

**INTERPOL3D - SISTEMA DE ANIMAÇÃO POR  
INTERPOLAÇÃO DE FORMAS DE OBJETOS  
DESCRITOS POR SUPERFÍCIES DE BÉZIER**

***Manuel Menezes de Oliveira Neto***  
***(CPGCC - UFRGS)***

***Anatônio Laschuk***  
***(CPGCC - UFRGS)***

Página em branco na versão original impressa.

INTERPOL3D - SISTEMA DE ANIMAÇÃO POR INTERPOLAÇÃO DE FORMAS  
DE OBJETOS DESCRITOS POR SUPERFÍCIES BÉZIER

Manuel Menezes de Oliveira Neto

Anatônio Laschuk

UFRGS - Universidade Federal do Rio Grande do Sul  
CPGCC - Curso de Pós-Graduação em Ciência da Computação  
Caixa Postal 1501 - 90001 - Porto Alegre, RS  
e-mail : menez@cunhan.inf.ufrgs.anrs.br

**ABSTRACT** - This work presents a tridimensional animation system for shape interpolation that offers control over the velocities of interpolation. It also establishes the correspondences between the pieces of the objects to be interpolated automatically.

## 1. INTRODUÇÃO

Interpolação de quadros-chaves constitui a técnica básica do processo de animação convencional. Conhecido como *inbetweening*, corresponde à geração de quadros intermediários para uma seqüência de imagens em movimento/transformação, dados um primeiro e último quadros. A essência deste processo corresponde à determinação de que partes ou pontos do primeiro quadro devem ser mapeados em quais partes ou pontos do último quadro [8]. Apesar de sua simplicidade conceitual, os processos de interpolação apresentam dificuldades no que diz respeito ao estabelecimento de correspondências entre as partes a serem interpoladas, especificação de acelerações/desacelerações dos movimentos e descontinuidades nas imagens de quadros vizinhos [8].

O processo de animação 2D auxiliado por computador apresenta várias restrições. Nele, a geração automática de quadros intermediários, a partir de quadros-chaves, é dificultada pelo fato de que os desenhos são projeções 2D de objetos 3D. Deste modo, perde-se informação preciosa para o processo de interpolação [2]. Outra deficiência é a necessidade do usuário especificar as correspondências entre linhas dos dois quadros, isto é, a necessidade de serem definidas correspondências entre linhas visíveis e outras, possivelmente não visíveis, de modo que

as transições possam se processar corretamente [2]. A dificuldade de um programa determinar automaticamente tais correspondências deve-se à perda de informação que acontece na passagem objeto 3D → projeção 2D. Animações 3D auxiliadas por computador têm sido freqüentemente utilizadas para obtenção de efeitos especiais em filmes como : "The Abyss", "The Last Starfighter", "Young Sherlock Holmes" [4], "Return of Jedi" e "TRON" [9].

Este trabalho apresenta um sistema para animação tridimensional baseado em objetos-chaves. Nele, objetos, descritos por superfícies com bordos [6], modelados por composições de patches Bézier bicúbicos, podem ter suas geometrias interpoladas. O sistema estabelece, automaticamente, as correspondências entre os vários patches que compõem os objetos a serem transformados. Além disso, oferece controle sobre acelerações/desacelerações durante o processo de interpolação, utilizando-se de interfaces bastante simples.

## 2. O SISTEMA DE ANIMAÇÃO INTERPOL3D

O sistema INTERPOL3D é parte integrante do ambiente BEZIER4D, também composto por um sistema de modelagem (BEZIER3D [1]) e por um sistema de visualização (RENDER3D).

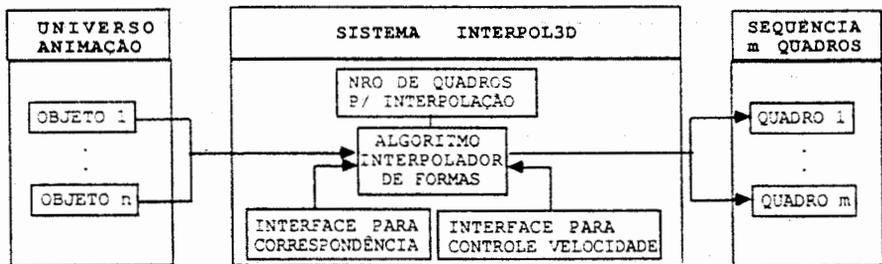
O sistema INTERPOL3D provê a geração de objetos intermediários para o procedimento de interpolação de forma entre pares de objetos definidos por superfícies com bordos [6]. Os elementos a serem interpolados podem apresentar geometrias "livres" (respeitadas as restrições estabelecidas por tais superfícies) e serem compostos por diferentes números de patches.

### 2.1. Estrutura do sistema

O sistema recebe como entrada um conjunto de objetos modelados por composições de patches Bézier bicúbicos, os quais definem o universo da animação. O usuário pode, então, especificar o número de quadros a serem gerados (o termo quadro não nos parece adequado para expressar um instante de uma animação tridimensional, a qual independe da posição do observador. Portanto, neste texto, aparecerá sempre em *itálico*),

estabelecer as correspondências entre pares de objetos para interpolação e as acelerações/desacelerações envolvidas no processo. Como resultado, o sistema fornece uma seqüência de quadros, que pode ser apresentada de duas maneiras : *preview*, onde os objetos são exibidos com remoção de linhas ocultas, dando uma boa idéia de como se processará a animação; ou saída em arquivo (em RGB) para posterior exibição com *rendering* pelo RENDER3D, o módulo de visualização do ambiente BEZIER4D. Veja figura 1.

Manipulando objetos no espaço tridimensional, o sistema elimina o inconveniente da perda de informação que ocorre na passagem objeto 3D  $\rightarrow$  projeção 2D, podendo, portanto, estabelecer, automaticamente, as correspondências entre os vários patches de objetos associados.



Figural. Estrutura do Sistema INTERPOL3D

## 2.2. O algoritmo interpolador de formas

O procedimento utilizado pelo módulo interpolador de formas consiste em: equacionar o número de patches dos objetos de cada par a ser interpolado; estabelecer, automaticamente, as correspondências entre esses patches e proceder a geração dos objetos intermediários por meio da interpolação dos pontos de controle das superfícies correspondentes. Tal equacionamento pode se obtido por sub-divisão de patches pré-existentes (utilizando o algoritmo de de Casteljaou [3]) e/ou por inclusão de novos patches

degenerados, sempre preservando as geometrias dos objetos originais (veja figura 2).

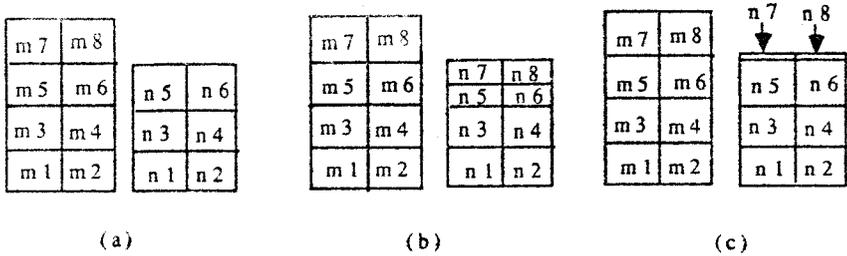


Figura 2. (a) Modelos planos de dois objetos antes do equacionamento do número de patches; Modelos planos dos mesmos objetos após equacionamento por: (b) subdivisão, (c) inclusão de patches degenerados.

### 3. A CLASSE DE OBJETOS VÁLIDOS

INTERPOL3D manipula representações de objetos descritos no espaço Euclidiano tridimensional  $E^3$ . Os objetos válidos, no contexto INTERPOL3D, são representados por superfícies com bordos [6], não sendo aceitas, portanto, realizações de superfícies variedade-2 em  $E^3$ . Uma superfície variedade-2 é um espaço topológico no qual todo ponto tem uma vizinhança topologicamente equivalente a um disco aberto em  $E^2$  [6]. Uma superfície  $S_1$  é dita uma realização da superfície variedade-2  $S_2$ , se  $S_1$  e  $S_2$  são topologicamente equivalentes [6]. Assim, são considerados objetos válidos para interpolação, todos os objetos topologicamente equivalentes a um disco ou a uma superfície cilíndrica no  $E^3$ .

#### 3.1. Superfícies com bordos

Mäntylä [6] apresenta a definição de um tipo especial de topologia que emprega o conceito de figura plana rotulada, bastante conveniente para o escopo deste trabalho. Uma figura plana rotulada é um modelo plano no qual arestas podem ser

rotuladas e orientadas. A superfície representada é obtida unindo-se as arestas de mesmos rótulos, respeitadas suas orientações. Assim, ambas as representações da figura 3 caracterizam a mesma superfície cilíndrica.

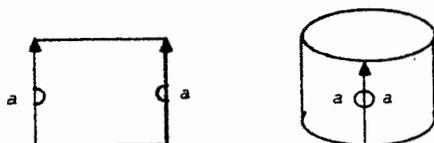


Figura 3. Representações de uma superfície cilíndrica (extraído de [6]). As arestas não rotuladas constituem os bordos da superfície.

A vizinhança dos pontos sobre as arestas não rotuladas são discos abertos de raio  $r$  no  $E^2$ . Por sua vez, a vizinhança dos demais pontos, inclusive aqueles situados sobre as arestas rotuladas, são discos no  $E^2$  [6]. Devido às particularidades dos pontos situados sobre as arestas rotuladas, estas superfícies são denominadas superfícies com bordos [6]. Embora objetos cujas superfícies sejam realizações de superfícies variedade-2 no  $E^3$  não pertençam ao domínio da aplicação, isto de modo algum representa uma restrição para os fins de produzir seqüências animadas. Por meio de superfícies com bordos podemos construir "representações" de objetos que sejam visualmente idênticas às suas representações variedade-2. Isto pode ser obtido fazendo com que os polígonos que definem os bordos das superfícies convirjam para pontos.

#### 4. O UNIVERSO DE UMA ANIMAÇÃO

INTERPOL3D utiliza objetos modelados pela composição de patches Bézier bicúbicos, sendo cada patch descrito por uma matriz de 16 pontos de controle [3,7,1].

O universo de uma animação é descrito de forma recursiva, sendo composto de objetos, os quais, por sua vez, podem ser compostos por subobjetos, e assim por diante, até que o usuário decida pela sua concretização como um conjunto de

superfícies (veja figura 4).

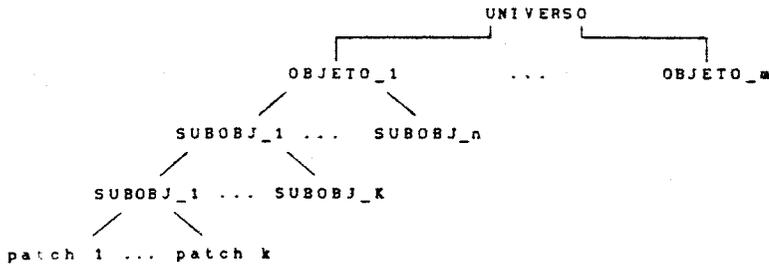


Figura 4. Hierarquia do universo de uma animação INTERPOL3D

## 5. INTERFACE DO SISTEMA

A interface INTERPOL3D é interativa e provê as seguintes opções : definição do número de quadros em que deve se processar a interpolação; indicação dos pares de objetos a serem interpolados e determinação das variações de velocidade durante o processo de interpolação.

### 5.1. Interface para correspondências

OBJETOS ORIGEM	OBJETOS ALVOS
OBJETO_1	OBJETO_1
.. / SUB-OBJ_1	.. / SUB-OBJ_1
... / SUB-OBJ_1	... / SUB-OBJ_1
... / ...	... / ...
... / SUB-OBJ_k	... / SUB-OBJ_k
.. / ...	.. / ...
.. / SUB-OBJ_n	.. / SUB-OBJ_n
...	...
OBJETO_m	OBJETO_m
NULO	NULO

Figura 5. Menu dinâmico para o estabelecimento dos pares de objetos a serem interpolados.

Para a determinação dos pares a interpolar, o sistema provê um menu dinâmico (figura 5), onde são mostradas duas colunas, cada uma contendo a relação de todos os objetos do universo, mais um objeto NULO. NULO permite ao usuário criar um objeto a partir do "nada" ou fazer um objeto "sumir" durante a

interpolação. A primeira coluna do menu corresponde aos objetos origem (elementos de partida para interpolação), enquanto que a segunda corresponde aos objetos alvo (elementos de chegada da interpolação). O procedimento de associar pares de objetos consiste em selecionar elementos terminais (não constituídos por outros sub-objetos) ou NULO da primeira coluna e associá-los a elementos terminais ou NULO da segunda coluna. Cada objeto terminal origem pode ser associado a, no máximo, um objeto terminal alvo (exceção feita ao objeto origem NULO, o qual pode encontrar-se associado à vários objetos terminais alvos), ao passo que um mesmo elemento terminal alvo pode estar associado a vários elementos terminais origem, significando que ao final do procedimento de interpolação teremos várias instâncias do mesmo elemento terminal alvo. O menu apresenta a relação de todos os objetos do universo e não apenas a relação dos objetos terminais, uma vez que um mesmo nome pode ser atribuído a mais de um objeto, desde que os "homônimos" não sejam objetos "irmãos".

## 5.2. Interface para controle de velocidade

Aos pares de objetos a serem interpolados, pode-se associar uma descrição de como se comportará a velocidade de interpolação ao longo do processo. Tal informação é especificada por meio de um gráfico (velocidade x quadros), no qual, o usuário indica o comportamento da velocidade por meio de uma curva Bézier cúbica. Uma curva Bézier cúbica é descrita por 4 pontos de controle [3,7].

O controle da variação de velocidade é realizado definindo-se, interativamente, a curva característica para velocidade de interpolação (como a apresentada na figura 6), enquanto o sistema exibe o número do quadro em que a transformação será concluída. Esta informação corresponde à abscissa do último ponto de controle da curva, relativamente ao eixo dos quadros. Isto permite ao usuário a possibilidade de especificar que pares de objetos podem ter suas interpolações concluídas após, por exemplo,  $n$  quadros, ao passo que um outro par de objetos, somente após  $m$  quadros. O comprimento total do eixo de quadros corresponde ao número total de quadros

especificados para interpolação.

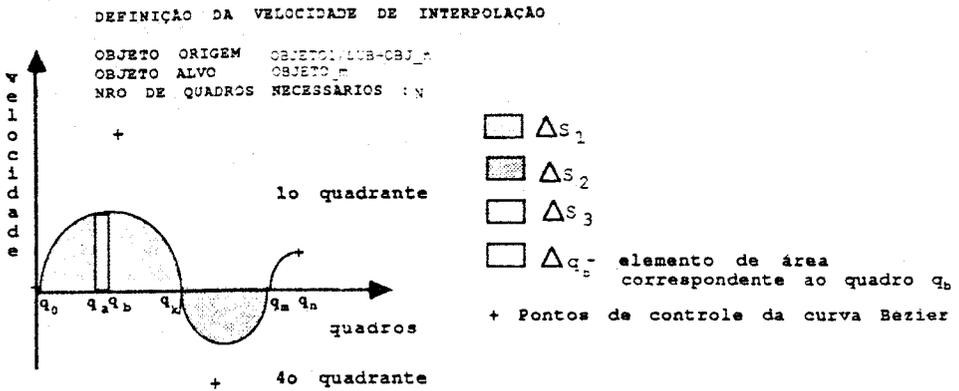


Figura 6. Curva de velocidade para interpolação de um par de objetos ao longo de  $n$  quadros.

Uma vez definida a curva, calcula-se  $\Delta S = \sum_{i=1}^a \Delta S_i - \sum_{j=0}^b \Delta S_j$ , onde:  $\Delta S_i$  corresponde à área do  $i$ -ésimo segmento de curva Bézier localizado no primeiro quadrante e  $\Delta S_j$  corresponde à área do  $j$ -ésimo segmento de curva Bézier localizado no quarto quadrante. O cálculo destas áreas pode ser realizado encontrando-se a representação algébrica (polinômio) da curva a partir de sua representação geométrica (pontos de controle) [7]. A partir do polinômio, os valores de áreas são obtidos por meio de integração paramétrica [5].

Como, em última instância, o algoritmo interpola pontos de controle de *patches* correspondentes, o processo de variação de velocidade realiza-se da seguinte forma: calcula-se, para cada quadro, a razão  $r = s \cdot \Delta q / \Delta S$ , onde  $\Delta q$  representa a área do elemento de área correspondente ao quadro a ser gerado (ver figura 6), e faz-se  $s = 1$  se o elemento de área encontra-se no primeiro quadrante, ou  $s = -1$ , caso contrário. Cada ponto de controle do objeto origem é, então, incrementado de  $r \cdot \Delta p$ , onde  $\Delta p = (\Delta x, \Delta y, \Delta z)$  corresponde à diferença dos pontos  $P_{alvo}(x, y, z) - P_{origem}(x, y, z)$  correspondentes. Isto garante que ao final do procedimento, os pontos do objeto origem coincidam

com os pontos do objeto alvo e que a cada quadro lhes sejam somados incrementos ( $\delta x, \delta y, \delta z$ ) proporcionais a  $\Delta p$  e à área relativa do elemento de área correspondente ao quadro ( $r \cdot \Delta p$ ).

O sistema estabelece como restrição que os pontos de controle da curva característica apresentem abscissas crescentes, de modo que a curva represente uma função velocidade. Observe-se que a técnica exposta acima permite não só controlar a variação da velocidade, como também produzir efeitos interessantes. Pode-se, por exemplo, fazer com que, durante a interpolação, um objeto "pulse". Considere a curva apresentada na figura 6. Suponha que  $\Delta s_1 = 11$ ,  $\Delta s_2 = 2$  e  $\Delta s_3 = 1$ . Deste modo, teremos  $\Delta S = 10$ . Suponha, ainda, que devamos interpolar, entre outros, o seguinte par de pontos:  $P_{origem}(0,0,0)$  e  $P_{alvo}(10,10,10)$ . Portanto,  $\Delta p = (10,10,10)$ . Assim, serão os seguintes os valores de coordenadas associados a instantes da interpolação:

$$q_1 : P_1 = P_0 + \left( \sum_{j=0}^w s_j \cdot \Delta q_j \right) \cdot \Delta p / \Delta S$$

$$q_0 : P_0 = P_0 + ( 0 ) \cdot 10/10 = ( 0, 0, 0 );$$

$$q_k : P_k = P_0 + ( 11 ) \cdot 10/10 = ( 11, 11, 11 );$$

$$q_m : P_m = P_0 + ( 9 ) \cdot 10/10 = ( 9, 9, 9 );$$

$$q_n : P_n = P_0 + ( 10 ) \cdot 10/10 = ( 10, 10, 10 );$$

Simultaneamente à interpolação de formas, realiza-se a interpolação de cores (RGB) dos objetos associados, utilizando-se a mesma técnica. Assim, obtém-se efeitos análogos aos descritos acima, inclusive uma "pulsação" de cores.

## 7. CONCLUSÕES

Este trabalho apresentou um sistema de animação por interpolação de formas baseado em objetos-chaves, que pode ser caracterizado por: manipular representações de objetos no  $E^3$ , permitindo que o sistema determine, automaticamente, as correspondências entre as diversas partes dos objetos envolvidos na animação; permitir um fácil controle sobre as variações de velocidade, uma vez que estas são descritas por curvas Bézier definidas interativamente; permitir a produção de efeitos de

"pulsação" sobre os objetos e suas cores ao longo da interpolação e gerar animações que sejam independentes da posição do observador.

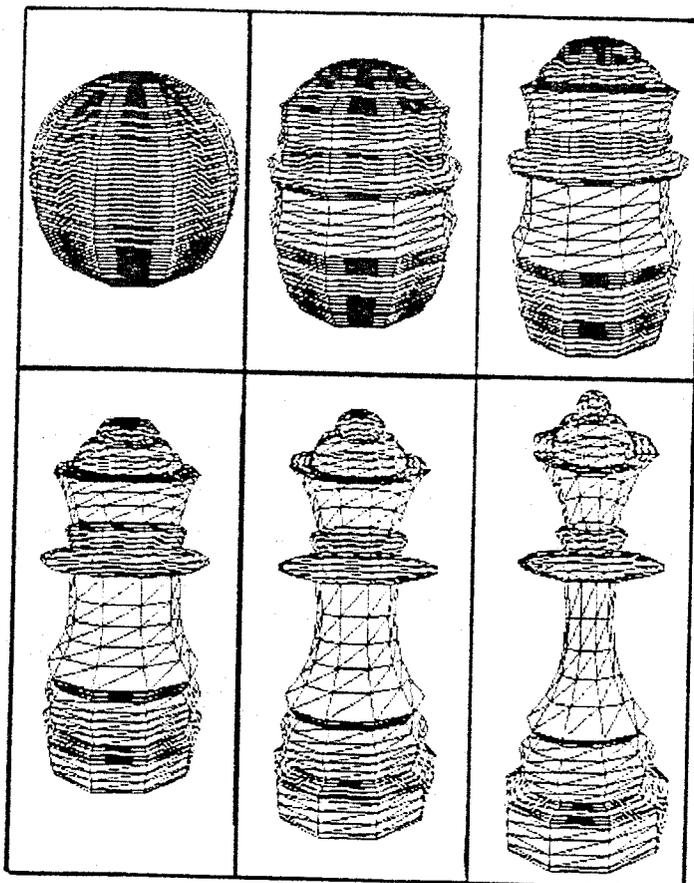


Figura 7. Interpolação de forma entre uma "esfera" e uma rainha do jogo de xadrez. Foi utilizada a técnica de subdivisão com velocidade de interpolação constante.

#### REFERÊNCIAS BIBLIOGRÁFICAS

- [1] BASTOS, R., NASCIMENTO, M. & OLIVEIRA NETO, M. M. BEZIER3D - Sistema para Modelagem e Visualização Tridimensional. Apresentado no WORKSHOP ON GEOMETRIC MODELING, IMPA, 9-11, jan. 1991.

- [2] CATMULL, Edwin. The Problems of Computer-Assisted Animation. In : Proceedings of SIGGRAPH'78, vol. 12, n. 3, aug 1978, pp. 348-353.
- [3] FARIN, Gerald. Curves and Surfaces for Computer Aided Geometric Design : A Practical Guide. 2nd. ed., Academic Press, San Diego, CA, 1990.
- [4] FOLEY, J.; van DAM, A.; FEINER, S. and HUGHES, J. Computer Graphics principles and practice, 2nd. ed., Addison Wesley, New York, 1990.
- [5] GRANVILLE, W.; SMITH, P. and LONGLEY, W. Elements of the Differential and Integral Calculus. The Atheneum Press. Boston, USA, 1941.
- [6] MANTYLA, Martti. An Intoduction to Solid Modeling, Computer Science Press, Rockville, Maryland, 1988.
- [7] MORTENSON, Michael. Geometric Modeling, John Wiley & Sons, New York, 1985.
- [8] REEVES, William. Inbetweening for Computer Animation Utilizing Moving Point Constraints. In : Proceedings of SIGGRAPH'81, vol. 15, n. 3, aug 1981, pp. 263-269.
- [9] THALMANN, N. & THALMANN, D. Computer Animation : Theory and Practice. Springer-Verlag, Tokyo, 1985.

#### AGRADECIMENTOS

Os autores gostariam de expressar seus agradecimentos aos colegas Rui Bastos e Gladimir Baranoski, pelas suas sugestões e incentivo, e Alceu Heinke Frigeri e Benhur Stein, pela edição da seqüência de imagens apresentada na figura 7.