localizados abaixo dele (informando que a adaptação foi concluída até aquele nível), ele realiza a modificação da taxa do fluxo multimídia.

A Figura 3 descreve os algoritmos dos dois tipos de controladores (a diferença básica está no passo 4). A nova taxa de bits (novos valores de parâmetros de QoS $\langle p_i^1, p_i^2, \dots, p_i^m \rangle$) calculada por um MCM_i no passo 1 é usada para notificar os demais controladores abaixo dele (através das mensagens *adapt*; passo 2), e esta é a taxa de bits máxima, permitida para esta sub-árvore *multicast*. Então, uma mensagem *adapt* enviada de MCM_i para MCM_j transporta os valores limites dos parâmetros de QoS para MCM_j . No passo 3, MCM_i precisa calcular a taxa de bits novamente, antes de realizar a adaptação, já que as condições das filas certamente foram modificadas. Portanto, ele tem que levar em conta a taxa de bits definida pelos controladores localizados acima dele, e todas mensagens *done* que transportam os valores dos parâmetros de QoS dos controladores localizados abaixo dele (estes são os valores limitantes). No passo 4, caso seja um controlador mestre, ele finaliza o procedimento agendando a próxima adaptação ($t_{now} + P$). Se for um controlador escravo, uma mensagem *done* é enviada para o próximo controlador, localizado acima dele, porque ele já recebeu todas as mensagens de resposta esperadas.

-
- 1) Calcula a nova taxa de bits.
 - 2) Se existirem controladores escravos abaixo dele:
 - a. Envia os novos parâmetros para os controladores abaixo dele (mensagens *adapt*).
 - b. Aguarda pelas respostas (mensagens *done*).
 - 3) Recalcula e atualiza a taxa de bits.
 - 4) Mestre: Agenda o próximo ciclo de adaptação.
Escravo: Envia uma mensagem *done* para o próximo controlador acima dele.
-

Figura 3: Algoritmo dos controladores mestre e escravo.

Uma mensagem *done* enviada de MCM_j para MCM_i transporta os valores dos parâmetros de QoS configurados no roteador j , através de MCM_j ($\langle p_j^1, p_j^2, \dots, p_j^m \rangle$). Estes valores são obtidos através de uma fusão das mensagens *done*, recebidas por MCM_j e enviadas pelos controladores localizados abaixo de j . O mecanismo de fusão gera $p_{j \rightarrow i}^k$ ($k = 1, 2, \dots, m$), levando em consideração os valores máximos¹ de todas as mensagens *done*, recebidas por MCM_j e os seus próprios valores calculados, reduzindo o número de mensagens de controle.

Finalmente, $Actuator_i$ configura (através de filtragem) os valores dos parâmetros de QoS. Com este mecanismo, enquanto é garantido que os links mais

¹Representando a melhor qualidade, mas requisitando mais largura de banda.

descarregados serão atendidos, evita o desperdício de largura de banda. Se considerarmos que todos os roteadores em uma topologia possuem *MCM*, um deles seria o controlador mestre (próximo ao emissor), e o resto seria controladores escravos. Portanto, o período pré-estabelecido (P) para o controlador mestre seria definido pela Equação 1.

$$P = [2 * D_{max}] + [d_{max} * C_{max}] \quad (1)$$

Onde:

- D_{max} : o atraso máximo da rede, entre o controlador mestre e um controlador escravo.
- d_{max} : a distância máxima (número de saltos) entre o controlador mestre (incluindo ele mesmo) e um receptor.
- C_{max} : o tempo máximo de filtragem.

Todos estes tempos são estimados. O primeiro termo da Equação 1 está relacionado com a transmissão das mensagens *adapt* e *done*, e o último termo está relacionado com o tempo de filtragem, consumido por cada controlador escravo localizado no caminho mais longo.

3. Avaliação do Esquema de Controle Descentralizado

Na maioria dos casos, como na Internet, nem todos os roteadores podem suportar a implementação do controlador proposto. Portanto, o controlador não está presente em todos os roteadores. Na seqüência, com o objetivo de discutir esta questão, a topologia apresentada na Figura 4 será utilizada.

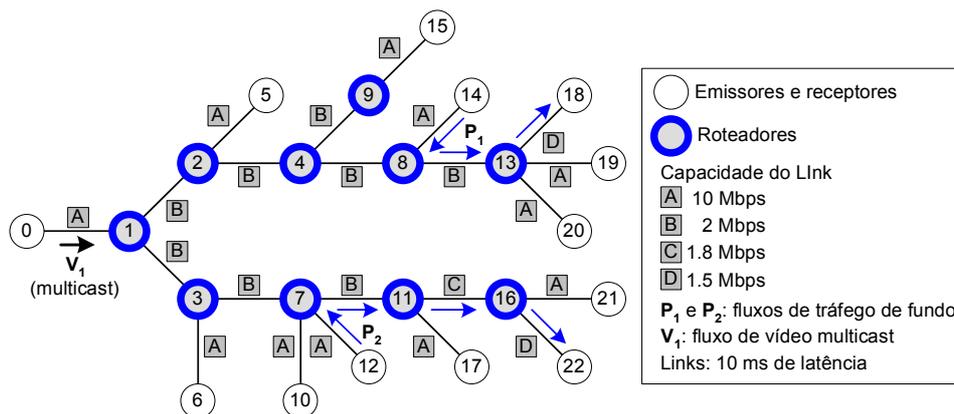


Figura 4: Cenário de uma rede.

Foram definidas quatro capacidades de *link*: duas para conexões locais (A e D), e duas para os links entre os roteadores (B e C). Existem 10 roteadores e 13 sistemas finais, incluindo a fonte de fluxo multimídia (V_1 no nó 0, transmitido para todos os nós, usando um grupo *multicast*) e duas fontes de tráfego de fundo (P_1 e P_2). A combinação das capacidades de *links* e do tráfego de fundo causam pontos de gargalo e, neste exemplo, existem 5 pontos de gargalo: 7, 8, 11, 13 e 16.

3.1. Localização dos controladores

Para avaliar a qualidade do fluxo multimídia nos receptores, foi assumido que a maior vazão significa a melhor qualidade. No entanto, também é preciso considerar as taxas de perda de pacotes, já que estas perdas podem implicar na degradação do fluxo multimídia e, conseqüentemente, na insatisfação dos usuários. Como poderá ser visto na seqüência, estes parâmetros (vazão e taxa de perda de pacotes) são sensíveis ao número e localização dos controladores. Os experimentos feitos definem três situações, mostradas na Figura 5 (emissores e receptores não são mostrados nesta figura).

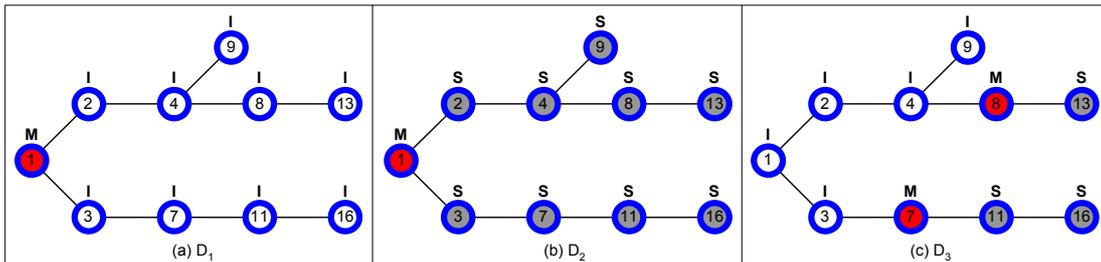


Figura 5: Três situações diferentes de localização dos controladores.

Na situação D_1 , mostrada na Figura 5(a), tem-se um controlador simples localizado no roteador 1. O objetivo da análise desta situação é comparar o esquema proposto com o esquema centralizado. De fato, a razão para estabelecer um controlador no nó 1 (e não no emissor) foi somente para facilitar as comparações.

Na Figura 5(b), representando a situação D_2 , o controlador mestre (M) é configurado no roteador 1, e controladores escravo (S) são configurados no restante dos roteadores. A vantagem desta abordagem é que o problema da heterogeneidade poderá ser tratado de uma melhor forma, pois um receptor em um caminho fim-a-fim descarregado irá receber uma melhor qualidade que qualquer outro em um caminho congestionado. No entanto, o parâmetro d_{max} da Equação 1 alcança o seu valor máximo, e uma adaptação pode levar muito tempo para ocorrer, dependendo da profundidade da árvore *multicast* (longos ciclos de adaptação não são interessantes).

Finalmente, na última situação (D_3), dois controladores mestre independentes são configurados nos nós 7 (com controladores escravo em 11 e 16) e 8 (com controladores escravo em 13). A vantagem do controle esparsos está na distância entre o controlador mestre e os receptores, e a distância entre o controlador mestre e os pontos de gargalo (o mestre pode estar bem próximo deles). Os nós 1, 2, 3, 4 e 9 não possuem qualquer controlador (I). Naturalmente, sem controladores em todos os roteadores, a heterogeneidade poderá não ser tratada completamente, dependendo dos nós onde é possível observar gargalos.

Em seguida, foi avaliado, através de simulações, o desempenho dos três casos: D_1 (abordagem centralizada), D_2 (controladores localizados em todos os roteadores) e D_3 (controladores localizados próximos aos nós de gargalo).

Para avaliar a qualidade do fluxo multimídia que chega nos receptores, foram utilizadas duas métricas propostas por [2]. Ambas estabelecem uma relação entre

a taxa de bits requerida por um receptor i (r_i) e a taxa de bits recebida (r_i^r), onde $r_i \geq r_i^r$. r_i é limitada à largura de banda máxima fim-a-fim que cada receptor pode observar. A primeira métrica é chamada *grau de satisfação de um receptor* (γ), e sua definição é dada pela Equação 2:

$$\gamma = \frac{r_i^r}{r_i} \quad (2)$$

Com o objetivo de avaliar a qualidade de uma forma global, foi utilizada a segunda métrica, chamada *degradação global* (δ), cuja definição é dada pela Equação 3:

$$\delta = \sum_{i=1}^R \delta_i \quad (3)$$

onde R é o número de receptores na sessão *multicast* e $\delta_i = (r_i - r_i^r)$.

4. Simulações e Resultados

É importante reforçar que o objetivo deste trabalho não é estabelecer o melhor controlador, mas avaliar a utilização do esquema de controle adaptativo proposto, bem como o mecanismo de sincronização. Para isto, foi feito um conjunto de simulações, usando o simulador de rede NS-2 [17]. As hipóteses a seguir foram adotadas em todos os experimentos.

- o tempo de filtragem consumido em cada controlador é igual a 10 milissegundos²;
- o fluxo multimídia é um tráfego *multicast* CBR, com uma taxa de transmissão máxima (emissor) de 1500 kbps, e uma taxa mínima de 350 kbps, iniciando no instante $t = 0$;
- as fontes de tráfego de fundo (Figura 6) seguem a função Pareto³ [14];
- todos os receptores na sub-árvore *multicast*, onde a raiz é um nó contendo um controlador, recebem a mesma qualidade, baseando-se na política do “pior caso” (priorizando os usuários com menos recursos); e
- cada simulação foi executada para um intervalo de 10 segundos.

O tráfego de fundo iniciado no nó 12 possui dois pontos ($t = 1.19$ e $t = 7.67$), onde vazão é nula por mais de um segundo, antes de ser comutada para o valor máximo (o valor médio é de 1000 kbps). O tráfego de fundo do nó 14 é iniciado em $t = 0.39$ e mantido a um valor médio de 700 kbps. Estes pontos são relacionados aos instantes que situações de gargalo poderão ser observadas.

²Este é um valor arbitrário, e o valor real depende da capacidade de processamento do roteador e das operações de filtragem.

³Uma fonte de tráfego Pareto gera um fluxo de acordo com uma distribuição *On/Off*, onde pacotes são enviados a uma taxa fixa durante os períodos de ativação (*On*), e nenhum pacote é enviado durante os períodos de desativação (*Off*). Estes dois períodos seguem uma distribuição Pareto, com pacotes de tamanho constante.

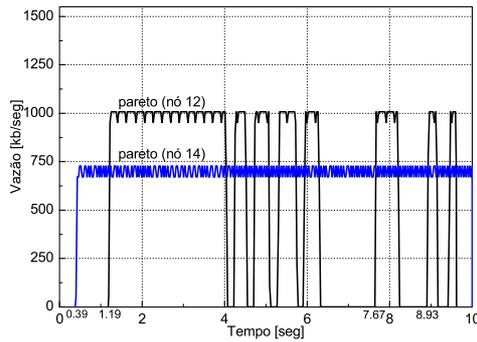


Figura 6: Tráfego de fundo iniciado nos nós 12 e 14.

Para todos os conjuntos de simulações (de acordo com as três situações apresentadas na Figura 5), foram avaliadas a vazão e o percentual de pacotes perdidos em todos os roteadores. Como foi dito anteriormente, o interesse foi maximizar o *tradeoff* entre a vazão e a taxa de perda de pacotes.

Através da utilização da Equação 1, os ciclos de adaptação foram definidos como:

- D_2 : $P = 120$ milissegundos; e
- D_3 : $P = 30$ milissegundos (para o controlador mestre do nó 8) e $P = 60$ milissegundos (para o controlador mestre do nó 7).

Para D_1 , a Figura 7(a) mostra a vazão observada em todos os receptores (abordagem centralizada). Para D_2 , as Figuras 7(b) e 7(c) mostram a vazão observada em cada nó controlado que está situado dentro da região de gargalo (nos nós 1, 2, 3, 4 e 9, não ocorrem perdas de pacotes e, portanto, a vazão é mantida em 1.5 Mbps). A figura relacionada aos nós 8 e 13 mostra o esforço em estabilizar a vazão por volta de 1.3 Mbps (para o nó 8) e 0.8 Mbps (para o nó 13), e estas são as larguras de banda disponíveis para os receptores conectados, considerando o tráfego de fundo e a política do “pior caso”. A figura relativa aos nós 7, 11 e 16 mostra a variação da vazão de acordo com a variação do tráfego de fundo (apresentada na Figura 6), já que cada controlador foi configurado para responder rapidamente à necessidade de reduzir a taxa de bits, e responder mais lentamente à necessidade de aumentar a taxa de bits. Isto é devido a necessidade de se fazer um teste cuidadoso quanto ao alcance de um ponto mais próximo possível do ponto de equilíbrio (valor ideal).

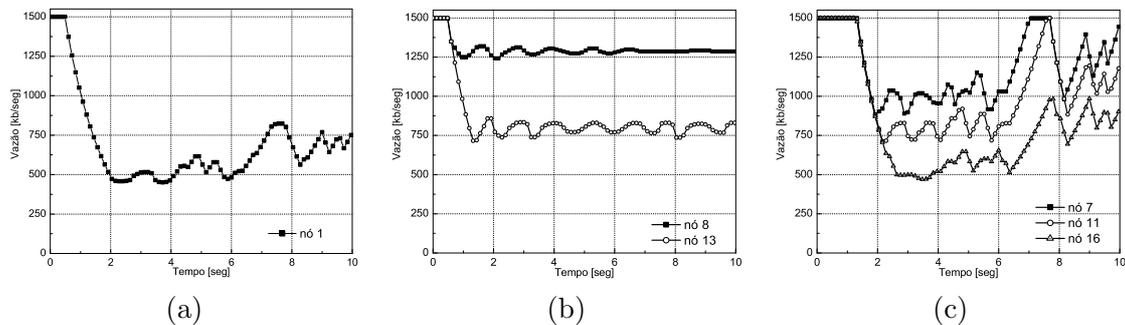


Figura 7: a) D_2 : Vazão no nó 1; b) D_2 : Vazão nos nós 8 e 13; c) D_2 : Vazão nos nós 7, 11 e 16.

O número de pacotes descartados em cada nó não foi relevante (menos de 5%). Mais ainda, os descartes de pacotes mais significantes foram observados nas mesmas situações ($t \cong 0.39$, $t \cong 1.19$ and $t \cong 7.67$), onde a vazão estava próxima do valor máximo (ou tentando alcançar este valor) e o tráfego pareto foi comutado (On).

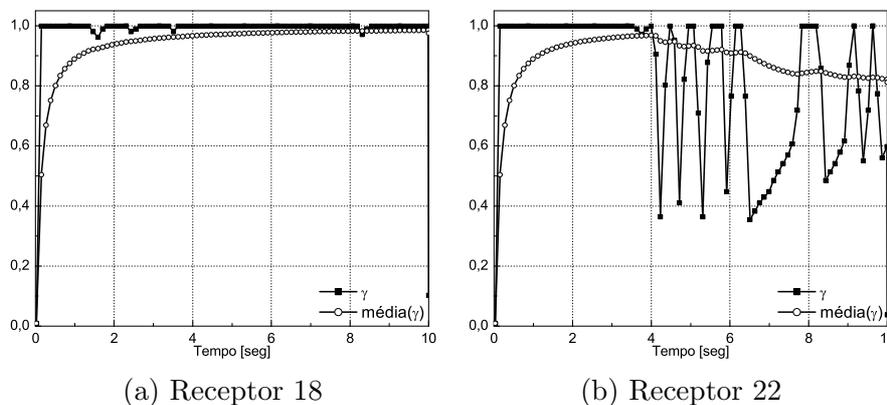


Figura 8: D_2 : Grau de satisfação para os receptores 18 e 22 (γ).

A Figura 8 mostra os valores⁴ do grau de satisfação γ somente para os receptores 18 e 22, em D_2 . Estes dois receptores estão em destaque por serem eles responsáveis pelos caminhos fim-a-fim que possuem os recursos mais restritos. Os bons resultados (do ponto de vista da largura de banda consumida e das perdas de pacotes) observados nestes receptores causam um impacto sobre os nós próximos deles (19, 20 e 21), devido à política do “pior caso”.

Esta abordagem não requer mecanismos de filtragem complexos e não aumenta o atraso fim-a-fim significativamente, se existe um número suficiente de nós envolvidos na transmissão, como é mostrado na Figura 9 (por exemplo, aplicações de conversação ao vivo necessitam a manutenção da natureza interativa, e o atraso e o *jitter* não podem ser superiores à 300 ms e 20 ms, respectivamente). Todavia, sua aplicação requer algoritmos de compressão específicos, e limita a qualidade da granularidade.

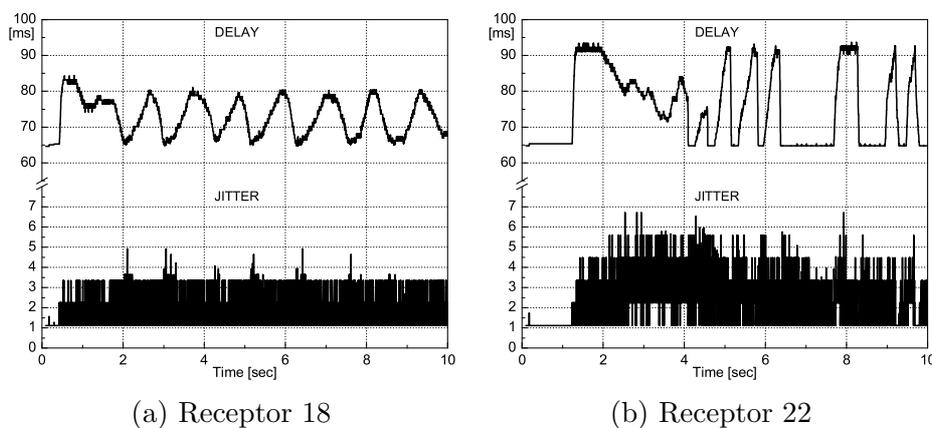


Figura 9: D_2 : Atraso e jitter para os receptores 18 e 22.

⁴O cálculo da média (Figuras 8 e 10) foi realizado levando-se em consideração a janela de valores formada deste o instante inicial até o instante no qual o cálculo é feito.

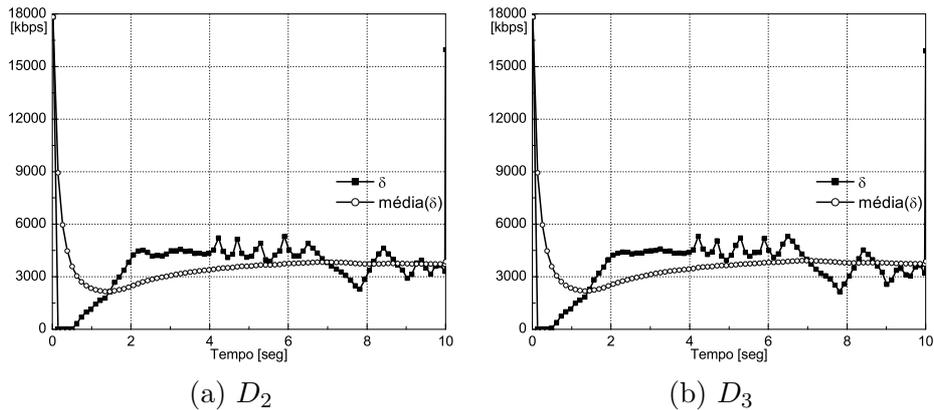


Figura 10: Degradação global para todos os receptores (δ), para D_2 e D_3 .

A Figura 10 apresenta os resultados da degradação global δ para D_2 e D_3 . O valor médio da degradação global, após um decréscimo no início da simulação, se mantém praticamente constante. De fato, o esquema proposto procura manter uma degradação global mínima e constante.

Para os casos D_2 e D_3 , as principais diferenças são:

- Em D_3 , existem controladores apenas próximos aos pontos de gargalo, e os receptores acima destes pontos sofrem uma variação de qualidade trivial (menos de 0,15%). A Figura 10(b) mostra a degradação global para todos os receptores;
- É possível calcular o percentual de degradação médio de um receptor, usando δ . A fórmula seria $\frac{\delta}{r_{mult}}$, onde r_{mult} é a taxa do fluxo multimídia. O percentual máximo alcançado em cada situação foi: 21.3% para D_2 e 21.7% para D_3 .

Analisando os resultados obtidos nas situações D_2 e D_3 , é possível observar que não há uma diferença significativa entre eles, o que implica em dizer que a situação D_3 possui um custo de implementação baixo (comparado com D_2), e um desempenho equivalente ao observado em D_2 . Outro aspecto importante está relacionado com a carga adicional gerada pelos pacotes de controle. O total de pacotes de controle não é significativo, comparado ao total de pacotes de dados (em todas as simulações, os pacotes de controle não ultrapassaram 2,85%). Este resultado pode ser comparado ao obtido com o protocolo RTCP, que utiliza 5% da taxa de dados nominal [13] com o auxílio de um limitador de envio de pacotes pelos receptores. Esta limitação se deve ao fato de que o aumento do tráfego RTCP está linearmente relacionado com o aumento do número de participantes da sessão *multicast*. Durante as simulações, não foi feita nenhuma limitação, mas esta seria solução interessante também para o esquema aqui proposto.

5. Conclusões

Neste artigo, foi descrito um mecanismo descentralizado para controle de congestionamento. O esquema proposto é baseado em um modelo sincronizado, que garante a adaptação de todos os controladores em um mesmo período (denominado ciclo de adaptação). Três diferentes situações também foram avaliadas, com o objetivo de lidar com diferentes condições de rede.

As simulações mostraram que o sistema responde bem aos cenários propostos, levando em consideração a política adotada (“piores caso”), e os resultados observados pelos receptores. A justiça é alcançada, considerando os nós da rede que possuem controladores implementados. Receptores conectados a um mesmo nó são tratados da mesma forma devido ao fato da ação do controlador, no caso estudado neste trabalho, ser executada da mesma maneira para todas as interfaces do roteador.

Todavia, é preciso realizar experimentos complementares de cenários de redes que seja próximos ao atual ambiente da Internet. É possível também implementar a abordagem proposta em redes baseadas em reserva de recursos (por exemplo, utilizando Serviços Integrados). Neste caso, a adaptação pode melhorar a qualidade dos links subutilizados. De fato, acredita-se que o controle de congestionamento não é suficiente para garantir uma qualidade mínima aceitável, já que a largura de banda liberada pelos controladores pode ser consumida pelas demais aplicações envolvidas. Estudos sobre adaptação e reserva de recursos se mostram como um caminho interessante.

Visando avaliar o atraso quando vários fluxos multimídia compartilham os mesmos *links*, é preciso considerar dois cenários: (1) uma sessão multimídia compartilhada entre vários membros; e (2) várias sessões multimídia compartilhando nós da rede. Para o primeiro cenário, o problema principal é a possibilidade de explosão de realimentações, devido ao tráfego de controle. Esta é uma possibilidade para qualquer esquema de controle de congestionamento, mas a descentralização e o mecanismo de fusão podem reduzi-la. Mais ainda, é possível reduzir o período de controle de acordo com o número de membros por sessão, como descrito em [15]. Para o segundo cenário, o problema é o atraso introduzido pelas operações de filtragem. Este atraso pode ser reduzido drasticamente através do estabelecimento de controladores nos *proxies*, no lugar dos roteadores [6], pois normalmente são máquinas potentes, como capacidades de processamento paralelo. O uso de operações simples de filtragem possui o problema da redução da granularidade da qualidade.

Finalmente, visando melhorar o desempenho, uma melhor escolha do tipo de controladores pode ser feita, sendo que este é um tópico para um próximo trabalho.

Referências

- [1] E. Amir, S. McCanne, and R. Katz. An active service framework and its application to real-time multimedia transcoding. *ACM SIGComm'98*, pages 178–189, September 1998. Vancouver - Canada.
- [2] M. D. Amorim, O. C. M. B. Duarte, and G. Pujolle. Application-aware Multicast. *IEEE Global Communications Conference - Globecom'01*, November 2001.
- [3] M. D. Amorim, O. C. M. B. Duarte, and G. Pujolle. Improving User Satisfaction in Adaptive Multicast Video. *IEEE/KICS - Journal of Communications and Networks*, 2002.
- [4] I. Busse, B. Deffner, and H. Schulzrinne. Dynamic QoS control of multimedia applications based on RTP. *Computer Communications*, 19(1):49–58, Jan 1996.
- [5] N. Duffield, K. Ramakrishnan, and A. Reibman. SAVE: an algorithm for smoothed adaptive video over explicit rate networks. *IEEE Transactions on Networking*, pages 717–728, April 1998.

- [6] M. Fry and A. Ghosh. Application Level Active Networking. *Computer Networks*, 31(7):655–667, May 1999. New York - USA.
- [7] K. Fukuda, N. Wakamiya, M. Murata, and H. Miyahara. On flow aggregation for multicast video transport. *IFIP 6th International Workshop on Quality of Service - IWQoS'98*, pages 13–22, May 1998. Napa Valley - USA.
- [8] P. A. S. Gonçalves, J. F. Rezende, and O. C. M. B. Duarte. An active service for multicast video distribution. *Journal of the Brazilian Computer Society*, pages 43–51, July 2000.
- [9] M. Hemy, U. Hengartner, P. Steenkiste, and T. Gross. MPEG systems streams in best-effort networks. *9th Packet Video Workshop (PV'99)*, October 1999. Ottawa - Canada.
- [10] D. Hull, A. Shankar, K. Nahrstedt, and J. W. S. Liu. An End-to-End QoS Model and Management Architecture. *IEEE Workshop on Middleware for distributed Real-Time Systems and Services*, pages 82–89, December 1997. San Francisco - USA.
- [11] V. Lakshman, P. P. Misshra, and K. K. Ramakrishnan. Transporting compressed video over ATM networks with explicit rate feedback control. *IEEE INFOCOM'97*, pages 38–47, April 1997. Kobe - Japan.
- [12] B. Metzler, T. Harbaum, R. Wittmann, and M. Zitterbart. AMnet: Heterogeneous Multicast Services based on Active Networking. *2nd Workshop on Open Architectures and Network Programming (OPENARCH'99)*, pages 98–105, March 1999. New York - USA.
- [13] K. W. Ross and J. F. Kurose. *Computer Networking: a top-down approach featuring the Internet*. Addison Wesley Press, 2001.
- [14] D. H. Von Seggern. *CRC Standard Curves and Surfaces*. CRC Press, December 1992.
- [15] D. Sisalem. End-to-End Quality of Service Control Using Adaptive Applications. *IFIP 5th International Workshop on Quality of Service - IWQoS'97*, May 1997. New York - USA.
- [16] D. Sisalem. Fairness of adaptive multimedia applications. *IEEE International Conference on Communications (ICC'98)*, June 1998. Atlanta - USA.
- [17] UCB/LBNL/VINT. *The Network Simulator - ns (version 2)*. <http://www.isi.edu/nsnam/ns/>, (May 2003), 2001.
- [18] L. Yamamoto and G. Leduc. An active layered multicast adaptation protocol. *2nd. International Working Conference on Active Networks - IWAN'2000*, pages 180–194, October 2000.
- [19] N. Yeadon, F. Garcia, D. Hutchinson, and A. Mauthe. Filters QoS support mechanisms for multipeer communications. *IEEE Journal on Selected Areas in Communications (JSAC)*, pages 1245–1262, September 1996.