

Modelação de Movimento no Sistema MS(E)

Rui Filipe Costa, DI.ISEP

Vasco Branco, DI.ISEP/INESC.Norte

Primeiro Encontro Português de Computação Gráfica

LNEC, JULHO de1988

LISBOA



1.Introdução

Uma das vertentes de desenvolvimento do sistema **MS(E)** - Modelador de Sólidos Experimental[BrF88] - prende-se ao estudo da modelação de movimento, partindo das descrições dos sólidos nesse sistema.

A detecção de interferências entre partes móveis de objectos, a simulação dinâmica do seu comportamento sob a acção de esforços, a simulação de maquinagens, a par da animação computacional são alguns dos motivos que justificam o interesse de dispôr, num sistema de modelação de sólidos, da possibilidade de associar movimento aos modelos dos objectos.

Partindo do sistema **ANIMED** desenvolvido no INESC-SUL, foi feita a sua adaptação ao sistema **MS(E)**, bem como a introdução de novas facilidades que permitem a definição do movimento de um objecto (simples ou composto) independentemente de outros, presentes na mesma cena.

2.Adaptação do sistema ANIMED ao MS(E)

No sistema **ANIMED**, o movimento dos objectos é obtido pela aplicação de um conjunto de transformações geométricas (rotações, translações, mudanças de escala), e pela definição do tempo de duração de cada sequência. Cabe ao utilizador definir interactivamente estes parâmetros através de um interpretador de comandos de linha, configurando os vectores de animação (v_{an}), atribuindo valores a parâmetros como o numero de imagens por segundo (ips), escolhendo opções como a espera entre 'frames' (wait), ou o tipo de visualização (com ou sem rasto) (path).

Para além da execução directa dos vectores de animação definidos, o sistema **ANIMED** pode trabalhar no modo **LEARN** para guardar num ficheiro uma sequência de vectores que serão executados pela invocação do mesmo através do modo **EXEC**.

Um vector de animação consta de um conjunto de valores que especificam a duração do movimento (t), as translações (t_x, t_y, t_z), rotações (r_x, r_y, r_z), mudanças de escala (s_x, s_y, s_z) e respectiva ordem de rotação (o_{rx}, o_{ry}, o_{rz}), bem como as leis de movimento (l_t, l_r, l_e), (uniforme/uniformemente

acelerado/etc), a atribuir a cada uma das transformações geométricas referidas. O 'van' terá então a configuração seguinte:

[tx ty tz t lt lr le orx ory orz rx ry rz ex ey ez]

A adaptação do sistema ANIMED ao MS(E) consistiu em:

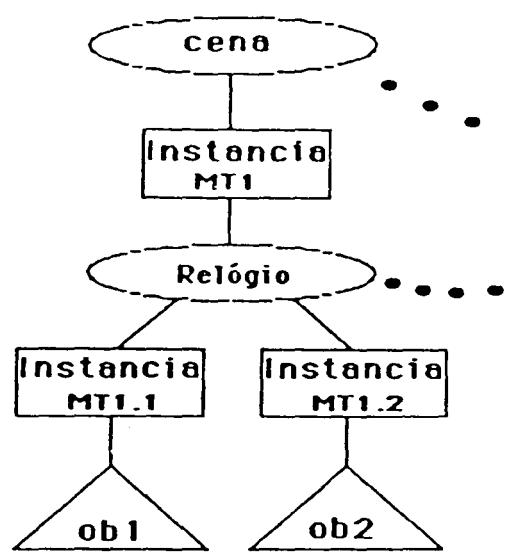
- reescrita da rotina de visualização em conformidade com a estrutura de dados de descrição dos objectos (hiérarquia topológica) e com o 'stack' de transformação de matrizes [BrF88].
- a matriz de transformação correspondente à definição do movimento é pós-concatenada às matrizes de instância antes de se proceder ao desenho da 'frame'. Na primeira versão, o movimento era comum para todos os objectos presentes na cena.
- introdução de novos comandos:
 - **VISIB**: invocação do algoritmo de 'ray-tracing' para 'pintar' a 'frame' produzida [AIB88].
 - **FIXAR**: salvaguarda da posição final do objecto pela alteração da sua matriz de instância. A nova matriz de instância é resultante da concatenação da matriz de instância anterior com a de movimento.
 - **EIXOS**: definição dos eixos de rotação (em x,y,z) dispondo da constante CENTRO (valor por defeito) para a rotação em torno do centro de massa.
 - **PLOTTER**: passagem a papel das 'frames' produzidas.
 - **HELP**: selectivo por nome de comando.

3. Extensão ao sistema ANIMED - actores e sub-actores

Um actor (objecto animado) [NDT85], no caso de ser um objecto complexo, é composto de varias partes, que para além de acompanharem o movimento do objecto que integram, cada uma delas poderá ser sujeita a um movimento próprio (sub-actores) Assim, quando um relógio acompanha o movimento de um pulso, todas as suas peças o seguem, no entanto cada um dos ponteiros segue a sua rotação própria.

A estrutura hierárquica de representação de objectos em MS(E) reflectiria uma cena deste tipo da seguinte maneira:

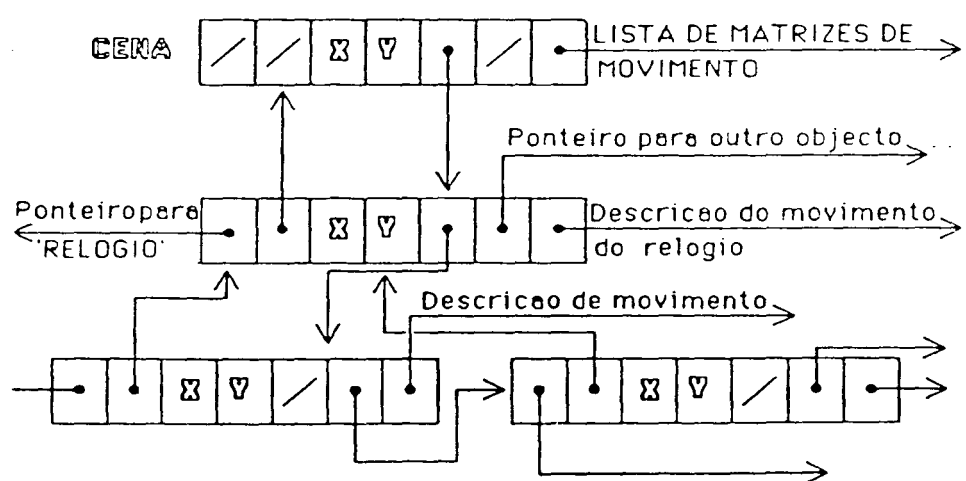




ob1 - Ponteiro dos minutos
 ob2 - Ponteiro das Horas

A matriz de instância MT1 é responsável pela colocação no espaço do objecto composto (relógio); a posição dos ponteiros é definida, no entanto, pelo produto $MT1.X * MT1$.

Outro aspecto importante, prende-se com a entrada em movimento e a paragem de cada um dos objectos (actores ou sub-actores) presente numa cena. Assim, foi definida uma estrutura de dados de suporte à animação de actores e sub-actores, que se liga a hierarquia topológica de MS(E) ao nível de instância.



X - duração do movimento
 Y - numero da frame de arranque



Como é visível cada célula desta estrutura contém, não só a descrição do movimento de um objecto (lista de matrizes, duração e arranque) como também um conjunto de apontadores que a ligam por um lado a informação contida na hierarquia topológica e por outro às outras células da estrutura (reflectindo as ligações presentes na hierarquia topológica).

O vector de animação foi alterado para incluir o nome do objecto a que diz respeito; o movimento global da cena pode ser definido por um 'van' atribuindo ao nome do objecto a constante 'CENA'.

A lista de matrizes de movimento para cada objecto é calculada a partir do 'van' [LoD87], mas posteriormente é feita a concatenação de todas as matrizes influentes na posição do objecto com cada uma das matrizes dessa lista, utilizando o 'stack' de matrizes do sistema MS(E).

Dois novos comandos foram introduzidos no sistema:

- **ANIMAR** [<nome>]['DE' <instante inicial>]['ATE' <instante final>]

Este comando é responsável pela execução de movimento que como é visível na sintaxe pode ser limitado a um só objecto ou a um determinado intervalo de tempo.

- **INICIAR** [<nome>]:

Permite fazer a destruição da lista de matrizes de movimento associada a um objecto (e 'reset' dos tempos de duração e arranque) ou de todas as listas constituídas.

4. CONCLUSÃO

A continuação deste trabalho prevê a inclusão da animação das cores dos objectos e a definição de camaras virtuais. Outra vertente do seu desenvolvimento passará pela alteração da 'interface' com o utilizador de modo a contemplar processos interactivos para a modelação de movimento [MeR88].

Agradecemos as facilidades e o apoio concedidos pelo grupo de CAD do INES-SUL e nomeadamente ao Eng. Pedro Lopes Faria.

5. BIBLIOGRAFIA

[NDT85] N.Magnenat-Thalmann,D.Thalmann,
"Computer Animation Theory and Practice",
Springer-Verlag,1985

[LoD86] P.Faria Lopes , J.M.S. Dias
"ANIMED - Animation Editor",
Relatório Técnico,RT 87-1106-1,INESC(SUL),Março de 1986

[BrF88] Vasco A.Branco, F. Nunes Ferreira
"Um modelador de sólidos experimental: o sistema MS(E)"
(a apresentar neste Encontro)

[AIB88] J. E. Almeida, V. A. Branco
"Adaptação de um algoritmo de 'ray-tracing' ao sistema MS(E)"
(a apresentar neste Encontro)

