

## Proposta e avaliação de uma arquitetura escalável para distribuição de TV na Internet

Tiago S. Proença<sup>1</sup>, Célio V. N. Albuquerque<sup>2</sup>

<sup>1</sup>Software Engineering Department – Siemens Corporate Research (SCR)  
08540 – Princeton – NJ – USA

<sup>2</sup>Instituto de Computação – Universidade Federal Fluminense (UFF)  
24.210-240 – Niterói – RJ – Brasil

tiago.proenca.ext@siemens.com, celio@ic.uff.br

**Abstract.** *This article deals with the problem of building a multicast tree in an Internet TV scenario, which is characterized by only one node sending streaming video to a large number of receivers. As a solution, it is proposed a P2P network-based architecture, herein defined as TVoIP (Television over IP), in which clients build an overlay network whose function is to receive and forward the video content, generating a distribution tree with well defined properties whose purpose is to perform application-level multicast communication in a efficient manner. Simulation results compare the performance of this proposal with the state of the art found in literature considering several performance metrics.*

**Resumo.** *Este artigo considera o problema de construção da árvore de distribuição multicast em um cenário de Internet TV, o qual é caracterizado por possuir somente um emissor transmitindo fluxos de vídeo para um grande número de receptores. Como solução, é proposta uma arquitetura baseada em redes P2P chamada TVoIP (Televisão sobre IP), na qual os clientes formam uma rede sobreposta para receber e repassar os fluxos de vídeo, gerando uma árvore de distribuição com propriedades bem definidas cujo propósito é realizar comunicação multicast em nível de aplicação de modo eficiente. Resultados de simulação comparam o desempenho desta proposta com alguns trabalhos encontrados na literatura em relação a várias métricas de desempenho.*

### 1. Introdução

Nos últimos anos, a popularização de serviços de distribuição de vídeo na Internet tem se tornado cada vez maior, devido à alta disponibilidade de banda passante no núcleo da rede e ao rápido crescimento do número de usuários com acesso à Internet em banda larga. Dentre os serviços mais populares, estão os de vídeo sob demanda, os de transmissão de vídeo ao vivo e o emergente serviço de notificação e envio automático de vídeos conhecido como vídeo *podcast*.

Estes serviços incluem-se em um grupo chamado de maneira genérica e abrangente de Internet TV. Aplicações de Internet TV normalmente são caracterizadas por possuírem somente um emissor transmitindo conteúdo para um grande número de receptores. Realizar a entrega de serviços Internet TV de forma escalável com uma qualidade

de vídeo aceitável tem sido um grande desafio, devido à inerente natureza de alto requisito de banda na transmissão de fluxos de vídeo. A solução empregada por provedores de conteúdo de pequeno e médio porte faz uso de servidores que transmitem fluxos de vídeo utilizando conexões ponto a ponto com cada um dos clientes requisitantes. Este tipo de solução é obviamente não escalável, pois com o aumento de popularidade do arquivo de mídia, o servidor acaba se tornando o gargalo.

O IP *multicast* foi proposto para melhorar a eficiência na comunicação multi-destinatária de um-para-muitos e muitos-para-muitos na Internet. No IP *multicast*, os dados são enviados uma única vez pelo emissor e alcançam cada um dos receptores sem a existência de duplicação desnecessária de pacotes em um mesmo enlace da rede, sendo portanto a forma mais eficiente para realizar comunicação em grupo. No entanto, uma vez que a implementação do IP *multicast* impõe dependência nos roteadores, sua implantação na Internet é restrita. Além disso, existe a dificuldade de implantação através de múltiplos sistemas autônomos, uma vez que os administradores de rede frequentemente bloqueiam tráfego *multicast* nos roteadores de borda de seus AS, devido a questões de desempenho, comerciais e administrativas. Outros assuntos associados com o IP *multicast* como confiabilidade fim-a-fim, controle de fluxo e congestionamento e modelo de cobrança oferecem desafios significantes para os quais ainda não surgiram soluções claras [Pendarakis et al. 2001].

O *multicast* na camada de aplicação procura resolver algumas das dificuldades encontradas na implantação do IP *multicast* na Internet. A vantagem deste tipo de solução é que a complexidade que estava antes nos roteadores é movida para os nós finais da rede, de forma que a solução seja totalmente independente da camada de rede. Embora aparentemente isto só eleve a funcionalidade de *multicast* à camada de aplicação, esta proposta atualmente revoluciona a forma como as aplicações de rede podem ser construídas. O *multicast* na camada de aplicação possibilita a implantação de novas funcionalidades no modelo de comunicação multi-destinatária, tais como gerência do grupo, segurança, privacidade, controle de fluxo e congestionamento, entre outras. É fácil perceber que este tipo de solução oferece grandes vantagens em relação ao IP *multicast*, principalmente porque um nó pode contribuir com vários outros recursos além da funcionalidade de roteamento. Além disso, o *multicast* na camada de aplicação utiliza os protocolos *unicast* já existentes na Internet, o que facilita sua implantação sem nenhuma modificação na estrutura atual de rede da Internet.

As propostas atuais de protocolos para *multicast* a nível de aplicação podem ser categorizadas em árvore ou malha dependendo do tipo de rede sobreposta construída. Uma rede em árvore possui um único caminho entre um par de nós, enquanto que em uma rede em malha pode existir mais que um caminho. O ALMI [Pendarakis et al. 2001], CoopNet [Padmanabhan et al. 2002], BTP [Helder e Jamin 2002], HTMP [Zhang et al. 2002] e Yoid [Francis et al. 2000] são exemplos de protocolos baseados em árvores. O Narada [Chu et al. 2000], TMesh [Wang et al. 2002] e Scattercast [Chawathe 2003] são exemplos de protocolos baseados em malhas. A existência de conexões redundantes na rede em malha requer o uso de um algoritmo de roteamento para construção de um único caminho sem laços entre um par de membros.

Alguns dos desafios encontrados nos protocolos para *multicast* a nível de

aplicação incluem escalabilidade da arquitetura, tempo de entrada no sistema, descon-tinuidades causadas por entradas e saídas de clientes e eficiência em relação ao número de pacotes duplicados nos enlaces. Uma das preocupações dos pesquisadores que está intimamente relacionada com estes desafios diz respeito a como a rede sobreposta será construída na arquitetura. Uma boa rede sobreposta deve refletir o máximo possível a rede física. Diversos trabalhos realizam esta tarefa partindo de uma rede inicial não otimizada, com nós muitas vezes ligados entre si de forma aleatória, realizando um refinamento incremental na rede para que o objetivo de se assemelhar à rede física seja alcançado.

Este trabalho propõe uma arquitetura escalável para transmissão ao vivo de fluxos de vídeo em um cenário de Internet TV, utilizando redes *peer-to-peer* (P2P) para comunicação multi-destinatária. A maior contribuição deste trabalho está na construção da rede sobreposta, pois introduz um novo algoritmo para construção da rede inicial de forma eficiente, sem utilizar medidas ativas de rede e fornecendo um menor tempo de entrada aos clientes do sistema. Através de resultados de simulações será possível observar que a rede sobreposta inicial proposta neste trabalho fornece os menores valores de atraso médio e de distribuição de carga entre os usuários do grupo quando comparada a arquiteturas semelhantes [Chu et al. 2000, Padmanabhan et al. 2002, Zhang et al. 2002].

As demais seções deste trabalho estão organizadas da seguinte forma. A Seção 2 propõe uma arquitetura para distribuição ao vivo de fluxos de vídeo utilizando redes P2P para comunicação multi-destinatária. A Seção 3 expõe alguns aspectos de implementação da arquitetura e apresenta o cenário utilizado para execução da simulação. A Seção 4 conduz uma análise experimental comparando o algoritmo de construção da rede sobreposta inicial da arquitetura com alguns dos trabalhos mais importantes da literatura. Por fim, a seção 5 apresenta a conclusão e futuras direções que podem ser tomadas para melhorar esse trabalho.

## 2. TVoIP: Televisão sobre IP

Nesta seção será apresentada em detalhes uma arquitetura de transmissão de vídeo ao vivo na Internet utilizando redes P2P para comunicação multi-destinatária, denominada TVoIP. O objetivo é fornecer uma arquitetura distribuída baseada em uma árvore de distribuição, descrevendo o mecanismo de construção da rede sobreposta em face do ambiente altamente dinâmico.

Aplicações de Internet TV normalmente são caracterizadas por possuírem somente um emissor transmitindo conteúdo para um grande número de receptores. Partindo deste princípio, consideramos três componentes para a arquitetura do nosso sistema: um conjunto de clientes, um nó de inicialização e uma fonte de vídeo. Sob a perspectiva da arquitetura, os clientes são um conjunto de nós heterogêneos que formam a rede sobreposta P2P, cuja função é receber e repassar o conteúdo de vídeo para outros nós. Em relação à heterogeneidade, consideramos os nós heterogêneos em relação à largura de banda no enlace de acesso de suas redes locais ao *backbone*. O nó de inicialização, também conhecido na literatura como *bootstrap node* ou *rendezvous point*, gerencia o processo de entrada e saída dos nós do sistema através de uma árvore de distribuição. A fonte de vídeo é a raiz da árvore de distribuição e fornece o fluxo de vídeo ao vivo aos clientes do sistema.

As próximas sub-seções irão descrever com maiores detalhes esses três componentes da arquitetura.

## 2.1. O Cliente

Nesta arquitetura, o cliente além de receber os fluxos de vídeo tem como função retransmitir estes fluxos para outros clientes presentes na árvore de distribuição. Esta característica fornece ao sistema o benefício de escalabilidade, uma vez que a fonte de vídeo não é sobrecarregada por um grupo com um grande número de usuários. A localização exata de onde o cliente consegue receber os fluxos de vídeo é enviada pelo nó de inicialização quando o cliente entra no sistema. A interação entre cliente, nó de inicialização e fonte de vídeo durante o processo de entrada na árvore de distribuição pode ser descrita da seguinte maneira:

1. Quando um cliente deseja receber o conteúdo de vídeo, ele envia uma mensagem ao nó de inicialização para se juntar à árvore de distribuição. Além do nome do conteúdo desejado, o cliente informa a sua capacidade de saída para servir futuras requisições (um valor que representa o número de clientes que o nó pode servir baseado em sua largura de banda disponível).
2. O cliente recebe uma mensagem do nó de inicialização informando a localização na qual pode receber o fluxo de vídeo. A localização pode ser o endereço da fonte de vídeo ao vivo ou de um outro cliente presente no sistema. Assim, o cliente entra em contato com o nó que lhe foi informado para que este comece a lhe enviar o fluxo de vídeo.

Uma das dificuldades encontradas é determinar a capacidade de saída do cliente, uma vez que é função da largura de banda disponível no nó. Então, em vez de medir a largura de banda disponível, tarefa que normalmente envolve um grande *overhead* e complexidade [Jain e Dovrolis 2003], cada cliente confia em um valor especificado pelo usuário que representa o número de clientes que ele pode e está disposto a servir. Alguns mecanismos de incentivo para que nós suportem outros clientes serão citados na conclusão.

No caso específico das mensagens de repasse de vídeo, o cliente só repassa o fluxo de vídeo para um número de clientes que seja menor ou igual a sua capacidade, ou seja, sem ultrapassar o limite definido no momento de conexão. Espera-se dessa forma que o cliente forneça um serviço de boa qualidade para os outros nós sem esgotar seus recursos.

## 2.2. A Fonte de Vídeo

A fonte de vídeo nesta arquitetura tem como função realizar a distribuição sincronizada dos fluxos de vídeo ao vivo para os clientes do sistema. Em relação à largura de banda, a fonte de vídeo não é sobrecarregada uma vez que a carga de distribuição do conteúdo de vídeo é compartilhada por todos os nós. Como a fonte de vídeo se localiza na raiz da árvore, os únicos clientes que recebem diretamente o fluxo de vídeo são os que se encontram no primeiro nível da árvore de distribuição. Os demais clientes recebem o fluxo de vídeo através da retransmissão dos pacotes pelos membros que se encontram em um nível acima na árvore, ou seja, pelos seus respectivos nós pai. Formalizando, podemos dizer que o cliente  $c_i$  só recebe fluxos de clientes  $c_{i-1}$ , onde  $1 \leq i \leq h$  considerando  $i$  como o nível da árvore,  $c_0$  como a fonte de vídeo e  $h$  como a altura da árvore.

## 2.3. O Nó de Inicialização

Nesta arquitetura assumimos a presença de um nó de inicialização bem conhecido, cujo propósito é auxiliar um novo cliente no processo de entrada no grupo *multicast*. O

endereço do nó de inicialização é parte da informação do grupo e é obtido *off-line* pela aplicação. O nó de inicialização funciona como uma espécie de servidor de diretório, no qual reside uma lista com todos os membros pertencentes ao grupo. Nesta lista, são considerados endereços dos clientes, capacidade de banda disponível, número de filhos e informação a respeito da hierarquia do nó na árvore de distribuição.

O nó de inicialização fornece para o novo cliente a localização onde ele poderá obter o fluxo de vídeo. Isto é possível porque o nó de inicialização é quem gerencia a árvore de distribuição *multicast*. O nó de inicialização não participa no encaminhamento dos dados, o que faz com que sua localização não tenha impacto significativo no desempenho da disseminação dos fluxos de vídeo. É possível que um nó de inicialização sirva vários grupos *multicast*, o que é desejável em um sistema de distribuição de TV, uma vez que cada grupo corresponde à transmissão de um programa televisivo diferente. O nó de inicialização pode ser uma simples estação de trabalho, um servidor dedicado, um *cluster* de servidores ou ainda estar localizado junto à fonte de vídeo.

Considerando o nó de inicialização, podemos observar que a centralização simplifica o protocolo mas introduz um único ponto de falha ao sistema. Entretanto, no contexto da proposta, a falha do nó de inicialização irá impossibilitar novos clientes de entrarem no grupo, mas não afetará a disseminação dos fluxos de vídeo entre os clientes que já participam do grupo. Além disso, funcionalidades como políticas de acesso, criptografia, segurança e várias outras podem ser implementadas no nó de inicialização.

#### 2.4. Construção da Rede Sobreposta

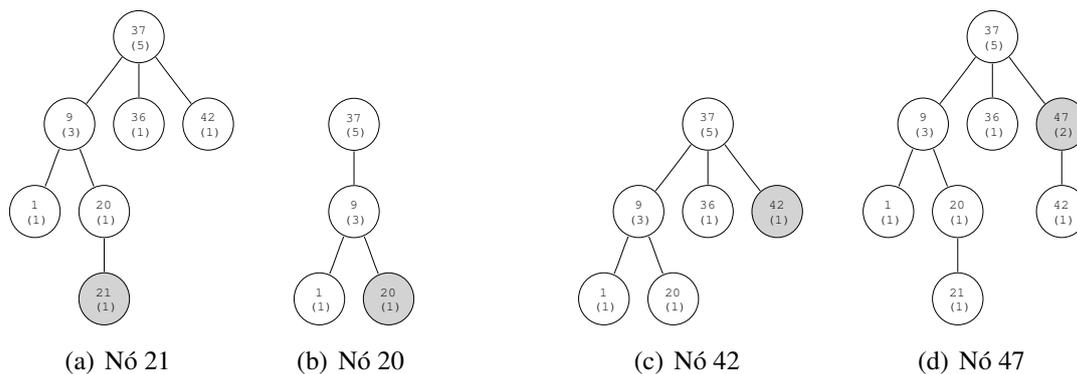
Para reduzir a saturação nos enlaces e o uso de recursos da rede, o TVoIP agrupa na rede sobreposta nós próximos na rede física. Para realizar esta tarefa, utilizamos como métrica de distância o valor absoluto da diferença entre o endereço de rede dos nós. Por não utilizar medidas ativas de rede, o uso desta métrica diminui o *overhead* das mensagens de controle do protocolo e contribui para um curto tempo de entrada dos clientes no sistema. Para atingir o objetivo de manter uma árvore baixa e larga, consideramos manter os nós com maior grau de saída logicamente conectados próximos à raiz da árvore (fonte de vídeo).

Para realizar o cálculo de proximidade, considere  $D_{A-B}$  como a distância entre os nós  $A$  e  $B$ . Assuma  $E_A$  como o endereço de rede do nó  $A$  e  $E_B$  como o endereço de rede do nó  $B$ , onde  $E$  representa um endereço de rede de 4 bytes (IPv4). Assim, o cálculo pode ser realizado através da seguinte expressão:

$$D_{A-B} = \| E_A - E_B \| \quad (1)$$

Esta métrica, embora não realística, apresenta bons resultados (Seção 4), pois usualmente nós que estão na mesma rede se localizam geograficamente próximos.

Para descrever o procedimento de entrada dos nós na árvore de distribuição, iremos usar como referência a Figura 1. Cada imagem representa o estado da rede sobreposta após a entrada de cada um dos nós marcados em cinza. O endereço de cada nó é ilustrado na parte superior do mesmo e sua capacidade de saída na parte inferior entre parênteses. Por questões de simplicidade, iremos adotar um endereçamento não hierárquico, no qual o endereço de rede é representado por um número natural. Assim, na Figura 1(a), o nó



**Figura 1. Estado da rede sobreposta após a entrada de cada um dos nós marcados em cinza.**

possui endereço 21 e capacidade 1. A seguir iremos ilustrar a entrada dos seguintes nós na árvore de distribuição: 21, 20, 42, 47.

Na Figura 1(a), o nó 21 é um novo cliente que deseja juntar-se ao grupo. Ele contacta o nó de inicialização para receber a localização do nó que irá ser seu pai na árvore de distribuição. O nó de inicialização tem total conhecimento da topologia da rede sobreposta. Assim, após receber a requisição do nó 21 e sua capacidade de saída, o nó de inicialização inicia o procedimento de entrada do nó 21 na árvore. O nó de inicialização assume a raiz 37 como um pai potencial. Em seguida calcula a distância entre o nó 21 e o nó 37 e entre o nó 21 e todos os filhos do nó 37 (nó 9, 36 e 42). Após, considera o nó mais próximo como sendo o novo pai potencial do nó 21, neste caso o nó 9. O processo é repetido e encontra-se o novo pai potencial, o nó 20. Neste caso, como o procedimento chegou em um nó que é folha da árvore, o nó de inicialização envia a localização do nó 20 para o nó 21 para que este possa entrar em contato com o nó 20 para receber o fluxo de vídeo.

Um outro exemplo considerado é ilustrado na Figura 1(b). Observe que o nó 20 possui como pai um nó interno da árvore. Após calcular a distância entre o nó 20 e o nó 9 e entre o nó 20 e todos os filhos do nó 9, o nó de inicialização verifica que o nó 9 está mais próximo do nó 20. De acordo com as informações sobre o nó 9 contidas em sua lista, o nó de inicialização sabe se o nó 9 tem capacidade suficiente para servir o novo nó 20. Neste caso, como o nó 9 ainda pode servir mais 2 nós, o nó de inicialização envia a localização do nó 9 ao nó 20. Se o nó 9 não pudesse servir o nó 20, o procedimento continuaria descendo a árvore utilizando como novo pai potencial o filho do nó 9 mais próximo do nó 20, de modo análogo ao descrito anteriormente. Procedimento semelhante ocorre na entrada do nó 42, conforme ilustrado na Figura 1(c).

Uma das características do algoritmo do procedimento de entrada é manter os nós com maior grau de saída próximos à raiz da árvore. A Figura 1(d) ilustra um exemplo em que podemos observar este comportamento. Observe que o nó 42 pode servir apenas 1 nó. Assumindo que o nó 47 pode servir 2 nós, quando o procedimento chegar ao nó 42, será verificado que o nó 42 possui um grau de saída menor que o do nó 47. Então o procedimento troca o nó 47 de lugar com o nó 42 e insere o nó 42 novamente na árvore a partir de seu antigo nó pai, que neste caso é o nó 37. Ao final do procedimento, observamos que o nó 42 é inserido como um filho do nó 47.

Todas as vezes que o procedimento move um nó de lugar dentro da topologia da árvore, o nó de inicialização adiciona este nó em uma lista de movidos. Após o término do procedimento, além do nó 47 receber uma mensagem contendo a localização de seu nó pai (nó 37), o nó 42 recebe uma mensagem de mudança de pai do nó 37 para o nó 47. Para que o nó 42 não experimente perda de pacotes durante a troca de pai, somente após receber pacotes de vídeo do nó 47 é que o nó 42 envia uma mensagem ao nó 37 notificando que mudou de pai. Assim o nó 37 atualiza sua lista de filhos, o que suspende o repasse dos fluxos de vídeo para o nó 42.

### 3. Modelo de Simulação

Para avaliação da proposta, foi desenvolvido um *framework* em linguagem C++ para uso no simulador de redes ns-2. Através do uso deste *framework*, foram implementados os protocolos das arquiteturas TVoIP, CoopNet (árvore centralizada), Narada (malha) e HMTP (árvore distribuída). Além disso, foi implementado um simulador próprio (*net-sim*) para realizar cálculos sobre o custo da árvore de distribuição dessas arquiteturas. As arquiteturas foram implementadas de acordo com as especificações descritas em suas respectivas publicações, sem realizar nenhum mecanismo de otimização na rede sobreposta. Desse modo, não foram considerados o algoritmo de refinamento da árvore do protocolo HMTP e o algoritmo de remoção e inserção dinâmica de enlaces na rede em malha do protocolo Narada. Para o CoopNet foi considerado somente uma árvore de distribuição, ou seja, o envio de um único descritor de vídeo.

Usando o gerador de topologias BRITE [Medina et al. 2001], geramos 5 topologias para a rede física seguindo o modelo de Waxman com 100 AS com 10 roteadores cada, totalizando uma rede com 1000 nós. Os parâmetros para geração das topologias resultaram em atrasos de propagação nos enlaces que variaram de 0.15 ms até 42 ms para todas as topologias. As taxas de transmissão dos enlaces foram configuradas de forma que os enlaces do *backbone* tivessem capacidade de 100 Mbps. Os enlaces de acesso possuem largura de banda variando de 3 Mbps a 10 Mbps, modelando enlaces típicos para interconexão de redes ao *backbone* (rádio a 3 Mbps, xDSL até 8Mbps, cabo de até 10 Mbps). As redes locais possuem capacidade de 10 Mbps modelando redes corporativas e residenciais.

A localização dos clientes e do nó de inicialização (*bootstrap node*) na topologia foi escolhida de maneira uniformemente aleatória utilizando somente os nós de borda da rede. O número de clientes do grupo *multicast* variou de 100 a 700, com incrementos de 100 clientes. Para cada cliente, o número de vizinhos na rede sobreposta (grau de saída) variou de 1 a 4. O processo de entrada na rede sobreposta foi modelado como uma distribuição de Poisson, pois consegue modelar satisfatoriamente eventos de entrada de usuários em uma rede P2P [He et al. 2003].

De forma a variar a qualidade do vídeo, neste trabalho foram utilizadas taxas de transmissão que variam de 150 kbps até 2 Mbps, modelando as taxas de transmissão de vídeo de codificadores de baixa qualidade até codificadores de alta qualidade. Com relação ao processo de perdas de pacotes, foram consideradas apenas perdas por congestionamento nas filas dos roteadores.

### 3.1. Métricas de Desempenho

Nos experimentos realizados para avaliação da arquitetura, a qualidade dos caminhos dos dados foi avaliada de acordo com os resultados obtidos através do simulador *netsim*. As métricas de desempenho utilizadas foram as seguintes:

- **Saturação do enlace:** Também conhecida como *link stress*, esta métrica mede o número de pacotes idênticos que trafegam nos enlaces. O objetivo é verificar a eficiência do TVoIP em distribuir a carga da rede nos enlaces físicos;
- **Penalidade relativa ao atraso (PRA):** Também conhecida como *stretch*, a penalidade relativa ao atraso entre emissor e receptor é definida como a razão do atraso entre eles na rede sobreposta e o equivalente utilizando *unicast*. Isto pode ser expresso como  $\frac{d_i^l}{d_i}$ , onde  $d_i^l$  representa o atraso do nó emissor até o nó  $i$  na rede sobreposta e  $d_i$  o atraso *unicast* entre eles na rede física. O objetivo é medir o aumento de atraso que os nós finais percebem ao utilizarem o TVoIP e os demais protocolos de *multicast* a nível de aplicação;
- **Uso de recursos normalizado (URN):** Esta métrica é definida como a razão entre o uso de recursos de um protocolo *multicast* a nível de aplicação e o uso de recursos do IP *multicast*. O uso de recursos pode ser definido como  $\sum_{i=1}^L d_i * s_i$ , onde  $L$  expressa o número de enlaces ativos na transmissão de dados,  $d_i$  o atraso referente ao enlace  $i$  e  $s_i$  a saturação do enlace  $i$ . O objetivo é avaliar o uso de recursos adicionais de rede consumidos pelos protocolos de *multicast* a nível de aplicação quando comparado ao IP *multicast*.

Para avaliar o desempenho nos nós finais, foram utilizados os resultados obtidos pelo simulador de redes ns-2. As métricas de desempenho utilizadas foram as seguintes:

- **Atraso médio da árvore de distribuição:** Considere  $d_i$  como sendo o atraso médio fim-a-fim sofrido pelos pacotes no caminho do nó emissor ao nó  $i$ . O atraso médio da árvore de distribuição é calculado através da média aritmética destes atrasos, ou seja, pela expressão  $\frac{1}{n} \sum_{i=1}^n d_i$ , onde  $n$  representa o tamanho do grupo *multicast*. O objetivo é avaliar o atraso percebido pelos usuários do TVoIP frente a outros trabalhos;
- **Carga média da árvore de distribuição:** Considere  $l_i$  como sendo a carga sofrida pelo nó  $i$ , calculada através da seguinte expressão:  $l_i = \frac{\text{filhos}_i}{\text{capacidade}_i}$ , onde  $\text{filhos}_i$  representa o número de nós que o nó  $i$  encaminha pacotes e  $\text{capacidade}_i$  o número de clientes suportados pelo nó  $i$  de acordo com sua largura de banda anunciada. O cálculo da carga média da árvore de distribuição pode ser expresso como a média aritmética da carga de todos os nós internos, isto é, dos nós que estão servindo clientes. Isto pode ser calculado através da expressão  $\frac{1}{q} \sum_{i=1}^q l_i$ , onde  $q$  expressa o número de nós internos na árvore. O objetivo é verificar a eficiência dos protocolos de *multicast* a nível de aplicação em distribuir a carga entre os nós da rede;
- **Perda de pacotes média da árvore de distribuição:** A perda de pacotes média é calculada através da média aritmética do número de pacotes perdidos por cada nó na árvore de distribuição. Isto pode ser calculado através da expressão  $\frac{1}{n} \sum_{i=1}^n p_i$  onde  $p_i$  expressa o número de pacotes perdidos pelo nó  $i$  e  $n$  o tamanho do grupo *multicast*. O objetivo é avaliar como o aumento da taxa de transferência de vídeo influencia o número de pacotes perdidos no TVoIP e nos outros trabalhos;

- **Tempo de entrada médio da árvore de distribuição:** O tempo de entrada médio da árvore de distribuição é calculado através da média aritmética do tempo de entrada de cada nó na árvore de distribuição. Isto pode ser calculado pela expressão  $\frac{1}{n} \sum_{i=1}^n t_i$  onde  $t_i$  representa o tempo de entrada do nó  $i$  e  $n$  o tamanho do grupo *multicast*. Assim podemos comparar os diferentes tempo de entrada na rede de cada uma dos trabalhos com o do TVoIP.

## 4. Resultados

Esta seção visa avaliar o desempenho da arquitetura TVoIP frente a outras arquiteturas de acordo com as métricas de desempenho descritas na Seção 3.1. Devido à característica aleatória de escolha dos clientes (Seção 3) e por utilizarmos 5 diferentes topologias de rede, foram executadas 100 iterações para cada experimento, sendo 20 iterações para cada topologia, e calculada a média aritmética com um intervalo de confiança de 95% para cada ponto encontrado nos gráficos. O simulador ns-2 foi configurado para executar durante um período de simulação de 900 segundos para cada iteração. Resultados adicionais deste trabalho estão disponíveis em [Proença 2006].

### 4.1. Qualidade dos Caminhos dos Dados

Esta seção tem como objetivo avaliar os caminhos percorridos pelos dados observando as características: aumento de atraso em relação ao caminho *unicast*, número de pacotes duplicados nos enlaces e uso de recursos de rede.

A Figura 2 ilustra a curva função distribuição cumulativa (*Cumulative Distribution Function* - CDF) da PRA para um grupo *multicast* com 700 clientes. Cada curva corresponde ao estado da rede após a entrada do último cliente no grupo *multicast*. Podemos observar claramente que o TVoIP possui o menor valor de PRA, seguido pelo CoopNet, Narada e HMTP. Isto indica que o TVoIP consegue construir redes sobrepostas com caminhos com um atraso menor que as outras propostas. Esta característica se deve ao fato do TVoIP construir árvores baixas, devido à sua propriedade de sempre manter os nós com maior grau de saída da árvore conectados logicamente próximos da raiz. Observe que o CoopNet tem a característica de preencher a árvore de distribuição por nível, o que garante que sua árvore também possua altura baixa. Os protocolos Narada e o HMTP possuem os piores resultados, pois constroem árvores altas.

A Figura 3 ilustra de forma mais clara como a penalidade relativa ao atraso (PRA) cresce com o aumento de clientes no grupo *multicast*. Cada ponto no gráfico representa o valor médio da PRA para um determinado tamanho de grupo com sua respectiva barra de erro. Observe que o coeficiente angular da curva do TVoIP possui um valor muito baixo (próximo de zero), o que sugere que o tamanho do grupo *multicast* não influencia na PRA para nossa arquitetura.

A Figura 4 apresenta os resultados do pior caso de saturação nos enlaces. Conforme esperado, o protocolo HMTP possui os melhores resultados, uma vez que procura conectar vizinhos próximos da rede física na rede sobreposta. O TVoIP também possui esta propriedade, visto que tenta agrupar nós de acordo com os endereços de rede. No entanto, o TVoIP prioriza a propriedade de construir uma árvore baixa antes de agrupar os endereços de rede, o que faz com que os nós com maior grau de saída da árvore sempre se localizem próximos da raiz, mesmo em situações onde poderiam estar melhor posicionados na árvore. Esta característica fornece ao TVoIP o segundo melhor resultado. As

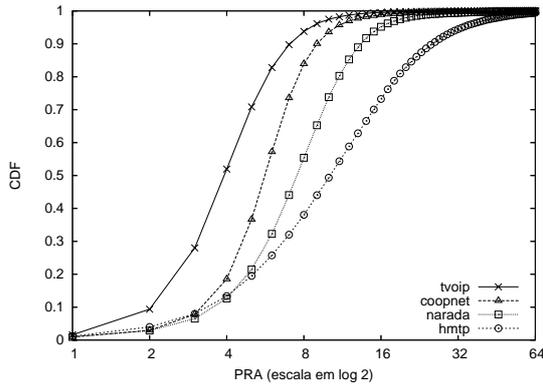


Figura 2. CDF da PRA para um grupo com 700 clientes.

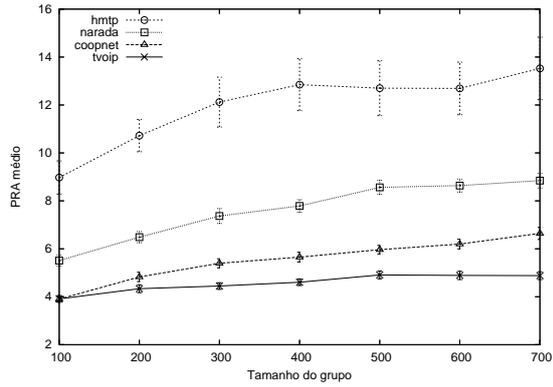


Figura 3. Valor médio da PRA para diferentes tamanhos de grupos multicast.

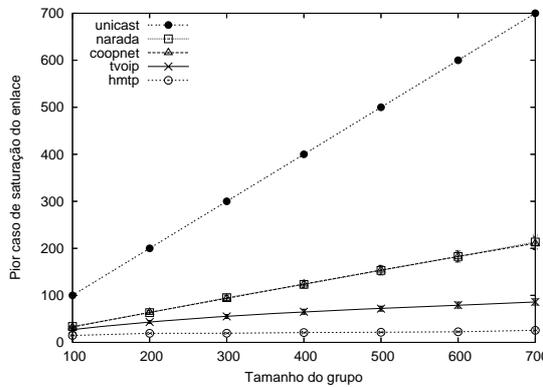


Figura 4. Pior caso de saturação de enlaces para diferentes tamanhos de grupos multicast.

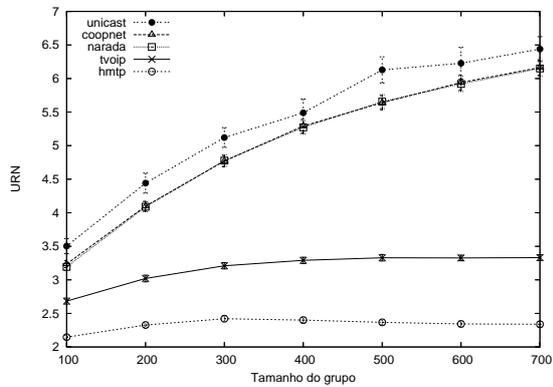


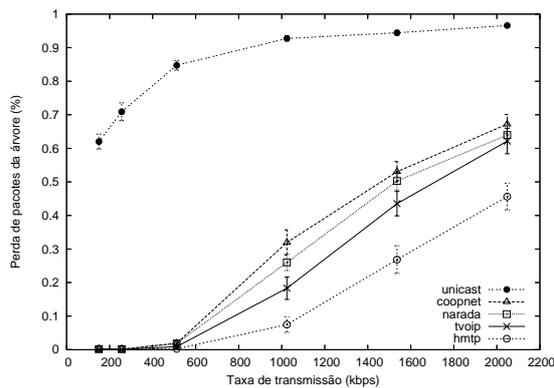
Figura 5. Valores de URN para diferentes tamanhos de grupos multicast.

arquiteturas Narada e CoopNet possuem valores de saturação similares, visto que ambas utilizam um processo de escolha aleatória do nó que irá atuar como pai do novo nó que deseja juntar-se à rede. A curva *unicast* possui os piores resultados, uma vez que os enlaces próximos ao emissor possuem maior probabilidade de saturação.

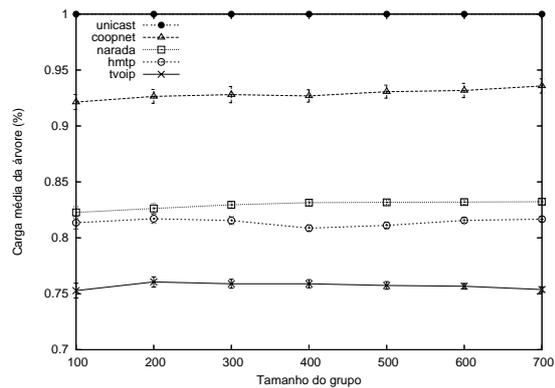
A Figura 5 ilustra a influência do tamanho do grupo *multicast* no uso de recursos da rede. Podemos observar que o protocolo HMTIP consome menos recursos de rede que os outros protocolos, consumindo em torno de 2,3 vezes mais recursos que o IP *multicast* e cerca de 42% menos recursos que o TVoIP. Isto pode ser explicado pelo fato do URN considerar a saturação dos enlaces no cálculo (Seção 3.1). Assim, podemos observar as mesmas características encontradas nos gráficos de saturação: o protocolo HMTIP possui os melhores resultados, seguido pelos protocolos TVoIP, CoopNet, Narada e *unicast* que possui os piores resultados.

#### 4.2. Desempenho nos Nós Finais de Rede

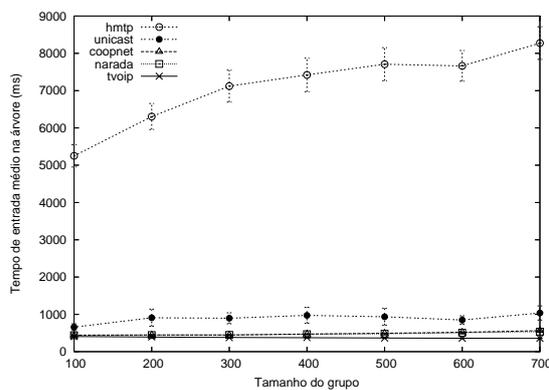
Esta seção tem como objetivo avaliar o desempenho nos nós finais de acordo com as características percebidas pelos usuários tais como atraso, perda de pacotes, tempo de entrada no sistema e carga em relação à largura de banda disponível.



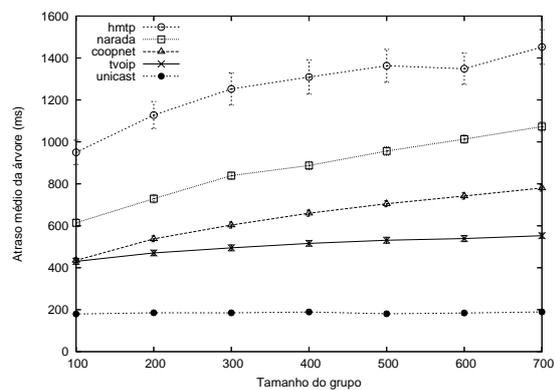
**Figura 6.** Relação entre a taxa de transmissão de vídeo e a perda de pacotes para um grupo de 100 usuários.



**Figura 7.** Relação entre o tamanho do grupo *multicast* e a carga média da árvore de distribuição para uma taxa de transmissão de vídeo de 150 kbps.



**Figura 8.** Relação entre o tamanho do grupo *multicast* e o tempo de entrada na árvore de distribuição para uma taxa de transmissão de vídeo de 150 kbps.



**Figura 9.** Relação entre o tamanho do grupo *multicast* e o atraso médio da árvore de distribuição para uma taxa de transmissão de vídeo de 150 kbps.

A Figura 6 ilustra a influência da taxa de transmissão de vídeo na perda de pacotes. Note que com exceção do *unicast*, a porcentagem de perdas para todos os protocolos começa a se tornar significativa quando a taxa de transmissão possui valores superiores a 512 kbps. Como esperado, os caminhos *unicast* possuem o maior valor de perda de pacotes. Este comportamento é justificado pela maior probabilidade de saturação nos enlaces próximos ao emissor, o que implica em maior probabilidade de congestionamento nestes enlaces. Independentemente da taxa de transmissão de vídeo, o protocolo HMTTP tem um desempenho notavelmente melhor do que os outros algoritmos, chegando a ter 36% menos perdas com uma taxa de transmissão de vídeo de 2Mbps que o TVoIP. Devido à característica de não construir uma topologia muito dispersa, o protocolo HMTTP consegue diminuir o tráfego entre domínios, minimizando desta forma o congestionamento nas filas dos roteadores dos enlaces de acesso. Mais uma vez, o TVoIP apresentou o segundo melhor resultado, visto que prioriza a propriedade de manter a árvore baixa para depois realizar o agrupamento dos nós. A arquitetura CoopNet possui a característica de

preencher a árvore de distribuição por nível, o que implica que um nível precisa estar completamente cheio para outro ser preenchido. Assim, conforme pode ser visto na Figura 7, os nós que utilizam o protocolo CoopNet experimentam a maior carga, o que implica em maior probabilidade de congestionamento nos enlaces e resulta no desempenho não satisfatório mostrado na Figura 6.

A Figura 7 ilustra a influência do tamanho do grupo *multicast* na carga média da árvore de distribuição. Observe que o TVoIP possui a menor carga na árvore de distribuição, com valores de aproximadamente 75%. Este comportamento já era esperado, devido a característica da arquitetura de sempre priorizar a propriedade de manter a árvore baixa, colocando os nós com maior grau de saída próximos à raiz da árvore. A solução *unicast* apresenta os piores resultados, com valores constantes de 100%, pois o *unicast* sobrecarrega o emissor com um número de conexões igual ao tamanho do grupo *multicast*. A arquitetura CoopNet, conforme discutido anteriormente, apresenta um resultado próximo de 93% de carga, pois preenche a árvore de distribuição por nível. Além disso, podemos observar que para todas as curvas da figura o coeficiente angular é muito baixo, indicando que o tamanho do grupo não influencia na carga média de distribuição.

A Figura 8 ilustra a influência do tamanho do grupo no tempo de entrada na árvore de distribuição. Podemos observar que o TVoIP possui o menor tempo de entrada na árvore, com valores variando de 350 ms a 400 ms. Além disso possui um tempo 50% melhor que o protocolo Narada, 57% que o protocolo CoopNet e 2218% que o protocolo HMTP para um grupo com 700 clientes. O pior desempenho do protocolo HMTP pode ser explicado através do seu processo de entrada na árvore de distribuição *multicast*, pois cada nó desce a árvore realizando medições de atraso até encontrar um bom nó para se conectar. Isto faz com que o tempo de entrada na árvore seja muito demorado, e que cresça de acordo com o tamanho do grupo *multicast*. Para se ter uma idéia, com um grupo de 700 nós, o tempo de espera médio de um nó foi de 8,27 s, o que é considerado um valor extremamente alto. Para os outros protocolos, as curvas possuem coeficiente angular muito baixo, indicando que o tamanho do grupo *multicast* não influencia o tempo de entrada médio da árvore.

A Figura 9 apresenta a influência do tamanho do grupo *multicast* no atraso médio da árvore de distribuição. Podemos observar que os caminhos *unicast* possuem os melhores resultados para o atraso, gerando uma curva quase que constante para diferentes números de clientes no grupo *multicast*. Embora o *unicast* forneça o menor atraso entre um par de nós, esta curva foi privilegiada devido à configuração de tamanho de fila de 100 pacotes no *buffer* dos roteadores do ns-2. Isto explica os altos valores de perdas de pacotes para o *unicast* na Figura 6, e os baixos valores para o atraso na Figura 9, pois são considerados somente os atrasos dos pacotes que conseguem chegar ao nó destino. Talvez os resultados pudessem ser diferentes com um tamanho de *buffer* maior, pois diminuiria o número de pacotes perdidos pelo *unicast* e conseqüentemente aumentaria o tempo de espera destes pacotes na fila, aumentando conseqüentemente o atraso. Entre os protocolos de *multicast* na camada de aplicação, o TVoIP apresentou os melhores resultados, devido à já mencionada propriedade de manter a árvore de distribuição baixa e ao fato de agrupar nós próximos de acordo com os endereços de rede. Observe que o coeficiente angular da curva do TVoIP possui um valor baixo, o que indica que o tamanho do grupo *multicast* influencia pouco no atraso para este protocolo. Isto mostra que a árvore do TVoIP cresce

umentando o número de ramificações, sugerindo uma árvore baixa e larga. Observe que este comportamento é desejado, visto que minimiza o atraso fim-a-fim entre emissor e receptor. O CoopNet por também possuir a característica de manter a árvore baixa, apresenta um bom resultado. O protocolo HMTP apresenta o pior resultado, pois não leva em consideração o grau de saída de um nó para construção da rede sobreposta, gerando árvores de distribuição altas. Com isto podemos concluir que árvores *multicast* baixas nos fornecem melhores resultados em relação ao atraso.

## 5. Conclusões e Trabalhos Futuros

Este trabalho apresentou uma arquitetura escalável para transmissão de vídeo ao vivo na Internet utilizando redes *peer-to-peer* para comunicação multi-destinatária. Foi considerado como caso de uso um cenário de Internet TV no qual existe somente um emissor transmitindo conteúdo para um grande número de receptores.

A principal contribuição deste trabalho foi o algoritmo de entrada de nós que constrói a rede sobreposta P2P. Este algoritmo utiliza o endereço de rede para cálculo da métrica de distância e sempre mantém os nós com maior grau de saída próximos à raiz da árvore. Além disso, evita sempre que possível a troca de mensagens entre os nós, para que o protocolo da nossa arquitetura tenha um baixo *overhead*. Esta característica contribui para a escalabilidade do sistema.

Através de simulações foi possível avaliar e comparar o desempenho do TVoIP com outras propostas bem conhecidas da literatura. Os resultados mostram que nossa arquitetura consegue construir redes sobrepostas com caminhos que fornecem o menor atraso em relação às outras propostas. Além disso, foi verificado que o tamanho do grupo *multicast* não influencia na penalidade relativa ao atraso para nossa arquitetura (Figura 3).

Outros experimentos mostram que nossa arquitetura consegue bons resultados para o uso de recursos da rede e saturação dos enlaces (Figuras 4 e 5). Sobre a carga média da árvore, foi verificado que o TVoIP fornece os melhores resultados (Figura 7), o que sugere uma boa distribuição da carga entre os nós do sistema. Isto contribui para um ambiente cooperativo mais justo.

A nossa arquitetura apresentou bons resultados para a perda de pacotes, conforme apresentado na Figura 6. Os resultados foram inferiores apenas para o protocolo HMTP. Isto porque o TVoIP prioriza a propriedade de manter a árvore baixa para depois realizar o agrupamento do nós, o que não ocorre com o HMTP. No entanto, aplicações multimídia são tolerantes a perdas ocasionais, que podem causar ruídos muitas vezes imperceptíveis no processo de reprodução.

Uma característica muito importante no projeto de um sistema com requisitos de escalabilidade é o tempo de entrada de um nó no sistema. Se o tempo de entrada cresce com o tamanho do grupo *multicast*, a aplicação pode se tornar inviável para grupos com um grande número de usuários. Os experimentos nos mostram que este é um ponto forte de nossa arquitetura (Figura 8), pois a mesma apresenta os melhores resultados quando comparada às arquiteturas dos outros trabalhos.

Uma vez que assumimos que cada cliente confia em um valor de capacidade de saída especificado pelo usuário (Seção 2.1), podemos ter dificuldades se o comportamento do usuário for egoísta. Mecanismos de incentivo a cooperação são apresentados

em [Fonseca e Manzato 2006] visando inibir o comportamento egoísta de usuários em redes P2P. Esperamos incluir um mecanismo de incentivo para resolver esta limitação em um trabalho futuro.

O uso de NATs e *firewalls* na Internet atual impõe restrições fundamentais sobre a conectividade de nós na rede sobreposta. Uma solução para este problema é encontrada em [Proença 2006] e por limitação de espaço não foi incluída neste artigo.

Uma outra contribuição deste trabalho é a possibilidade de inclusão do nosso nó de inicialização em outros trabalhos tais como CoopNet, Narada e HMTP. Isto não iria interferir no funcionamento dessas arquiteturas e iria fornecer um tempo de entrada constante ao nós que desejam entrar no sistema, como acontece em nossa arquitetura.

Podemos concluir com este trabalho que o uso de um nó de inicialização que forneça uma boa localização de entrada para um novo nó que deseja juntar-se ao grupo fornece uma melhora significativa de desempenho na rede sobreposta das arquiteturas de *multicast* em nível de aplicação, sejam elas de natureza centralizada ou distribuída.

## Referências

- Chawathe, Y. (2003). Scattercast: an adaptable broadcast distribution framework. *Multimedia Systems*, 9(1):104–118.
- Chu, Y., Rao, S., e Zhang, H. (2000). A case for end system multicast. In *Proceedings of SIGMETRICS*, pp. 1–12, New York, NY, USA. ACM Press.
- Fonseca, N. e Manzato, D. (2006). Incentivos em redes peer-to-peer de fluxo de media com distribuição sincronizada. In *Anais do SBRC*, pp. 1–16.
- Francis, P. et al. (2000). Yoid: Your own internet distribution. Unreferred report.
- He, Q. et al. (2003). Mapping peer behavior to packet-level details: A framework for packet-level simulation of peer-to-peer systems. In *MASCOTS*, pp. 71–78.
- Helder, D. e Jamin, S. (2002). End-host multicast communication using switch-trees protocols. In *Proceedings of CCGRID*, pp. 419, Washington, DC, USA.
- Jain, M. e Dovrolis, C. (2003). End-to-end available bandwidth: measurement methodology, dynamics, and relation with TCP throughput. *IEEE/ACM TON*, 11(4):537–549.
- Medina, A. et al. (2001). BRITE: Universal topology generation from a user's perspective. Technical Report 2001-003, Boston University.
- Padmanabhan, V., Wang, H., e Chou, P. (2002). Distributing streaming media content using cooperative networking. In *Proceedings of NOSSDAV*, pp. 177–186.
- Pendarakis, D. et al. (2001). ALMI: An application level multicast infrastructure. In *Proceedings of USITS*, pp. 49–60.
- Proença, T. (2006). Proposta e avaliação de uma arquitetura escalável para distribuição de TV na Internet. Master's thesis, Universidade Federal Fluminense (UFF).
- Wang, W. et al. (2002). Overlay optimizations for end-host multicast. In *Proceedings of NGC*.
- Zhang, B., Jamin, S., e Zhang, L. (2002). Host multicast: A framework for delivering multicast to end users. In *Proceedings of INFOCOM*, pp. 1366–1375.