

# Experimento de Controle Colaborativo em Ambiente Virtual Tridimensional

Éder Torres Patrício<sup>1</sup>, Manoel Ribeiro Filho<sup>1</sup>, Marcos Paulo Alves de Sousa<sup>1,2</sup>, Orlando Fonseca Silva<sup>1</sup>

<sup>1</sup>Departamento de Engenharia Elétrica e de Computação – Universidade Federal do Pará (UFPA)  
Caixa postal 479 – 66.075-110– Belém – PA – Brazil

<sup>2</sup>Centro Universitário do Pará (CESUPA)- Área de Ciências Exatas e Tecnologia.  
Av. Gov. José Malcher, 1963 - CEP 66.060-230 -Belém - PA - Brasil  
eder@ufpa.br, mrf@ufpa.br, marcosp@ufpa.br, orfosi@ufpa.br

**Abstract:** *The aim of this paper is to present the development of a virtual collaborative laboratory to be used as a tool for education of undergraduate engineering students in the area of control systems. A three-dimensional environment representing a "Ball-Beam" dynamic system is developed using the API (Application Programming Interface) Java™ 3D. This system offers a multiple users simulation environment, where an instructor represented by an avatar coordinates the students' actions in the process of controlling the "Ball-Beam" dynamic system. The students either can execute the program using the already known parameters of the plant or they can change their value to simulate different operational conditions of the system.*

**Resumo:** *Este trabalho tem como objetivo o desenvolvimento de um laboratório virtual colaborativo como ferramenta de auxílio no ensino de disciplinas de sistemas de controle dos cursos de engenharia. Foi construído um ambiente tridimensional que representa um sistema dinâmico "Ball-Beam" utilizando-se a API Java™ 3D. O sistema oferece um ambiente de simulação compartilhada, onde um professor representado por um avatar no ambiente tridimensional dirige a experiência realizada por alunos também representados por avatares na cena 3D. Os alunos podem executar a simulação utilizando parâmetros da planta e ganhos de realimentação já calculados, porém, intermediados pelo professor, podem alterar estes valores e então carregá-los no programa, dessa forma, podem projetar um novo controlador e observar o comportamento do sistema em tempo real.*

## 1. Introdução

O Experimento de Controle Colaborativo em Ambiente Virtual Tridimensional tem por objetivo explorar os recursos da interface *não-imersiva* de realidade virtual (Burdea et al, 1994), oferecendo um ambiente de aprendizagem colaborativo (Perucia et al, 2002) entre alunos e professores os quais realizam simulações de um sistema de controle tipo "ball-beam". A escolha do sistema dinâmico não-linear "ball-beam" deu-se porque trata-se de um dos problemas clássicos da engenharia de controle (Ogata, 1996). Este sistema tem sido usado para testar técnicas de controle (Faria et al, 1997), (Hauser et al, 1992) e construção de protótipos didáticos (Rothe et al, 1999) para serem usados em laboratórios de ensino e pesquisa.

Nesse ambiente colaborativo, todo o processo de aprendizado e simulação é intermediado pelo professor e compartilhado pelos alunos em tempo real. O professor ou instrutor tem a função de orientar os alunos quanto à configuração adequada dos valores de cada

elemento do sistema “*ball-beam*” e instruí-los e organiza-los quanto à simulação do ambiente virtual representado.

Os objetivos do ambiente colaborativo são:

- Oferecer um ambiente interativo que complemente o processo de aprendizagem teórica aliada a prática de experiências realizadas em laboratório.
- Promover maior motivação nos alunos quanto a prática laboratorial.
- Facilitar a disseminação do conhecimento do professor aos alunos compartilhando experiências de laboratório.
- Oferecer um ambiente de simulação de controle compartilhada.
- Prover um ambiente colaborativo que permita a cada aluno expressar seu conhecimento adquirido por meio de simulações e diálogos (*chat*).
- Professores e estudantes representados por seus avatares podem navegar pelos elementos tridimensionais do mundo virtual.

A seguir será apresentado o projeto e a implementação de um laboratório virtual (Lopes et al, 2002) de um sistema dinâmico de controle tipo “*ball-beam*”, em seguida o projeto e implementação do ambiente colaborativo, os resultados encontrados e as conclusões.

## **2. Laboratório Virtual de um Sistema de Controle tipo Ball-Beam**

Um Laboratório Virtual de um Sistema de controle tipo “*ball-beam*” monousuário, cujos detalhes de projeto e implementação são encontrados em (Patrício et al 2004) serão aqui resumidos.

A estratégia de controle empregada foi por alocação de pólos via realimentação de estados. Dessa forma foi implementado, em Linguagem Java, um simulador que resolve as equações diferenciais do modelo não linear considerando um atuador e sensores ideais e o controlador como sendo contínuo. Os resultados deste simulador se mostraram satisfatórios para os ganhos de realimentação de estados calculados.

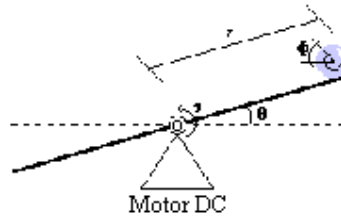
No ambiente virtual os usuários podem observar o processo de controle de um sistema com ganhos previamente calculados, mas também podem alterar parâmetros da planta, desenvolver os cálculos de seus próprios ganhos de realimentação de estados e então incorporá-los diretamente ao sistema, isto é, no mundo tridimensional que representa o sistema “*ball-beam*”. Nele, os usuários poderão obter resposta visual imediata a qualquer alteração do projeto, proporcionando assim um melhor entendimento do problema.

Usando técnicas de modelagem 3D construiu-se uma sala onde se colocou uma mesa e sobre esta se encontra o sistema “*ball-beam*”, como mostra a Figura 11. Dessa forma o usuário poderá navegar pela sala, como se estivesse em um laboratório real. Através de botões e campos de texto poderá interagir com o sistema, ordenando o início do processo de controle com valores padrões ou poderá alterá-los, além disso, pode-se visualizar a plotagem de gráficos das variáveis de estado e ainda rotacionar o ambiente tridimensional, obtendo assim diversos ângulos de visualizações.

### **2.1. Modelamento do sistema “Ball-Beam”**

O modelamento do sistema “*Ball-Beam*” requer uma análise física e matemática do sistema, então, seja a Figura 1 representativa do sistema “*ball-beam*”, considera-se que a barra pode girar

livremente em torno de seu centro quando submetida a um torque e a bola, por sua vez, é livre para rolar ao longo da barra, sem deslizamento (Faria et al, 1997).



**Figura 1: Sistema “Ball-Beam”**

Definindo-se:

$r$  : deslocamento linear da bola [m]

$\theta$  : deslocamento angular da barra [rad]

$\tau$  : torque de entrada [N.m]

$\Phi$  : ângulo de rotação da bola [rad/s]

$\dot{\theta}$  : velocidade angular da barra [rad/s]

O modelo pode ser obtido a partir da equação de Lagrange (Houpis, 1981), e resulta nas equações (1) e equações (2).

$$0 = \left( \frac{J_b}{R_b^2} + m_b \right) \ddot{r} + m_b g \sin \theta - m_b r \dot{\theta}^2 \quad (1)$$

$$\tau = (m_b r^2 + J + J_b) \ddot{\theta} + 2m_b r \dot{r} \dot{\theta} + m_b g r \cos \theta \quad (2)$$

Estas caracterizam o sistema como não linear. Assim uma das técnicas para linearizá-lo é o uso da matriz jacobiana (Ogata, 1990).

A estratégia de realimentação de estados consiste em determinar ganhos de realimentação de estados tal que os pólos do sistema em malha fechada assumam valores previamente especificados. Naturalmente, a escolha dos pólos de malha fechada desejada deve ser feita de tal modo que especificações como sobre sinal, tempo de estabilização etc... sejam satisfeitas.

Para verificação do desempenho do sistema em malha fechada, utilizou-se o simulador do Matlab™ (Matlab, 2000) denominado Simulink para implementar as equações dinâmicas não lineares do sistema em conjunto com os valores obtidos para os ganhos de realimentação de estados (Faria et al, 1997). Os resultados encontrados foram satisfatórios, assim, foi elaborado um programa em Java, que realizasse o mesmo papel desse simulador, isto é, que resolvesse as equações do modelo não linear considerando um atuador ideal e o controlador como sendo contínuo.

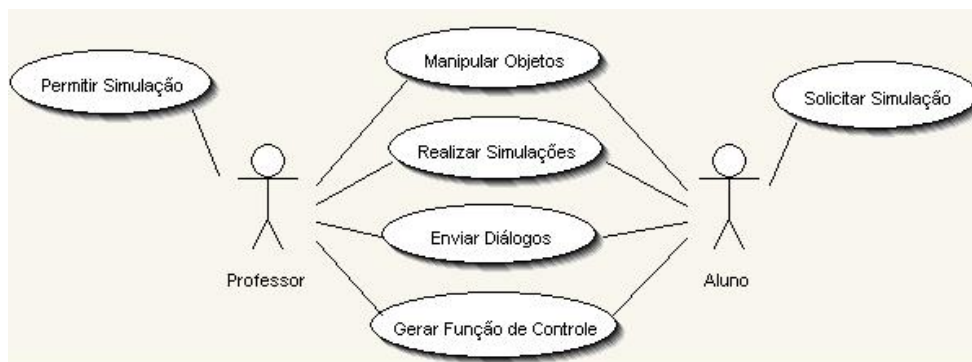
Os valores de K foram definidos como os ganhos de realimentação de estados padrões, isto é, quando um estudante carregar o software da animação 3D do sistema “Ball-Beam” tais valores serão passados ao simulador que realizará o controle.

### 3. Projeto do Sistema Colaborativo

O sistema é multi-usuário e colaborativo (Vidal et al, 2003) permitindo aos usuários realizarem simulações no sistema de controle *ball-beam*, onde toda simulação é compartilhada pelos demais usuários.

A Figura 2 exibe o diagrama de Caso de Uso apresentando os papéis desempenhados pelo atores do sistema colaborativo. Observam-se os atores Professor e aluno assumindo papéis semelhantes no ambiente virtual, como *Visualizar Ambiente Virtual*, *Manipular Objetos*

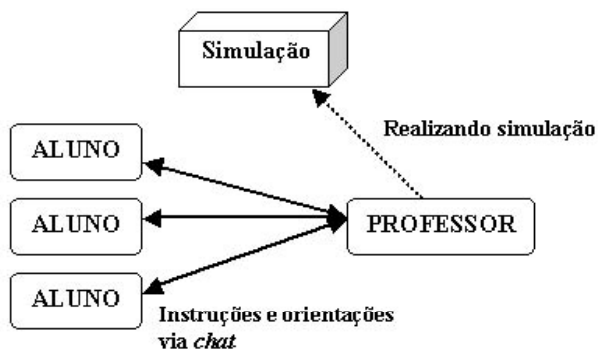
*Tridimensionais, Realizar Simulações, Enviar Diálogos, Gerar Funções do Sistema de Controle* para análise dos resultados. Além desses papéis, o professor e o aluno possuem seus papéis particulares quanto a permissão de alunos na simulação do ambiente virtual e quanto a solicitação ao professor para simulação.



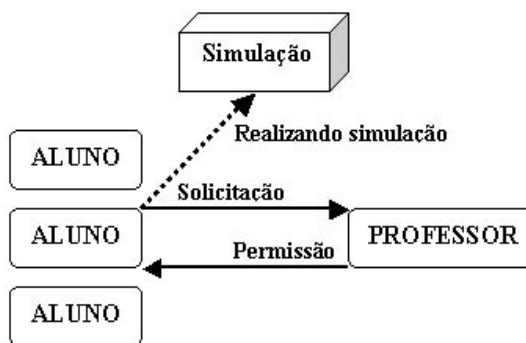
**Figura 2: Diagrama de Caso de Uso do sistema colaborativo**

### 3.1 Solicitação e Permissão de Simulação

Qualquer simulação no sistema é organizada e controlada pelo professor, e apenas um aluno de cada vez poderá realizar as simulações conforme a ordem de solicitação, os demais alunos permanecem desabilitados enquanto estiver ocorrendo a simulação. Na Figura 3 é mostrada a troca de instruções e orientações entre o professor e o aluno via *chat*, e para mostrar o funcionamento “na prática” do sistema de controle, o professor realiza simulações e aguarda a manifestação dos alunos para esclarecer dúvidas e também realizar simulações. Conforme a Figura 4, o professor tem o poder de permitir qual aluno poderá realizar a simulação no ambiente.



**Figura 3: Professor instruindo os alunos e simulando o sistema de controle.**



**Figura 4: O aluno solicitando a permissão de simular no ambiente.**

As Figuras 5 e 6 mostram as interfaces utilizadas no sistema para executar o processo de permissão e solicitação entre o professor e o aluno. Na Figura 5, o professor gerencia as simulações verificando a listagem dos alunos ordenada conforme a ordem de solicitação, e também os status da simulação do aluno é exibido, e informações importantes como o tempo de acesso do aluno dentro programa e o tempo de simulação realizada por um aluno especificado, e, além disso, o professor poderá retirar e desconectar o aluno do sistema caso este esteja impedindo o processo de aprendizagem dos demais usuários. A Figura 6 mostra como o aluno deverá solicitar a simulação ao professor e outras informações como o status do pedido e a ordem de simulação dos demais usuários.

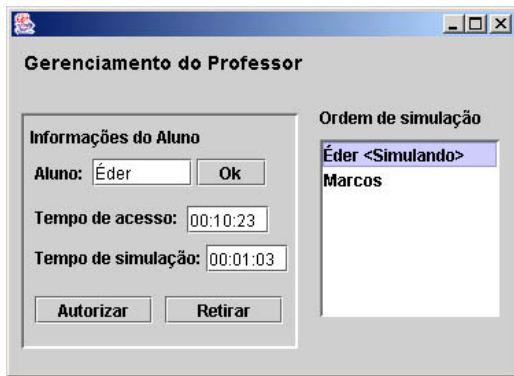


Figura 5: Gerenciamento do professor.

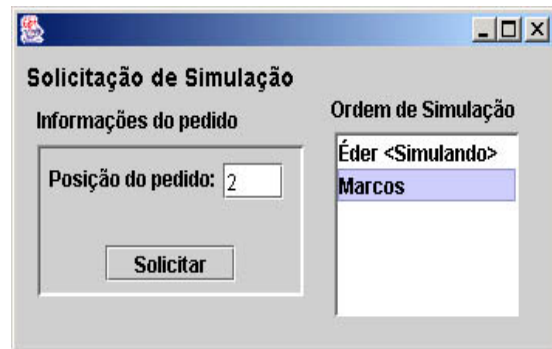


Figura 6: Solicitação de simulação do aluno.

### 3.2 Arquitetura multi-usuário de comunicação do sistema colaborativo

A arquitetura de comunicação do sistema é baseada no modelo *Cliente/Servidor*, veja a Figura 7. A troca de mensagens entre o módulo usuário e o servidor ocorre via *Socket* da API Java (Sun, 2004) sendo o servidor uma aplicação que comporta *threads* para receber múltiplas conexões de usuários remotos. O módulo Usuário é responsável por renderizar o ambiente virtual usando as bibliotecas do Java 3D e transmitir as informações dos estados dos objetos do ambiente virtual para o servidor.

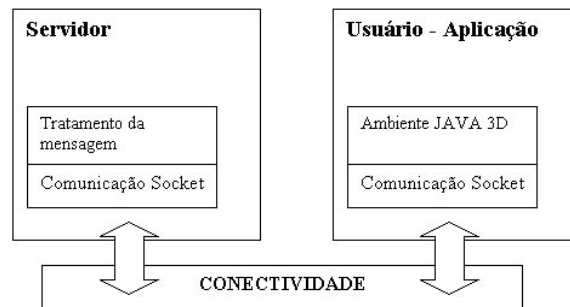


Figura 7: Arquitetura de comunicação

O servidor tem como função realizar o tratamento de todas mensagens que são transmitidas na rede, mantendo a consistência e a replicação dos dados entre os ambientes das máquinas dos usuários utilizando o protocolo de comunicação mostrado na Figura 8, onde as mensagens são enviadas junto à camada *TCP*, e é formada pelos campos *ID*, que se refere a um identificador de tipo operação do sistema, e o *CONTEÚDO* possui os dados necessários para a execução de cada tipo de operação.

A Figura 9, refere-se ao conteúdo da mensagem de operação de *login* do usuário, tendo como dado *LOG* no campo de identificador de operação, e no outro campo, o nome do usuário. Já a Figura 10 mostra o conteúdo da mensagem de operação de atualização de objetos, identificado pelo dado *ATL*, e no conteúdo são enviadas as informações dos objetos que sofreram atualização representada pelo *ID\_OBJ*, e seus respectivos dados atualizados formado pelo *PS\_OBJ*.

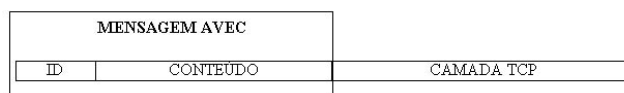


Figura 8: Protocolo de comunicação

MENSAGEM AVEC	
ID	CONTEÚDO
LOG	NOME_USUÁRIO

Figura 9: Operação de login

MENSAGEM AVEC		
ID	CONTEÚDO	
ATL	ID_OBJ	PS_OBJ

Figura 10: Operação de atualização dos objetos tridimensionais

### 3.2 Implementação Computacional

Todo o sistema foi implementado em linguagem Java (Sun, 2004) que é uma plataforma de programação livre e traz consigo a característica da portabilidade, isto é, Java usa a idéia “escreva uma vez, execute em qualquer lugar” o que é um recurso importante para qualquer aplicação. O ambiente 3D foi implementado utilizando a API Java 3D (Selman, 2002), (Bouvier, 2002) que se apresenta bastante difundida no âmbito do desenvolvimento de Sistemas de Realidade Virtual. A interface gráfica foi desenvolvida usando o pacote swing presente nas GUI (Graphical User Interface) do Java. Sendo que a parte de navegação do usuário através do ambiente tridimensional foi feita usando a classe KeyNavigator da API Java 3D. Os usuários podem observar os gráficos das variáveis de estado, para isso foi incorporado ao projeto a API gráfica Java JFreeChart (JfreeChart, 2004) que é responsável por todo o processo de plotagem.

### 4. Resultados

Quando o módulo do servidor estiver ativo, os usuários da aplicação poderão se conectar e visualizar o ambiente virtual e os valores da configuração do sistema de controle compartilhado. A Figura 11, mostra o ambiente do professor compartilhado por dois outros alunos. Neste ambiente o professor visualiza os avatares representado os alunos. Na Figura 12, o aluno *Marcos* visualiza o professor *Manoel* e o outro aluno *Éder*.

O professor realiza a aula virtual através de troca de diálogos com os alunos via *chat*, e também através de simulações no sistema de controle *Ball-Beam*. Os alunos podem trocar diálogos entre os participantes, e enquanto o professor estiver realizando a simulação os demais alunos serão apenas expectadores, e somente quando o professor autorizar e permitir, diante a solicitação do aluno, é que este terá permissão para simular.

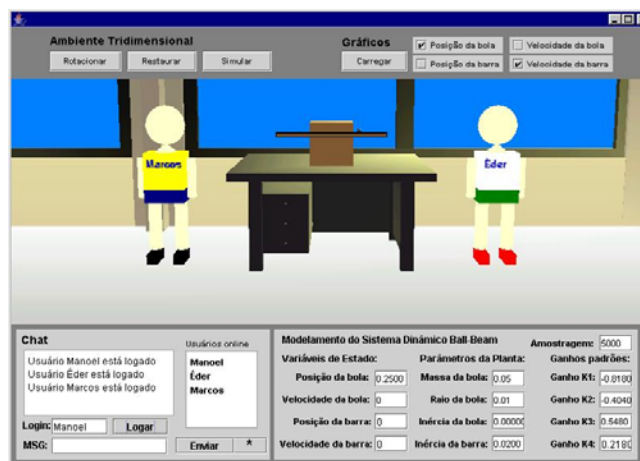
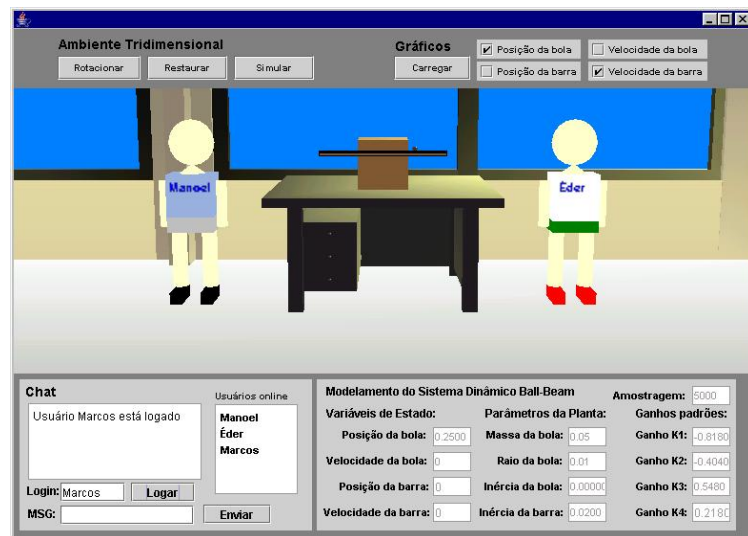
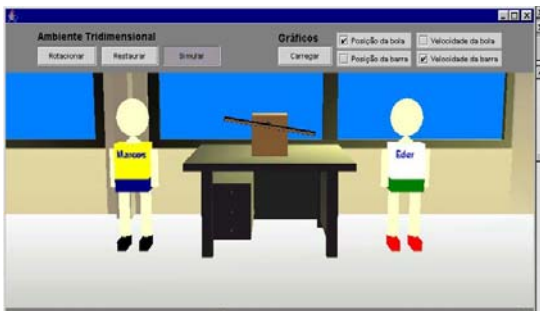


Figura 11: Ambiente do professor compartilhado pelos alunos *Marcos* e *Éder*

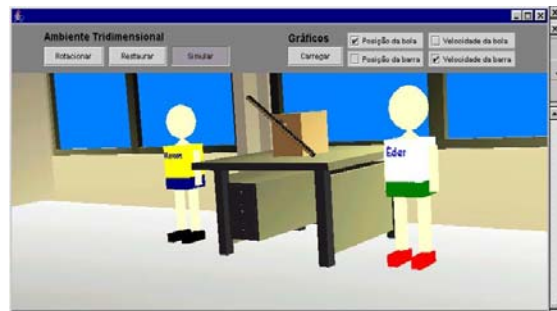


**Figura 12: O aluno Marcos visualiza o professor Manoel e o outro aluno Éder**

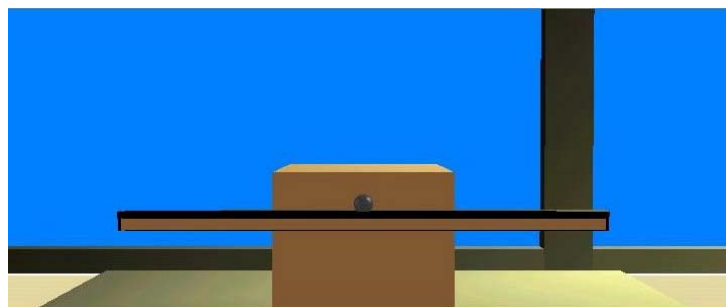
Nas Figuras 11 e 12, observa-se os valores padrões para os parâmetros da planta de controle, como a massa da esfera e a distância em relação origem da barra e os respectivos ganhos. A seguir o professor realiza uma simulação com esses valores padrões, clicando no botão “Simular” localizado na parte superior esquerda da janela da Figura 11. A Figura 13 mostra o sistema durante o controle, a partir daí, a barra, em forma de calha, faz um movimento de gangorra, visando não deixar a bola cair por uma de suas extremidades, lembrando que a interação entre barra e bola é realizada a partir do modelo matemático do sistema. A Figura 14 mostra também o sistema durante a simulação, porém com o ambiente rotacionado. A Figura 15 mostra o sistema em equilíbrio, visto pelo aluno Marcos, observe que a Figura 12 é o ambiente virtual também visto pelo aluno Marcos, porém na Figura 15 este navegou com o uso do teclado pelo mundo virtual.



**Figura 13: Sistema durante a simulação**



**Figura 14: Durante a simulação e com o ambiente rotacionado**



**Figura 15: Sistema em equilíbrio**



Como confirmação dos cálculos feitos pelo sistema de controle, pode-se ainda observar os gráficos das quatro variáveis de estado da planta, posição-velocidade da barra e da bola, para tanto se deve configurar os CheckBox's localizados na parte superior direita da janela e clicar no botão "Carregar". O gráfico da Figura 16 mostra posição e velocidade da barra o que ratifica realmente que o sistema é amortecido.



**Figura 16: Gráfico das variáveis de estado da barra**

A partir desse momento os alunos, com a permissão do professor, poderão simular a experiência com novos valores para as variáveis da planta, que também serão compartilhadas por todos os usuários conectados no ambiente. Alguns valores poderão ser alterados sem a necessidade de se fazer um novo projeto para o simulador. Um desses valores é a posição da esfera, que como se pode observar nas Figuras 11 e 12 tem como valor padrão 0.25. Os usuários podem mudar essa posição para valores entre

-0.25 e 0.25, clicar no botão Restaurar para o sistema configurar esse novo valor e depois clicar em Simular para realizar uma nova simulação, que mostrará que o sistema tenderá para o equilíbrio. Para a mudança em outros valores, como a massa da bola, será necessário fazer um novo projeto para o simulador, nesse caso serão encontrados outros valores para os ganhos (K1,K2,K3,K4) que deverão se repassados para o sistema.

## 5. Conclusão

Neste trabalho apresentou-se o projeto, a implementação e os resultados de um experimento virtual colaborativo de controle, especificamente desenvolvido para o sistema dinâmico "Ball-Beam". A construção de um ambiente virtual tridimensional representando o sistema "Ball-Beam" faz parte do interesse de desenvolver um laboratório virtual para disciplinas de Controle que contará também com outras plantas, como o "Pêndulo Invertido" (Cavalcanti, 1999), que já está em desenvolvimento, tendo como ponto de partida as mesmas características interativas desse trabalho.

É importante salientar que é possível, aos alunos ou professores realizarem a alteração nos parâmetros da planta: inércia da barra, inércia da bola, massa da bola e raio da bola, sendo dessa forma, também necessário que o usuário recalcule os ganhos de realimentação de estados, ou seja, neste caso para cada execução do programa tem-se um novo sistema "Ball-Beam" e um novo projeto do controlador, além de permitir alterações nas variáveis de estado inicial: posição ou velocidade linear da bola e posição ou velocidade angular da barra bem como a taxa de amostragens.

Os resultados encontrados mostram que o ambiente colaborativo funciona adequadamente, onde as simulações do experimento rodam de maneira compartilhada para todos os usuários conectados, sendo que como o sistema é não imersivo, optou-se por não se



mostrar o avatar no ambiente 3D visto pelo usuário corrente, ou seja, o professor visualiza em primeira pessoa os avatares dos alunos conectados, não vendo o seu avatar e um aluno conectado não visualiza o seu avatar, mas sim o do professor e os dos outros alunos conectados. A presença dos avatares aumenta o sentimento de ambiente compartilhado e a visão em primeira pessoa amplia a sensação de imersão.

## Referências

- Bouvier, D. J. (2002). "Getting Started with the Java 3D API", Online: disponível na Internet via <http://java.sun.com/products/java-media/3d/collateral>.
- Burdea, Grigore e Coiffet, Philippe. (1994) "Virtual Reality Technology", John Wiley & New York, NY.
- Cavalcanti, J. H. F., Alsina, P. J., Ferneda, E., (1999). Posicionamento de um Pêndulo Invertido Utilizando Algoritmos Genéticos. SBA Controle & Automação, Vol. 10, No. 01.
- D. Selman. (2002) "Java 3D Programming", Manning Publications CO.
- E. Borges, A Cardoso, e E. Lamounier Jr. (2002) "Investigando técnicas interativas para aprimorar o uso de realidade virtual no ensino de geometria espacial". Anais do V Symposium on Virtual Reality, SVR 2002, pp.78-89, Fortaleza.
- Faria, C. G. e Moscoso, M.N.A. (1997) "Simulação Híbrida de um Sistema Dinâmico do tipo Ball-Beam com a Implementação de um Controlador Digital em Tempo Real", Trabalho der Conclusão de Curso, DEEC-CT-UFPA.
- Houpis, D'Azzo. (1981) "Análise e Projeto de Sistemas de Controle Lineares", 2a edição, McGraw-Hill, Inc.
- J. Hauser, S. Sastry, and P. Kokotovic. (1992) "Nonlinear control via approximate input-output linearization: The ball and beam example" - IEEE Trans. Automat. Contr, vol. 37, pp. 392-398.
- JfreeChart class library,(2004) Online: na Internet via <http://www.jrefinery.com> .
- Lopes, A. S.; Oliveira, J. C.; Santos, M. P. L. (2002) "Laboratórios Virtuais: uma nova alternativa na preparação dos alunos para as aulas experimentais". In: XXX Congresso Brasileiro de Ensino de Engenharia, 2002, Piracicaba. Anais. Piracicaba: COBENGE/2002, p. 1-10.
- Matlab. The language of Technical Computing, Version 6.0.0.88 Release 12, The MathWorks, Inc. September 22, 2000.
- Ogata, Katsuhiko. (1990) "Engenharia de Controle Moderno", 2a edição, PHB-Prentice Hall do Brasil.
- Ogata, Katsuhiko. (1996) "Projeto de Sistemas Lineares de Controle com Matlab". Prentice - Hall do Brasil, Rio de Janeiro.
- Patrício, Éder Torres; Ribeiro Filho, Manoel e Silva, Orlando Fonseca (2004), "Laboratório Virtual De Controle: Uso De Técnicas De Realidade Virtual Para Representação E Controle De Um Sistema Ball-Beam" ., artigo aceito para ser publicado nos anais do XV Congresso Brasileiro de Automática, da Sociedade Brasileira de Automática, a ser realizado em Gramado, Rio Grande do Sul, Brasil em Setembro de 2004.
- Perucia, Alexandre Souza e Pinho, Márcio Serolli(2002), "Ferramenta de Suporte à Construção de Ambientes Virtuais Colaborativos", anais do V Symposium on Virtual Reality, SVR2002, pgs, 115 a 125.

Rothe-Neves, M. (1999) “Construção de protótipo para laboratório de ensino e pesquisa: sistema ball-beam”, Trabalho de Conclusão de Curso, DEEC-CT-UFPA.

Sun Microsystem. Online (2004): site oficial <http://www.sun.com/> .

Vidal, Creto A.; dos Santos, Emanuele Marques; Leite Júnior, José M.; Almendra, Camilo C. e Borges, Vlândia Maria C.(2003), “Collaborative Virtual Environments for Language Learning”, anais do VI Symposium on Virtual Reality, SVR2003, pgs, 309 a 320.