

## Modelagem, Visualização e Simulação de Manipuladores Mecânicos

Anderson Maciel  
Departamento de  
Informática – Universidade de  
Caxias do Sul  
Rua Francisco Getúlio Vargas,  
1130  
Caxias do Sul - RS – 95070-560 -  
Brasil  
+55 054. 212.1133  
Anderson@nl.com.br

Gilda Aparecida de  
Assis  
Departamento de  
Informática – Universidade de  
Caxias do Sul  
Rua Francisco Getúlio Vargas,  
1130  
Caxias do Sul - RS – 95070-560 -  
Brasil  
+55 051.286.4708  
gaassis@ucs.tche.br

Ricardo Vargas  
Dorneles  
Departamento de  
Informática – Universidade de  
Caxias do Sul  
Rua Francisco Getúlio Vargas,  
1130  
Caxias do Sul - RS – 95070-560 -  
Brasil  
+55 054.212.1133  
rvdornel@ucs.tche.br

### 1. Resumo

Este artigo apresenta o ASIMOV, um ambiente educacional para o desenvolvimento e simulação de manipuladores mecânicos. Os principais aspectos referentes à modelagem, visualização e simulação dos manipuladores mecânicos são apresentados e discutidos.

### 2. Palavras-chave

Robótica, Computação Gráfica, CAD, Manipuladores Mecânicos, Ambientes Educacionais.

### 3. Introdução

O ASIMOV é um ambiente educacional integrado para auxiliar no projeto e simulação de manipuladores mecânicos.

Um manipulador mecânico é uma máquina programável de propósito geral. Usualmente, os manipuladores são comparáveis, quanto à forma e capacidade de movimento, ao braço humano. Os manipuladores executam atividades rotineiras, como pintura, montagem, carga, etc.

Há uma variedade de ferramentas de software para o projeto e simulação de manipuladores mecânicos. Usualmente, estas ferramentas são voltadas para etapas específicas do projeto e não são integradas [7]. Neste contexto, o projeto ASIMOV foi desenvolvido. O ASIMOV integra recursos computacionais para auxiliar o projeto e simulação de manipuladores mecânicos. Os recursos disponibilizados são:

- Base de dados de componentes;
- Módulo de CAD;
- Módulo de Simulação;
- Módulo de Programação;
- Módulo de Ensino.

Este trabalho apresenta a descrição em detalhes das técnicas de modelagem, visualização e animação presentes nos módulos de CAD e Simulação. A figura 1 representa a estrutura geral do Ambiente ASIMOV. As elipses localizam os módulos de CAD e Simulação dentro da estrutura geral do Ambiente ASIMOV.

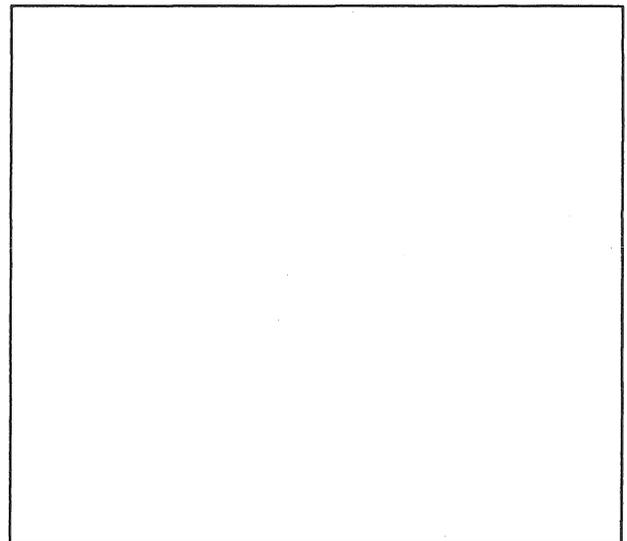


Figure 1

Este artigo está organizado da seguinte forma: A próxima seção inicia com uma visão geral da estrutura de dados utilizada. Na seção 4 o módulo de ensino é apresentado. Na seção 5 são apresentadas as operações de modelagem incluídas no módulo de CAD de manipuladores. A seção 6 apresenta as operações de visualização 3D que são utilizadas nos módulos de CAD de manipuladores e de Simulação. A seção 7 apresenta uma visão geral do módulo de Simulação. Na seção 8 o método de rendering é apresentado e na seção 9 é descrita a detecção de colisões. A seção 10 apresenta aspectos da implementação do Ambiente ASIMOV. Finalmente, a seção 11 apresenta algumas conclusões sobre o projeto.

#### 4. Estrutura de Dados

Esta seção descreve a estrutura geométrica e hierárquica dos manipuladores

As características do projeto ASIMOV pedem uma estrutura de dados capaz de representar elementos geométricos em três dimensões, o agrupamento destes elementos em um complexo sistema hierárquico, as informações físicas destes elementos (massa, momento de inércia), informações necessárias ao mecanismo de visualização 3D (dimensões da janela, posição da câmara sintética, ângulo de abertura da câmara sintética) e todas as informações relacionadas à simulação de movimentos.

Os objetos primitivos são obtidos a partir de arquivos no formato DXF. Os objetos primitivos são obtidos a partir de arquivos no formato DXF. A estrutura geométrica é composta de pontos, que aos pares formam segmentos. Estes segmentos em grupos com  $n$  elementos formam faces, que são montadas para formar poliedros convexos. Cada um destes poliedros é denominado um objeto primitivo. (Figura 2).

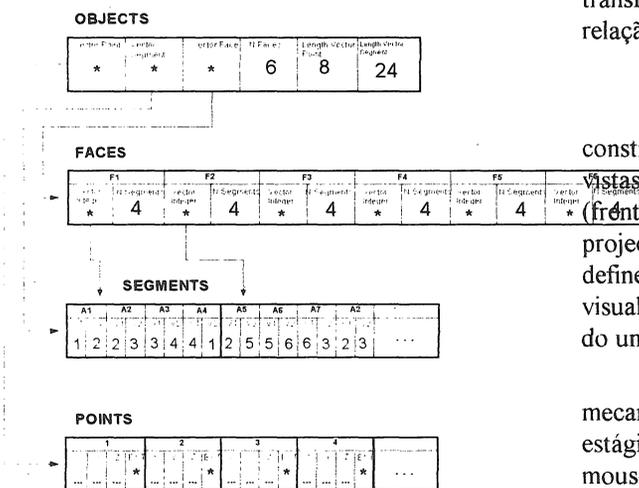


Figure 2

Os objetos primitivos são montados em grupos. Cada um dos objetos primitivos de um grupo é descrito no sistema de referência cartesiano do grupo. Um grupo é uma peça de um manipulador mecânico. Os objetos primitivos de uma peça não têm movimento relativo entre eles. A estrutura geométrica de um manipulador mecânico é composta de um conjunto limitado de peças.

O movimento de uma peça relativo a outras peças é definido através de uma estrutura hierárquica. As peças são identificadas por um número inteiro. A primeira peça é fixa no sistema de referência do universo. Assim, esta peça não se move e seu identificador 0 (zero).

A maioria dos robôs industriais são articulados. Eles são compostos por juntas de revolução e translação. Apenas um conjunto de peças não define uma hierarquia, é necessário também definir um conjunto de juntas. Os identificadores das juntas são inteiros. O identificador da primeira junta é o de número 1 (um). Cada junta agrega

duas peças. O par de peças montado em uma junta é classificado como a peça fixa e a peça móvel. A peça fixa tem o identificador menor, enquanto que a peça móvel tem o identificador maior. Quando uma junta se move, é a peça móvel e seus objetos primitivos que se movem. Por exemplo, se o manipulador mecânico representa um braço humano e a junta correspondente ao cotovelo humano se move então o braço move-se mas o antebraço continua fixo, o que não é o resultado desejado. Então um método é utilizado para relacionar as juntas.

A notação de Denavit-Hartenberg, originalmente utilizada para manipulação de robôs, é utilizada para estabelecer uma inter-relação entre as juntas. [3]. Tal formalismo descreve a cinemática de cada elemento em relação a seu vizinho associando um sistema de coordenadas a cada elemento. Isto significa que cada sistema de referências cartesianas de uma junta está associado às coordenadas da junta vizinha predecessora em uma hierarquia de juntas. Assim, quatro parâmetros são produzidos. Estes parâmetros definem uma matriz de transformação linear, a matriz DH, que estabelece a relação entre juntas consecutivas.

#### 5. Operações de Modelagem

Quatro vistas diferentes são utilizadas na construção geométrica dos manipuladores mecânicos: 3 vistas originadas das projeções ortográficas paralelas (frente, superior e lateral) e uma vista originada da projeção em perspectiva. Na vista perspectiva, o usuário define os atributos de uma câmara sintética para visualizar o manipulador mecânico de diferentes posições do universo.

O módulo de CAD de manipuladores apresenta mecanismos para seleção e manipulação de objetos no estágio de modelagem. Objetos são selecionados com o mouse e a seleção de partes é feita através da seleção do sistema de referência cartesiano da peça. As operações de manipulação implementadas são:

Translação;

Rotação;

Escala.

Estas operações podem ser utilizadas para definir a posição, a dimensão e para reorientar os objetos primitivos e as peças do manipulador mecânico.

#### 6. Operações de Visualização

As operações de visualização podem ser divididas em duas classes:

Modelagem

Simulação

##### 6.1 Visualização na Etapa de Modelagem

Na etapa de modelagem do manipulador mecânico não há informação sobre a estrutura hierárquica

do manipulador. Nesta etapa, as peças permanecem livres, com nenhuma ligação física entre elas.

Nesta etapa, a visualização corresponde aos passos necessários para mapear a informação geométrica 3D que está armazenada na estrutura de dados do Sistema ASIMOV, em linhas de um monitor CRT, que forma as imagens do manipulador mecânico. A figura 4 apresenta o pipeline de visualização na etapa de modelagem, isto é, todos os passos na seqüência de execução.

### 6.2 Visualização na Etapa de Simulação

O módulo de visualização permite ao usuário selecionar um manipulador já definido da base de dados e simular seu funcionamento. Portanto, é necessário ter uma estrutura de dados representando a hierarquia de manipuladores. Assim, o processo de visualização na etapa de simulação é diferente do processo na etapa de modelagem.

Estas diferenças são a inclusão de alguns passos novos e a divisão do processo de visualização em dois blocos na etapa de simulação. Estes blocos são:

- Passos executados antes do início da simulação
- Passos executados em cada loop da simulação

Os passos executados antes do início da simulação geram as estruturas para representar a hierarquia de peças existente no manipulador.

Os passos executados em cada loop de simulação analisam esta hierarquia e extraem a informação geométrica necessária. Depois disso, o processo de visualização na etapa de simulação é similar ao processo de visualização na etapa de modelagem.

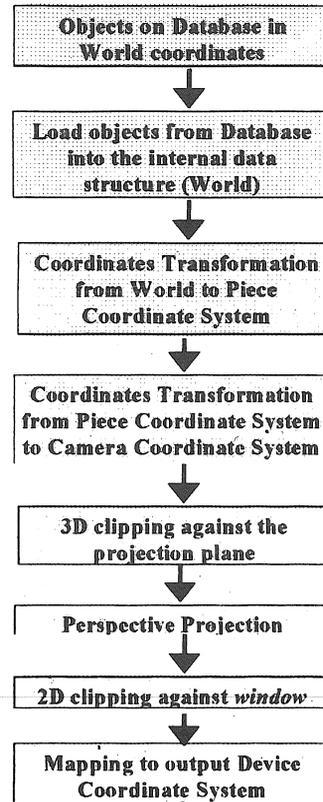


Figure 4

A Figura 5 apresenta a seqüência de passos na visualização na etapa de simulação.

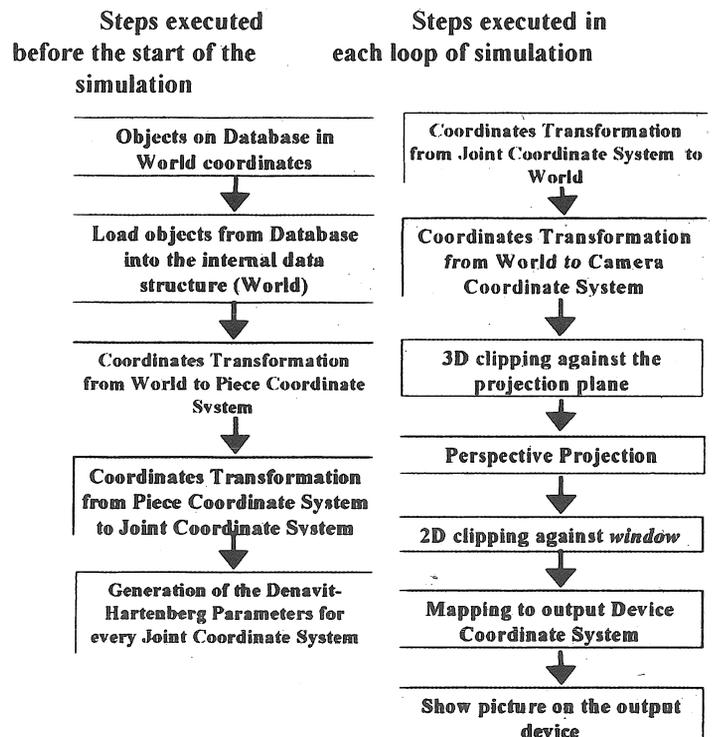


Figure 5

## 7. Módulo de Simulação

As ações que o manipulador realizará são determinadas por um programa. A simulação de um programa inicia por sua definição. O Ambiente ASIMOV apresenta um editor de programas orientado a sintaxe integrado ao ambiente. A linguagem utilizada no Sistema ASIMOV para a programação de manipuladores é a VAL II [10].

Após a edição do programa, o mesmo é submetido a um processo de análise para verificar sua correção e uma estrutura mais apropriada à simulação é gerada. Da estrutura gerada pelo processo de análise, um conjunto de pontos alvo e as respectivas coordenadas cartesianas são gerados. Após, a conversão destes pontos para coordenadas de juntas é feita através de algoritmos de cinemática inversa. A cinemática inversa usa transformações homogêneas para gerar um sistema de equações no qual o resultado é uma função das coordenadas cartesianas do ponto alvo e das outras posições das juntas [11]. Para obter a solução destas equações um algoritmo iterativo é necessário. Entretanto, considerando que a programação de tarefas para robôs relativamente simples gera uma quantidade muito grande de pontos, o uso destes algoritmos demanda um grande esforço computacional. Em face desta dificuldade, soluções fechadas foram desenvolvidas para a maioria dos manipuladores mecânicos disponíveis.

A figura 6 mostra o ASIMOV durante a simulação de um programa. Pode-se ver na figura o painel do teach pendant, o painel dos sensores e o manipulador sendo simulado.

Figure 6

## 8. Rendering

Alguns elementos que não pertenciam ao projeto ASIMOV original foram acrescentados. Estes componentes são implementações de métodos de rendering e detecção de colisões. Estes elementos foram incluídos no projeto ASIMOV devido a necessidades identificadas durante a validação do sistema. Esta validação foi efetuada por especialistas em Engenharia Elétrica e Mecânica. Estes componentes foram implementados como trabalhos de conclusão de curso por alunos de graduação do Bacharelado em Ciência da Computação na UCS.

Os métodos de rendering aproximam a imagem gerada pelo computador da realidade. Os métodos de rendering implementados são a remoção de elementos ocultos e o cálculo de iluminação e sombreamento.

Entre os vários métodos de rendering encontrados na literatura, o método selecionado foi o sombreamento de Gouraud, aplicado a um modelo de iluminação local. O sombreamento de Gouraud interpola a intensidade na superfície. Em um modelo de

iluminação local, apenas a primeira reflexão de luz é considerada. O sombreamento de Gouraud calcula as intensidades de iluminação para cada vértice dos polígonos que formam os objetos primitivos e usa uma interpolação entre os vértices para calcular a iluminação em outros pontos. O sombreamento de Gouraud elimina o efeito de intensidades descontínuas e dá um aspecto de suavidade aos objetos curvos representados por uma malha de polígonos.

Este método não é o mais realístico, entretanto, sua implementação é simples e sua execução é mais rápida que outros métodos de realismo. No caso do projeto ASIMOV, o principal objetivo é a simulação de movimento. Assim, o sombreamento de Gouraud é adequado, uma vez que:

- É o método mais rápido para geração de imagens;
- É adequado à estrutura de dados usada para representar os manipuladores mecânicos no sistema ASIMOV;
- Fornece o grau de realismo necessário para que o projeto ASIMOV atinja seus objetivos.

## 9. Detecção de Colisões

Outra funcionalidade acrescentada ao projeto ASIMOV original é a detecção de colisões. Em um ambiente de operação real, um manipulador mecânico pode atingir outros objetos ou pessoas. Por isso, é importante durante a simulação de um manipulador mecânico verificar se ocorreram colisões.

Esta verificação pode ser feita através do uso de projeções perspectivas, obtidas de diferentes posições do visualizador no ambiente durante a simulação. Isto pode ser conseguido com o uso de uma câmara sintética.

Esta alternativa envolve busca exaustiva e erros podem ocorrer. Para evitar isso, um procedimento automático de detecção de colisões é utilizado. Este procedimento verifica se um objeto ocupa o espaço já ocupado por outro objeto. Este teste é feito a partir da descrição geométrica dos objetos e de sua posição e orientação no universo.

Após um estudo dos algoritmos existentes na literatura sobre detecção de colisões, concluiu-se que um algoritmo em quatro passos, apresentado por Garcia-Alonso [12], tinha grandes vantagens sobre os outros. Duas razões principais justificam a escolha do método: a adequação à representação das estruturas geométricas e eficiência (o método deve assegurar que uma simulação em tempo-real é possível).

Este método executa uma seqüência de quatro passos onde cada passo é um filtro de possibilidades de colisão. O último passo raramente será executado, apenas quando alguma possibilidade de colisão não pode ser

detectada pelos passos anteriores. Além disso, um conjunto de procedimentos é executado na etapa de pré-processamento. Esta etapa de pré-processamento reduz a quantidade de processamento necessário durante o loop de simulação. O envelope minimax e a voxelização são dois métodos utilizados pelo algoritmo de Garcia-Alonso com o objetivo de otimizar o processo de detecção de colisões de modo que o mesmo possa ser executado em um computador pessoal.

### 10. Implementação

O Ambiente ASIMOV foi desenvolvido utilizando a linguagem C++. O ASIMOV roda nos sistemas operacionais Windows 95 e Windows NT. O compilador utilizado foi o Borland C++ 5.01 da Inprise Inc. As rotinas da base de dados de componentes também foram desenvolvidas em C++ utilizando o SGBD Oracle.

### 11. Conclusões

Normalmente, para avaliar o comportamento de um manipulador mecânico, é necessário construir um protótipo físico para verificar se ele atende os objetivos de projeto. Para a construção deste protótipo, normalmente uma grande quantidade de tempo é necessária. Se há alterações no projeto inicial, novos protótipos devem ser construídos. Por outro lado, com um protótipo nem sempre é possível obter todas as informações necessárias para analisar o projeto. Um ambiente para representar e manipular objetos físicos é importante porque ele elimina, ou ao menos reduz, a necessidade de construir protótipos físicos. Em um sistema com suporte ao desenvolvimento de protótipos virtuais, os usuários podem analisar e avaliar seu projeto em diferentes situações, utilizando menos tempo, esforço e recursos.

O projeto ASIMOV foi desenvolvido neste contexto. O Ambiente ASIMOV é um ambiente integrado para auxiliar no projeto e simulação de manipuladores mecânicos e no ensino dos fundamentos da robótica. Pode ser utilizado tanto nos níveis técnico e acadêmico, ou mesmo em um ambiente industrial como um complemento para o profissional que desenvolve ou opera este tipo de equipamento.

Atualmente, o ASIMOV está sendo utilizado nos cursos de Automação Industrial e Engenharia Mecânica, na UCS, e tutores para o ensino de eletricidade e pneumática estão sendo desenvolvidos.

O próximo passo no desenvolvimento do ASIMOV é sua integração em um ambiente de simulação global simulando uma linha de produção, e algum trabalho ainda deve ser feito para considerar a dinâmica na simulação.

### 12. Agradecimentos

Agradecemos à UCS, CNPq e FAPERGS pelo suporte a este trabalho.

### 13. Bibliografia

[1] MACIEL, A. Relatório Técnico do Módulo de Programação do Projeto ASIMOV para o PIBIC/CNPq. Caxias do Sul: UCS, 1998. (Relatório Técnico)

[2] PELLIN, A. F.; ORTIGARA, V. Interface Cliente/Servidor para os Módulos de CAD de Base de Dados de Componentes do Projeto ASIMOV. Caxias do Sul: UCS, 1997. (Trabalho de Conclusão de Curso)

[3] DENAVIT, J.; HARTEMBERG, R. S. A Kinematics Notation for Lower-Pair Mechanisms Based on Matrices. *J. App. Mech.*, v.77, p.215-221. 1955.

[4] FACHIN, N. Obtenção de Imagens de Objetos Geométricos Modelados por Computador Através de Técnicas de Cálculo de Iluminação e Métodos de Sombreamento, 1997. (Trabalho de Conclusão de Curso)

[5] MACIEL, A. Detecção de Colisões entre Pares de Poliedros Rígidos Aplicada ao Projeto ASIMOV, 1998. (Trabalho de Conclusão de Curso)

[6] WEBBER, C. G. Representação do Conhecimento sobre o Domínio em um Sistema Tutorial Inteligente de Hidráulica, Caxias do Sul: UCS, 1994. (Trabalho de Conclusão de Curso)

[7] SOUZA, A. *et al.* Educational Framework for the Development of Mechanical Manipulators. In: INTERNATIONAL IFIP 9.9 CONFERENCE, 1997, Florianópolis: *Proceedings...*, Florianópolis: [S. N.], 1997

[8] BUTIGNOL, C. M. Táticas de Ensino para um Sistema Tutorial Inteligente em Hidráulica. Caxias do Sul: UCS, 1995. (Trabalho de Conclusão de Curso)

[9] VIECELLI, G. Estratégias de Ensino para um Tutor Inteligente em Hidráulica. Caxias do Sul: UCS, 1994. (Trabalho de Conclusão de Curso)

[10] GROOVER, M. P. *et al.* Industrial Robotics. New York: McGraw-Hill, 1986.

[11] FU, K. S. *et al.* Robotics: Control, sensing, vision and intelligence. New York: McGraw-Hill, 1987.

[12] GARCIA-ALONSO, A. *et al.* Solving the Collision Detection Problem. In: IEEE Computer Graphics and Applications, v.14, n.4, p. 36-43, 1994.

