Uma Plataforma para Sistemas Distribuídos de Realidade Virtual *

Renata Cruz Teixeira
renata@gta.ufrj.br

Otto Carlos M. B. Duarte
otto@gta.ufrj.br

Grupo de Teleinformática e Automação - GTA
PEE/COPPE-EE/UFRJ
C. P. 68504 - 21945-970 - Rio de Janeiro - Brasil

Resumo

Este artigo propõe e analisa o desempenho de uma plataforma para sistemas distribuídos de realidade virtual. São apresentados a arquitetura e o protocolo propostos para esta plataforma junto com os princípios que guiaram seu desenvolvimento. Os principais objetivos da plataforma são: escalabilidade e flexibilidade. Estes objetivos foram avaliados através de simulações. Os resultados mostram que o particionamento do ambiente melhora o atraso para entrada no mundo e para trocas de grupos e diminui o número de mensagens trocadas na rede. Este particionamento permite aumentar a velocidade dos participantes e o número de participantes, garantindo uma interatividade menor que 300 ms.

Abstract

This paper describes and analyzes the performance of a platform for distributed virtual reality systems on the Internet. We present the principles that have guided the platform design, together with an architecture and a protocol for this platform. The main goals of this platform are: scalability and flexibility. We investigate these goals via simulation. The results show that the division of the virtual environment improves the delay both to enter and to exchange groups, and reduces the number of messages exchanged. This partition allows increasing both participants' speed and the number of participants, assuring an interactivity lower than 300 ms.

*Este trabalho foi realizado com recursos da CAPES, UFRJ, FUJB, CNPq, REENGE e COFECUB

1 Introdução

Os sistemas distribuídos de realidade virtual buscam a imersão de seus usuários em um ambiente tri-dimensional gerado por computador. Neste ambiente, também chamado de mundo virtual, usuários geograficamente dispersos podem interagir entre si e com o próprio ambiente de forma similar à interação no mundo real. Assim, a realidade virtual oferece novas perspectivas para a interação homem-máquina, tornando-se uma ferramenta bastante atrativa.

Porém, apesar de todo seu potencial, os sistemas de realidade virtual distribuída sofrem diversas restrições que dificultam seu funcionamento de forma eficiente. Um destes obstáculos é a rede de computadores [6, 9] que possui limitações como: banda passante, atraso e perdas de pacotes. Como os participantes podem se movimentar pelo mundo virtual, interagir com ele (criando ou modificando objetos) e interagir com outros participantes, torna-se necessário que os sistemas, para suportar este tipo de aplicação, possuam mecanismos para garantir que todos os usuários tenham uma visão consistente deste ambiente compartilhado.

Cada participante do ambiente virtual tem um estado associado, que deve ser notificado a todos os outros participantes. Esta atualização permanente é importante para a manutenção de consistência do mundo virtual. Com o objetivo de otimizar a utilização do sistema de comunicação, diminuindo o volume de dados gerado por estas atualizações, surgiu a técnica de filtragem de dados. Esta técnica explora o fato de um usuário não interagir com todos os participantes de um mundo virtual simultaneamente. Assim, pode-se delimitar o escopo de um participante e reduzir o número de mensagens de atualização trocadas. O escopo de interação de um participante descreve os limites da informação que ele deseja obter e pode ser visto como um filtro para as informações a serem recebidas.

Alguns trabalhos em realidade virtual distribuída propõem a utilização da técnica de filtragem aliada a um protocolo multidestinatário, mapeando os escopos de interação em grupos multipontos. O NPSNET [7] é um sistema de simulação para treinamento militar baseado no protocolo para simulações interativas distribuídas (*Distributed Interactive Simulation* – DIS) e no particionamento do mundo em classes. A comunicação do DIVE (*Distributed Interactive Virtual Environment*) [5] usa um protocolo multiponto confiável com reconhecimentos negativos (NACKs). O sistema MASSIVE (*Model, Architecture and System for Spacial Interaction in Virtual Environments*) [4] adota um modelo espacial para realizar o particionamento dos dados entre as estações clientes. O SPLINE (*Scalable Platform for Large Interactive Networked Environment*) [1] divide o mundo em locais, que são processados separadamente.

Este trabalho propõe uma plataforma para atender aos requisitos das aplicações de realidade virtual distribuída. A plataforma é responsável pela distribuição dos dados, mantendo a consistência do ambiente virtual. Para suportar um maior número de usuários, é feita a divisão dos participantes em grupos multipontos. Um estudo do impacto da filtragem de dados no comportamento da plataforma é apresentado em [10]. Neste artigo é realizada a análise do desempenho da plataforma para aplicações de realidade virtual distribuída com requisitos diferentes. Foram gerados cenários de simulação buscando representar alguns requisitos mais críticos para o desempenho da plataforma: o particionamento do ambiente virtual, a velocidade de deslocamento dos participantes e o número de usuários.

Este artigo está organizado da seguinte forma. A seção 2 apresenta as principais características da plataforma, a arquitetura e o protocolo propostos para esta plataforma. A seção 3 descreve a infra-estrutura e o modelo adotados para as simulações realizadas. O desempenho da plataforma em condições diversas, considerando os requisitos de aplicações diferentes é analisado na seção 4. A seção 5 apresenta algumas conclusões.

2 Plataforma Proposta

A plataforma proposta utiliza os serviços dos protocolos de transporte disponíveis na Internet, visando atender aos requisitos das aplicações de realidade virtual distribuída. O serviço oferecido pela plataforma é flexível, ou seja, a plataforma é configurada de acordo com a aplicação desejada. A aplicação é responsável por fornecer os cenários, as imagens para representação dos usuários (avatares) e algumas informações sobre a estrutura do ambiente, específicas para a aplicação. A plataforma baseia-se nestas informações para realizar a entrega dos dados de forma eficiente, garantindo a manutenção da consistência do mundo. Assim, para a aplicação a distribuição dos dados e o controle dos objetos do ambiente virtual são transparentes.

Além da manutenção da consistência e do controle de concorrência do acesso aos objetos do mundo, outro aspecto importante para as aplicações de realidade virtual distribuída é a possibilidade de interação direta entre os participantes através de múltiplas mídias, como o áudio e o vídeo. Esta interação aumenta a sensação de realismo no ambiente virtual. Por isso, a arquitetura proposta para a plataforma é modular, divida em funções para interação direta entre participantes (multimídia), controle de concorrência (interação com o mundo virtual) e funções do ambiente virtual (controle de consistência e envio de atualizações de estado dos participantes).

2.1 Projeto da Plataforma

A plataforma foi desenvolvida com dois objetivos principais: flexibilidade, podendo ser adotada por diferentes aplicações de realidade virtual distribuída, e escalabilidade, permitindo que o número de participantes cresça sem prejudicar a interatividade entre eles. Para prover escalabilidade, a plataforma divide os participantes em escopos de interação de acordo com a estrutura definida pela aplicação. Estes escopos são mapeados em grupos multipontos, que são o domínio para realização do controle de consistência.

A consistência do ambiente é mantida por uma estação especial chamada líder. A estação líder é eleita entre uma das estações participantes comuns, acumulando as funções de líder. Cada escopo de interação possui exatamente um líder. Quando um participante troca de um escopo de interação para outro, mudando conseqüentemente de grupos multipontos, diz-se que ele efetua um *handoff*. O líder é responsável pelo controle das entradas, saídas e *handoffs* no seu escopo de interação. Além disto, o líder controla o acesso aos objetos do mundo virtual.

A divisão dos participantes em grupos multipontos reduz a carga no sistema de comunicação, realizando a filtragem das mensagens de atualização trocadas. Conforme os participantes se movimentam no mundo virtual, seu escopo de interação modifica-se e, conseqüentemente, seu grupo multiponto, havendo a necessidade de *handoffs* dinâmicos. Este dinamismo é importante para que não haja interrupções na percepção do ambiente virtual com a realização de um *handoff*.

Cada participante possui uma cópia dos dados referentes a seu escopo de interação. Para que a informação dos participantes de um escopo permaneçam consistentes entre si, é necessário que as mensagens sinalizando alterações no escopo sejam transmitidas de forma confiável. Logo, seria interessante que o protocolo de transporte utilizado oferecesse um serviço multidestinatário confiável. Porém, os protocolos multidestinatários apresentam restrições à escalabilidade quando é necessária confiabilidade [8]. Sendo assim, propõe-se a separação das operações do ambiente virtual em operações críticas, que precisam de confiabilidade, e operações não críticas. Todas as comunicações com o líder para entrar no mundo, sair, trocar de grupos são consideradas críticas. Já as operações de atualização de estado dos participantes e as interações multimídias entre participantes, que possuem como maior restrição o atraso, são trocadas sem confiabilidade.

A divisão das operações causa a distinção entre três tipos de comunicação utilizados na plataforma. As mensagens não críticas devem ser recebidas por todos os participantes em um escopo com o menor atraso possível. As mensagens críticas precisam ser entregues com confiabilidade e podem ser de dois tipos: ponto-a-ponto, para entrega do banco de dados, por exemplo; ou multiponto, como no envio de um avatar para um grupo. Com a finalidade de atender a todos estes requisitos, é necessária a utilização de três protocolos de transporte diferentes em conjunto com um serviço de rede que permita a entrega de pacotes para grupos multipontos. Então, o IP Multiponto aliado ao UDP é adotado para a transmissão das mensagens não críticas. Para as mensagens críticas multidestinatárias o protocolo de transporte utilizado é o SRM (*Scalable Reliable Multicast*) [3], um protocolo de transporte confiável com reconhecimento negativo que visa a escalabilidade. O TCP é usado para a troca de mensagens confiáveis entre dois participantes.

Resumindo, a escalabilidade é alcançada através da distribuição do controle de consistência, realizado pelos líderes, e da redução da troca de mensagens com confiabilidade, através da divisão das operações.

2.2 Arquitetura

A arquitetura proposta para a plataforma visa a uniformidade entre todas as estações que a executam, inclusive as estações líderes. Assim, qualquer participante pode ser eleito líder durante uma sessão. A única distinção entre os líderes e os participantes é o fato destes primeiros realizarem, além das funções comuns, funções de controle. A arquitetura é apresentada na figura 1. Pode-se observar que as funções referentes aos líderes estão em um módulo separado, sendo acionadas apenas quando o participante torna-se líder. As funções da estação líder são totalmente transparentes ao usuário que participa e interage com o mundo virtual como qualquer outro participante conectado a uma estação comum (não-líder).

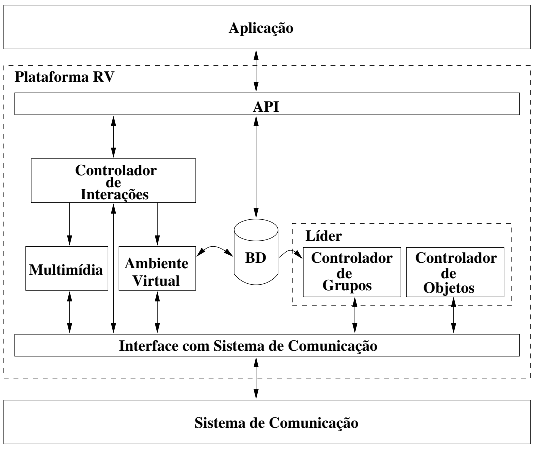


Figura 1: Arquitetura da plataforma.

A divisão do mundo virtual em escopos é definida pela aplicação. Estes escopos não variam no decorrer de uma sessão. Assim, tendo-se a posição inicial de um participante no ambiente virtual, tem-se o escopo a que ele pertence e, conseqüentemente, seu líder. Os dados do ambiente virtual estão parcialmente replicados no banco de dados (BD) de acordo com esta divisão. O BD contém, além da representação de cada objeto, sua posição e comportamento, as informações sobre o cenário do mundo virtual e sua divisão em escopos. Quando ocorre alguma modificação no usuário local, a aplicação atualiza o BD através da interface com a aplicação.

Os módulos responsáveis por manter a consistência do ambiente são: o módulo ambiente virtual e o módulo controlador de grupos do líder. No participante, o módulo ambiente virtual envia as modificações feitas na sua BD local para os outros participantes e recebe as atualização dos demais participantes. No líder, o controlador de grupos é responsável pelas adesões e desligamentos dos participantes dos grupos. O módulo ambiente virtual pode executar um algoritmo de *dead-reckoning*[1].

As interações diretas entre usuários e as interações com o mundo são gerenciadas pelo módulo controlador de interações. Este módulo é responsável pelo estabelecimento da conexão entre os participantes, que passam a interagir diretamente através do módulo multimídia. O controlador de interações também realiza, em conjunto com o módulo controlador de objetos do líder, o controle de concorrência no acesso aos objetos. Este último contém as permissões de acesso a todos os objetos do escopo.

2.3 Protocolo

Um dos objetivos da plataforma é a flexibilidade. Para alcançá-lo o protocolo definido para esta plataforma deve ser genérico, sem incorporar características específicas de nenhuma aplicação. Assim, as unidades de dados do protocolo (PDUs) especificadas podem ser utilizadas por qualquer tipo de aplicação de realidade virtual distribuída.

Neste artigo estão apresentadas apenas as PDUs necessárias para realizar o controle de consistência. Este controle engloba as funções determinantes para a escalabilidade do sistema – entrada e saída do mundo, atualizações de estado e *handoffs*. Por isto, a tabela 1 mostra as PDUs especificadas para realizar o controle de consistência na plataforma.

Todo participante do mundo virtual é membro de um grupo multiponto. Este grupo é determinado de acordo com a sua posição no mundo virtual. Os

[1] O *dead-reckoning* é uma técnica de predição de movimentos utilizada para calcular a posição dos outros participantes baseada nos movimentos anteriores. Assim, um participante não precisa enviar uma mensagem para cada novo movimento.

Tabela 1: Unidades de dados do protocolo.

PDU	*Função*
JR	Pedido de adesão a um grupo
JI	Indicação de que um novo avatar aderiu ao grupo
JC	Confirmação de um pedido de adesão, envio do BD
LR	Desligamento do grupo
FNL	Indicação de que um participante deve iniciar um *handoff*
ES	Atualização de estado de um avatar

líderes participam de um grupo multiponto de controle, com endereço conhecido. Assim, para começar uma sessão, i.e., para entrar no mundo virtual, os participantes devem enviar uma mensagem multiponto com uma PDU JR para este endereço pedindo para aderir ao grupo. A PDU JR contém a posição inicial e as informações do novo avatar. Ao receber uma PDU JR, os líderes verificam se a posição requisitada está dentro do seu escopo de interação. O líder responsável pela posição responde ao pedido enviando uma PDU JC com o BD contendo o estado atual do escopo. Após a entrega do BD, o líder envia uma mensagem multiponto com a PDU JI para todos os participantes de seu escopo, contendo as informações do novo avatar.

Após a fase de adesão, os participantes trocam atualizações de estado através do envio de mensagens multipontos com a PDU ES para o seu escopo de interação. O líder, assim como todos os participantes de seu escopo, recebe as atualizações de estado de todos os outros membros de seu escopo. Quando o líder detecta que um participante está cruzando os limites de seu escopo, ele envia uma PDU FNL para este participante, sinalizando que um *handoff* deve ser iniciado. Assim, o participante envia uma mensagem multiponto com uma PDU JR e reinicia a fase de adesão. Após receber uma PDU JC de seu novo líder, o participante desliga-se de seu grupo antigo através do envio de uma PDU LR para este grupo. É importante ressaltar que os *handoffs* são controlados pelos módulos da plataforma sem o conhecimento da aplicação. Para terminar uma sessão, um participante envia uma mensagem multiponto com uma PDU LR para seu escopo, pedindo a remoção de seu avatar do ambiente.

3 Modelo de Simulação

A fim de avaliar o desempenho da plataforma para aplicações com diferentes características foram realizadas simulações em uma topologia similar a estrutura da Internet. Estas simulações compreenderam as operações críticas e as atualizações de estado.

A ferramenta escolhida para simular a plataforma foi o *ns* (*Network Simulator*) [2], que engloba vários modelos de protocolos usuais na Internet e outros protocolos mais novos que poderão ser usados na Internet. Os protocolos de transporte adotados foram o UDP, o TCP e o SRM.

A topologia para a simulação foi gerada através do gerador de topologias GT-ITM (*Georgia Tech Internetwork Topology Models*) [11]. Assim, obteve-se um formato mais próximo do encontrado atualmente na Internet, ou seja, sub-redes interconectadas por núcleos, que representam os *backbones*. Para a simulação foi usada uma topologia com quarenta nós – oito no núcleo e mais oito sub-redes com quatro nós. As latências nos enlaces variam entre 0.3 e 3.8 ms e a banda passante é 2 Mbps. Nas simulações, a plataforma só é executada nos nós das sub-redes. Logo, o número máximo de estações simultâneas participando do mundo virtual para esta topologia é 32.

Nas simulações, o ambiente virtual é divido em áreas de tamanho fixo. O espaço do mundo virtual, ou seja, o número de posições possíveis, é mantido fixo, enquanto o número de grupos varia. Logo, com o aumento do número de grupos, a área de cada grupo diminui. Esta divisão é feita entre todos os líderes, de forma que não haja interseção entre os escopos dos diferentes líderes e que a união de todos os escopos resulte na totalidade do mundo virtual. A escolha dos líderes é feita aleatoriamente[2] entre todos os possíveis participantes. A posição inicial dos participantes também é escolhida aleatoriamente entre todas as posições possíveis no ambiente.

A movimentação dos participantes através do mundo está associada a uma probabilidade de movimentação (P_m). Assim, com probabilidade P_m um participante movimenta-se uma posição para frente, ou seja, incrementa em uma unidade sua posição. Esta probabilidade está relacionada a velocidade do participante, logo, participantes mais velozes têm maior probabilidade de movimento. Neste artigo, assume-se uma aplicação em que todos os participantes têm a mesma probabilidade de movimento.

As simulações foram realizadas baseadas neste modelo utilizando a topologia descrita. Os parâmetros de simulação utilizados para obter os resultados são apresentados na tabela 2. O tempo de simulação de cada sessão utilizando a plataforma foi de cinco minutos.

[2]Todos os valores aleatórios foram gerados utilizando as variáveis aleatórias disponíveis no *ns*.

Tabela 2: Parâmetros utilizados na simulação.

Parâmetros	*Valores*
Tamanho inicial BD	10 k octetos
Tamanho avatar	1,5 k octetos
Tamanho atualizações	100 octetos
P_m	0,2

4 Desempenho da Plataforma

A plataforma tem como principais objetivos a escalabilidade e a flexibilidade. Por isso, a avaliação do seu desempenho foi realizada observando estes dois aspectos. As simulações foram realizadas para verificar o impacto do particionamento em escopos, da velocidade dos participantes e do número de participantes no funcionamento da plataforma.

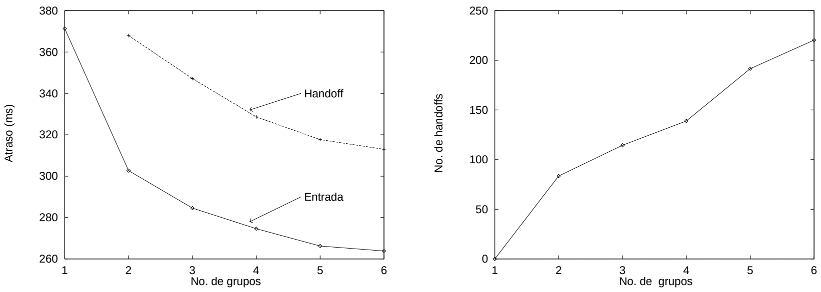
4.1 Particionamento do Ambiente Virtual

O particionamento do ambiente virtual em escopos deve ser realizado de acordo com as características especificadas pela aplicação. Para verificar o impacto deste particionamento na plataforma foram realizadas simulações variando-se o número de grupos.

Quando o número de grupos é igual a um, tem-se apenas um líder para todos os participantes. Neste caso, a plataforma atua como um sistema centralizado e não há filtragem de mensagens entre os participantes do mundo. Como o mundo não é particionado, não há trocas de grupo e, conseqüentemente, também não há interrupções na percepção do mundo. Os participantes sofrem apenas o atraso para entrar no mundo virtual.

Por outro lado, quando o número de grupos é maior do que um, o controle de consistência passa a ser distribuído entre os líderes. Quanto maior o número de grupos, menor o número de participantes a ser controlado por cada líder, maior o nível de distribuição do controle entre estes líderes e menor é o número de mensagens de atualização trocadas em cada grupo. Porém, número de interrupções na percepção do ambiente cresce, devido às trocas de grupo (*handoffs*) mais freqüentes.

A figura 2 mostra o atraso para *handoffs* e entradas no mundo e o número total de *handoffs* em uma sessão. As simulações foram realizadas para 32 participantes, utilizando os parâmetros definidos na seção 3. A figura 2(a) mostra o atraso médio para a entrada de participantes no mundo virtual e para *handoffs*. É importante ressaltar que não existe valor de atraso de *handoffs*



(a) Atraso de entrada e de *handoff*.   (b) Número de *handoffs*.

Figura 2: Impacto do particionamento em escopos.

quando o valor do número de grupos é igual a um. A diferença entre o atraso de *handoff* e o de entrada reflete o atraso necessário para o líder sinalizar ao participante que ele deve sair do grupo e este participante iniciar o processo para entrar no novo grupo.

Ao aumentar o número de grupos observa-se uma queda tanto no atraso para entrada no mundo quanto no atraso de *handoffs*. Esta melhora do atraso é causada pela repartição entre os líderes da responsabilidade de enviar a base dados para os participantes. Com a menor quantidade de participantes em cada escopo, os líderes ficam menos sobrecarregados e podem responder mais rapidamente aos pedidos de adesão. O principal componente destes atrasos é a entrega do banco de dados pelo protocolo TCP, causado pelo atraso de transmissão.

A figura 2(b) apresenta o número total de *handoffs* realizados por todos os participantes durante o período de simulação. Pode-se observar no gráfico que com o aumento do número de grupos, aumenta a freqüência com que as trocas de grupo são realizadas. Este efeito pode ser facilmente explicado pelo fato do aumento do número de grupos corresponder a uma menor área para cada grupo. Assim, para a mesma probabilidade de movimento, isto significa um maior número de *handoffs*.

A figura 3 apresenta o número total de mensagens trocadas. Nota-se que o número total de mensagens trocadas, mensagens de controle mais mensagens de atualização (ES), reduz com o aumento do número de grupos. Este aumento do número de grupos significa escopos de interação menores, logo, as mensagens de atualização são enviadas para um grupo menor de participantes, otimizando o uso dos recursos do sistema de comunicação. Paralelamente, a freqüência de *handoffs* aumenta, aumentando o número de mensagens de controle trocadas para realizar estes *handoffs*. Porém, a redução nas mensagens de atualização é mais significativa que o aumento na troca de mensagens de controle.

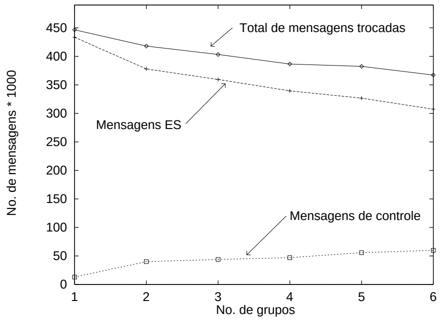


Figura 3: Número de mensagens trocadas.

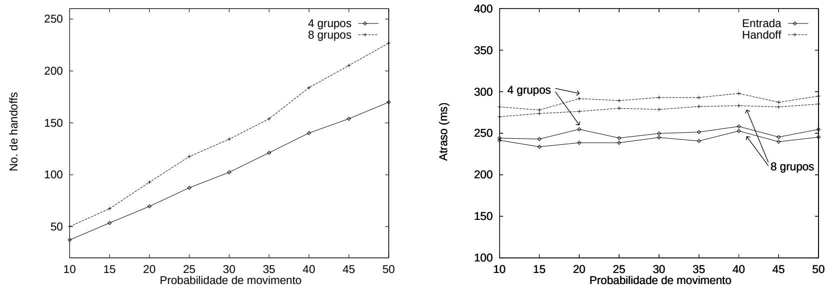
4.2 Velocidade dos Participantes

Para verificar o impacto da velocidade dos avatares no funcionamento da plataforma, foram realizadas simulações variando-se a probabilidade de movimento (P_m), parâmetro que determina a velocidade dos participantes. Os outros parâmetros foram mantidos como apresentados na seção 3 e o número de usuários simulados foi 16. As simulações foram realizadas tanto para o número de grupos igual a 4, quanto para o número de grupos igual a 8.

A figura 4(a) apresenta o número de *handoffs* realizados por todos os participantes durante toda uma sessão. Pode-se reparar que o número de *handoffs* cresce linearmente com a probabilidade de movimento. Com o aumento da velocidade dos avatares, os participantes trocam mais de grupos, realizando mais *handoffs* por sessão. O número de *handoffs* é maior para as simulações com o mundo virtual divido em 8 grupos do que as com 4 grupos. Isto ocorre devido a redução da área de cada grupo, como discutido na seção 3.

No entanto, o aumento no número de trocas de grupos, gerado por probabilidades de movimento maiores, não tem grande influência no atraso para os participantes entrarem no mundo virtual e nem para realizaram *handoffs*. Isto pode ser verificado na figura 4(b). Pode-se ressaltar que os atrasos para a simulação com 8 grupos são menores do que os da simulação com 4 grupos. Assim, as interrupções causadas pelos *handoffs* são mais curtas quando o ambiente esta mais particionado.

A figura 5 verifica o impacto da variação da velocidade dos participantes na carga da rede, ou seja, no número de mensagens trocadas. Com o aumento

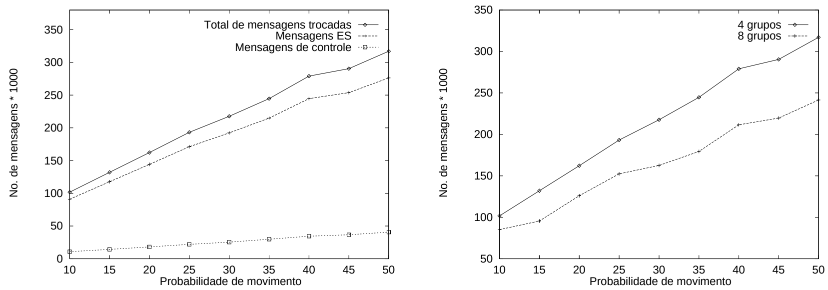


(a) Número de *handoffs*.

(b) Atraso de entrada e de *handoff*.

Figura 4: Variação da velocidade dos participantes.

no número de *handoffs*, aumenta também o número de mensagens de controle trocadas. Além disto, como os participantes movem-se com maior velocidade, o número de mensagens de atualização (PDU ES) enviadas também aumenta. Na figura 5(a) nota-se que as mensagens de atualização são o principal componente do número total de mensagens. Por isto, a utilização de um algoritmo de dead-reckoning poderia reduzir substancialmente a carga na rede. A divisão em grupos também é importante para esta redução, como pode-se reparar na figura 5(b), que compara o número total de mensagens trocadas com 4 e 8 grupos multipontos.



(a) Número de mensagens para 4 grupos.

(b) Comparação entre 4 e 8 grupos.

Figura 5: Impacto da variação de velocidade na troca de mensagens.

4.3 Número de Participantes

A plataforma deve suportar aplicações com números diferentes de participantes. Para verificar como a variação no número de participantes reflete no desempenho da plataforma foram realizadas simulações para 4 e 8 grupos. Os parâmetros utilizados são os apresentados na seção 3.

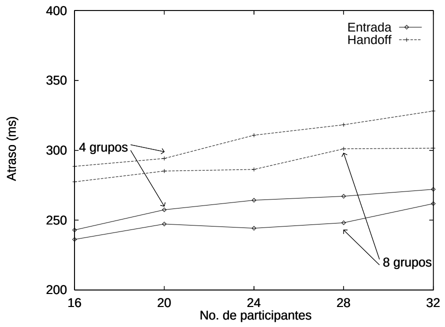
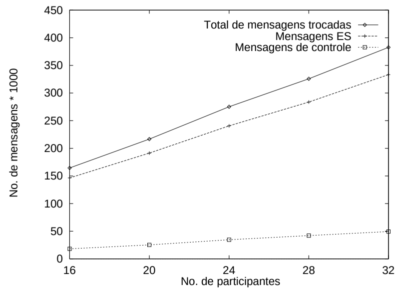
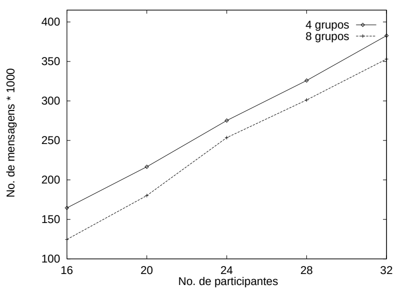


Figura 6: Atraso de entrada e de *handoffs*.



(a) Número de mensagens para 4 grupos.



(b) Comparação entre 4 e 8 grupos.

Figura 7: Número de mensagens trocadas.

A figura 6 mostra a variação do atraso quando o número de participantes cresce. Tanto o atraso para entrar no mundo quanto o atraso para realização de um *handoff* crescem com o aumento do número de usuários. Este aumento acorre porque o número de líderes foi mantido, assim, quando o número de usuários aumenta, o número por líder também aumenta. Os líderes ficam mais sobrecarregados, demorando mais a responder aos pedidos de adesão e a detectar os *handoffs*. Porém, nota-se no gráfico que o crescimento do atraso desacelera com o aumento do número de usuários.

Quanto maior o número de usuários maior o número de mensagens devem ser trocadas tanto para notificar o estado destes participantes quanto para controlar adesões e desligamentos de escopos. Este crescimento no número de mensagens é mostrado na figura 7(a). A figura 7(b) apresenta uma comparação entre o número de mensagens trocadas para a simulação com os participantes divididos em 4 e em 8 grupos, onde a simulação com 8 grupos sobrecarrega menos o sistema de comunicação.

5 Conclusões

Os resultados de simulação da plataforma proposta mostram que o particionamento do ambiente virtual provoca uma diminuição no atraso para as operações de entrada no mundo e troca de grupos. Esta diminuição beneficia a interatividade entre os participantes e reduz a possibilidade de inconsistências. Há um compromisso entre este ganho em interatividade e as interrupções geradas pelos *handoffs* quando o número de grupos aumenta. Porém, para todos os valores de número de grupos simulados, as interrupções não foram significativas, representando menos de 1% do tempo total da sessão de um participante. Assim, a maior carga na rede devido ao aumento no número de participantes pode ser compensada em aplicações que permitam um ambiente mais particionado.

Nas aplicações em que os avatares movimentam-se rapidamente, o número de interrupções devido a *handoffs* e a carga na rede são maiores. Para diminuir a carga na rede é interessante particionar mais o ambiente. Considerando o pior caso, quando P_m igual a 0.5, a redução do número de mensagens quando aumenta-se o número de grupos de 4 para 8 foi de 23.19%. Porém o particionamento aumenta o número de interrupções na percepção do ambiente virtual. Para P_m igual a 0.5, as interrupções representaram 2.69% da sessão de um participante para a simulação com 8 grupos. Enquanto na simulação com 4 grupos, as interrupções representam 1.37%. Novamente, o tempo gasto em interrupções devido aos *handoffs* não foi significativo. Este compromisso entre o número de mensagens trocadas e as interrupções no mundo virtual deve ser considerado de acordo com a aplicação.

Pode-se concluir que a plataforma proposta, com a adoção de um modelo distribuído entre líderes, e uma divisão do ambiente em vários escopos separados, permitiu uma boa interatividade sem prejudicar a continuidade da aplicação.

Referências

[1] Barrus, J. W., Waters, R. C., e Anderson, D. B. Locales: Supporting Large Multiuser Virtual Environments. *IEEE Computer Graphics and Applications 16*, 6 (novembro de 1996), 50–57.

[2] Fall, K., e Varadhan, K. NS Notes and Documentation. Relatório técnico, The VINT Project, novembro de 1997.

[3] Floyd, S., Jacobson, V., Liu, C.-G., McCanne, S., e Zhang, L. A Reliable Multicast Framework for Light-weight Sessions and Application Level Framing. *IEEE/ACM Transactions on Networking 5*, 6 (dezembro de 1997), 784 – 803.

[4] Greenhalgh, C., e Benford, S. MASSIVE: A Distributed Virtual Reality System Incorporating Spatial Trading. In *Proceedings IEEE DCS'95* (Vancouver, Canada, 1995).

[5] Hagsand, O. Interactive Multiuser VEs in the DIVE System. *IEEE Multimedia 3*, 1 (1996), 30–39.

[6] Macedonia, M. R., e Zyda, M. J. A Taxonomy for Networked Virtual Environment. *IEEE Multimedia 4*, 1 (1997).

[7] Macedonia, M. R., Zyda, M. J., Pratt, D. R., Brutzman, D. P., e Barham, P. T. Exploiting Reality with Multicast Groups. *IEEE Computer Graphics and Applications* (setembro de 1995), 38 – 50.

[8] Pingali, S., Towsley, D., e Kurose, J. A Comparison of Sender-Initiated and Receiver-Iniciated Reliable Multicast Protocols. *IEEE Journal on Selected Areas in Communications 15*, 3 (abril de 1997), 398 – 406.

[9] Teixeira, R. C., e Duarte, O. C. M. B. Requisitos de Ambientes Virtuais Distribuídos de Larga Escala. In *I Workshop de Realidade Virtual* (São Carlos, novembro de 1997), pp. 26 – 35.

[10] Teixeira, R. C., e Duarte, O. C. M. B. Multicast Communication for Scalable Distributed Virtual Environments. Relatório Técnico GTA-99-11, COPPE/UFRJ - Grupo de Teleinformática e Automação, março de 1999.

[11] Zegura, E. W., Calvert, K., e Bhattacharjee, S. A Quantitative Comparision of Graph-Based Models for Internet Topology. *IEEE/ACM Transactions on Networking 5*, 6 (dezembro de 1997), 770 – 783.