

ANÁLISE COMPARATIVA DE ALGORITMOS DE ROTEAMENTO MULTICAST PARA COMUNICAÇÃO MULTIMÍDIA EM REDES ATM

Frank Meylan^(*), Luis Gustavo Gasparini Kiatake^(*), Marcelo Zanoni Santos^(*),
Sergio Takeo Kofuji^(*), Jean Pierre Courtiat^(**)

^(*) LSI-USP Laboratório de Sistemas Integráveis – Universidade de São Paulo
Av. Prof. Luciano Gualberto 158 tr.3 05508-900 São Paulo – SP – Brasil
Tel. +55 11 818-5589 Fax: +55 11 2114574
meylan, kiatake, kofuji, zanoni @ lsi.usp.br

^(**) LAAS-CNRS 7, avenue du Colonel Roche 31077 Toulouse Cedex 04 - France
Tel : +33 5 61 33 62 44 Fax : +33 5 61 33 64 11
courtia@laas.fr

Abstract

For a long time, the quality of the distributed multimedia applications was restricted by the limitations of the computer networking technologies. This scenery has changed a lot with the introduction of ATM (Asynchronous Transfer Mode). Characteristics like quality of service (QoS), support to different categories of traffic (voice, video and data), high scalability, among others, make ATM one of the most appropriate technologies for multiservice networks.

Another important characteristic is the support to multicast transmissions. In ATM, the use of multicast communication, as well as the resources' reservations are specified during the connection establishment phase. In this phase, connection establishment requests are routed among the switches in order to assure the necessary resources for the provision of the required communication services. In this case, the routing task involves aspects like quality of service (QoS), multicast communication, and resources reservation, becoming a complex task with specific characteristics.

This paper will analyze in details the multicast routing with QoS restriction, presenting a comparative study about the solutions that has been proposed to deal with this new problem. This study will cover theoretical analyses and simulations of four heuristics: KPP, CSPT, CCET and Hybrid. Parameters related to the order of complexity, cost of the multicast tree, delay, number of switches, execution time and maximum number of simultaneous multicast sessions will be evaluated, analyzed and compared.

Resumo

Durante muito tempo, a qualidade das aplicações multimídia distribuídas esteve restringida às limitações das tecnologias de rede utilizada para a interligação dos computadores. Este cenário mudou bastante com a introdução das redes ATM (Asynchronous Transfer Mode). Características como garantia de qualidade de serviço (QoS), suporte a diferentes tipos de tráfego (voz, vídeo e dados), alta escalabilidade, entre outras, fazem do ATM uma das alternativas mais adequadas para a implantação das chamadas redes multisserviço.

Uma outra característica importante é o suporte às transmissões multicast. Nas redes ATM, a utilização da comunicação multicast, bem como a reserva dos recursos são especificados durante a fase do estabelecimento da conexão. Nesta fase, requisições de estabelecimento de conexão são roteadas entre os comutadores de maneira a assegurar os recursos necessários para a provisão dos serviços de comunicação desejado. Neste caso, o roteamento em questão envolve aspectos como qualidade de serviço (QoS), comunicação multicast, reserva de recursos, tornando-se uma tarefa complexa e com características particulares.

Este artigo abordará em detalhes a questão do roteamento multicast com restrição de QoS, apresentando um estudo comparativo sobre as soluções que estão sendo propostas para tratar deste novo problema. Tal estudo será conduzido através de análises teóricas e simulações entre quatro heurísticas: KPP, CSPT, CCET e Híbrida. Parâmetros relativos à ordem de complexidade, custo da árvore multicast gerada, atraso inserido, número de comutadores, tempo de execução e número máximo de sessões multicast simultâneas, serão avaliados, analisados e comparados.

Palavras Chave: Roteamento Multicast, Redes ATM, QoS

1 Introdução

Recentemente, as aplicações multimídia distribuídas, como por exemplo vídeo sob demanda, ensino a distância, videoconferência, trabalho cooperativo, têm-se desenvolvido e disseminado de maneira acentuada. Tais aplicações possuem características e necessidades especiais que devem ser consideradas pela rede de transmissão, de forma a se obter um serviço com a qualidade esperada.

A transmissão de voz e vídeo em uma videoconferência, por exemplo, requer certa quantidade de banda disponível e preferencialmente garantida durante a existência da sessão, além de atraso controlado dentro de limites estabelecidos. Estas características ou exigências impostas pelas aplicações são chamadas de parâmetros de qualidade de serviço (QoS). De acordo com esses parâmetros e os recursos disponíveis na rede, é possível avaliar se um serviço pode ser prestado com a qualidade requisitada.

Estas aplicações envolvem, em muitos casos, vários participantes simultâneos em uma mesma sessão, como por exemplo cinco pessoas discutindo através de uma videoconferência. Isto exige da rede um mecanismo eficiente de comunicação *multicast*.

A tecnologia ATM apresenta algumas características que a destaca no suporte às transmissões multimídia, como altas taxas de transmissão, categorias de serviços diferenciadas e garantia de qualidade de serviço. Esta tecnologia é orientada a conexão, sendo que durante o estabelecimento desta, os recursos são alocados, desde a origem, passando por todos os equipamentos intermediários, até o destinatário, permitindo desta forma que a continuidade do serviço negociado seja garantida.

O roteamento nas redes TCP/IP é uma tarefa relativamente simples, uma vez que esta tecnologia não permite a atribuição de QoS e poucas são as aplicações que utilizam o conceito de *multicast*. Logo, os algoritmos de roteamento são relativamente simplificados, porém suficientes.

Nas redes ATM, o roteamento é executado durante o estabelecimento da conexão, sendo que a escolha das rotas poderá levar em consideração aspectos de qualidade de serviço, como banda requisitada e o atraso máximo suportado. Portanto, o roteamento torna-se uma tarefa mais complexa e custosa, exigindo algoritmos eficientes e elaborados.

Diversas propostas têm sido apresentadas com o intuito de resolver o problema de roteamento *multicast* quando parâmetros de QoS estão envolvidos. Porém, são escassos os trabalhos que comparam segundo uma metodologia definida estas propostas. Assim, este artigo abordará a questão do roteamento *multicast* com restrição de QoS, também conhecido como roteamento *multicast* restrito, apresentando um estudo comparativo sobre as soluções que estão sendo propostas para tratar deste novo problema. Este estudo será conduzido através de análises teóricas e simulações sobre as principais heurísticas descritas na literatura: KPP [Kompella93], CSPT [Sun95], CCET [Waters94]. Em especial, uma nova heurística de roteamento *multicast* proposta por Waters, heurística Híbrida [Waters97], será apresentada e comparada com a demais.

Esta heurística tem como característica inovadora o fato de agregar vantagens de outras heurísticas otimizadas para casos específicos do problema de roteamento *multicast* com QoS. Entretanto, esta não foi avaliada no trabalho original, segundo uma metodologia que possibilitasse sua comparação de uma forma abrangente com as heurísticas já existentes. Logo, este trabalho apresenta como contribuição principal uma análise detalhada da heurística Híbrida, assim como sua comparação com as heurísticas existentes mais importantes.

O artigo é estruturado da seguinte forma: a seção 2 apresenta os conceitos básicos sobre a representação do modelo de redes *multicast*, a seção 3 descreve o problema do roteamento *multicast* com aspectos de QoS envolvidos, a seção 4 descreve o simulador e a metodologia de análise, a seção 5 analisa os resultados obtidos através das simulações, na seção 6 são formuladas as conclusões e por último, na seção 7, serão indicados possíveis trabalhos futuros.

2 Conceitos Básicos

Para viabilizar o estudo da comunicação *multicast* em redes de computadores, é necessário que se utilize uma linguagem de representação do modelo em questão. Desta forma, uma rede de computadores será representada por um grafo conectado $N=(V,E)$, onde V representa um conjunto de nós e E representa um conjunto de enlaces entre estes nós. A existência de um enlace $e=(u,v)$ do nó u até o nó v , implica também na existência de uma ligação $e'=(v,u)$ para qualquer $u,v \in V$, ou seja, está sendo assumido que os enlaces são *full-duplex*.

Para todo enlace $e=(u,v) \in E$ são definidas uma função custo $C(e)$ e uma função de atraso $A(e)$. A **função custo** $C(e)$ pode ser entendida como custo monetário ou custo em função do tráfego existente em um enlace. Neste artigo a função custo estará associada ao tráfego existente na ligação, uma vez que o objetivo é estudar o gerenciamento eficiente dos recursos da rede. Desta forma, quanto maior for a utilização de uma ligação, maior será o seu custo. Neste trabalho também é assumido que **as ligações são assimétricas**, ou seja, $C(e) \neq C(e')$, dado que este é um fato real em redes de computadores. A **função atraso** $A(e)$ expressa o atraso que os pacotes adquirem ao passar por um enlace e . Este atraso pode ser atribuído ao tempo de transmissão, devido ao enfileiramento, roteamento ou comutação dos pacotes.

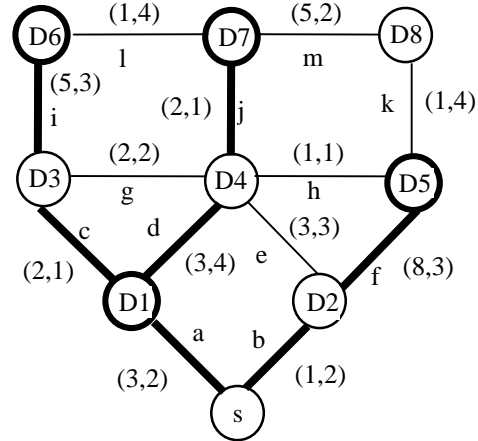


Figura 1 – Grafo G, (custo,atraso)

Uma sequência alternada de nós e enlaces é chamado **caminho**. Assim define-se um caminho $P(v_0, v_k) = v_0, e_1, v_1, e_2, \dots, v_{k-1}, e_k, v_k$ onde todo $e_i = (v_{i-1}, v_i) \in E, 1 \leq i \leq k$. Quando um caminho contém repetição de nós, diz-se que este possui ciclos (*loops*). Se não houver tal repetição, o caminho é livre de ciclos.

O **custo de um caminho** $P(v_0, v_k)$ é definido como a soma dos custos dos enlaces que constituem o caminho:

$$Custo(P(v_0, v_k)) = \sum_{e \in P(v_0, v_k)} C(e)$$

De acordo com esta definição, o custo de $P(s, D6)$, figura 1, pode ser calculado como: $C(P(s, D6)) = C(P(s, D1)) + C(P(D1, D3)) + C(P(D3, D6)) = 3 + 2 + 5 = 10$.

Analogamente, define-se o **atraso fim-a-fim do caminho** $P(v_0, v_k)$ como sendo a soma dos atrasos de cada enlace que constitui o caminho:

$$Atraso(P(v_0, v_k)) = \sum_{e \in P(v_0, v_k)} A(e)$$

O atraso do caminho $P(s, D6)$ é dado por: $A(P(s, D6)) = A(P(s, D1)) + A(P(D1, D3)) + A(P(D3, D6)) = 2 + 1 + 3 = 6$.

Um **grupo multicast** $G = \{g_1, g_2, \dots, g_n\} \subseteq V$, onde $n = |G| \leq |V|$, é formado por um conjunto de nós participantes de uma mesma sessão. Todo grupo *multicast* possui um nó fonte $s \in V$, que pode ou não pertencer a este grupo. Novamente na figura 1, o grupo *multicast* G é definido por $G = \{D1, D5, D6, D7\}$.

Uma **árvore multicast** $T(s, G) \subseteq E$ que possui uma fonte específica é um conjunto de enlaces que formam caminhos livres de ciclos com uma única fonte s , também chamada raiz, que está conectada, direta ou indiretamente, a todos os membros do grupo *multicast* G . A árvore que interliga os nós do grupo *multicast*, no exemplo da figura 1, é dada por $T(s,G)=\{a,b,c,d,f,i,j\}$.

Então, o **custo total da árvore** é dado pela soma dos custos dos enlaces dos caminhos que a formam:

$$Custo(T(s,G)) = \sum_{e \in T(s,G)} C(e)$$

Desta forma, o custo da árvore destacada na figura 1 é dado por $C(T(s,G))=C(a) + C(b) + C(c) + C(d) + C(f) + C(i) + C(j) = 3 + 1 + 2 + 3 + 8 + 5 + 2 = 24$.

E seu **atraso fim-a-fim máximo** é o atraso máximo da fonte até cada membro do grupo *multicast*:

$$Max_Atraso(T(s,G)) = \max_{g \in G} \left(\sum_{e \in P_T(s,g)} A(e) \right)$$

onde $P_T(s, g)$ é o caminho de s até g sobre a árvore $T(s, G)$. Em outras palavras, para se conhecer o atraso máximo da árvore *multicast*, deve-se calcular o atraso dos caminhos que interligam a fonte até cada membro do grupo, selecionando aquele que obtiver o maior valor. Assim $Max_Atraso(T(s,G)) = \text{Max}(A(P(s,D1)), A(P(s,D5)), A(P(s,D6)), A(P(s,D7))) = \text{Max}(2, 5, 6, 7) = 7$.

3 Heurísticas de Roteamento *Multicast* com QoS

A primeira experiência de transmissão *multicast* em redes de longa distância ocorreu em 1992, quando foi realizado pelo IETF (*Internet Engineering Task Force*) um *audiocast* para vinte participantes simultâneos em três continentes [Deering92]. A partir de então, muitas pesquisas estão sendo realizadas no sentido de desenvolver e aperfeiçoar algoritmos e protocolos de roteamento *multicast* que considerem as seguintes características:

- Parâmetros de qualidade de serviço exigidos pelas aplicações multimídia de tempo real;
- Gerenciamento eficiente dos recursos da rede;
- Escalabilidade.

3.1 Principais Heurísticas de Roteamento *Multicast* Restritas

Dentre as diversas heurísticas existentes, este trabalho se restringirá à análise das heurísticas de atraso restrito, mais adequadas para as aplicações multimídia de tempo real. As heurísticas de atraso restrito têm como objetivo criar uma árvore *multicast* ligando todos os participantes do grupo, sendo que esta árvore tenha o custo mínimo e ao mesmo tempo respeite o limite de atraso máximo definido. Utilizando uma linguagem mais formal:

$$\begin{cases} Atraso_Max(T(s,G)) \leq \Delta \\ Custo(T(s,G)) = \text{mínimo} \end{cases}$$

Este é um problema NP-completo, logo algumas heurísticas foram desenvolvidas para a sua resolução. A seguir as heurísticas KPP, CSPT, CCET e Híbrida, principais a abordar essa problemática, são apresentadas.

3.1.1 Heurística KPP

Kompella, Pasquale e Polyzos propuseram em [Kompella93] uma heurística baseada no modelo de Árvore de Steiner Restrita, que será denominada neste trabalho de KPP.

O primeiro passo da heurística é encontrar para cada par de nós pertencente ao grupo multicast, o caminho de menor custo e que respeita o limite de atraso. Este procedimento é realizado para todos os valores inteiros de atraso, a partir de 1 até o limite máximo Δ . Isto é feito com o objetivo de aumentar o número de caminhos a serem avaliados em função de seu custo, conseqüentemente aumentar a possibilidade de se aproximar do caminho de custo mínimo. Em outras palavras, é encontrado o fechamento do grafo formado pelos participantes do grupo *multicast*.

Encontra-se, então, uma árvore restrita, tendo a fonte como raiz, a partir do fechamento. Para isso, é utilizado um algoritmo do tipo *greedy*, ou seja, as arestas desnecessárias são sucessivamente retiradas do grafo baseando-se na função custo. Existem, também, mecanismos de seleção alternativos, baseados tanto na função custo quanto na função atraso, ou ainda, em ambas. Os enlaces da árvore de espalhamento (*spanning tree*) são, então, novamente mapeados para o grafo original.

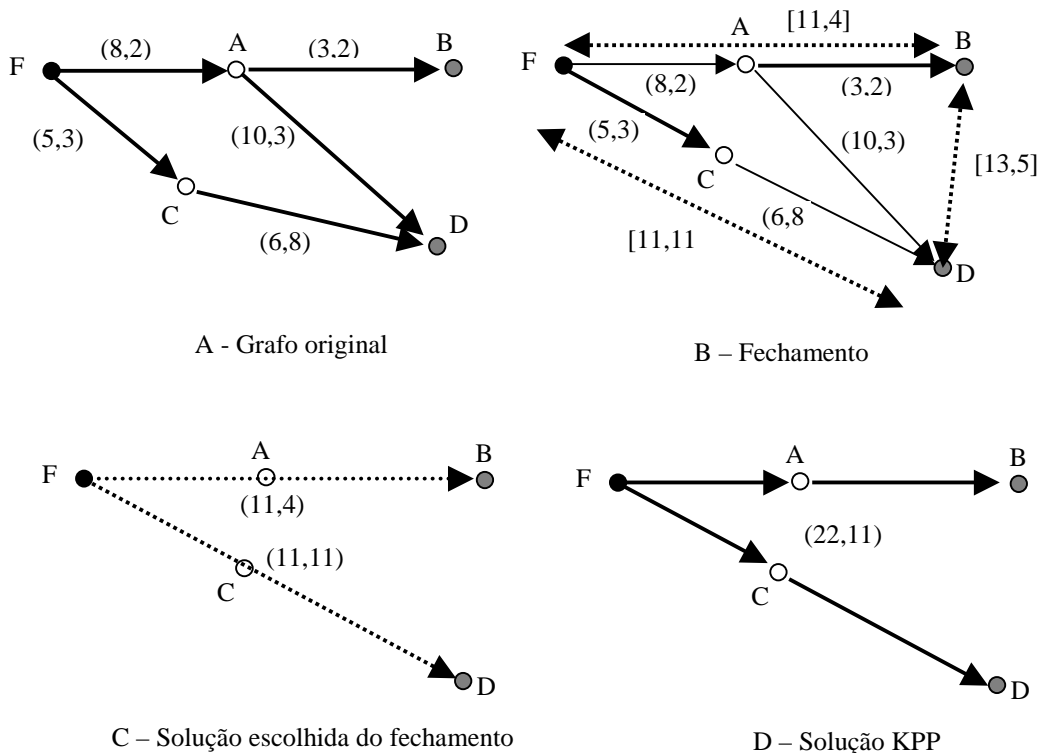


Figura 2 – Evolução da solução KPP

A computação dos menores caminhos restritos durante a primeira fase da heurística é a que consome mais tempo, tendo uma complexidade de $O(\Delta n^3)$, onde n é o número de vértices do grafo. O segundo passo da heurística, definição da árvore restrita, tem complexidade de $O(m^3)$, onde m é o número de nós do grupo *multicast*. O último passo, mapeamento da solução para o grafo original, tem uma complexidade de no máximo $O(n^2)$. Portanto a complexidade total da heurística é dada por $O(\Delta n^3)$.

A evolução da solução KPP pode ser acompanhada através da figura 2. A comunicação *multicast* tem como origem o emissor F e os membros do grupo *multicast* estão identificados na figura com os rótulos B e D. O atraso máximo estabelecido é 12. O fechamento do grafo formado pelos participantes do grupo *multicast*, primeiro passo da heurística, pode ser

identificado através das setas pontilhadas no item B da figura 2. É através deste fechamento que o algoritmo selecionará a solução. No exemplo acima, a solução será formada pelos enlaces (F,B) e (F,D) somando um custo de 22 e um atraso máximo de 11, item C. Esta é uma solução válida, dado que satisfaz a restrição de atraso. O estágio final do KPP consiste em mapear a solução do fechamento novamente no grafo original, resultando nos enlaces (F,A), (A,B), e (F,C), (C,D), formando a árvore de *multicast* restrita apresentada no item D da figura 2.

3.1.2 Heurística CSPT

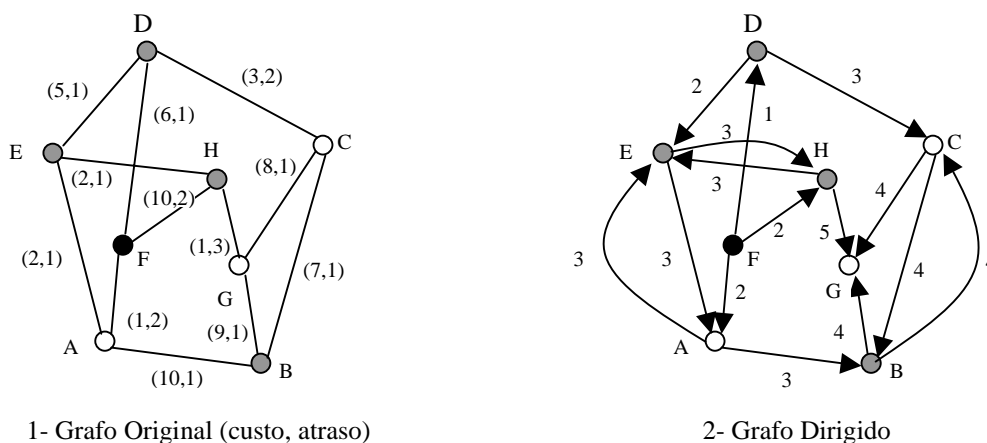
Quan Sun e Horst Langendörfer publicaram em [Sun95] a heurística CSPT (*Constrained Shortest Path Trees*). Esta é composta por três fases, as quais são descritas. Inicialmente é construída uma árvore de espalhamento de baixo custo, através do algoritmo de Dijkstra, composta pelo maior número possível de componentes pertencentes ao grupo *multicast* e sempre limitando o atraso máximo. Como nem sempre todos os membros do grupo serão cobertos por esta árvore, uma segunda execução do algoritmo de Dijkstra procurará criar uma árvore formada por caminhos de menor atraso, abrangendo os elementos não cobertos pelo passo anterior. A última fase consiste em juntar estas duas árvores, removendo os eventuais *loops* e tomando o cuidado de nenhum caminho exceder o limite máximo de atraso.

Tanto o primeiro quanto o segundo passo dependem basicamente do algoritmo de Dijkstra com algumas alterações. Logo a complexidade destas fases será da ordem de $O(n^2)$. A última fase, de concatenação das árvores, limita-se a $O(n)$. Portanto, a heurística tem a complexidade total da ordem de $O(n^2)$.

3.1.3 Heurística CCET

A heurística CCET (*Constrained Cheapest Edge Tree*) foi descrita pela primeira vez em [Waters94]. Alguns aperfeiçoamentos foram feitos desde então. Na proposta original a restrição de atraso estava relacionada com o atraso mínimo de *broadcast*, ou do grupo *multicast*. Na versão abordada neste trabalho, esta restrição pode ter um valor arbitrário.

Primeiramente, a heurística calcula, através do algoritmo de Dijkstra, o valor do atraso de cada nó a partir da fonte, criando um grafo dirigido. Durante essa fase, as arestas cujo atraso inserido ultrapasse o limite definido, são removidas da árvore (figura 3, item 2). Então, os nós são ordenados inversamente em relação ao seu atraso. Iniciando pelo nó de maior atraso, a heurística procura o enlace de menor custo, criando um caminho de volta à fonte. É bom ressaltar que, como as redes são assimétricas, o enlace é escolhido em relação ao seu custo sempre na direção da fonte para o nó. Estes caminhos são inseridos numa árvore de *broadcast*



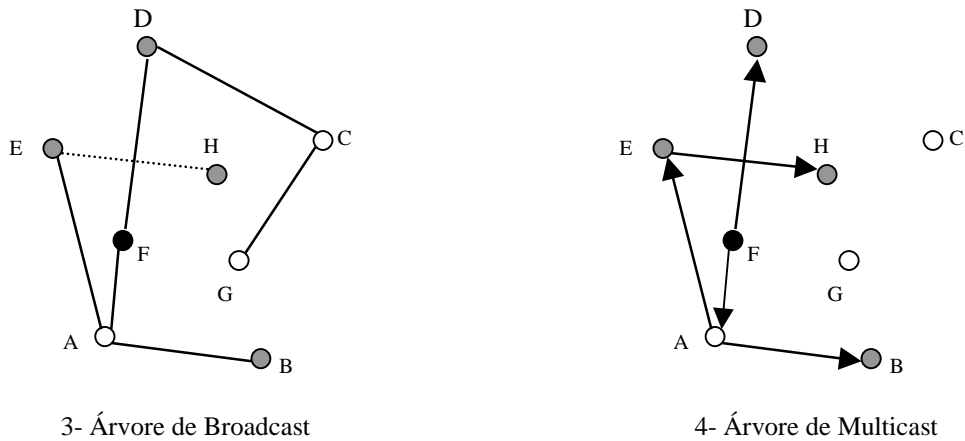


Figura 3 – Estágios da heurística CCET

(figura 3, item 3). Ao final deste processo, os nós que não pertencem ao grupo *multicast* são retirados da árvore, formando a árvore de *multicast* (figura 3, item 4).

Em relação à ordem de complexidade desta heurística, verifica-se que no primeiro estágio, que utiliza o algoritmo de Dijkstra, mantém-se na ordem de $O(n^2)$. Para a criação da árvore de *broadcast*, é necessário, para cada nó, executar uma busca em profundidade na árvore até a fonte, o que consome $O(\max(N, |E|))$, onde N representa o número de nós e E o conjunto de arestas da árvore de busca, sendo esta formada por todas as possíveis rotas de volta à fonte[Gibbons89]. Portanto a ordem de complexidade total da heurística será de $O(n^2)$.

3.1.4 Heurística Híbrida

A heurística híbrida foi proposta por John Crawford e Gill Waters em [Waters97]. Esta heurística é formada basicamente pela integração do algoritmo de Dijkstra e as heurísticas CCET e CSPT. Esta integração foi proposta em função dos bons resultados obtidos pelas heurísticas para casos específicos do problema de roteamento *multicast* restrito.

O cálculo da árvore *multicast* é realizado através da aplicação do algoritmo de Dijkstra criando simultaneamente as árvores de menor atraso e menor custo. Em seguida, aplicam-se os métodos de seleção utilizados nas heurísticas CCET e CSPT para gerar as respectivas árvores *multicast*. Neste ponto existem três possíveis árvores para serem consideradas: i-) árvore gerada pelo algoritmo de Dijkstra; ii-) árvore gerada pelo mecanismo de seleção do CCET; iii-) árvore gerada pelo mecanismo de seleção do CSPT. Logo a análise procede da seguinte forma:

- Se alguma árvore contiver um caminho que ultrapasse o limite de atraso máximo, é automaticamente descartada do processo de análise;
- Calcula-se o custo total das árvores e aquela que possuir o menor custo será selecionada.

A ordem de complexidade desta heurística é dominado pelo CCET, o que resulta em $O(n^2)$.

4 O Simulador e a Metodologia de Análise

Entre os diversos parâmetros passíveis de estudo na simulação, alguns deles, como a ordem de complexidade, a quantidade de informações de estado armazenadas, etc., devem ser analisados através de modelos analíticos. Contudo, outros de fundamental importância, sobretudo para ambientes de redes reais, devem ser considerados. Entre estes podemos citar: tempo de execução, número de falhas, escalabilidade e qualidade dos resultados produzidos. De forma a reproduzir o ambiente real de transmissão de fluxos multimídia em redes ATM,

alguns procedimentos e definições foram criados e implementados gerando um simulador de transmissões *multicast* sobre redes ATM.

4.1 Simulador

O simulador MCRSIM utilizado neste trabalho foi originalmente desenvolvido por Hussein Salama como parte de sua tese de doutorado [Salama96] apresentado à Universidade Estadual da Carolina do Norte – Estados Unidos. Este foi devidamente aperfeiçoado e modificado para realizar as simulações da heurística Híbrida em conjunto com as já implementadas.

Seguindo o modelo de rede ATM, o simulador considera que as ligações são *full-duplex*, e com capacidade de transmissão de 155Mbits (OC-3). Foi definido que os nós da rede são gerados aleatoriamente dentro de um retângulo de 4000x2400 Km² de área, correspondendo aproximadamente à área territorial dos Estados Unidos.

Um ponto importante que deve ser tratado com muito cuidado quando se trabalha com simulações diz respeito ao gerador aleatório utilizado. Neste trabalho foi utilizado um gerador aleatório baseado no modelo de gerador de Waxman [Waxman88] para a geração das redes (nós e ligações). O gerador foi otimizado de forma que a rede resultante fosse composta por nós cujos graus sempre fossem maior ou igual a dois, ou seja, cada nó estaria ligado a pelo menos dois outros nós. Alguns parâmetros do gerador foram ajustados para que, assim como em redes reais, a probabilidade da existência de uma ligação de menor distância seja maior do que uma de maior distância, e que a média dos graus dos nós seja 4, como é encontrado hoje na Internet.

Cada nó da rede representa um comutador ATM não bloqueante. A velocidade de propagação das ligações foi estimada em dois terços da velocidade da luz. Não foram considerados atrasos relativos ao enfileiramento de células, assim pode-se dizer que os atrasos de uma dada ligação são simétricos, ou seja, $A(u,v) = A(v,u)$.

Para simular o tráfego multimídia, foram utilizadas conexões *multicast* com categoria de serviço VBR (*Variable Bit Rate*), supondo o transporte de fluxos de vídeo MPEG-2 de alta resolução. Para cada fluxo de vídeo que passa por uma dada ligação, é reservada uma fração da banda desta ligação correspondente à necessidade do fluxo.

Desta forma a função custo de um enlace foi associado à utilização do mesmo, ou seja, quanto maior for a utilização de um enlace, maior será o seu custo. O custo de uma ligação varia, então, conforme as sessões que a utilizam são estabelecidas ou liberadas. Um enlace pode aceitar novas sessões até que a soma das reservas de banda atinja 85% de sua capacidade. A partir deste ponto, o enlace torna-se saturado e novas sessões são rejeitadas. Esta política de controle de admissão, apesar de simples, permite multiplexação estatística e uma utilização eficiente dos recursos da rede, além de ser suficiente para os estudos em questão.

As sessões de vídeo e voz interativos, têm exigências rígidas em relação ao atraso máximo suportado. Segundo as considerações feitas em relação às dimensões da rede e ao tempo de propagação, é possível estimar que o atraso imposto pela rede é de 0,02 s numa transmissão de leste a oeste e 0,012 s numa transmissão de norte a sul. Vide figura 4.

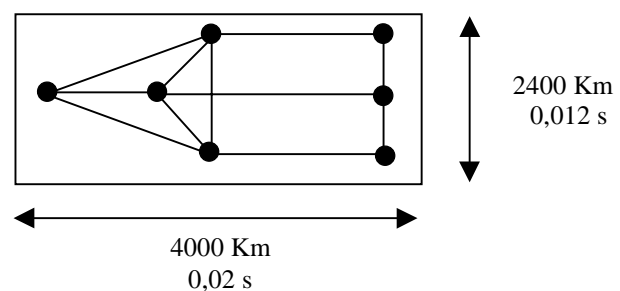


Figura 4 – Dimensões e atrasos impostos pela rede

Tendo isso em vista, foi adotado um valor para Δ (atraso máximo entre fonte e receptor) de 0,03 segundos que representa um limitante superior para o tempo de propagação fim-a-fim através da rede. Este valor relativamente baixo foi utilizado de forma que os protocolos de camadas superiores tenham tempo suficiente de processar as informações transmitidas sem que o serviço seja prejudicado. Além disso, espera-se que este valor sirva de diferencial para os algoritmos mais eficientes.

Reproduzindo ambientes reais, o simulador gera um tráfego de fundo nas ligações, também de forma aleatória. Este tráfego não tem uma característica definida, apenas estabelece uma banda mínima B_{min} e uma banda máxima B_{max} de utilização. Assim, quando do estabelecimento de uma nova conexão, além dos tráfegos gerados por conexões já existentes, o tráfego de fundo também deverá ser considerado pelos mecanismos de controle de tráfego.

4.2 Adaptações feitas no simulador

De forma a viabilizar a comparação da nova heurística apresentada na seção anterior com aquelas já implementadas no simulador, foram executadas algumas alterações no programa original.

O primeiro passo consistiu em portar o simulador, escrito originalmente para as plataformas Sun, DEC, RS6000, para a plataforma Linux/Intel. Esta plataforma foi escolhida devido ao fato de ser aberta e mais acessível, além de possuir reconhecido espaço no meio acadêmico. Em seguida foi necessário escrever a nova heurística, proposta por Waters, no simulador mantendo a estrutura de dados já existente e utilizada pelas outras heurísticas. Esta preocupação de manter a mesma estrutura de dados, teve por objetivo criar a coerência de implementação necessária para que os dados obtidos expressassem fielmente a situação em teste, possibilitando uma comparação justa com as outras heurísticas.

Além destas alterações, foram desenvolvidos módulos que implementam testes específicos utilizados durante as simulações. Dentre estes, destaca-se o módulo de simulação contínua de algoritmos, ou seja, para cada rede gerada aleatoriamente, as quatro heurísticas são executadas utilizando as mesmas condições de tráfego de fundo, grupo *multicast* e fonte transmissora. Os respectivos resultados são armazenados em arquivos de saída, de acordo com o parâmetro analisado na simulação.

4.3 Parâmetros Relevantes para Comparação de Heurísticas *Multicast*

É necessário que se defina um critério para realizar a comparação das heurísticas de roteamento *multicast*. Para a definição deste critério, é preciso levantar os parâmetros relevantes e adotar uma métrica aplicável a todas as heurísticas [Tobagi94]. Neste sentido, destacam-se os principais pontos que devem ser considerados na análise de uma heurística:

- Gerenciamento eficiente dos recursos da rede;
- Parâmetros de QoS considerados;
- Complexidade do algoritmo;
- Quantidade de informações armazenadas necessárias;
- Aplicabilidade para redes simétricas e assimétricas.

Como nem sempre é possível a utilização de redes reais para o desenvolvimento e aperfeiçoamento destas heurísticas, as simulações têm se mostrado o método preferido pelos pesquisadores para a análise dos mesmos.

Para que as simulações representem as condições reais das redes de computadores, é necessário que se execute o algoritmo em diferentes topologias e tamanhos de redes, utilizando cenários reais ou gerados aleatoriamente. Condições de congestionamento, atrasos,

diferentes tipos de tráfego, etc. devem se aproximar o quanto possível do que se observa na realidade. Não se tendo o cuidado apresentado, os resultados obtidos através de simulações, podem expressar uma aparente eficiência ou ineficiência para casos especiais e, portanto, não podendo ser generalizados.

Como já foi amplamente explorado na literatura, os resultados das simulações de algoritmos de roteamento variam bastante em função do tamanho da rede considerada. Logo, uma tendência muito encontrada em trabalhos deste tipo é a divisão em redes pequenas de até vinte nós e redes grandes de até duzentos nós, onde o comportamento dos algoritmos é analisado em ambos os casos separadamente.

Em cada caso, é estudada a influência do tamanho do grupo *multicast* sobre o desempenho do algoritmo utilizado. Duas configurações foram especificadas em relação ao tráfego de fundo existente nos enlaces. Uma varia a banda alocada entre $B_{min}=45$ Mbps e $B_{max}=85$ Mbps. A outra avalia os algoritmos quando existe uma maior variação do tráfego de fundo, estando este entre $B_{min}=5$ Mbps e $B_{max}=125$ Mbps. Como os custos de um enlace, que é *full-duplex*, são diretamente proporcionais à sua utilização, uma maior diferença entre os valores de B_{min} e B_{max} pode implicar em uma maior variação nos custos deste enlace. Este fato caracteriza uma maior assimetria dos enlaces. Logo, nas simulações, redes com maior assimetria significam redes cujos enlaces possuem tráfego de fundo com $B_{min}=5$ Mbps e $B_{max}=125$ Mbps.

Em cada execução do simulador, os algoritmos são analisados em função da qualidade da árvore *multicast* gerada, ou seja, o seu custo total, o atraso médio imposto, número médio de comutadores presentes em cada caminho, tempo de execução e número de falhas. A falha de uma heurística pode ser causada pela existência de um caminho que exceda a restrição do limite máximo de atraso entre a fonte e um receptor, ou por não encontrar um enlace com banda disponível para a transmissão do fluxo de vídeo.

Para que os resultados apresentados demonstrem a confiabilidade esperada, cada configuração é executada trezentas vezes, ou seja, são geradas trezentas redes aleatórias cada uma com um diferente tráfego de fundo e grupo *multicast*. Estudos anteriores [Salama96], demonstram que em simulações deste tipo, entre duzentos e cinquenta e trezentas execuções, atinge-se uma confiança de 95% com uma variação máxima de 5% em relação aos resultados obtidos.

5 Resultados e Análises

Os próximos itens demonstrarão os resultados e análises das simulações realizadas com as heurísticas de roteamento *multicast* restritos. Os seguintes parâmetros foram analisados: estudo da complexidade, custo total, atraso inserido, número de comutadores envolvidos, tempo de execução e número máximo de sessões estabelecidas.

5.1 Estudo da complexidade dos algoritmos

O estudo da complexidade dos algoritmos tem como objetivo avaliar a escalabilidade e estimar o tempo de execução dos mesmos em função do número de nós que compõe a rede. Desta forma, é possível prever se um determinado algoritmo tem condições de ser empregado em implementações de protocolos ou equipamentos. Como o foco deste artigo é o roteamento *multicast* para aplicações multimídia, a questão do tempo de execução do algoritmo é essencial para um desempenho eficiente do cálculo das rotas. A tabela seguinte resume os resultados de cada algoritmo.

Algoritmo	Ordem de complexidade
KPP	$O(n^3)$
CSPT	$O(n^2)$
CCET	$O(n^2)$
Híbrida	$O(n^2)$

Figura 5 – Tabela referente à ordem de complexidade dos algoritmos

Independentemente dos resultados práticos, é possível prever, em função da figura 5, que o algoritmo KPP terá um tempo de execução mais alto que os outros algoritmos. Além disso, este tempo de execução aumentará, em função do aumento do número de nós da rede, segundo uma taxa maior que os outros, comprometendo assim, sua escalabilidade. Consequentemente, sua utilização em redes com um elevado número de nós fica prejudicada.

5.2 Custo total da árvore gerada

O custo total da árvore *multicast* construída pelo algoritmo demonstra sua habilidade em alocar os recursos dos computadores de maneira inteligente e transmitir sobre enlaces de baixa utilização.

Redes pequenas (até 20 nós)

Redes grandes (até 200 nós)

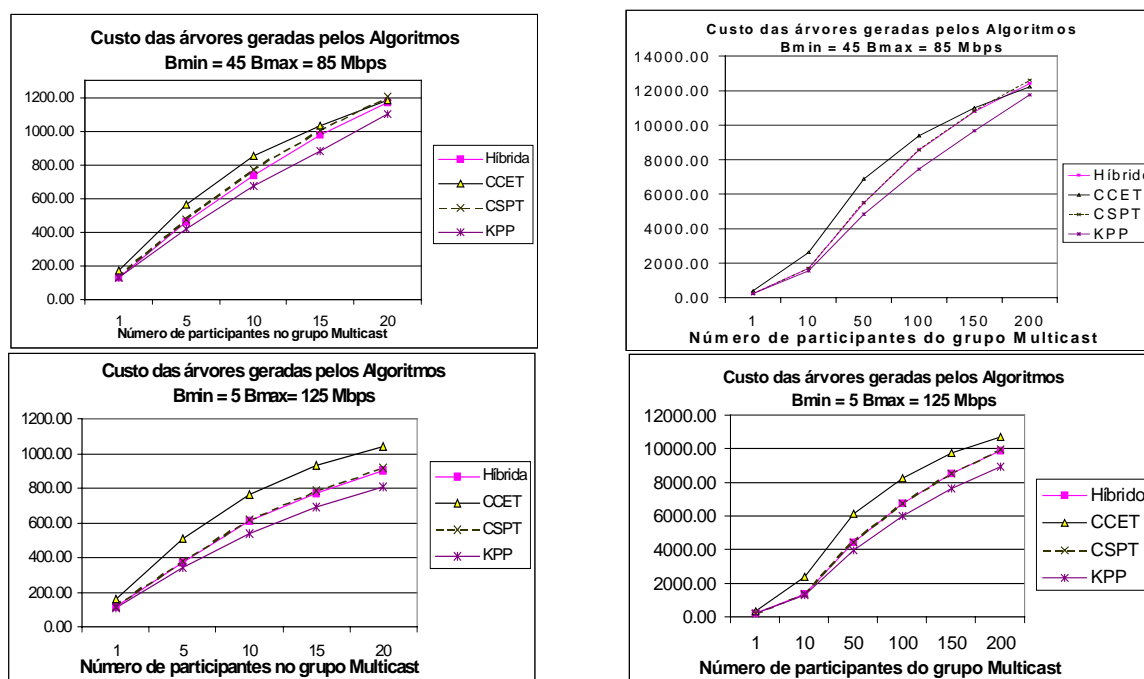


Figura 6 – Custo das árvores geradas pelos algoritmos

Pelos gráficos da figura 6, observa-se que a heurística KPP gera as árvores de menor custo total em todos os casos testados, o que já era esperado, pois é objetivo das heurísticas que implementam a técnica de árvore de Steiner minimizar a utilização total dos recursos. Entre as heurísticas de menor caminho (Híbrida, CCET e CSPT), a Híbrida foi a que apresentou os melhores resultados quando aplicada em redes pequenas, até 5% melhor que a CSPT e 20% melhor que a CCET. Em redes grandes, a Híbrida praticamente se iguala à CSPT, e estas obtiveram resultados até 28% melhores que a CCET.

Um comportamento interessante que pode ser percebido, é a diferença de desempenho dos algoritmos em relação à maior assimetria dos enlaces, que pode ser observado nos gráficos

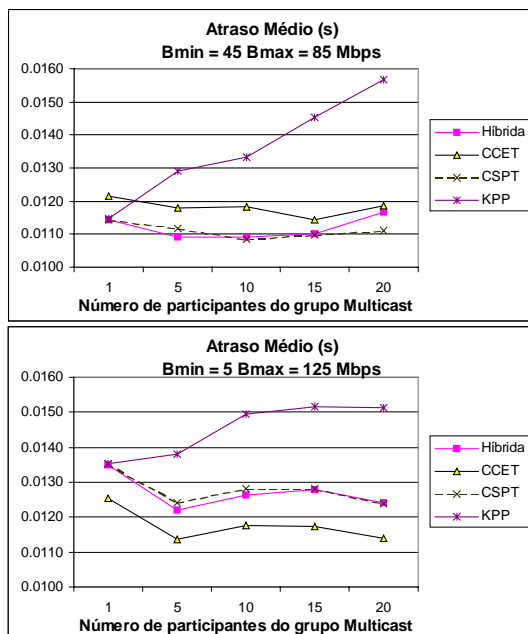
com $B_{min}=5$ Mbps e $B_{max}=125$ Mbps. Neste caso, em redes pequenas, a diferença entre a Híbrida e a CSPT diminui bastante, enquanto a diferença destas com a CCET aumenta.

5.3 Atraso da árvore *multicast*

O atraso inserido pela árvore *multicast* é um parâmetro importante de comparação entre os algoritmos, pois através dele é possível avaliar a capacidade do algoritmo em utilizar enlaces que insiram o menor atraso na transmissão dos fluxos de vídeo. Apesar de todos os algoritmos restringirem o atraso máximo, é interessante que se tenha este parâmetro com o menor valor possível. Pois, em se tratando de tráfegos gerados por aplicações multimídia, além da restrição temporal natural destas aplicações, deve-se ter em mente que estas consomem muito tempo de processamento das estações receptoras para receberem os fluxos, desencapsular os pacotes de dados e apresentá-los ao usuário.

A seguir são apresentados os resultados referentes ao atraso médio das árvores geradas, variando-se o número de participantes do grupo *multicast*.

Redes pequenas (até 20 nós)



Redes grandes (até 200 nós)

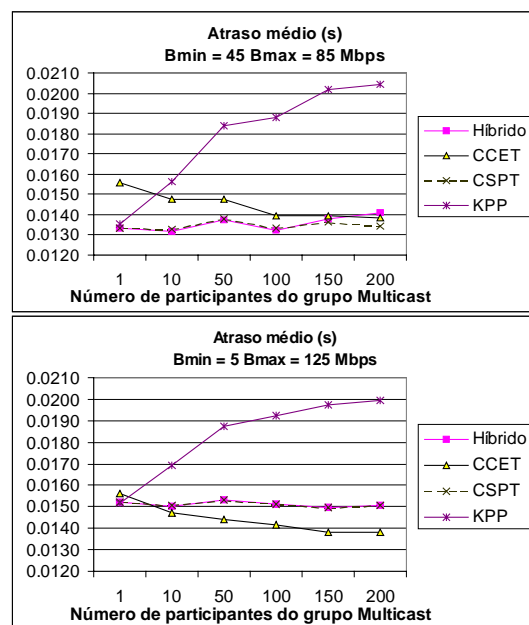


Figura 7 – Atraso médio das árvores geradas

Pelos gráficos da figura 7, conclui-se que a heurística KPP demonstrou ser a mais ineficiente em relação ao atraso médio das árvores geradas tanto em redes pequenas quanto em grandes. Isto se deve ao fato da heurística KPP procurar otimizar apenas o custo da árvore, desconsiderando outros parâmetros.

Das heurísticas de menor caminho, pode-se dizer que tanto em redes pequenas quanto em redes grandes, observou-se que com pouca assimetria, as heurísticas Híbrida e CSPT, apresentaram resultados sensivelmente melhores do que a CCET. Este comportamento praticamente se inverte quando estas são aplicadas em redes com elevada assimetria, ou seja, quando os custos dos enlaces tem grande variação em função do sentido da transmissão.

Seria necessário um estudo mais aprofundado, abrangendo inclusive aspectos de implementação, para explicar os motivos do comportamento das heurísticas em relação à maior ou menor simetria dos enlaces da rede.

5.4 Número de comutadores em cada caminho

O número médio de comutadores presentes nos caminhos que formam a árvore *multicast* é um importante parâmetro para determinar o consumo de recursos da rede pelos algoritmos, assumindo que um comutador também é um recurso da rede. Apesar deste artigo não estar avaliando o atraso imposto pela comutação em cada equipamento, este trabalho pode servir de ponto de partida para um estudo mais detalhado em relação a todos os atrasos envolvidos na transmissão de fluxos de voz ou vídeo.

Redes pequenas (até 20 nós)

Redes grandes (até 200 nós)

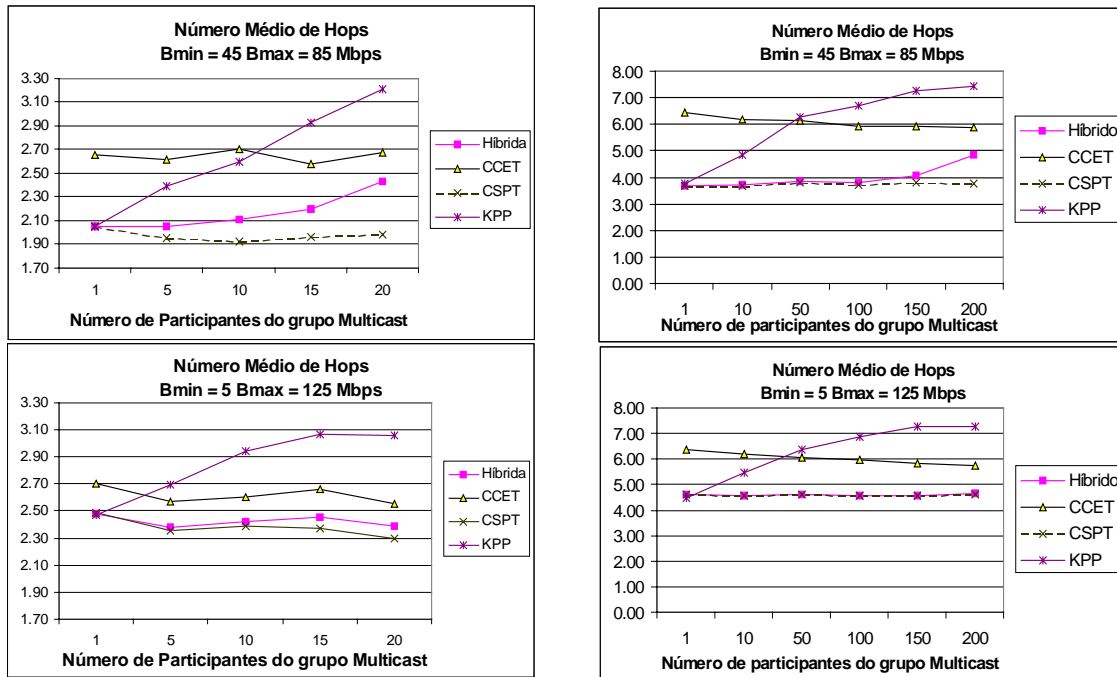


Figura 8 – Número de médio de comutadores nos caminho da árvore *multicast*

Com relação ao número médio de comutadores alocados nos caminhos da árvore *multicast* gerada pelos algoritmos, pode-se concluir, através da figura 8, que a CSPT foi a heurística que apresentou o melhor desempenho, ou seja, alocou o menor número de comutadores. A Híbrida obteve um desempenho de até 19% pior que a CSPT em redes pequenas e praticamente igual em redes grandes com grande assimetria. A heurística CCET obteve o pior resultado entre as de menor caminho, cerca de 42% maior em comparação com a CSPT em redes grandes com pouca assimetria. A única heurística que implementa a técnica de árvore de Steiner testada (KPP) obteve resultados crescentes em função do número de participantes do grupo *multicast*, e em geral maiores que as de menor caminho.

5.5 Tempo de execução dos algoritmos

A análise do tempo de execução dos algoritmos é essencial para comparar os resultados por eles produzidos com o “custo” (tempo de processamento) necessário para seu cálculo. Não menos importante, este estudo serve de base para o estudo da viabilidade de implementação do algoritmo em protocolos ou equipamentos específicos, tanto em redes de pequeno porte quanto em redes de alcance global. É bom lembrar, entretanto, que nem a heurística Híbrida nem as já existentes no simulador foram implementadas visando otimizar o tempo de execução.

O resultados descritos na figura 9, foram obtidos executando o simulador em um Pentium II com o sistema operacional Linux. Porém, o que interessa para este estudo não são os valores propriamente, mas a diferença entre eles.

Fica claro, em todos os casos analisados, que a heurística KPP possui um tempo de execução muito superior em relação às heurísticas de menor caminho. Este fato, já descrito na literatura, ratifica a tese de que algoritmos que implementam a técnica de árvore mínima de Steiner não são aplicáveis às redes grandes, apesar de seus resultados serem melhores.

Redes pequenas (até 20 nós)

Redes grandes (até 200 nós)

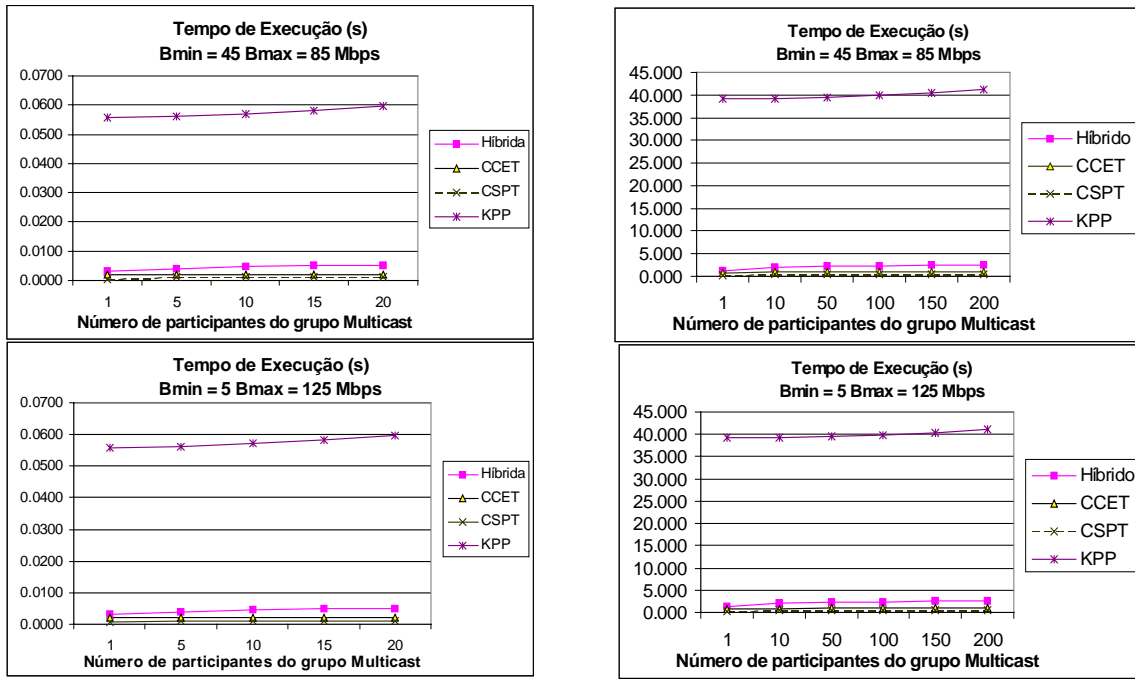


Figura 9 – Tempo de execução dos algoritmos

Entre os algoritmos de menor caminho, a heurística Híbrida apresentou o maior tempo de execução, próximo da soma dos tempos de execução das duas outras heurísticas. Isto pode ser explicado pelo seu mecanismo de funcionamento que após criar a árvore de *broadcast*, utiliza os dois mecanismos de seleção, um utilizado na CCET e o outro na CSPT, para criar duas árvores *multicast*, para então comparar o custo de ambas e escolher aquela que possuir o menor custo. De qualquer forma, o custo da Híbrida ainda é muito inferior ao da KPP, possibilitando sua utilização tanto em redes pequenas quanto em redes globais.

5.6 Número de sessões estabelecidas com sucesso

Com o objetivo de avaliar a eficiência dos algoritmos em relação à alocação de recursos dos comutadores e sua habilidade em lidar com situações de saturação de enlaces, foram conduzidos testes de número máximo de sessões de vídeo simultâneas.

A simulação consiste em gerar uma rede aleatória, sem qualquer tipo de tráfego de fundo. Sobre esta rede serão criados seqüencialmente, também de forma aleatória, grupos *multicast* com uma respectiva fonte emissora de fluxo (VBR) de vídeo, formando assim uma sessão *multicast*. Vale lembrar, que a sessão *multicast* só é estabelecida se houver recursos disponíveis em todos os comutadores intermediários pertencentes à árvore *multicast*. As sessões são criadas continuamente até que os algoritmos falhem em encontrar rotas que possuam banda suficiente para o tráfego de vídeo, ou ainda, que não respeitem o limite de atraso máximo imposto.

Para cada tamanho de grupo *multicast*, são geradas trezentas redes diferentes, onde, em cada uma destas, tenta-se estabelecer mil sessões *multicast*. Os resultados descritos no gráfico da figura 10 representam a média dos valores obtidos nas simulações.

As simulações referentes à heurística KPP não puderam ser realizadas devido ao seu elevado tempo de execução. Entre as heurísticas de menor caminho, tanto a CSPT quanto a Híbrida demonstraram resultados semelhantes, onde a principal causa de falha foi devido à saturação dos enlaces. Já na heurística CCET, foi estabelecido um número até 39 % menor de sessões simultâneas. Este fato se deve ao grande número de falhas devido à violação do atraso máximo imposto (0,03 segundos), além de sua menor habilidade em selecionar enlaces menos congestionados.

Fica claro, a partir da figura 10, que conforme o número de participantes do grupo *multicast* cresce, menor é o número de sessões que os algoritmos conseguem estabelecer antes da saturação da rede. Isto pode ser explicado devido ao fato de um maior grupo *multicast* exigir uma maior quantidade de enlaces para formar a árvore *multicast*. Logo os recursos da rede tendem a acabar mais rapidamente com grandes grupos *multicast*.

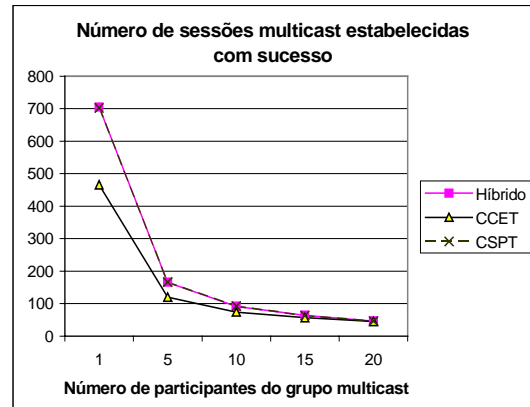


Figura 10 – Número de sessões *multicast* estabelecidas com sucesso

6 Conclusões

Este artigo analisou os principais algoritmos de roteamento *multicast*, realizando estudos analíticos e simulações, de modo a explorar as principais restrições envolvidas em uma comunicação multimídia distribuída, tais como atraso máximo da árvore *multicast*, número de comutadores envolvidos, número máximo de sessões multimídia. Outros parâmetros mais gerais também foram avaliados, tais como complexidade, custo total da árvore e tempo de execução.

Os resultados apresentados mostraram que, em relação à qualidade das árvores geradas, ficou claro que a heurística KPP apresentou os melhores resultados de custo total da árvore *multicast*, tendo em contrapartida um tempo de execução muito alto, o que praticamente inviabiliza sua utilização em redes com um grande número de nós. A heurística Híbrida não apresentou resultados tão bons quanto a KPP, porém a relação custo/tempo de execução faz com que ela tenha mais condições de gerar protocolos ou equipamentos de roteamento *multicast* com QoS.

O atraso médio da árvore de *multicast* foi o único parâmetro onde a heurística CCET obteve alguma vantagem em relação às concorrentes. As heurísticas Híbrida e CSPT obtiveram resultados muito próximos, enquanto a KPP introduziu os maiores atrasos.

A heurística CSPT obteve o menor número de comutadores intermediários nos caminhos formadores da árvore *multicast*, seguido de perto pela Híbrida, ressaltando a habilidade destas em relação ao menor consumo de recursos da rede. Este mesmo resultado se repetiu em relação ao número máximo de sessões *multicast* simultâneas, teste que não pôde ser realizado para a heurística KPP devido seu elevado tempo de execução.

Conclui-se, portanto, que a heurística Híbrida demonstrou resultados iguais ou superiores às concorrentes na maioria dos parâmetros avaliados, mantendo ainda um tempo de execução que não inviabiliza sua utilização, sendo assim a melhor solução para o roteamento *multicast* para comunicação multimídia em redes ATM.

7 Trabalhos Futuros

Dado que o ATM Forum não dispõe de nenhuma padronização referente ao roteamento *multicast* com restrição de QoS, existe muito espaço para pesquisas nessa área. Logo o estudo de novos algoritmos ou, como foi abordado neste artigo, a integração de abordagens já existentes, servirão para que, em um futuro próximo, seja possível especificar protocolos, ou até mesmo equipamentos de comutação, com funcionalidades de roteamento *multicast* com QoS.

Os algoritmos analisados neste artigo não consideraram a inclusão ou exclusão de novos elementos do grupo *multicast*, o que acarreta alterações nas rotas, sendo uma possível extensão deste trabalho.

Também foi assumido durante as simulações que as transmissões de vídeo utilizavam a categoria de serviço VBR. Uma análise comparativa com o CBR ou o ABR em relação à alocação de banda nos enlaces poderia originar resultados sobre o comportamento dos algoritmos quanto a diferentes políticas de reserva de recursos.

Por fim, foi visto que os algoritmos que implementam a técnica de árvore mínima de Steiner possuem como grande desvantagem, o seu tempo de execução. Logo, poderia se pensar em modificar estes algoritmos de forma que seu processamento pudesse ser feito de forma distribuída ou paralela.

Referências Bibliográficas

- [Crawford94] J. S. Crawford, "Multicast Routing: Evaluation of a New Heuristic", dissertação de mestrado, Universidade de Kent, Inglaterra, 1994
- [Deering92] S. Casner e S. Deering, "First IETF Internet AudioCast", in Proceedings of ACM SIGCOMM, Computer Communications Review, vol. 22, no.3, Julho de 1992
- [Gibbons89] Alan Gibbons, "Algorithmic Graph Theory", Cambridge University Press, 1989
- [Kompella93] V. Kompella, J. Pasquale e G. Polyzos, "Multicast Routing for Multimedia Communications", IEEE/ACM Transactions on Networking, 1(3):286-292, 1993
- [Salama94] H. F. Salama, D. Reeves, Y. Viniotis e T. L. Sheu, "Comparison of Multicast Routing Algorithms for High Speed Networks", Tech. Rep. TR 29.1930, IBM, Setembro de 1994
- [Salama96] H. F. Salama, "Multicast Routing for Real-Time Communication on High-Speed Networks", Ph.D. dissertation, Department of Electrical and Computer Engineering, N. C. State University, Novembro de 1996
- [Sun95] Q. Sun e H. Langendoerfer, "Efficient Multicast Routing for Delay Sensitive Applications", in Proceedings of the Second Workshop on Protocols for Multimedia Systems (PROMS '95), pp. 452-458, Outubro de 1995
- [Tobagi94] C. Noronha e F. Tobagi, "Evaluation of Multicast Routing Algorithms for Multimedia Streams", in Proceedings of the IEEE International Telecommunications Symposium, Agosto de 1994
- [Waters94] G. Waters, "A New Heuristic for ATM Multicast Routing", in 2nd IFIP on performance Modelling and Evaluation of ATM Networks, pages 8/1-8/9, Julho 1994
- [Waters97] G. Waters e J. Crawford, "A Hybrid Approach to Quality of Service Multicast Routing", Fifth IFIP Workshop on Performance Modelling and Evaluation of ATM Networks, 1997
- [Waxman88] B. Waxman, "Routing of Multipoint Connections", IEEE Journal on Selected Areas in Communications, vol 6, no. 9, pp. 1617-1622, Dezembro de 1988