

# Análise, Simulação e Correções do Protocolo MMA para Multicast em Redes Móveis Estruturadas

Renato Maia, Vagner Sacramento, Markus Endler

Departamento de Informática – Pontifícia Universidade Católica do Rio de Janeiro

Rio de Janeiro – RJ – Brazil

{maia,vagner,endler}@inf.puc-rio.br

**Abstract** - In this article we describe the prototyping and analysis by simulations of the MMA protocol (Multicast by Multicast Agent), which aimed at analyzing its behavior and evaluating its deficiencies and limitations related to the multicast message delivery for wireless mobile networks. Based on a rigorous analysis and tests in deterministic scenarios we identified that the protocol had serious problems. We made the necessary modifications to the protocol, and then compared the original and the modified versions in several stochastic scenarios. The results showed that our version has a better performance than the original MMA protocol

**Resumo** - Neste artigo foi realizada uma prototipação e análise por simulações do protocolo MMA (Multicast by Multicast Agent), com intuito de analisar o seu comportamento e avaliar suas deficiências e limitações quanto à entrega de mensagens multicast para redes móveis sem fio. A partir da análise rigorosa e de testes realizados em cenários de simulação determinísticos identificamos que este protocolo apresentava sérios problemas de funcionamento. Fizemos as modificações necessárias e em seguida comparamos a versão original e a modificada em vários cenários de simulação estocásticos. Os resultados mostraram que a nossa versão apresenta um desempenho melhor do que o protocolo MMA original

## I. INTRODUÇÃO

Muitas aplicações distribuídas necessitam transmitir a mesma informação para múltiplos destinos, como por exemplo, aplicações de video-conferência, processamento distribuído, jogos, dentre outras. Uma maneira eficiente de enviar dados de uma fonte para múltiplos destinos é a utilização de um serviço de *multicast*. Alguns protocolos de *multicast* existente na rede fixa (rede cabeada) são: DVMRP [DEERING 88] e MOSPF [MOY 94].

Um serviço de *multicast* provê uma forma eficiente de comunicação multiponto entre vários *hosts*, pois possibilita mais rapidez na entrega das mensagens e principalmente o uso adequado dos recursos de rede (e.g. largura de banda). Sendo assim, este serviço é apropriado para diferentes tipos de rede, incluindo as redes móveis infra-estruturadas, onde os enlaces sem fio apresentam largura de banda reduzida e maior probabilidade de falhas. Além de implementar as funcionalidades requeridas para a disseminação eficiente das mensagens na parte cabeada da rede móvel, os protocolos de

Multicast para redes móveis devem também tratar a migração de hosts entre pontos de acesso da rede (*handoff*<sup>1</sup>).

Existem várias propostas [Suh, Shin and Kwon 2001], [Sadok, Cordeiro and Kelner 2000], [Acharya 1996], [Acharya 2000] para protocolos multicast que atendam as exigências das redes móveis sem fio infra-estruturadas.

Um dos principais problemas a serem tratados pelos protocolos de Multicast para redes móveis sem fio é realizar a entrega de datagramas multicast pelo melhor caminho levando em conta a mobilidade dos hosts. Para comunicação unicast de datagramas o protocolo IP móvel [Perkins 1997] é uma solução que garante a transparência de mobilidade. Isto é feito através da utilização de um Home Agent (HA) para cada host móvel, que trata do encaminhamento de datagramas para a localização corrente deste host. Para cada host móvel que esteja fora da sua rede origem, o HA correspondente mantém em um cache a informação sobre a rede na qual este host se encontra no momento (rede visitada). Este endereço é denominado care-of-address. Todo datagrama destinado ao host móvel é primeiramente encaminhado para o seu HA, e quando este identifica que o host móvel (MH) está fora da rede de origem, o HA encapsula o datagrama original em um novo datagrama IP (tunelamento) e o re-encaminha para um Foreign Agent (FA), na atual rede visitada pelo host móvel. Este re-encaminhamento é realizado com base no care-of-address. Ao receber os datagramas, o FA desencapsula e re-encaminha os datagramas para o host móvel visitante.

Neste artigo, foi realizada uma prototipação e análise por simulações do protocolo MMA (Multicast by Multicast Agent) proposto por [Suh, Shin and Kwon 2001], com o objetivo de compreender melhor o seu funcionamento e avaliar a sua complexidade e eventuais limitações. Como resultado desta análise identificamos uma série de problemas na especificação deste protocolo, e propusemos e implementamos soluções para corrigir estes problemas, fazendo com que o protocolo possa de fato ser usado em uma rede móvel. Para a prototipação e simulação do protocolo MMA original e do protocolo modificado utilizamos o ambiente MobiCS [Rocha and Endler 2001].

O restante do artigo está estruturado da seguinte forma: a seção 2 apresenta alguns trabalhos relacionados; a seção 3 descreve resumidamente o protocolo MMA; a seção 4 descreve a nossa implementação do protocolo

<sup>1</sup> Operação realizada durante a migração de um *host* móvel de uma rede para outra.

MMA, apresentando brevemente o framework MobiCS e alguns cenários utilizados na simulação, onde são apresentados os problemas encontrados e as soluções propostas para o protocolo MMA; a seção 5 apresenta uma simulação estocástica do protocolo e a seção 6 contém a conclusão deste trabalho.

## II. TRABALHOS RELACIONADOS

### II.1. IP MÓVEL

A especificação do IP Móvel contempla duas abordagens para suportar serviços de *multicast* para os *hosts* móveis [Xylonmenos and Polyzos 1997]: *multicast* baseado no *foreign agent* e *multicast* baseado no *home agent* [Chikarmane and Williamson 1998]. Estas abordagens utilizam os mesmos princípios do IP móvel para prover a transparência de mobilidade de *hosts* para datagramas *multicast*.

No *multicast* baseado no *foreign agent*, o *host* móvel tem que se registrar no grupo *multicast* sempre que ele move para uma rede visitada. Este esquema tem a vantagem de oferecer um caminho de roteamento ótimo e a inexistência de datagramas duplicados, além de não ser necessário qualquer encapsulamento para encaminhamento da mensagem. Contudo, quando o *host* móvel apresenta alta taxa de mobilidade, o gerenciamento da árvore de *multicast* pode gerar muito overhead.

No *multicast* baseado no *home agent*, o datagrama *multicast* é encaminhado através do tunelamento *unicast* do IP móvel via HA. Quando o *home agent* recebe um datagrama *multicast* destinado para um *host* móvel, ele encapsula este datagrama duas vezes (uma vez com o endereço IP do *host* móvel, e outra com o care of address do FA) para então transmiti-lo para o *host* móvel como um datagrama *unicast*. O encapsulamento duplo aumenta o tamanho do pacote, e os datagramas geralmente acabam sendo enviados por um caminho que não seja ótimo, uma vez que a rota de encaminhamento sempre passa pelo HA. Além do mais, se vários *hosts* móveis que pertencem à mesma rede local estão na mesma rede visitada (rede externa), cópias duplicadas dos datagramas *multicast* chegarão ao destino. Assim, nenhuma das duas propostas do IP móvel se mostram como uma boa solução para implementar serviços de *multicast*.

### II.2. MOM (MOBILE MULTICAST)

O MoM [Harrison 1997] propõe um protocolo *multicast* baseado no *home agent*, sendo este responsável pelo tunelamento de datagramas *multicast* para os *hosts* móveis. O MoM essencialmente define soluções para alguns dos problemas existentes no *multicast* baseado no *home agent* do IP móvel, discutido na seção anterior.

No esquema de *multicast* baseado no *home agent* apresentado na seção 2.1, um *home agent* repassa uma cópia separada do datagrama *multicast* para cada *host* móvel. Se todos *hosts* móveis que desejam receber o datagrama *multicast* estiverem na mesma rede visitada, n cópias duplicadas serão encaminhadas para o mesmo *foreign agent*. Para resolver este problema, no protocolo MoM, o *home agent* repassa uma única cópia do

datagrama *multicast* para cada rede visitada que contém algum *host* móvel do grupo *multicast*. Ao receber o datagrama *multicast*, o *foreign agent* o encaminha para os *hosts* móveis usando um link em nível de *multicast*. Este esquema reduz o número de datagramas *multicast* duplicados e, economiza a banda passante nos links wireless e wired.

Um outro problema resolvido pelo protocolo MoM é a convergência de múltiplos túneis de *home agents* diferentes estabelecidos com um mesmo *foreign agent*, mostrado na Figura 1. Quando múltiplos *home agents* têm *hosts* móveis na mesma rede visitada, uma cópia de cada datagrama é repassada para o mesmo *foreign agent* por cada *home agent*. Como solução para este problema, o MoM propõe que o *foreign agent* defina somente um único *home agent* como sendo o seu Designated Multicast Service Provider (DMSP) para um dado grupo *multicast*. Assim, o DMSP envia um único datagrama através do túnel para o *foreign agent*, enquanto os outros *home agents* que não são um DMSP ficam aguardando sua vez para transmitir. A solução proposta através do DMSP é mostrada na Figura 2.

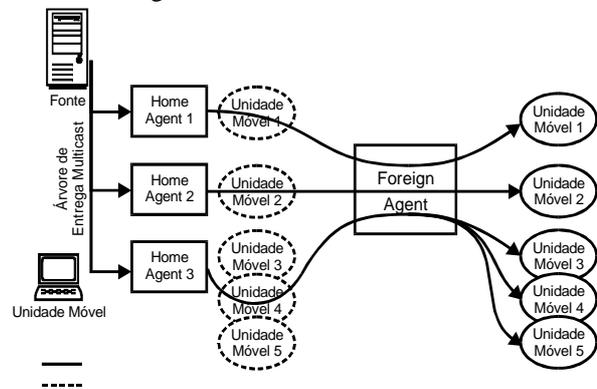


Figura 1. Problema de convergência de tunelamento.

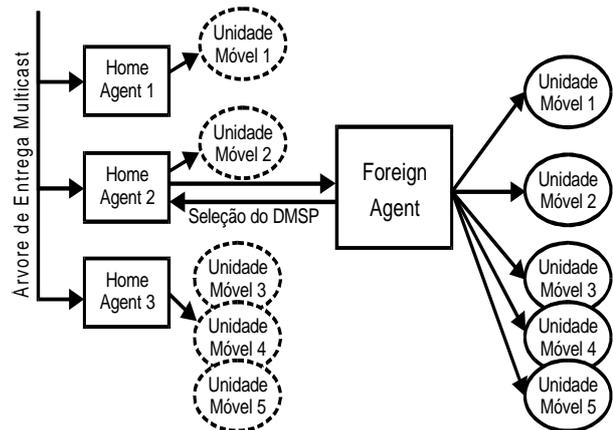


Figura 2. Seleção do DMSP no protocolo MoM.

Apesar de reduzir o tráfego *multicast*, o MoM apresenta problemas de duplicação de pacotes, e problemas na escolha de caminhos otimizados para entrega do datagrama *multicast* [Suh, Shin and Kwon 2001].

## III. O PROTOCOLO MMA

O protocolo analisado, chamado de MMA (*Multicast by Multicast Agent*), propõe um novo esquema de

comunicação multicast, introduzindo os conceitos de *Multicast Agent (MA)* que vai estar presente em uma estação de suporte de mobilidade (*Mobile Support Station – MSS*) e *Multicast Forwarder (MF)* [Suh, Shin and Kwon 2001]. Essencialmente, os MAs são *foreign agents* que são responsáveis por repassar os datagramas multicast para os hosts móveis presentes na sua rede local. Um MF de um grupo multicast é um MA que além disto faz parte da árvore de multicast do grupo. Cada MA define um MF do qual deverá receber retransmissões de datagramas para os hosts móveis de um grupo *multicast*. Esta associação de um único MF por MA é definida dinamicamente, levando-se em consideração também a indicação de um MF por um host móvel a cada *handoff*. Quando a rede local de um MA faz parte da árvore de entrega *multicast*, o próprio MA pode fazer o papel de MF. Mas o MF associado a um MA cuja rede não faz parte da árvore *multicast* deverá ser um MA de outra rede (que pertença ao grupo *multicast*). Além disto, a associação de um MF para um MA é específica para cada grupo *multicast*.

No protocolo MMA, são utilizados dois métodos distintos para a definição da fonte de datagramas multicast, dependendo se a rede sendo visitada por um MH pertence ou não a uma árvore de multicast. Se a rede sendo visitada pertence à árvore multicast, os dados repassados ao host móvel são obtidos pelo MA diretamente do roteador multicast local da rede. Se a rede sendo visitada não pertence à árvore multicast, os dados multicast são obtidos através de um túnel do MF, ou seja, um MA que esteja incluso na árvore do grupo multicast e que esteja localizado em uma rede próxima da rede sendo visitada pelo host móvel.

### III.1. DESCRIÇÃO DO PROTOCOLO

Essencialmente, o protocolo MMA funciona da seguinte maneira: quando um *host* móvel migra de uma rede (e.g. N1) para outra rede (e.g. N2), este envia a informação sobre o seu MF (MA da rede N1) para o MA na rede N2 durante o processo de registro na nova rede, informação esta que é utilizada pelo MA para selecionar o novo MF. Se N2 pertence à árvore de entrega *multicast*, o próprio MA de N2 se torna o MF, e tanto o MA quanto o *host* móvel atualizam os seus registros sobre o *MF corrente* com a identificação do MA de N2. Se o MA de N2 não pertence à árvore de entrega *multicast* e o MA não possui informação sobre MF no grupo *multicast*, o registro de *MF corrente* que do *host* móvel é usado também em N2. Se o MA em N2 não pertence a árvore de entrega *multicast*, mas o MA tem informação sobre um MF no grupo *multicast*, há duas possibilidades: a) Na primeira o valor de MF que o *host* móvel possuía em N1 é usado diretamente como MF em N2 (seleção do MF mais antigo), ou b) Na segunda, o MA seleciona o agente que está mais próximo a ele dentre aquele informado pelo *host* móvel e aquele anteriormente conhecido pelo MA (seleção do MF mais próximo). Em ambos os casos, o MA e o *host* móvel são atualizados com o valor do MF escolhido.

O registro na árvore multicast deve ser requisitado pelos hosts móveis visitantes durante o processo de registro em um determinado grupo multicast. Sempre que

um *host* móvel migra para uma nova rede e se registra com um novo MA, o novo MA executa o processo de registro à árvore multicast caso tal funcionalidade tenha sido requisitado pelo *host* móvel. Este processo é similar ao dos protocolos multicast baseados em *foreign agents* (conforme descrito na seção 2.1).

Se o *host* móvel se “conecta” a um MA que não faz parte da árvore multicast, este MA estabelecerá um túnel com o MF para que este retransmita as mensagens encaminhadas para o grupo multicast do qual o *host* móvel faz parte. A confirmação do estabelecimento deste túnel no MA da rede visitada ocorre através do recebimento do primeiro datagrama multicast do grupo dentro de um intervalo de tempo pré-estabelecido.

## IV. IMPLEMENTAÇÃO

Nesta seção apresentaremos o ambiente utilizado para a prototipação e simulação do protocolo MMA, e explicaremos algumas simplificações sobre a rede que fizemos em nossas simulações. Em seguida, discutiremos os principais problemas detectados na especificação original do protocolo MMA (contida em [Suh, Shin and Kwon 2001]) a partir de cenários específicos de execução, e finalmente apresentaremos as nossas soluções para os problemas, que foram implementadas em uma versão modificada do MMA.

### IV.1. MOBICS

MobiCS [Rocha and Endler 2001] é um *framework* desenvolvido em Java que permite a criação de um ambiente customizado para a prototipação e simulação de protocolos distribuídos para redes móveis infra-estruturadas. O *framework* permite que usuário defina as propriedades dos elementos de rede (hosts móveis, estações fixas, pontos de acesso, enlaces, etc.) e a configuração da rede. Além disto, induz o usuário a programar o seu protocolo de forma modular e totalmente independente do(s) modelo(s) de simulação usado.

Um modelo de simulação define todos os efeitos externos ao protocolo sendo prototipado, tais como o padrão de mobilidade dos hosts móveis, a relação de vizinhança entre as redes sem fio, o padrão de conectividade dos hosts móveis e propriedades dos enlaces sem fio, entre outros. MobiCS permite tanto o uso de modelos pré-concebidos, como também a criação de novos modelos de simulação, dando assim ao usuário a liberdade de criar um ambiente de simulação com um grau variável de detalhes e com abstrações adequadas para avaliar certas propriedades de seu protocolo. Por exemplo, para alguns protocolos pode ser necessário definir e programar o roteamento de mensagens/pacotes entre hosts fixos, enquanto que em outros protocolos isto pode ser irrelevante, bastando que se defina uma latência média para comunicação entre quaisquer dois hosts fixos.

Em MobiCS, são previstos dois modos de simulação: o determinístico e o estocástico. No modo determinístico o usuário define um (ou mais) script de simulação, cada qual descrevendo uma situação (ou cenário) bem específica, na qual deseja testar o seu protocolo. Neste script, são indicados explicitamente eventos como

migrações de hosts, desconexões e reconexões, bem como as requisições do serviço implementado pelo protocolo sendo testado (por exemplo, uma entrega multicast) provenientes de uma suposta “aplicação cliente”. Além disto, define-se nestes scripts a ordem relativa (causal) de ocorrência destes eventos. Simulando o protocolo segundo cada um destes scripts, o usuário é capaz de detectar se o seu protocolo apresenta o comportamento esperado para o cenário correspondente. Portanto, o modo determinístico é geralmente utilizado para a depuração de protocolos.

No modo estocástico o objetivo é testar o comportamento e avaliar a complexidade de um protocolo em cenários maiores e semi-aleatórios. Neste modo, a geração dos eventos acima citados é determinada por probabilidades escolhidas pelo usuário. Para isto, o usuário programa geradores de eventos aleatórios para cada tipo de evento (migração, desconexão, atraso, falha, etc.). Por exemplo, para um host móvel, os eventos relevantes são geralmente migrações, desconexões e reconexões. Em ambos os modos, o usuário é capaz de instrumentar o seu simulador a fim de poder contar número de ocorrências de eventos, ou visualizar o estado das estruturas de dados usadas em seu protocolo.

#### IV.2. SIMPLIFICAÇÕES ADOTADAS

Ao invés de simular o MMA para uma configuração de rede complexa e possivelmente com centenas de parâmetros, adotamos algumas simplificações a fim de manter a simulação pequena e gerenciável. No entanto, as simplificações adotadas foram definidas de forma a não influenciar ou induzir um determinado comportamento do protocolo. Nesta sub-seção são descritas algumas simplificações adotadas para a simulação do protocolo MMA no ambiente MobiCS. Apesar destas simplificações, objetivou-se a conformidade total com à especificação do protocolo original. As simplificações adotadas foram definidas de forma a não alterar o princípio de funcionamento do protocolo, estas são descritas abaixo:

**Deteção de migração dos hosts móveis:** A descrição original do protocolo sugere que a migração dos hosts móveis seja detectada através de um time-out na rede de origem. Dessa forma é necessário que os hosts móveis enviem periodicamente um beacon (ou uma mensagem IamAlive) para o atual MA informando que ainda se encontram (ativos) na rede deste MA. Entretanto esta abordagem se mostrou difícil de ser implementada no MobiCS. Por esta razão, abstraiu-se desta deteção, e em vez disto, implementou-se que no momento da migração para a nova rede, o host móvel informa ao novo MA a rede da qual ele está vindo. Assim, o novo MA pode enviar uma mensagem ao antigo MA informando-o da migração do host.

**Seleção do ótimo MF:** A descrição do protocolo MMA não define um critério específico para selecionar o MF mais próximo/adequado no momento da migração de um host móvel que traga consigo o registro de um MF. Na implementação apresentada neste trabalho é feita a seleção do último MF (oldest selection), que consiste em sempre escolher o MF trazido pelo host móvel.

**Topologia da rede cabeada:** Todos os cenários avaliados são baseados numa rede composta por um conjunto de estações de suporte à mobilidade (MSSs, ou MAs, na terminologia do MMA) interligadas pelo meio cabeado (wired). Dentre as quais existe um subconjunto dessas estações que estão ligadas com um roteador multicast (MR), podendo portanto fazer parte de alguma árvore de entrega multicast. O roteador multicast é o núcleo da rede e é responsável por receber e repassar as mensagens de multicast para os MAs conforme ilustrado na Figura 3.

**MHs estacionários:** Para evitar que alguns MAs saiam da árvore de multicast em virtude da migração de todos os seus MHs, em alguns casos um MH é mantido estacionário em cada MA, de forma que este nunca saia da árvore de multicast.

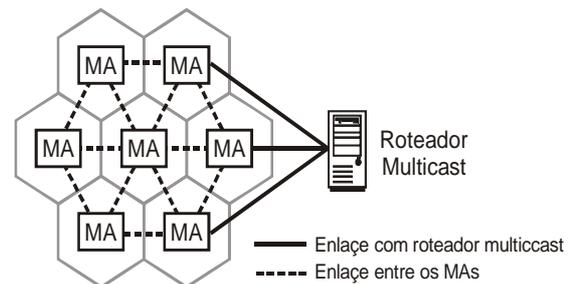


Figura 3. Conjunto de MAs interligados a um roteador multicast.

#### IV.3. CENÁRIOS DETERMINÍSTICOS

Nesta seção, são apresentados os cenários determinísticos que permitiram a deteção dos problemas do protocolo MMA. Antes de programar o *script* que descreva o cenário específico a ser simulado no MobiCS, deve-se determinar qual é exatamente o comportamento do protocolo a ser testado em cada cenário, escolher o conjunto mínimo de elementos de rede e definir a ordem (causal) da ocorrência de eventos externos ao protocolo que sejam necessários para criar a situação desejada. Em seguida, a sequência de eventos são codificadas no *script*. O MobiCS provê várias primitivas para a programação do *script*, tais como, pontos de sincronização global, ocorrência de uma migração ou de uma desconexão/reconexão de um MH, habilitar/desabilitar a recepção de mensagens ou eventos de *timer*, dentre outros.

O MMA foi simulado em vários cenários diferentes com o intuito de testar o seu comportamento em várias situações extremas ou possivelmente não previstas pelos seus autores, como por exemplo, quando mensagens são perdidas ou recebidas fora da ordem esperada. Nestas simulações avaliamos a eficiência e a corretude do protocolo, por exemplo, se ocorre duplicação e/ou perda de mensagens no serviço de multicast.

##### IV.3.1. PROBLEMAS DO PROTOCOLO ORIGINAL

###### CENÁRIO 1 - PROBLEMA DA SAÍDA DEFINITIVA DO MA DA ÁRVORE DE MULTICAST

O protocolo MMA determina que quando todos os MHs de um determinado MA migram para uma outra

rede sem fio (célula) e, este MA pára de repassar mensagens multicast para outros MAs por tunelamento, então ele deve sair da árvore de multicast através do envio de uma mensagem PRUNE. Esta característica pode levar a diversas situações indesejáveis, como por exemplo, que um dado MA saia da árvore de roteamento multicast (devido ao recebimento de uma mensagem ForwardSTOP) e em seguida receba um pedido (atrasado) de tunelamento de um outro MA para um MH que o tenha indicado como MF. Esta situação é ilustrada no diagrama espaço-tempo da Figura 4, onde, após a migração do MH0 da rede N2 para N3, o MA N1 sai da árvore

multicast (de-registrando-se junto ao MR) pois supostamente não possui mais qualquer MH do grupo sob sua responsabilidade, e além disto recebe a mensagem ForwardSTOP antes da mensagem ForwardREQUEST.

Note que este problema impossibilita a utilização do protocolo, pois não há meios de se garantir que um dado MA tenha sempre algum MH membro do grupo em sua rede, ou então mantenha sempre a função de MF para MHS em outras redes. Em sua versão original, portanto, o protocolo MMA possibilitaria que o serviço multicast se tornasse indisponível de forma permanente em vários pontos de acesso.

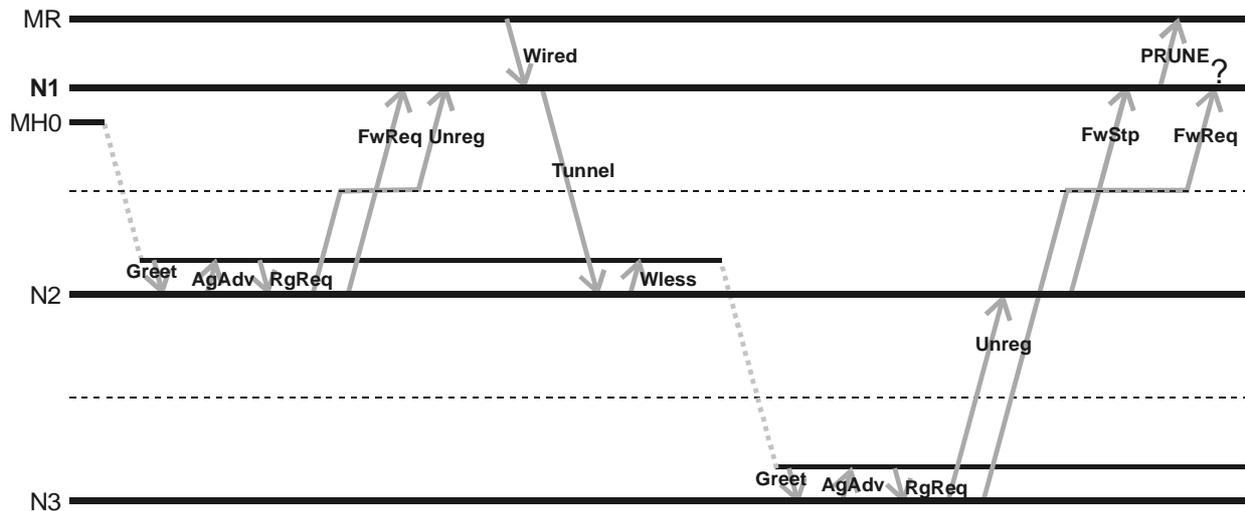


Figura 4. Cenário mostrando o problema da saída prematura do MA da árvore multicast.

**CENÁRIO 2 – PROBLEMA DOS MÚLTIPLOS TÚNEIS ABERTOS**

A especificação do protocolo MMA define que o estabelecimento de túneis entre os MAs seja feito pelo envio de mensagens de controle (i.e. ForwardREQUEST) que podem ser perdidas caso não haja rota disponível entre os MAs. Desta forma, é necessário algum mecanismo para a confirmação do estabelecimento do novo túnel, a fim de permitir que o túnel anterior/antigo seja encerrado. Essa confirmação ocorre no momento do recebimento do primeiro datagrama tunelado proveniente do novo MF antes de um determinado tempo de espera (i.e. timeout). Entretanto, a especificação do protocolo MMA não trata a situação na qual este primeiro datagrama enviado pelo novo túnel chega após o tempo de espera. Assim sendo, o protocolo MMA permite que vários túneis para um mesmo MH em um grupo sejam mantidos abertos simultaneamente, o que acarreta um problema crítico de duplicação de mensagens.

Esta situação é ilustrada pelo cenário da Figura 5, onde o MA da rede N3 inicialmente está recebendo datagramas tunelados pelo MA da rede N1 (supondo-se que N1 tenha sido supostamente indicado como MF de MH1 (Fases 1 e 2)). Entretanto, quando o MH0 (originalmente sitiado na rede N1) migrar da rede N2 para a rede N3 (Fase 5), um novo pedido de tunelamento será enviado para o MA de N2. Sendo assim, o MA da rede N3 terá associado a si temporariamente dois túneis para o mesmo grupo multicast. O túnel estabelecido com N1 só será finalizado quando a confirmação de estabelecimento

de tunelamento de N2 for recebida. De acordo com o cenário ilustrado, o MA de N3 receberá mensagens duplicadas (Fase 6), uma enviada pelo MA de N1 e outra enviada pelo MA de N2, e ambas serão repassadas para todos MH's locais que fazem parte do grupo multicast.

Essa situação tem uma agravante, pois um número arbitrário de mensagens duplicadas pode ser continuamente recebidas e repassadas para os MH's locais. A duplicação de datagramas ocorrerá também sempre que a confirmação de estabelecimento do tunelamento enviada por N2 chegar depois que o timeout ocorrer, pois o protocolo MMA não trata adequadamente este caso e, portanto, nunca enviará espontaneamente a mensagem ForwardStop para N1. Ou seja, N3 continuará recebendo datagramas multicast através de dois túneis indefinidamente. Novamente, o problema está no fato de que a especificação do protocolo MMA não define um procedimento adequado para encerrar os túneis abertos.

Este problema faz com que o protocolo MMA não satisfaça a propriedade essencial do protocolo multicast, que é o de evitar que a mesma mensagem seja transmitida a partir de várias fontes diferentes para o mesmo destino. Além do mais, há um desperdício da banda passante na rede cabeada, e também no meio sem fio devido à retransmissão de mensagens duplicadas para os hosts móveis.

**CENÁRIO 3 – PROBLEMA DO RETORNO DO MH**

Quando um MH migra para uma nova rede, durante o processo de registro do MH junto ao novo MA é

selecionado o melhor MF dentre o atual utilizado pelo novo MA e aquele informado pelo MH. O protocolo MMA verifica se o MF selecionado é igual ao MF informado pelo MH, e neste caso é capaz de deduzir o seguinte: (a) o MA está recebendo as mensagens de multicast por um túnel com outro MA, pois se o MA estivesse incluso na árvore de multicast, o MF escolhido

seria o próprio MA, que sempre seria selecionado como MF ótimo por possuir um roteador multicast disponível. (b) um novo túnel deve ser estabelecido com o MF informado pelo MH, pois este foi o MF escolhido. Com base nestas informações o protocolo MMA decide estabelecer um novo túnel com o MF escolhido.

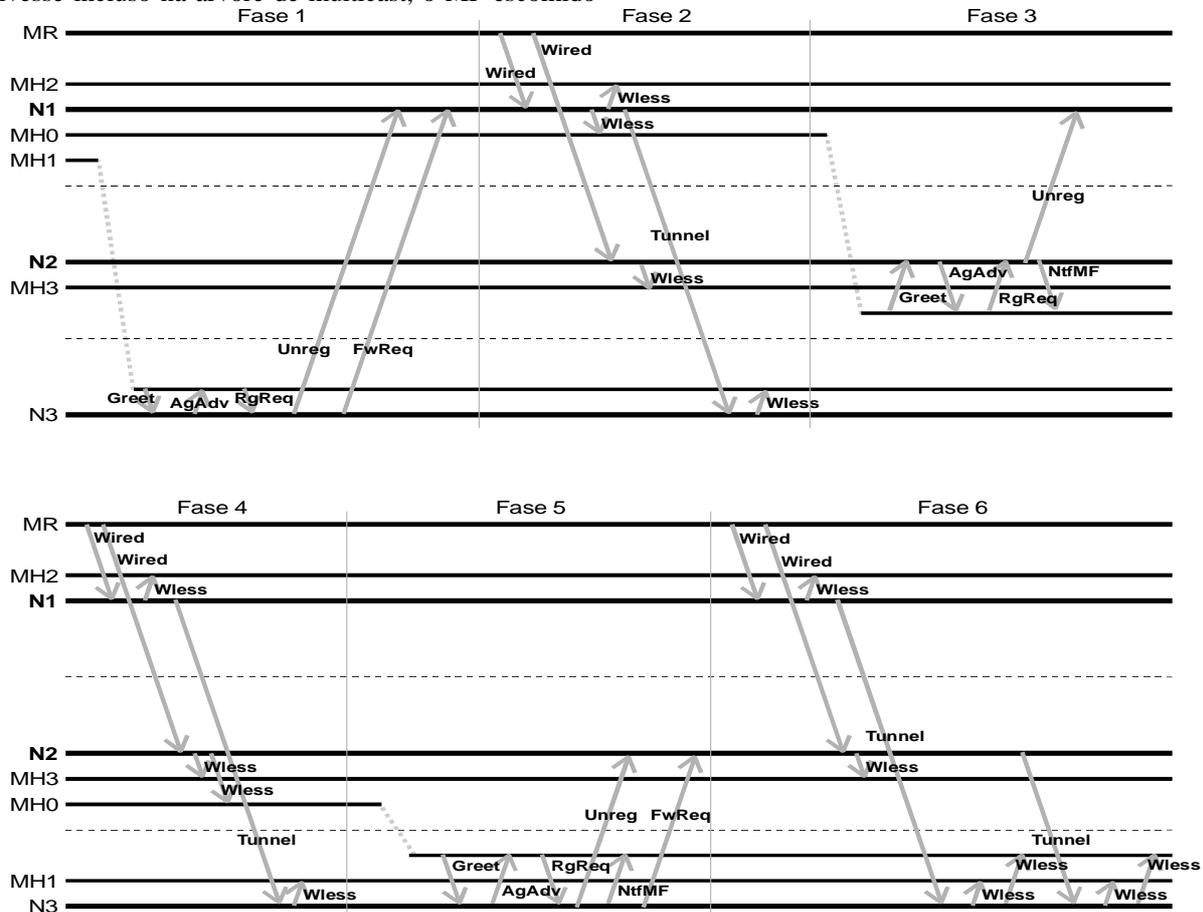


Figura 5. Cenário mostrando o problema da abertura de múltiplos túneis.

É importante notar que as deduções discutidas anteriormente somente podem ser feitas corretamente se o MF informado pelo MH for diferente do MF utilizado pelo MA. Entretanto isso nem sempre ocorre, pois mesmo após migrar para outras redes o MH ainda pode manter o mesmo MF, pois este pode também ser selecionado como MF ótimo. Desta forma, é possível que o MH retorne a uma rede trazendo consigo o mesmo MF associado ao MA. Neste caso, podem ocorrer duas situações: (a) se o MA estiver recebendo datagramas através de um túnel com outro MA, então será feita uma solicitação para o estabelecimento de um novo túnel com o MF atual. Entretanto, o MA aguardará a confirmação do

estabelecimento do “novo” túnel. Se um datagrama multicast for enviado antes do tempo para confirmação o novo túnel, o túnel atual será encerrado. Note que neste caso o serviço de entrega multicast se torna indisponível até que o MA decida trocar de MF. (b) se o MA fizer parte da árvore, então será feita uma solicitação para o estabelecimento de um túnel com o próprio MF. A formação de túneis em círculo é extremamente desastrosa, pois o recebimento de um novo datagrama faz com que o MA fique bloqueado indefinidamente num loop de envio de mensagens tanto pelo meio cabeado (através de túneis com outros MAs) quanto pelo meio sem fio, como ilustrado na Figura 6.

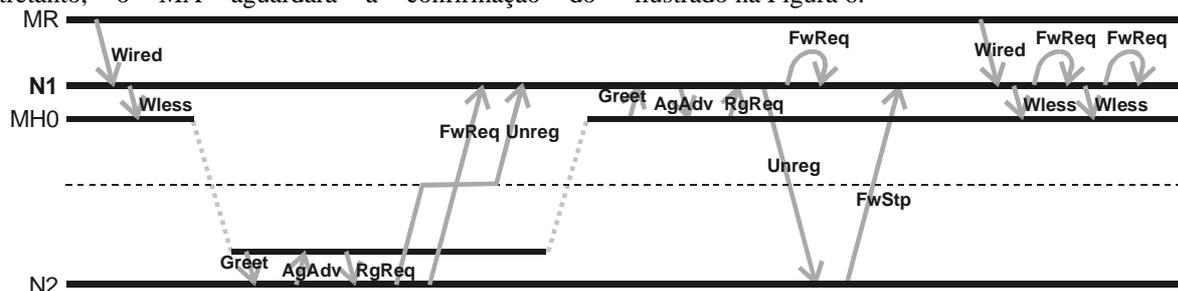


Figura 6. Cenário mostrando o retorno do MH utilizando o mesmo MF.

### IV.3.2. O PROTOCOLO MMA MODIFICADO

Nesta seção apresentaremos algumas correções e extensões do protocolo que solucionam os problemas discutidos na seção anterior. As alterações foram incorporadas preservando a estrutura do protocolo e mantendo a essência de sua funcionalidade para comunicação multicast para redes móveis.

O problema apresentado no Cenário 1 se deve ao fato do protocolo MMA não tratar a situação de um MH eventualmente precisar usar um MA como MF que já tenha saído da árvore de multicast. É importante ressaltar que é correto fazer com que um MA saia da árvore de multicast caso não existam mais MH's em sua rede ou túneis associados ao grupo multicast, pois isso evita que datagramas multicast sejam enviados pelo meio cabeado desnecessariamente. Deve-se observar também que o quanto maior for o número total de MHs no sistema, menor será a probabilidade de um MA sair da árvore de multicast.

A solução proposta para a correção do problema consiste em fazer com que o MA se reintegre a árvore multicast caso receba um novo pedido de tunelamento de outro MA. Apesar da operação de reintegração à árvore ser computacionalmente custosa, essa abordagem se parece ser mais eficiente do que manter um MA na árvore multicast mesmo que para efeito da entrega multicast isto não seja necessário. Além do mais, acreditamos que o Cenário 1 seja relativamente raro, de forma que haverá poucos casos em que um MA saia e tenha que se reintegrar à árvore de multicast em um curto espaço de tempo. A implementação desta solução é mostrada no trecho de pseudo-código do algoritmo MMA para um MA (no papel de MF), ilustrado na Figura 7.

```

IF(Forwarding STOP message from a foreign MA is received){
  Delete the sender MA from FA_List from
  the group entry;
  IF(Group member does not exist in group
  entry anymore){
    Sends PRUNE message;}
}ELSE IF(Forwarding REQUEST message from a foreign MA
is received){
IF(Group entry does not exist in Group Information Table)
{
  Make new group entry;
  Send join message to multicast delivery tree;
}
  Add the sender MA to FA_List in group
  entry;
}

```

Figura 7. Correção do problema da saída prematura do MA da árvore de multicast.

O problema descrito no cenário 2 trata do estabelecimento de múltiplos túneis, pois o protocolo MMA não trata a possibilidade da confirmação do estabelecimento do túnel não ser recebida no tempo esperado (i.e. timeout). Sendo assim, datagramas multicast serão recebidos por túneis com MFs diferentes do MF corrente. Portanto, se um datagrama for recebido nessa circunstância, induz-se que este datagrama veio através de um túnel antigo, que então deve ser encerrado. Desta forma, elimina-se o problema de datagramas

duplicado nas mensagens subsequentes transmitidas para um dado grupo multicast.

A solução proposta é descrita no trecho de pseudo-código do algoritmo MMA mostrado na Figura 8, que demonstra o procedimento realizado quando um datagrama multicast é recebido.

O problema descrito no cenário 3 é devido o protocolo MMA não tratar o fato de um MH informar o mesmo MF utilizado pelo MA. Ao receber um novo pedido de registro de um MH não é necessário selecionar um novo MF se o informado pelo MH for o mesmo utilizado pelo MA.

```

IF(datagram is forwarded from the MF before time-out
occurs){
  Send Forwarding STOP message to old MF;
  Notify new MF to all mobile hosts;
  Update its MF with new MF;
}ELSE{
  Send Forwarding STOP message to the MF;
  Ignore tunnelled datagram;
}

```

Figura 8. Correção dos problemas de múltiplos túneis abertos.

```

Registration Request Message is received from MH
Take the information (Group_ID, MF, JOIN_Option);
IF(Group ID is already registered in Group Information
Table){
  Add MH to MH_List;
  IF(MF received from MH different to MF){
    Inform MH of its MF value;
    Select new optimal MF;
    IF(Selected MF equals to MF received from MH){
      Send Forwarding REQUEST message to new MF;
      IF(datagram is forwarded from the MF before
      time-out occurs){
        Send Forwarding STOP message to old MF;
        Notify new MF to all mobile hosts;
        Update its MF with new MF;
      }
    }
  }
  * On receiving a registration request.
  Registration Request Message is received from MH
  Take the information (Group_ID, MF, JOIN_Option);
  IF(Group ID is already registered in Group Information
  Table){
    Add MH to MH_List;
    IF (MF received from MH is not MA itself) {
      Inform MH of its MF value;
      Select new optimal MF;
      IF(Selected MF equals to MF received from MH)
      Send Forwarding REQUEST message to new MF;
    }
  }ELSE{
    Make new group entry;
    Add MH to MH_List;
    IF (MF received from MH is not MA itself) {
      Send Forwarding REQUEST message to MF of MH;
      Set up MF value with MF of MH;
    } ELSE
    Send join message for connection to multicast
    delivery tree;
  }
}

```

Figura 9. Correção para os problemas de processamento de requisições para estabelecimento de tunelamento

Para resolver o problema descrito no cenário 3, é realizada uma validação do MF informado pelo MH. Se o MF especificado pelo MH na mensagem RegistrationRequest for o MF corrente utilizado pelo MA, as seguintes condições devem ser analisadas: Se o MA está na árvore multicast, o MH deve somente ser adicionado na lista de MHs, pelo contrário, o MA deve enviar uma mensagem Join para o MR, com o propósito de juntar-se a árvore para receber as mensagens multicast destinadas ao MH. Deve ser notado que nesta ocasião o MA nunca envia uma mensagem de ForwardRequest para o seu próprio endereço, conforme o problema abordado no cenário 3. Ao invés disso, outros procedimentos são tomados, dependendo se o MA está ou não na árvore multicast. Ressalta-se que estes procedimentos serão executados somente na ocasião em que um determinado MH, sair da rede que o MA se encontra, e em um momento posterior retornar para esta mesma rede tendo o MA como seu MF. É muito provável que os MHs troquem de MF durante o seu deslocamento pelas outras redes. A solução proposta resolve o problema abordado e não aumenta o custo de gerenciamento da árvore de multicast, pois ela vai ser executada somente em situações muito específicas. O código da Figura 9 mostra a implementação desta solução.

## V. TESTES ESTOCÁSTICOS

Nesta seção é feita uma avaliação do impacto das alterações feitas no protocolo MMA sobre a sua performance através de simulações estocásticas. Nestas, foram feitas comparações do número de datagramas perdidos e duplicados no protocolo original e no protocolo modificado. Como o problema ilustrado no Cenário 3 permite a formação de um *loop* (MA tunela datagrama para si mesmo) para a simulação alteramos o protocolo original a fim de evitar este problema.

Nas simulações, criou-se uma infraestrutura composta por 19 redes (células) hexagonais, e onde todos os MAs das redes possuem uma ligação com o roteador multicast. Entretanto, apenas 10 redes selecionadas aleatoriamente compõem inicialmente a árvore multicast. Foram feitas simulações com 150 e 300 hosts móveis distribuídos uniformemente entre as redes que compõem a árvore multicast inicial. Não foi utilizado o Join Option, pois as modificações feitas no MMA não abrangem o tratamento deste recurso. De fato, a utilização do Join Option ameniza os problemas do protocolo original, uma vez que permite que um MA que tenha saído da árvore (Cenário 1) possa se reintegrar à mesma.

Nas simulações usou-se quatro taxas de mobilidade: muito baixa, baixa, média-baixa e média, isto é, a probabilidade de um host móvel migrar a cada 100 Unidades de Tempo Simuladas (UTS) é de 10%, 20%, 40% e 50%, respectivamente. Em cada simulação 200 datagramas são enviados em multicast para todos os hosts móveis, sendo que a probabilidade de um datagrama ser enviado (também a cada 100 UTS) é de 50%. O timeout utilizado para a confirmação do estabelecimento de um túnel foi de 200 UTS, fazendo com que a probabilidade do pedido de estabelecimento de um túnel seja confirmado no tempo esperado seja de 75%.

Para cada conjunto de parâmetros escolhido foram feitas 30 simulações. Os resultados apresentados para o MMA original e o modificado (corrigido) são a média aritmética dos valores medidos e as barras verticais indicam o desvio padrão. Foram medidos o número total de datagramas enviados por tunelamento, o número de datagramas perdidos, (i.e. a soma do número total de datagramas que não foram recebidos por cada host móvel) e o número de datagramas repetidos enviados pelo meio cabeado e pelo meio sem fio.

O gráfico da Figura 10 mostra o número de datagramas não recebidos pelos hosts móveis. Apesar do problema ilustrado pelo Cenário 1 indicar que pode ocorrer uma perda de um grande número de datagramas, este cenário é bastante raro, especialmente devido ao fato de que o protocolo original permite que um número maior do que necessário de túneis permaneçam abertos (Cenário 2), tornando menos provável que algum MA tenha sua entrada de grupo sem nenhum membro cadastrado.

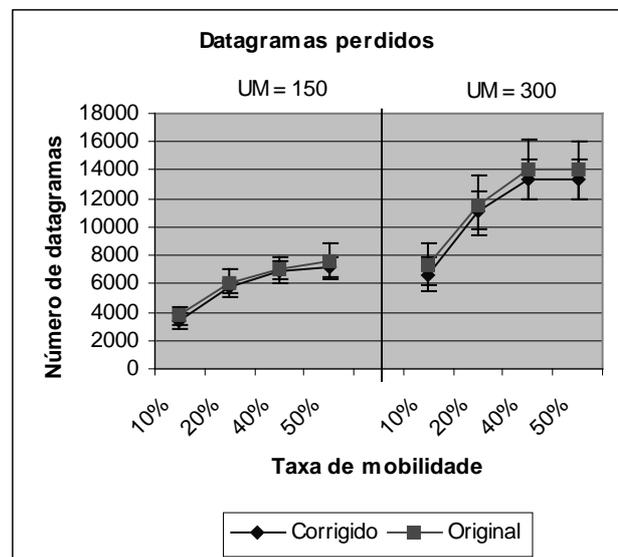


Figura 10. Número de datagramas não recebidos pelos hosts móveis em simulações feitas com 150 e 300 hosts móveis.

Nas simulações feitas, a saída de MA's da árvore multicast ocorre muito raramente e apenas com poucos hosts móveis. O fato do número de pacotes perdidos no MMA modificado ser apenas um pouco menor do que no MMA original, (apesar do problema discutido no Cenário 3) pode ser explicado da seguinte maneira, como no MMA original são deixados abertos desnecessariamente vários túneis (conforme mostrado no Cenário 2), esta redundância de envio de datagramas naturalmente reduz as chances de perda de datagramas, apesar do problema mencionado acima.

A Figura 11 mostra o gráfico do número de datagramas tunelados repetidos provenientes de diferentes túneis. Apesar da diferença entre o número de pacotes ser considerável, também neste caso o problema de duplicação de datagramas devido a múltiplos túneis (Cenário 2) pode estar sendo amenizado pelo fato de que a migração de um MH, que traz como MF um MA que mantém um túnel duplicado, pode fazer com que esse túnel seja utilizado corretamente e posteriormente seja encerrado de forma adequada.

O problema de replicação de datagramas se mostra ainda mais grave no meio sem fio, pois levando-se em consideração que cada datagrama redundante recebido por um MA resultará em vários datagramas duplicados sendo transmitidos pelo meio sem fio aos hosts registrados no MA. O gráfico da Figura 12 mostra o número de datagramas duplicados enviados pelo meio sem fio.

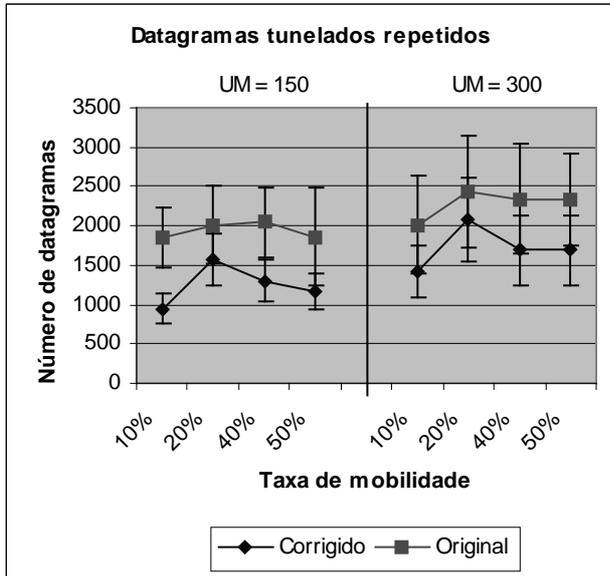


Figura 11. Número de datagramas repetidos recebidos pelos MA.

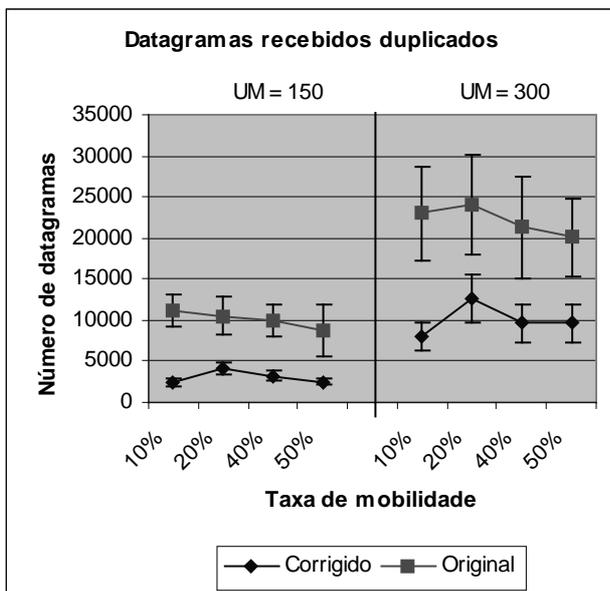


Figura 12. Número de datagramas repetidos recebidos pelos hosts móveis.

É interessante notar que os efeitos dos problemas do MMA original se amenizam mutuamente, pois o grande número de datagramas repetidos faz com que menos datagramas sejam perdidos e as situações que ocasionam a perda de datagramas permitem que menos datagramas repetidos sejam entregues. Por outro lado, isto não torna os problemas do protocolo original menos críticos, especialmente se levarmos em consideração a maior sobrecarga de comunicação (e processamento) causada por mensagens de controle trocadas entre os MAs para o

estabelecimento e encerramento de túneis (como pode ser visto no gráfico da Figura 13).

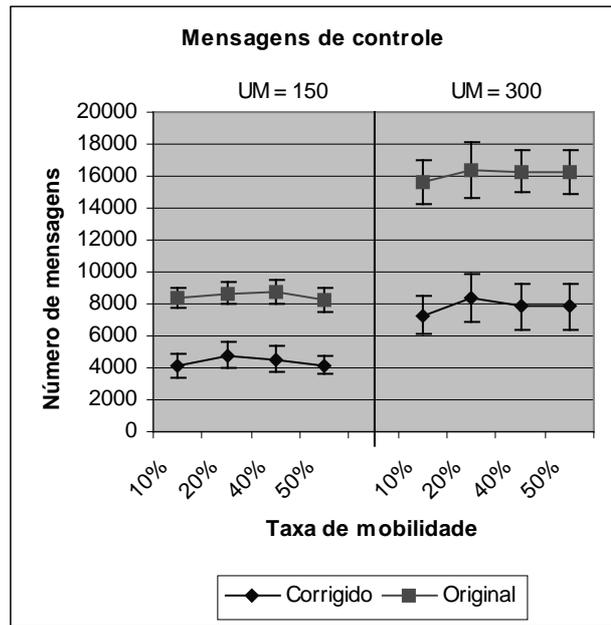


Figura 13. Número de mensagens trocadas entre os MAs para o estabelecimento e encerramento de túneis

## VI. CONCLUSÕES

O desenvolvimento de protocolos para redes móveis é uma tarefa bastante complexa, para a qual são necessárias ferramentas para a especificação, o projeto, a prototipação, a simulação, a depuração e análise de desempenho destes protocolos. Devido à grande quantidade de variáveis que influenciam o comportamento de um protocolo em uma rede móvel, simulação tem sido uma abordagem amplamente utilizada para estes tipos de rede.

Neste artigo apresentamos o protocolo MMA para multicast em redes móveis, e discutimos alguns dos problemas deste protocolo para prover multicast de forma eficiente e correta, tais como, por exemplo, a duplicação de túneis, loops e o possível não atendimento dos hosts móveis. Para estes problemas propusemos pequenas modificações no protocolo que foram validadas através de uma prototipação e simulações do protocolo. Além disto, usamos simulações para comparar o MMA original e o modificado, e conseguimos mostrar vantagens do MMA modificado com relação à eficiência do serviço de multicast.

Entretanto, pudemos comprovar o quanto ferramentas para prototipação e simulação de protocolos, tais como por exemplo o MobiCS, podem ser úteis também para estudar e analisar protocolos já existentes ou publicados (como o MMA, em um journal conceituado) e possivelmente até corrigir ou otimizá-los. Mas estamos cientes também de que devido à necessidade de escolha de um determinado modelo de simulação (com certas simplificações/suposições) naturalmente, nunca será possível analisar todos os aspectos de um protocolo com igual ênfase. No nosso caso, o objetivo do trabalho foi analisar o aspecto de gerenciamento de mobilidade do

MMA, e acreditamos que os resultados obtidos foram bastante interessantes.

Como trabalhos futuros pretendemos comparar o MMA modificado com outros protocolos para multicast, como o MoM, e o protocolo proposto em [Sadok, Cordeiro and Kelner 2000].

## VII. REFERÊNCIAS BIBLIOGRÁFICAS

- [Acharya 1996] Acharya A., Bakre A. and Badrinath B., IP multicast extensions for mobile internetworking, in: *IEEE INFOCOM'96* (1996).
- [Acharya 2000], Acharya A. and Badrinath B., A framework for delivering multicast message in networks with mobile hosts, *Mobile Networks and Applications* (2000) 199–219.
- [Bhatti 1999] Bhatti, N. and Schlichting. Configurable Communication Protocols for Mobile Computing, Proceedings of the 4<sup>th</sup> International Symposium on Autonomous Decentralized Systems, pages 220-227, Tokyo, March 1999
- [Chikarmane and Williamson 1998] Chikarmane, V. and Williamson, C.L.; Multicast support for mobile host using mobile IP: design issues and proposed architecture, *Mobile Networks and Applications* (1998) 365–379.
- [DEERING 88] DEERING, S. E. Multicast Routing in Internetworks and Extended LANs. Proceedings of ACM SIGCOMM'88 Conference, 1988, pp55-64.
- [Harrison 1997] Harrison T., Williamson C., W. Mackrell and R. Bunt, Mobile multicast (MOM) protocol: multicast support for mobile hosts, in: *Proc. Of ACM MOBICOM'97* (1997) pp. 151–160.
- [MOY 1994] MOY, J. Multicast Extensions to OSPF, Request For Comments 1584, Mar 1994, 102pp.
- [Perkins, 1997] Perkins, C. "Mobile IP," *IEEE Comm.*, Vol. 35, No. 5, 1997, pp. 84-99.
- [Rocha and Endler 2001] Rocha, Ricardo C.A. and Endler Markus. MobiCS: An Environment for Prototyping and Simulating Distributed Protocols for Mobile Networks. In Proc. 3rd IEEE Intern. Conference on Mobile and Wireless Communications Networks (MWCN2001).
- [Sadok, Cordeiro and Kelner 2000] Sadok Djamel H., Cordeiro Carlos de M. and Kelner Judith; *A Reliable Subcasting Protocol for Wireless Environments*; SBRC - Simpósio Brasileiro de Redes de Computadores (2000);
- [Suh, Shin and Kwon 2001] Suh Young-Joo, Shin Hee-Sook, and Kwon Dong-Hee, "An Efficient Multicast Routing Protocol in Wireless Mobile Networks" *ACM Wireless Networks*, vol.7, no.5, pp.443-453, Sep. 2001.
- [Xylonmenos and Polyzos 1997] Xylonmenos G. and Polyzos G., IP multicast for mobile hosts, *IEEE Communications Magazine* (January 1997) 54–58.

## SOBRE OS AUTORES

Renato Maia é Mestre em Informática pela PUC-Rio e Bacharel em Ciência da Computação pela UFPA. Atualmente é analista do laboratório de tecnologia em computação gráfica da PUC-Rio (TeCGraf). Suas áreas de pesquisa incluem Sistemas Distribuídos, Adaptação Dinâmica e Linguagens de Programação.

Vagner Sacramento é doutorando em informática pela PUC-Rio, Mestre em Sistemas e Computação pela UFRN, Bacharel em Sistemas de Informação pela Ulbra. Professor da disciplina Redes de Computadores no curso de especialização APGS na CCE/PUC-Rio. Experiência como consultor na área de redes e segurança. Áreas de pesquisa de interesse: Protocolos e Sistemas Distribuídos, Computação Móvel, Middlewares, Colaboração entre usuários Móveis, Segurança de Redes de Computadores.

Markus Endler é Bacharel em Matemática e Mestre em Informática pela PUC-Rio (1985 e 1987), e obteve os títulos de Dr.rer.nat. pela Universidade Técnica de Berlim (TU Berlin) em 1992, e de Professor livre-docente da Universidade de São Paulo em 2001. De 1994 to 2000 ele foi professor do Instituto de Matemática e Estatística da USP, e desde 2001 ele é professor em tempo integral do Departamento de Informática da PUC-Rio, onde ele coordena o Laboratory for Advanced Collaboration (LAC). Suas principais áreas de pesquisa são Sistemas e Protocolos Distribuídos, Computação Móvel e Ubíqua, Middlewares, Serviços baseados em localização e Colaboração Móvel. É membro da SBC e da ACM, e participa de comitês técnicos de vários simpósios e workshops nacionais e internacionais em Redes e Sistemas Distribuídos.