Modelagem e manipulação de objetos de geometria complexa

Luis Antonio Rivera Escriba

rivera@inf.puc-rio.br

Paulo Cezar Pinto Carvalho

pcezar@visgraf.impa.br

Luiz Velho

lvelho@visgraf.impa.br

PUC-RioInf.MCC06/99 January, 1999

Abstract: In this paper, we focus the modeling and manipulation of surfaces of complex object of arbitrary form, with the objetive of simulating the deformation of its surfaces. The object is modeled using closed cubic B-splines having a large number of control points. For the local deformation of the surface it is used a direct manipulation technique applied at a point that is determined by an adequate binary search, which also gives the segment of surface containing the point. In order to produce a realistic deformation of the surface, we explore the multirresolution representation of the object, applying the local deformation through the multirresolution surfaces level.

Keywords: Complex Objet, Arbitrary Form, Close B-splines Curves, Direct Manipulation, Multirresolution Objets, Deformation of Objet, Automatic Deformation

Resumo: Neste artigo, enfoca-se a modelagem e manipulação de superfície de um objeto complexo de forma arbitrária, com o objetivo de simular deformações reais de sua superfície. O objeto é modelado usando-se grande número de pontos de controle para curvas B-splines cúbicas de características fechadas. Para a deformação local da superfície, é usada uma técnica de manipulação direta sobre um ponto da superfície que é determinado por um algoritmo de busca binária adequada, que também fornece o segmento da superfície contendo o ponto. Para se lograr uma deformação real da superfície, explora-se a representação multiresolução do objeto, aplicando-se a deformação local através das superfícies em multiresolução.

Palavras-Chave: Objeto Complexo, Forma Arbitrária, Curvas B-splines Fechadas, Manipulação Direta, Multiresolução de Objetos, Deformação de Objeto, Deformação Automática

Modelagem e manipulação de objetos de geometria complexa

1 Introdução

Entende-se objetos complexos como sendo aqueles que não possuem descrição geométrica através de formas simples. Estes corpos podem ser modelados de várias maneiras: por composição de polígonos, por operações lógicas de primitivas geométricas, etc. Neste trabalho, um objeto complexo 2D será modelado por curvas fechadas B-splines cúbicas, com um grande número de pontos de controle.

Um objeto assim definido pode se tornar ainda mais complexo manipulando-se a sua forma morfológica (deformando-o). Para este objetivo, discutem-se as técnicas de deformação de curvas, tendo como foco a deformação automática de determinados segmentos da curva. Estas deformações podem ser locais ou globais. Para defini-las se usa a manipulação direta de curvas, aplicada sobre sua representação em multiresolucao, de forma a se conseguir uma deformação real, quase física, do objeto.

Para atingir o objetivo, aborda-se na seção 2 a geração de curvas fechadas, discutindo as funções bases adequadas para este caso, e as maneiras mais fáceis de gerar as curvas fechadas para definir um objeto complexo de forma arbitrária. A seguir, na seção 3, discutem-se as formas de deformar segmentos de superfície do objeto, para por fim apresentar a técnica de manipulação direta. Também, se discutem os algoritmos de determinação de ponto no qual se aplica o agente externo deformador. Já na seção 4, enfoca-se a representação multiresolução do objeto, enfatizando sua decomposição e composição em vários níveis para a sua posterior manipulação. Finalmente, na seção 5 tratam-se as deformações da superfície do objeto usando a manipulação direta em multiresolução.

2 Funções base B-splines cúbicas periódicas contínuas

As curvas de bases B-splines apresentadas nas literaturas, como em [RogAda90, PieTil95, Farin90], podem ser, dependendo da distribuição de seu domínio, contínuas ou não contínuas, periódicas ou não periódicas. Entre elas, as curvas afetadas por seus extremos, que são baseadas em bases não periódicas, são mais apropriadas para curvas ou superfícies finitas.

Una superfície fechada não é possível de se obter com análise de extremos; por este motivo, pela natureza de objetos manipulados neste trabalho, se faz referência somente as bases B-splines periódicas e contínuas.

A função base B-spline, $N_i^k(t)$, de ordem k e grau $k-1$, define-se como

$$N_i^1(t) = \begin{cases} 1 & \text{se } x_i \leq t < x_{i+1} \\ 0 & \text{outros} \end{cases}$$

e

$$N_i^k(t) = \frac{(t - x_i).N_i^{k-1}(t)}{x_{i+k-1} - x_i} + \frac{(x_{i+k} - t).N_{i+1}^{k-1}(t)}{x_{i+k} - x_{i+1}}. \quad (1)$$

Os valores x_i, conhecidos como *nós*, são elementos de vetor de nós X, satisfazendo a condição $x_i \leq x_{i+1}$ ($[x_i, x_{i+1})$ é um intervalo de domínio). Quando se constrói uma curva qualquer usando estas funções, o número de nós depende de n (número de pontos de de controle da curva). Assim, o vetor de nós de um intervalo aberto para n pontos de controle teria $n + k$ pontos, $X = [x_0, x_1, x_2, ..., x_{n+k-1}]$. Estes elementos tais que a distância entre elementos consecutivos é constante; em geral são números inteiros positivos incluindo zero.

As funções bases B-spline uniforme periódica não são afetadas pelos elementos extremos ou limites; isto quer dizer que o vetor de nós se considera infinitamente aberto. Cada função é a mesma, só deslocada em uma posição (um nó) sobre o intervalo X.

$$N_{i-1}^k(t-1) = N_i^k(t) = N_{i+1}^k(t+1).$$

A Figura 1 mostra as curvas de base B-spline cúbica uniforme periódica. Em cada intervalo de domínio, a curva base é um segmento de uma curva de ordem 4, sendo que para cada base $N_i^4(t)$, há 4 intervalos de domínio ($[x_i, x_{i+4})$).

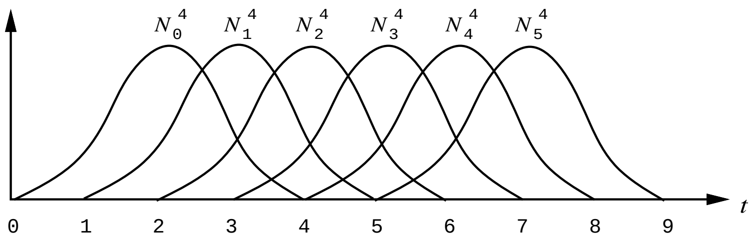


Figura 1: Curvas base B-splines cúbica periódicas.

As funções bases interpoladas linearmente, com os pontos de controle B_i, definem a curva B-splines $f(t)$ de grau $k-1$. Desta forma, cada base B-spline é asociada com um ponto de controle B_i. Assim cada ponto só afeta uma porção da curva limitada pelo domínio $[x_i, x_{i+k})$ da função base $N_i^k(t)$. Tem-se então:

$$f(t) = \sum_{i=0}^{n} B_i.N_i^k(t), \tag{2}$$

onde $f(t)$ é uma função de grau $k-1$ em cada intervalo de domínio $x_i \leq t < x_{i+1}$, já que foi obtida pela interpolação linear das funções base de grau $k-1$.

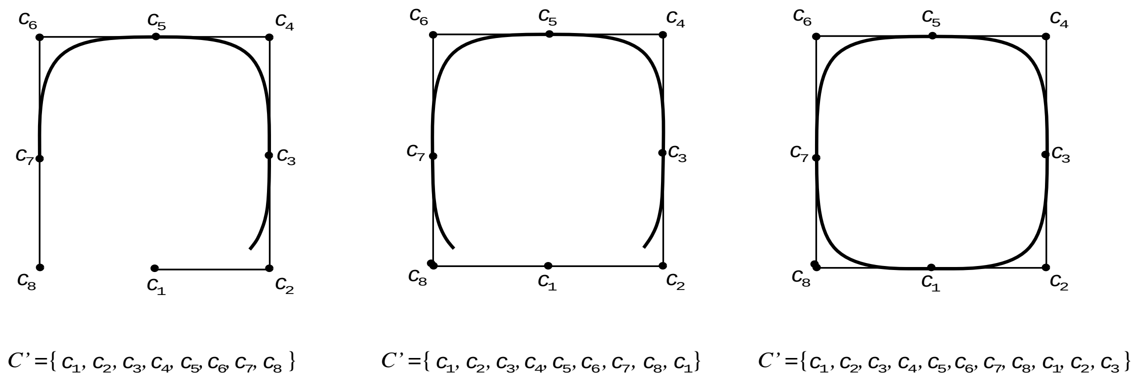
Futuramente, na seção 4.1, para efeitos de análise multiresolução, esta função será denotada por $f^n(t)$ como superfície do objeto refinado, com maior número de pontos de controle, e por $f^j(t)$ para objeto decomposto, que depende do nível j. Em qualquer dos casos, a função $f(t)$ vai depender diretamente dos pontos de controle B_i, segundo a fórmula apresentada acima (expressão (2)).

2.1 Curvas fechadas B-splines

As curvas fechadas B-splines são geradas aproveitando as características das bases contínuas B-splines, já que uma curva fechada é contínua sem tratamentos especiais de limites (não são segmentos finitos). Para isto, são considerados os pontos de controle adicionando-se sobre os n pontos a ocorrência repetida dos $k-1$ primeiros pontos de controle nas últimas posições; ver maior detalhe em [RogAda90]. Assim, uma curva fechada com n pontos de controle, $C = \{c_1, c_2, ..., c_n\}$, é editada com $n+k-1$ pontos de controle $C' = \{c_1, ..., c_n, c_1, ..., c_{k-1}\}$. O vetor de nós X para C' terá $n+k-1+k = n+2k-1$ pontos, $X = [x_0\, x_1 \ldots x_{n+2k-2}]$. O domínio do parâmetro t que gera a curva $f(t)$ varia de $x_{k-1} \leq t < x_{n+k-1}$.

A Figura 2 mostra a seqüência de formação de uma curva fechada, para uma curva B-spline cúbica de oito pontos de controle. Observe-se que a curva fechada é obtida repetindo os primeiros 3 pontos de controle nas últimas posições (imagem (c)). A imagem (b) mostra uma curva aberta gerada com polígono fechado em c_1, enquanto a imagem (a) mostra uma curva aberta gerada por polígono aberto.

Para gerar uma curva fechada $f(t)$ com n pontos de controle, onde n é da ordem de centenas, como na modelagem de objetos complexos não é aceitável usar a expressão (2), porque analisando a distribuição das funções base no vetor X, cada ponto da curva só envolve k funções bases significativas, e o resto das funções tem uma contribuição zero. Essas k funções base são chamadas "funções base relativas ao ponto". Interpolar somente k elementos para calcular um ponto sobre a curva é computacionalmente ótimo, para o propósito deste trabalho, já que vai incluir a manipulação de um grande número de curvas



(a) (b) (c)

Figura 2: Geração de uma curva B-spline cúbica com oito pontos de controle: (a) curva aberta com polígono de controle aberto; (b) curva aberta com polígono de controle fechado; (c) curva fechada.

fechadas, cada curva tendo um grande número de pontos de controle. As mesmas k funções base intervêm no cálculo de todos os pontos de um segmento de curva, $f_j(t)$, determinados pelo parâmetro $t \in [x_{j+k-1}, x_{j+k})$ do domínio. Com este enfoque, a expressão (2) pode-se escrever como

$$f_j(t) = \sum_{i=0}^{k-1} B_i^j N_{i,j}^k(t), \quad \text{com} \quad t \in [x_{j+k-1}, x_{j+k}) \quad \text{para} \quad j = 1, ..., n. \tag{3}$$

A representação matricial, ver em [RogAda90], é outro enfoque que permite calcular a curva rapidamente. Para este propósito, o domínio de parâmetros deve ser reparametrizado, $t \in [0, 1)$. As k funções base relativas aos pontos do segmento da curva $f_j(t)$, também reparametrizadas, são denotadas pela matriz $\tilde{N}^k(t)$ e os k pontos de controle relativos a estas funções estão representados pelo vetor B_j^k, com estes critérios, a expressão (3) é equivalente a

$$f_j(t) = \tilde{N}^k(t).B_j^k, \tag{4}$$

onde $\tilde{N}^k(t) = \frac{1}{(k-1)!}[t^{k-1}\, t^{k-2} \ldots 1][n_{i,r}^k]$ com $0 \leq t < 1$ e para $j = 1, ..., n.$ Os elementos $n_{i,r}^k$ são inteiros, e são obtidos pela fórmula de Cohen e Riesenfeld [RogAda90]. Especificamente para $k = 4$, a matriz $[n^k_{i,r}]$ gerada é

$$[n^4_{i,r}] = \begin{pmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 0 & 3 & 0 \\ 1 & 4 & 1 & 0 \end{pmatrix}, \qquad (5)$$

onde cada coluna r da matriz é formada pelos coeficientes da função base reparametrizada $\tilde{N}^4_r(t)$. A matriz coluna B^4_j, de 4 elementos, teria a forma

$$B^4_j = [c_{j-1}\ c_j\ c_{j+1}\ c_{j+2}]^T .$$

Para efeitos de balanceamento dos pontos de controle em relação ao ponto a se gerar, o segmento de curva $f_j(t)$ é relativo aos pontos de controle c_j e c_{j+1}. Desta forma, para n pontos de controle existem n segmentos de curva, com $0 \leq t < 1$ para cada segmento, e não é necessário criar um vetor de nós de $n + 2k - 1$ elementos que a expressão (3) requer.

É necessário definir as derivadas de curvas B-splines, porque serão usadas posteriormente na manipulação de curvas. A forma matricial também permite calcular, em forma simples, as derivadas em um ponto qualquer da curva. Como uma curva B-spline de grau $k - 1$ é derivável $k - 2$ vezes, então em particular a curva B-spline cúbica é contínua até a segunda derivada.

A primera derivada da curva $f_j(t)$ é formulada como

$$f'_j(t) = \tilde{N}^{k-1}(t).B^{k-1}_j. \qquad (6)$$

Para $k = 4$, as matrizes são:

$$\tilde{N}^3(t) = \frac{1}{2}[t^2\, t\, 1] \begin{pmatrix} 1 & -2 & 1 \\ -2 & 2 & 0 \\ 1 & 1 & 0 \end{pmatrix} \quad \text{e} \quad B^3_j = \begin{pmatrix} c_j - c_{j-1} \\ c_{j+1} - c_j \\ c_{j+2} - c_{j+1} \end{pmatrix}.$$

A segunda derivada é dada por

$$f''_j(t) = \tilde{N}^{k-2}(t)B^{k-2}_j. \qquad (7)$$

Para $k = 4$, as matrizes são:

$$\tilde{N}^2(t) = [t\, 1] \begin{pmatrix} -1 & 1 \\ 1 & 0 \end{pmatrix} \quad \text{e} \quad B^2_j = \begin{pmatrix} c_{j+1} - 2c_j + c_{j-1} \\ c_{j+2} - 2c_{j+1} + c_j \end{pmatrix}.$$

Para maiores detalhes de dedução das derivadas, ver [RogAda90, PieTil95].

Um objeto complexo, bidimensional de forma arbitrária, será definido por uma curva fechada (formulação (4)), sendo esta a superfície do objeto, vide Figura 3.

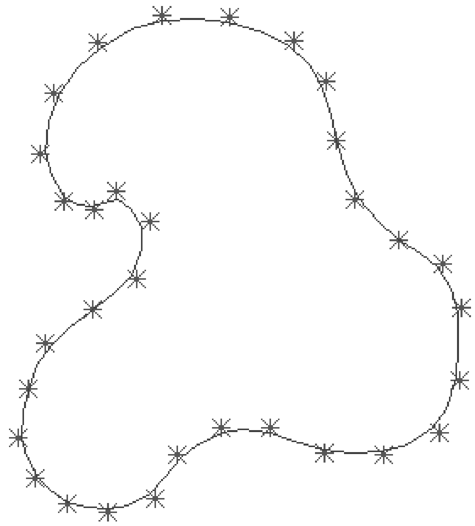


Figura 3: Um objeto complexo limitado por uma curva fechada B-spline cúbica com 32 pontos de controle.

3 Alteração de forma do objeto

Um objeto definido por n pontos de controle pode ter uma forma complexa e arbitrária sem usar operações de primitivas regulares[1]. Este objeto com maior número de pontos de controle definidos por refinamento, explicados posteriormente na seção 4.1.4, permite maior manipulação de sua característica morfológica.

A necesidade de alterar a forma de um objeto é importante, em muitas aplicações. Um caso pode ser na etapa de modelagem dos objetos complexos, de modo que a partir de um objeto inicial pode se obter objetos ainda mais complexos, talvez mantendo alguna característica deste, tal como manter fixo o número de pontos de controle. Uma outra aplicação, que se encaminha neste trabalho, é a deformação automática da forma de um objeto, sob a influência de um agente externo tal como uma força. Com as alterações de forma dos objetos (também, deformação do objeto), estes ainda continuam tendo formas arbitrárias e complexas.

Existem várias abordagens em relação a deformação de curvas e superfícies dadas em [Piegl89, PieTil95, RogFog89, ForBar88], nas que baseam-se na mudança de posição de

[1] Operações lógicas AND, OR e NOT entre primitivas regulares como círculos, quadrados, etc. para se obter objetos complexos.

pontos de controle, alteração (inserção, eliminação e mudança de posição) de elementos de nós, variando o valor de pesos em casos de NURBS [Piegl89, PieTil95], ou combinando estas operações. Qualquer destas operações seria correta para resolver o problema de um designer, de modificar a forma de uma curva para uma outra forma desejada.

Um outro enfoque seria de refinar os pontos de controle do objeto, até que estes estejam sobre a própria curva, e com a ajuda de algoritmo Boor [PieTil95] determinar esse ponto e tratar como se fosse um ponto de controle. Mas este método não é conveniente para o propósito deste trabalho, porque a inserção de nós em subsegmentos de domínio faría que as bases fossem não periódicas.

Para deformações automáticos, como em uma simulação dinâmica de corpos em movimentos, as deformações de corpos (segmentos de superfície) como conseqüência de contatos não poderá ser controlada diretamente modificando os pontos de controle, nem os nós; porque os contatos entre os corpos ocorrem em pontos arbitrários da superfície dos corpos. Para este propósito, é necessário enfocar as deformações de superfícies sub influência de restrições que são abordados, com diferentes enfoques, em [GuLiTa+97, HsHuKa92, LycMor87], ou manipulação direta de curvas e superfícies [BarBea89, BarFow93].

3.1 Manipulação direta da superfície

A forma como o objeto é definido, usando k pontos de controle para cada segmento de curva ($f_j(t)$, pela expressão (4)), permite imaginar que qualquer mudança de posição de um ponto sobre este segmento acarreta o reposicionamento de seus pontos de controle relativos. Este enfoque, abordado por Bartels e Beaty [BarBea89], pode afetar a alteração de todos os segmentos em que participam os k pontos de controle reposicionados, sendo estes $k + (k - 1)$ segmentos. Esta deformação não requer refinar os pontos de controle, nem usa operações nos elementos de nós, senão reposicionar minimamente os pontos de controle relativos ao ponto da curva que se movimenta, de tal forma que permita levar este ponto a sua nova posição. Para atingir este objetivo, calculam-se a variação Δc_{j+i-1} de cada ponto de controle c_{j+i-1} relativo ao segmento j da curva, para os $i = 0, ..., k - 1$ pontos, com a expressão seguinte:

$$\Delta c_{j+i-1} = \Delta f_j(t) \frac{N^k_{j+i-1}(t)}{\sum\limits_{l=0}^{k-1} (N^k_{j+l-1}(t))^2} \quad \text{para } i = 0, ..., k - 1. \tag{8}$$

A nova posição de ponto c_{j+i-1} é dada por

$$c'_{j+i-1} = c_{j+i-1} + \Delta c_{j+i-1}. \tag{9}$$

Note que as bases $N^k_{j+i-1}(t)$ são as mesmas funções definidas pela expressão (1). Cada variação de ponto de controle (Δc_{j+i-1}), para $k = 4$, na expressão (8), é facil de se obter. Depois de atualizadas a novas posicões dos k pontos de controle relativos ao ponto $f_j(t)$, recalcula-se a curva total modificada.

A expressão (8) também é apresentado, como um caso simples, em [BarFow93] considerando que a restrição afeta em um ponto só, dada pela expressão

$$\Delta B = \Delta f_j(t) N^T (N.N^T)^{-1}. \tag{10}$$

Esta expressão é equivalente à expressão (8), onde $\Delta B = [\Delta c_{j-1}\, \Delta c_j\, \Delta c_{j+1}\, \Delta c_{j+2}]$.

A representação matricial, para a equação (8), permite operar com facilidade, porque cada coluna da matriz (expressão (5)) é formada por coeficientes de $N^4_{j+i-1}(t)$ reparametrizado. Assim, teria-se a expressão que relaciona cada função base B-spline reparametrizada da seguinte forma:

$$\begin{bmatrix} N^4_{j-1}(t) \\ N^4_j(t) \\ N^4_{j+1}(t) \\ N^4_{j+2}(t) \end{bmatrix} = \frac{1}{6}[t^3\, t^2\, t\, 1] \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 0 & 3 & 0 \\ 1 & 4 & 1 & 0 \end{bmatrix}. \tag{11}$$

Usando as expressões (8) e (11) para o cálculo da expressão (9), obtem-se a deformação do corpo complexo. A Figura 4 mostra a deformação de três segmentos da superfície de um corpo complexo definido por 32 pontos de controle.

3.1.1 Deformações locais

Um corpo complexo definido por pontos que estão com uma vizinhança mínima (pontos com uma tendência a B-splines interpolantes), não vão permitir uma deformação substancial da superfície, sua ação será simplesmente sobre uma pequena porção desta, causando uma deformação local, tal como mostrado na Figura 5. Uma deformação automática do objeto nem sempre vai ser superficial. Ela pode ser pronunciada, acarretando reposicionamentos de uma vizinhança maior que k pontos de controle. Existem abordagem que tenta de resolver este problema, para manipulação direta de superfícies, tal como [ForBar88] com o refinamento de pontos de controle em hierarquias.

Neste trabalho, a forma de permitir o reposicionamento de maior número do que k pontos de controle, será conseguido utilizando a multiresolução de superfícies, que se discutem na seção 4.1. A manipulação direta será aplicada sobre a curva de resolução menor, e este ao se compor em resoluções maiores permitirá o reposicionamento de maior número do

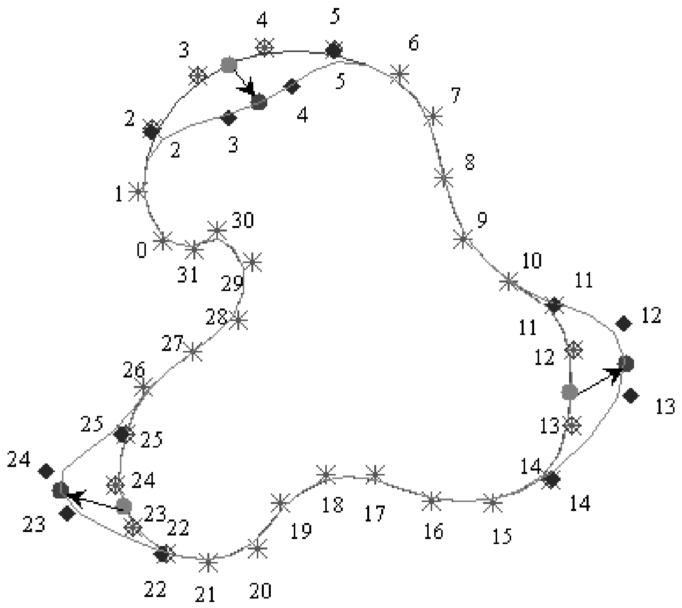


Figura 4: Uma curva fechada B-spline cúbica deformada em três pontos de diferentes segmentos da curva.

que k pontos de controle. Assim, usar a manipulação direta em multiresolução vai fornecer uma série de facilidades como operações locais e globais na deformação das superfícies dos objetos.

3.2 Determinação de um ponto sobre a curva

Um ponto dado deve ser identificado em relação a um segmento da curva para considerar se este pertence ou não à curva. Para este propósito, deve-se supor que o ponto indicado não necessariamente está exatamente sobre a curva, se usa uma tolerância permitida para se considerar que o ponto forme parte da curva (muito próximo). Assim, nesta etapa se determinará se o ponto pode ser considerado como sendo parte da curva, e se for assim, se determinará seu parâmetro correspondente.

Seja $p = (x, y)$ um ponto supostamente sobre a curva $f(t)$, pretende-se determinar o valor de t_p talque $p = f(t_p)$. Como o ponto p não necessariamente está sobre a curva, então será necessário analisar esta aproximação vía "projeção de um ponto sobre uma curva". O valor de t_p será determinado por iteração, avaliando-se em cada iteração a distância Euclid-

Figura 5: Deformação local de um objeto.

iana de sua projeção sobre o vetor tangente da curva no parâmetro (t_i) que se aproxima. O valor do parâmetro t_i, obtido em cada iteração, deve convergir para o parâmetro t_p. Com este parâmetro convergido, se verifica, com uma tolerância permitida, a proximidade do ponto p à curva $f(t)$.

Um ponto p é próximo a uma curva $f(t)$ quando existe um ponto $f(t_p)$ sobre a curva que é a projeção do ponto p. O segmento entre estes dois pontos deve ser perpendicular à tangente da curva no ponto de projeção $f(t_p)$. Este fato é verificado pela projeção do segmento sobre o vetor tangente igual a zero, o que é equivalente a dizer que o coseno entre o segmento e o vetor tangente se aproxima a zero. A magnitude do segmento deve ser próximo a zero. Vide Figura 6.

Para determinar o parâmetro t_p, se seguem dois passos:

- Determinar o segmento da curva ao qual possivelmente pertence o ponto p.

- Calcular o t_i em cada iteração, que finalmente vai convergir em t_p.

O processo de iteração para convergir em t_p pode ser baseado na *iteração de Newton* [PieTil95] ou na *subdivisão binária*. Neste caso, se usará a subdivisão binária por ser mais simples, embora o método de Newton seja de convergência rápida sempre que a heurística usada para calcular o primeiro valor t_i é adequada (como saber se a heurística usada é a adequada?). Os cálculos de restrições para subdivisão binária, formulada neste trabalho, são baseados nos critérios de convergência de Newton, por isto se chamou *subdivisão binária* *adequada*. Por esta razão, se detalha primeiramente, o método de Newton para depois explicar os cálculos da subdivisão binária adequada.

A determinação do segmento da curva relativo ao ponto p vai fornecer o índice j do segmento $f_j(t)$, e por tanto, os pontos de controle relativos a este. O método usado para detetar o índice j se explica na seção 3.2.3 já que se usam algumas formulações do método de Newton da seção 3.2.1.

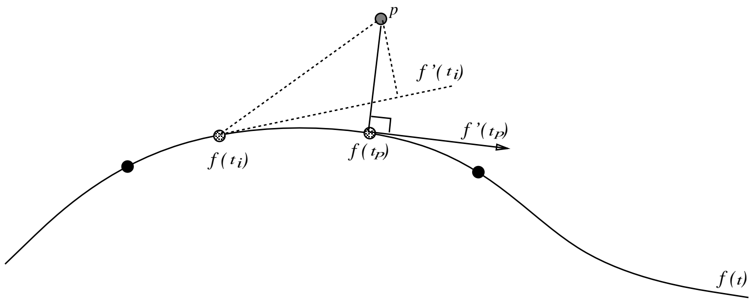


Figura 6: Projeção do ponto p sobre a curva $f(t)$

3.2.1 Método iterativo de Newton

O método iterativo de Newton requer um parâmetro inicial, t_0. Para isto se subdivide o intervalo $[a, b)$, domínio do segmento de curva candidato, em n segmentos iguais $[t_1, ..., t_n]$. Para se obter o valor adequado de n usa-se uma heurística adequada, porque deste cálculo depende a eficiência do método de Newton. Como segundo passo deste método, calculam-se as distâncias de p para cada ponto sobre a curva, $f(t_i^n)$, e o parâmetro t_i^n corresponde ao menor das distâncias que será o dado inicial (t_0) para a iteração.

Tendo em vista que a derivada de f representa o vetor tangente no ponto $f(t)$, a função distância da projeção do ponto p sobre um ponto $f(t)$ da curva é dada por

$$d(t) = f'(t).(f(t) - P). \tag{12}$$

A distância Euclidiana é mínima quando esta projeção é nula ($d(t) = 0$). Em outras palavras, quando p se projeta perpendicularmente sobre o vetor tangente em $f(t)$.

O parâmetro t_{i+1} de cada iteração será calculado por

$$t_{i+1} = t_i - \frac{f'(t_i).(f(t_i) - P)}{f''(t_i).(f(t_i) - P) + |f'(t_i)|^2}. \tag{13}$$

São usados dois indicadores de convergência, como dito anteriormente, (ϵ_1 para tolerância da distância Euclidiana; ϵ_2 para tolerância de cosenos).

Os critérios de convegência são dados por:

- variação insignificante
$$|(t_{i+1} - t_i).f'(t_i)| \leq \epsilon_1, \tag{14}$$

- ponto coincidente
$$|f(t_i) - P| \leq \epsilon_1, \tag{15}$$

- coseno zero
$$\frac{|f'(t_i).(f(t_i) - P)|}{|f'(t_i)|.|f(t_i) - P|} \leq \epsilon_2. \tag{16}$$

A verificação de convergência do método de Newton é na ordem:

1. Verificar ponto coincidente, expressão (15).

2. Verificar coseno zero, expressão (16).

3. Se (2) nao for satisfeito, calcula-se o novo t_{i+1} usando a equação (13), e depois verificar que $t_{i+1} \in [a, b)$, para o caso de curva fechada:

   - se $t_{i+1} < a$ então, $t_{i+1} = b - (a - t_{i+1})$

   - se $t_{i+1} > b$ então, $t_{i+1} = a + (t_{i+1} - b)$

4. O parâmetro não varia significativamente, expressão (14).

5. Se são satisfeitos (1) e (2) ou (4), convergiu a valor $t_p = t_i$.

3.2.2 Método da subdivisão binária adequada

A subdivisão binária, neste caso, é uma adequação do método de Newton. Em lugar de gerar o parâmetro t_i usando a equação (13), calcula-se pela divisão binária.

Para a subdivisão binária adequada, se considera inicialmente o intervalo do segmento de domínio $[a, b)$, que gera o segmento da curva, como segmento inicial $[min, max]$.

A iteração é feita como segue:

1. Calcular o valor médio $m = (min + max)/2$.

2. Verificar ponto coincidência, expressão (15).

3. Verificar coseno zero, expressão (16).

4. Se (3) não for satisfeito, estabelecer o novo intervalo $[min, max]$. Para isto, avaliar a função $d(t)$ da expressão (12) nos pontos min e max, assim:

   se $d(min) < d(max)$, $max = m$

   senão , $min = m$

5. O parâmetro m não varia significativamente, expressão (14)

6. Se são satisfeitos (2) e (3) ou (5), se obteve convergência do valor de $t_p = m$.

7. Quando ocorre (5) e não (2), quer dizer que o ponto não pertence à curva.

3.2.3 Determinação do segmento da curva relativo ao ponto

Uma curva B-spline de ordem k tem a propriedade importante chamada *propriedade fecho convexo forte* [PieTil95, RogAda90], que diz: *Se $t \in [t_i, t_{i+1})$ então $f(t)$ está dentro do fecho convexo gerado pelos pontos de controle relativos a este ponto*.

Em curvas B-splines cúbicos, os pontos de controle relativos ao segmento de curva $f_j(t)$ para $t \in [0, 1)$, segundo a formulação (4), formam um fecho convexo, no qual, o $f_j(t)$ para esses pontos de controle está no seu interior (vide Figura 7). Com este critério, pode-se aproveitar a verificação dos pontos próximos a este segmento de curva, para sua manipulação no processo de *inversão de pontos*[2].

Um ponto próximo à curva cúbica deve pertencer a algum dos fechos convexos determinados por cada quatro pontos de controle consecutivos, mas para a verificação formal deste caso, deve se usar o método de inversão de pontos, especificamente o método de projeção de um ponto sobre uma curva.

O fecho convexo forte vai permitir determinar o índice j do segmento da curva ao qual possivelmente pertence o ponto em análise. Podem existir vários fechos convexos incluindo o ponto p (fechos convexos candidatos), mas o fecho convexo forte é aquele fecho onde a projeção do ponto p à curva é mínima. Para isto, será necessário fazer dois testes para cada segmento relativo a cada fecho convexo candidato.

1. Projeção no interior do segmento da curva $f_j(t)$

   - avaliar coseno em $t = a$ e $t = b$ usando a expressão (16).

[2] inversão de pontos é o processo de calcular o valor do parâmetro t dado um ponto p tal que $f(t) = p$.

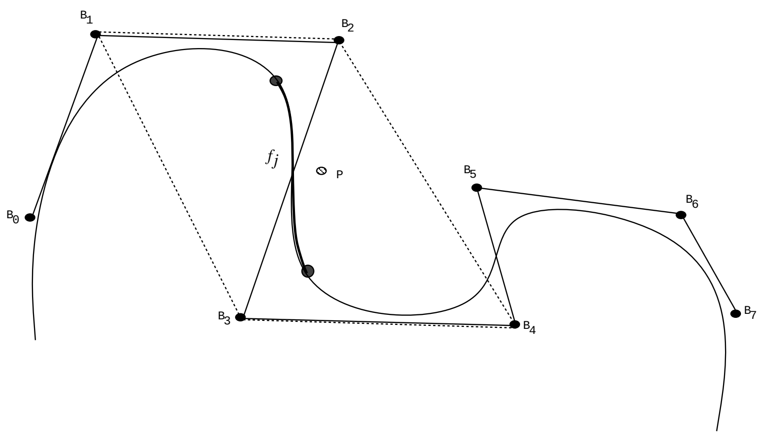


Figura 7: Um fecho convexo forte para curva b-spline cúbica: o segmento f_j incluido no fecho convexo formado por quatro pontos de controle; o ponto p relativo ao segmento.

- se o produto dos cosenos é negativo, então a projeção de p cai no interior do segmento $f_j(t)$. Foi encontrado o índice do segmento $f_j(t)$.

2. Se houver mais de um intervalo cumprindo (1), procurar o intervalo cuja projeção de p para seu interior seja menor. Sempre será um intervalo só que cumpra esta condição, mas se existir mais de um segmento de curva nas proximidades do ponto p, este critério será considerado.

4 Multiresolução do objeto

Seja $f(t)$ a função que define a superfície de um objeto complexo, com 2^n pontos de controle escritas em uma matriz coluna $C^n = [c_0^n \; c_1^n \; c_2^n \; ... \; c_{2^n-1}^n]^T$, onde cada elemento c_s^n, no plano, é um ponto de coordenadas x e y $(c_s^n = (x_s, y_s) \, \epsilon \, R^2)$. Para efeitos de sua representação em multiresolução, esta função, com estas características, e definida como descrita na seção 2.1, será denotada por $f^n(t)$, onde n representa o nível de resolução. Neste caso é a função límite ou a superfície refinada do objeto, enquanto as outras $f^j(t)$, com $j < n$, são representações grosseiras do objeto em versão j. Em outras palavras, f^j é a projeção de f no espaço V^j (a definir na próxima seção).

Um objeto de uma resolução j $(j < n)$ pode ser obtido a partir de seus respectivos coeficientes de escala (pontos de controle), porque os pontos de controle de f^j e os coeficientes de sua diferência $(o^j = f^{j+1} - f^j)$ constituem os coeficientes de escala e os coeficientes de detalhe, respectivamente, do nível j. Desta forma se usará a teoria de wavelets para a representação de um objeto em multiresolução. Na seção a seguir formulam-se, sem entrar em maior detalhe de teoria de wavelets, os cálculos e teorias relevantes para se obter objetos em várias resoluções.

4.1 Decomposição e composição do objeto em varias resoluções

Para se obter uma função $f^{j-1}(t)$, é necessário determinar a matriz coluna C^{j-1} de pontos de controle do objeto a partir de C^j. O filtro para este processo, de levar um objeto de um nível de resolução j para um imediato menor $j-1$, é denominado filtro de análise A^j, que é uma matriz de constantes de dimensão $2^{j-1} \times 2^j$, que é chamado também "matriz de análise de coeficientes".

$$C^{j-1} = A^j C^j \tag{17}$$

Os coeficientes de detalhe, desta resolução, são capturados em uma outra matriz coluna $D^{j-1} = [d_0^{j-1}\ d_1^{j-1}\ d_2^{j-1}\ ...\ d_{2^{j-1}-1}^{j-1}]^T$, como

$$D^{j-1} = B^j C^j. \tag{18}$$

O número de elementos de D^{j-1} é o mesmo de C^{j-1}, ou seja 2^{j-1}. A matriz B^j, de dimensão $2^{j-1} \times 2^j$, é a "matriz de análise de detalhe". As duas matrizes, A^j e B^j, são chamadas "matrizes de filtro de decomposição".

O processo de decomposição de C^j em C^{j-1} e D^{j-1}, para $min \le j \le n$ (sendo min o nível mínimo requerido para manter forma de um objeto mínimo), é aplicado recursivamente sobre C^{j-1}.

Para a composição do objeto a partir dos coeficientes de resolução menor que n, cujo processo é conhecido como transformada inversa, se usarão as "matrizes de filtro de composição" P^j e Q^j, cujas dimensões são $2^j \times 2^{j-1}$, sendo a sua formulação

$$C^j = P^j C^{j-1} + Q^j D^{j-1}. \tag{19}$$

A Figura 8 mostra a decomposição dos coeficientes C^j em C^{j-1} e D^{j-1} e a composição de C^{j-1} e D^{j-1} em C^j.

O número de pontos de controle C^n deve ser potência de 2, já que o processo de decomposição sempre vai gerar metade dos pontos em cada recursão.

As matrizes de filtro A^j, B^j, P^j e Q^j são calculadas associando a função que define $f^j(t)$ com os coeficientes de escala C^j, para $0 \le t \le 1$, tal como definido pela

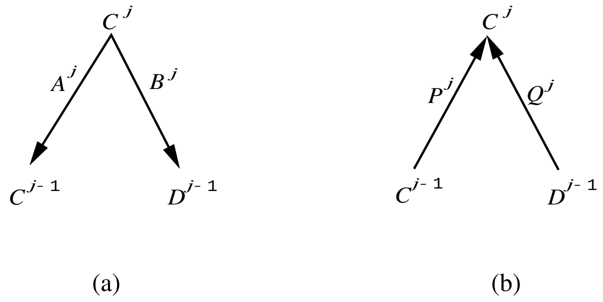


Figura 8: Estrutura de transformação direta e inversa:(a) decomposição; (b) composição.

expressão (4).

$$f^j(t) = \Phi^j(t)C^j \tag{20}$$

O vetor $\Phi^j(t) = [\phi_1^j(t), ..., \phi_{2^j}^j(t)]$ é chamado *função base de escala*. Neste caso, é um vetor das funções B-splines cúbicas definidas na seção 2, que é refinável pela matriz de filtro P^j.

$$\Phi^{j-1}(t) = \Phi^j(t)P^j \tag{21}$$

As bases Φ^j geram o espaço V^j, as funções refinadas Φ^{j-1} geram o espaço V^{j-1} tal que $V^{j-1} \subset V^j$, e assim recursivamente. Do mesmo modo, os espaços complementos W^{j-1}, talque $V^{j-1} \oplus W^{j-1} = V^j$, são gerados por *funções base wavelets* $\Psi^{j-1}(t)$.

$$\Psi^{j-1}(t) = \Phi^j(t)Q^j \tag{22}$$

Q^j é uma matriz de filtro de síntese de wavelets. Com estas formulações obtém-se a expressão que relaciona os quatro filtros A^j, B^j, P^j e Q^j, que finalmente se expressam como

$$\left[\frac{A^j}{B^j}\right] = [P^j|Q^j]^{-1}. \tag{23}$$

Desta forma, dados os valores dos filtros de síntese, é imediato calcular os valores dos filtros de análise. Os filtros P^j e Q^j são calculados definindo o produto interno entre as bases, de forma que V^{j-1} e W^{j-1} sejam ortogonais.

Os espaços $\{V^j\}$, tal que ... $\subset V^{j-1} \subset V^j \subset$ ..., definem multiresolução de f, já que a sua projeção sobre cada V^j é f^j (função de nível j).

Para evitar a manipulação de grandes matrizes, tanto na análise como na síntese, na seguinte seção se estabelecem a relação dos elementos das matrizes de filtros e os coeficientes.

4.1.1 Relação de coeficientes e filtros

Uma função de escala $\phi_s^j(t)$ é definida pelas operações de traslação e dilatação de uma "função mãe" $\phi(t)$, explicadas em diversos trabalhos, entre outros, [StDeSa96, SweSch95, GomVel98], como

$$\phi_s^j(t) = \phi(2^j t - s). \tag{24}$$

A relação de um nível para o outro da função mãe é expressa usando um coeficiente de filtro p_m, como na expressão (21),

$$\phi(t) = \sum_m p_m \phi(2t - m). \tag{25}$$

O processo de refinamento das funções base obtem-se substituindo (25) em (24),

$$\begin{aligned} \phi_s^j(t) &= \sum_m p_m \phi(2(2^j t - s) - m) \\ &= \sum_m p_m \phi(2^{j+1} t - (2s - m)) \\ &= \sum_m p_m \phi_{2s-m}^{j+1}(t). \end{aligned} \tag{26}$$

Fazendo uma mudança nos índices, pode-se re-escrever a expressão acima na forma

$$\phi_s^j(t) = \sum_m p_{m-2s} \phi_m^{j+1}(t). \tag{27}$$

Refinando a função $\psi_s^j(t)$, considerando a relação

$$\psi_s^j(t) = \sum_m q_m \phi(2t - m), \tag{28}$$

e operando da forma parecida acima, chega-se à expressão

$$\begin{aligned} \psi_s^j(t) &= \sum_m q_m \phi_{2s-m}^{j+1}(t) \\ &= \sum_m q_{m-2s} \phi_m^{j+1}(t). \end{aligned} \tag{29}$$

Como a síntese constrói a projeção da função f sobre o um espaço, por exemplo, V^{j+1}, este seria da forma

$$\begin{aligned} f^{j+1} &= f^j + o^j \\ &= \sum_s c_s^j \phi_s^j + \sum_s d_s^j \psi_s^j. \end{aligned}$$

Substituindo pelas expressões(27) e (29), tem-se

$$
\begin{aligned}
f^{j+1} &= \sum_s c_s^j (\sum_m p_{m-2s}\phi_m^{j+1}) + \sum_s d_s^j (\sum_m q_{m-2s}\phi_m^{j+1}) \\
&= \sum_m (\sum_s p_{m-2s}c_s^j)\phi_m^{j+1} + \sum_m (\sum_s q_{m-2s}d_s^j)\phi_m^{j+1} \\
&= \sum_m (\sum_s p_{m-2s}c_s^j + \sum_s q_{m-2s}d_s^j)\phi_m^{j+1} \\
&= \sum_m c_m^{j+1}\phi_m^{j+1}
\end{aligned}
$$

Em conseqüência, obtem-se a relação dos coeficientes de composição, como

$$c_m^{j+1} = \sum_s p_{m-2s}c_s^j + \sum_s q_{m-2s}d_s^j. \tag{30}$$

Critério similar será usado para se obter a relação dos filtros e coeficientes de decomposição (análise) chegando às expressões:

$$c_m^j = \sum_s a_{s-2m}c_m^{j+1} \tag{31}$$

e

$$d_m^j = \sum_s b_{s-2m}c_m^{j+1}. \tag{32}$$

Observe-se que, nos cálculos dos coeficientes c_m^j e d_m^j (decomposição), são capturados elementos relativos a $2m$ de C^{j+1}. Quer dizer de que em cada recursão de decomposição são gerados a metade dos coeficientes de C^{j+1}. Os elementos a, b, p e q são elementos de uma coluna das matrizes A, B, P e Q respectivamente, e não dependem do nível j.

Os valores dos filtros são calculados definindo o produto interno de tal forma que as bases ϕ^j e ψ^j são ortogonais, cumprindo a condicão de espaços de multiresolução $V^j \bigoplus W^j = V^{j+1}$ (definidos na seção anterior). Estes valores podem ser calculados pelo método de Lifting [SweSch95], ou pelo método de domínio de frequências [GomVel98]. Estes algoritmos não são abordados neste trabalho porque não se pretende construir análise multiresolução, mas somente usa-la. Os valores dos filtros usados são apresentados no Anexo A, para transformações wavelets biortogonais que são apresentadas brevemente na seção a seguir.

4.1.2 Multiresolução Biortogonal

Wavelets Biortogonais constituem uma generalização de wavelets ortogonais, apresentados brevemente na seção anterior, porque nem todos os wavelets são ortogonais, nem simétricos. A clase de wavelets biortogonal tem qualidades de wavelets ortogonais e é mais flexível que a ortogonal. Estas características são fundamentais para expressar um objeto complexo em multiresolução, para a manipulação de sua superfície.

Em multiresolução biortogonal, são usadas as bases biortogonais ao invés das ortogonais: uma para análise e outra para a síntese. Isto é, para o cálculo de c_s^{j-1} e d_s^{j-1} são usadas as bases *primais* ϕ_s^j e ψ_s^j, respectivamente, e para síntese são usadas as bases *duais* $\tilde{\phi}_s^j$ e $\tilde{\psi}_s^j$

$$f^{j-1} = \sum_m c_s^{j-1} \tilde{\phi}_s^{j-1} \quad \text{e} \quad o^{j-1} = \sum_m d_s^{j-1} \tilde{\psi}_s^{j-1}.$$

As funções ϕ_s^j e $\tilde{\phi}_s^j$ são bases dos espaços V^j e $\tilde{V}^j$, respectivamente, assim como ψ_h^j e $\tilde{\psi}_s^j$ são bases de W^j e $\tilde{W}^j$. Desta forma, teria-se dois espaços de multiresolução

$$... \subset V^{j-1} \subset V^j \subset ...$$

$$... \subset \tilde{V}^{j-1} \subset \tilde{V}^j \subset ...$$

Com $V^{j-1} + W^{j-1} = V^j$ e $\tilde{V}^{j-1} + \tilde{W}^{j-1} = \tilde{V}^j$.

A ortogonalidade é dada entre primal e dual

$$V^j \perp \tilde{W}^j,$$

$$\tilde{V}^j \perp W^j$$

e

$$W^j \perp \tilde{W}^i, \quad \text{com} \quad j \neq i.$$

Isto implica que seus produtos internos sejam como

$$\langle \tilde{\phi}(t), \psi(t-s) \rangle = 0$$

$$\langle \tilde{\psi}(t), \phi(t-s) \rangle = 0$$

e

$$\langle \tilde{\phi}(t), \phi(t-s) \rangle = \delta_s$$

$$\langle \tilde{\psi}(t), \psi(t-s) \rangle = \delta_s.$$

Pode se estender a

$$\langle \tilde{\phi}_s^j, \phi_s^j \rangle = \delta_{s,m}$$

$$\langle \tilde{\psi}_s^j, \psi_m^i \rangle = \delta_{j,i} \delta_{s,m}.$$

Com estas características de biortogonalidade das bases são relacionados os filtros com as bases, de forma parecida as expressões (25) e (28), como

$$\phi(t) = \sum_s h_s \phi(2t - s) \quad \text{e} \quad \psi(t) = \sum_s g_s \phi(2t - s),$$

similarmente

$$\tilde{\phi}(t) = \sum_s \tilde{h}_s \tilde{\phi}(2t - s) \quad \text{e} \quad \tilde{\psi}(t) = \sum_s \tilde{g}_s \tilde{\phi}(2t - s).$$

Finalmente, combinando estas expressões com os produtos definidos acima, obtem-se a relação dos coeficientes e filtros, para decomposição

$$c_m^j = \sum_s h_{s-2m} c_s^{j+1}, \tag{33}$$

$$d_m^j = \sum_s g_{s-2m} c_s^{j+1} \tag{34}$$

e para a composição

$$c_m^{j+1} = \sum_s \tilde{h}_{m-2s} c_s^j + \sum_s \tilde{g}_{m-2s} d_s^j. \tag{35}$$

Observa-se que, na decomposição, as expressões (31) e (32) com (33) e (34), respectivamente, são idênticas. A mesma característica se observa no processo de composição entre as expressões (30) e (35). Neste sentido, para efeitos de manipulação de decomposição e composição, os coeficientes de filtro são considerados como $a = h$, $b = g$, $p = \tilde{h}$ e $q = \tilde{g}$. Os coeficientes de relação podem ser intercambiados; assim, dados os filtros, calculados em termos de multiresolução biortogonal B-splines cúbico, apresentados no Anexo A, pode se operar sem problema tanto no processo de análise como no síntese, pelo processo de transformada rápida formulada pelas expressões (37) e (38) na secão a seguir. O método de cálculo dos filtros para este tipo de multiresolução é apresentado em [GomVel98].

4.1.3 Pontos de controle de objetos em multiresolução

Considerando que uma curva fechada B-spline é definida por bases B-splines periódicas e contínuas, não existe tratamento de extremos. Cada ponto de uma curva fechada sempre tem pontos vizinhos em ambos os lados, assim não existe um extremo absoluto. Dados os pontos de controle $C = \{c_1, c_2, ..., c_n\}$, os pontos c_1 e c_n (supostamente extremos) são vizinhos. Sendo assim, a seqüência de pontos é cíclica já que sempre um ponto de controle tem vizinho esquerdo e direito, tal como ilustrado pela Figura 9.

Para os efeitos de análise, considera-se os elementos significativos das matrizes de filtro, em que cada coluna é simétrica, e se repetem em cada coluna com um deslocamento em duas posições. As matrizes de filtro de análise seriam da forma

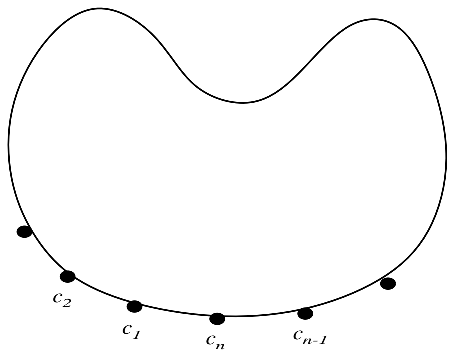


Figura 9: Um objeto com n pontos de controle, tratados em ciclo.

$$A = \left(\begin{array}{ccc} a_{-r} & & \\ \cdot\cdot & & \\ \cdot\cdot & a_{-r} & \\ \cdot\cdot & \cdot\cdot & \\ a_{-1} & \cdot\cdot & \\ a_0 & \cdot\cdot & \\ a_1 & a_{-1} & \\ \cdot\cdot & a_0 & \\ \cdot\cdot & a_1 & \\ \cdot\cdot & \cdot\cdot & \cdot\cdot \\ a_r & \cdot\cdot & \\ & \cdot\cdot & \\ & a_r & \\ & & \end{array} \right) \quad \text{e} \quad B = \left(\begin{array}{ccc} b_{-x} & & \\ \cdot\cdot & & \\ \cdot\cdot & b_{-x} & \\ \cdot\cdot & \cdot\cdot & \\ b_{-1} & \cdot\cdot & \\ b_0 & \cdot\cdot & \\ b_1 & b_{-1} & \\ \cdot\cdot & b_0 & \\ \cdot\cdot & b_1 & \\ \cdot\cdot & \cdot\cdot & \cdot\cdot \\ b_x & \cdot\cdot & \\ & \cdot\cdot & \\ & b_x & \\ & & \end{array} \right).$$

Para poupar espaço de armazenamento das matrizes e sua manipulação nos cálculos de novos pontos de controle, considerou-se só uma coluna significativa (semicoluna) e aproveitou-se sua simetria. Assim, os elementos significativos de A e B são expressos por

$$A_a = \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_r \end{pmatrix} \quad \text{e} \quad B_x = \begin{pmatrix} b_0 \\ b_1 \\ \vdots \\ b_x \end{pmatrix}.$$

O mesmo critério será aplicado para substituir as matrizes de síntese P e Q.

$$P_p = \begin{pmatrix} p_0 \\ p_1 \\ \vdots \\ p_v \end{pmatrix} \quad \text{e} \quad Q_q = \begin{pmatrix} q_0 \\ b_1 \\ \vdots \\ b_w \end{pmatrix}.$$

Com os elementos de filtros definidos nesta seção, a decomposição do objeto é simples, ao invés de operar as expressões matriciais. Dado o vetor de pontos de controle cíclico $C^j = [\ldots\ c^j_{2s-2}\ c^j_{2s-1}\ c^j_{2s}\ c^j_{2s+1}\ c^j_{2s+2}\ \ldots]$, um elemento c^{j-1}_s de C^{j-1} é determinado como

$$c^{j-1}_s = a_0 c^j_{2s} + \sum_{i=1}^{r} a_i (c^j_{2s-i} + c^j_{2s+i}), \tag{36}$$

e um elemento d^{j-1}_s de D^{j-1} é calculado como

$$d^{j-1}_s = b_0 c^j_{2s-1} + \sum_{i=1}^{x} b_i (c^j_{2s-i} + c^j_{2s+i}). \tag{37}$$

Para o processo de composição (reconstrução), de modo análogo à decomposição, pode-se simplificar a operação da expressão (19) em um cálculo linear simples, e adequando a expressão (30) como

$$\begin{aligned} c^j_{2s} &= p_0 c^{j-1}_s + \sum_{i=1}^{v} p_{2i}(c^{j-1}_{s-i} + c^{j-1}_{s+i}) + \sum_{i=0}^{w} q_{2i-1}(d^{j-1}_{s-i} + d^{j-1}_{s+i+1}) \\ &\text{e} \\ c^j_{2s+1} &= \sum_{i=0}^{v} p_{2i-1}(c^{j-1}_{s-i} + c^{j-1}_{s+i+1}) + q_0 d^{j-1}_{s-1} + \sum_{i=1}^{w} q_{2i}(d^{j-1}_{s-i+1} + d^{j-1}_{s+i+1}). \end{aligned} \tag{38}$$

As operações simplificadas, expressadas por (36) e (37) para decomposição, e (38) para a reconstrucão são apresentados em [GomVel98] em forma de um segmento de código em C como a “trasformação rápida de wavelets”. Também se discute o grau de complexidade que é linear em relação ao número dos coeficientes m e o número de elemento do filtro r, sendo O(mr). A Figura 10 mostra a decomposição em dois níveis de um objeto de 32 pontos de controle.

4.1.4 Refinamento dos pontos de controle do objeto

Refinar pontos de controle de um objeto é inserir mais pontos de controle para o mesmo objeto, sem alterar a sua forma. Existem abordagem em curvas B-splines para este problema, via inserção de nós seqüêncial ou paralela [PieTil95]. Mas usar os métodos de reconstrução em multiresolução resulta mais fácil, especificamente para o propósito deste trabalho.

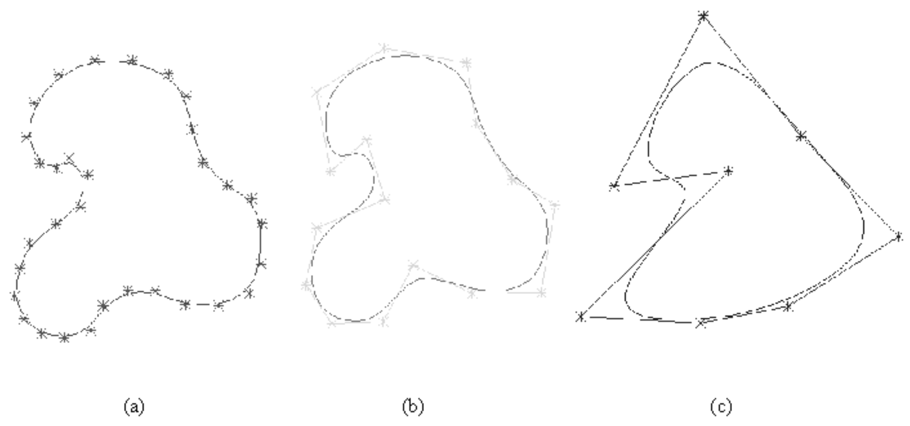


Figura 10: Decomposicão de um objeto em dois níveis: (a) objeto original; (b) objeto decomposto no primeiro nível; (c) objeto decomposto em segundo nível.

O objetivo de adicionar mais vértices do objeto é permitir que estes pontos sejam os mais densos possível e que estejam mais próximos de sua superfície. Esta característica do objeto, em relação a seus vértices, permitirá uma manipulação real de sua morfologia.

Pode-se obter um objeto refinado, sem usar inserção de nós, aproveitando a técnica de composição de objetos em multiresolução, já que a expressão (26) permite o refinamento das bases. Isto é, dada a superfície definida pela curva $f^n(t)$, supostamente refinada, com 2^n pontos de controle, pode-se refinar mais ainda esta curva para $f^{n+1}(t)$ com 2^{n+1} pontos de controle, representando a mesma superfície, para o qual, é necessário considerar o D^n nulo (com elementos zeros). Assim, cada objeto definido no sistema pode ser refinado para o número de pontos desejados.

A Figura 11 mostra o refinamento de um objeto de 32 pontos de controle para ter 64 pontos. A morfologia do objeto continua sendo a mesma.

5 Alteração de forma dos objetos em múltiplas resoluções

Se aproveita a idéia de edição de objetos em multiresolução, comentados brevemente em [FinSal94, KazElb97], para gerar alguma mofidicação ou deformação da forma do objeto fechado em multiresolução. Esta deformação será observada na superfície mais refinada do objeto após de sua reconstrução. Estas alterações podem ser feitas na sua projeção em qualquer nível, sendo as mudanças de posição de determinados pontos de controle que vão

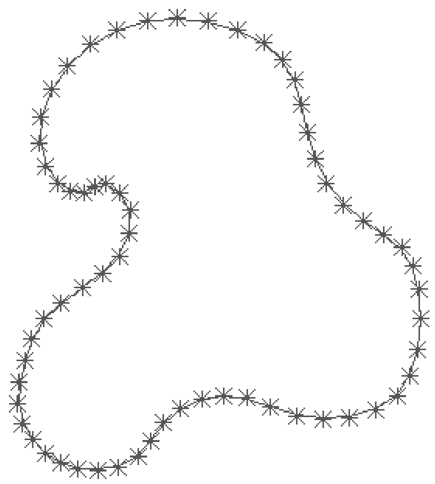


Figura 11: Um objeto inicial de 32 pontos de controle refinado para 64 pontos.

afetar um segmento da superfície.

Seja $f^n(t)$ a superfície refinada do corpo, e seja $f^j(t)$ a projeção de $f^n(t)$ no espaço V^j. Uma alteração de $f^j(t)$ em $\Delta f^j(t)$ vai afetar as versões mais refinadas, sendo consideradas as mudanças na reconstrução, como

$$\tilde{f}^j(t) = f^j(t) + \Delta f^j(t)$$

$$\tilde{f}^{j+1}(t) = \tilde{f}^j(t) + o^j(t)$$

$$\ldots$$

$$\tilde{f}^n(t) = \tilde{f}^{n-1}(t) + o^{n-1}(t).$$

Neste processo, a alteração é feita nos coeficientes C^j, já que as funções bases para os espaços V^j e W^j são inalteráveis. Assim, no processo de reconstrução, as variações se refletem nos coeficientes respectivas $(\tilde{C}^{j+1})$.

$$\tilde{C}^j = C^j + \Delta C^j \quad \text{implica} \quad \tilde{C}^{j+1} = P^{j+1}(C^j + \Delta C^j) + Q^{j+1}(D^j).$$

Suponha-se que ainda existe uma variação nos coeficientes de detalhe em ΔD^j, então

$$\begin{aligned} \tilde{C}^{j+1} &= P^{j+1}C^j + Q^{j+1}D^j + P^{j+1}\Delta C^j + Q^{j+1}\Delta D^j \\ &= C^{j+1} + P^{j+1}\Delta C^j + Q^{j+1}\Delta D^j. \end{aligned} \tag{39}$$

Isto implica que os coeficientes dos níveis superiores variam do seguinto modo:

$$\begin{aligned}
\tilde{C}^{j+2} &= C^{j+2} + P^{j+2}(P^{j+1}\Delta C^j + Q^{j+1}\Delta D^j) \\
\tilde{C}^{j+3} &= C^{j+3} + P^{j+3}P^{j+2}(P^{j+1}\Delta C^j + Q^{j+1}\Delta D^j) \\
&\dots \\
\tilde{C}^{n} &= C^{n} + P^{n}...P^{j+3}P^{j+2}(P^{j+1}\Delta C^j + Q^{j+1}\Delta D^j).
\end{aligned} \tag{40}$$

Finalmente, observa-se que a variação de um ponto no nível j determina o grau da deformação da superfície refinada do objeto. A expressão (40) mostra que, quando j se aproxima a zero, a mudança de um ponto afeta quase toda a superfície, enquanto se j for próximo a n a porção da superfície afetada pela mudança é pequena. Assim, a profundidade do nível manipulado determina o grau de deformação, que podem ser pequenas deformações, ou deformações globais.

5.1 Orientação de vetores de detalhe

Os elementos de detalhe são as diferenças do objeto, que adicionados aos coeficientes do mesmo nível, produzem os coeficientes mais refinados. Tendo em vista que os coeficientes são, neste caso, bidimensionais, cada elemento d_s^j de D^j será tratado como um vetor. A cada vetor d_s^j será associado um Sistema de Coordenada Local (SCL), de tal forma que, qualquer alteração deste vetor, seja escalamento e/ou rotação, por efeito do agente externo que produz a deformação, seja feito em relação a esta coordenada local. Na deformação do corpo, é necessário manter as características de seus detalhes ou controlar suas orientações, para se obter um efeito real da deformação, o qual é dificil de se obter se os detalhes estão expressados em termos de Sistema Referencial Global (SRG). Estes enfoques são sugeridos por [FinSal94, ForBar88, ThuWut97], cada um de uma forma e com um propósito diferente. Para o propósito deste trabalho, operar com os elementos de detalhe vai permitir controlar os efeitos que podem causar um corpo ao se aplicar uma força com uma direção sobre um determinado ponto de sua superfície.

Modificando alguns coeficientes de detalhe em um nível de baixa resolução, que poderia ser em $D^{j-1}(t)$, o afetado seria o coeficiente C^j que representa a superficie de versão mais refinada, segundo a expressão (40).

Uma alternativa para a representação do sistema de referência local de cada vetor d_k^j é usar o método similar ao formulado por Forsey e Bartels [ForBar88] para a representação de detalhe em relação à tangente e normal à curva no nível grosseira $j-1$. Especificamente, para computar a SCL de referência para a orientação de um coeficiente de detalhe d_i^j, usa-se a tangente e normal da curva $f^{j-1}(t)$ na posição do parâmetro t_0 correspondente ao máximo valor de wavelet $\psi_i^j(t_0)$.

Neste trabalho, se consideram as coordenadas do SCL de cada detalhe d_s^{j-1} como sendo o eixo x paralelo à aresta s do polígono de controle de f^{j-1}, formado pelos pontos c_s^{j-1} e c_{s+1}^{j-1}; o eixo y do sistema é o vetor perpendicular externo à aresta s.

A orientação de cada detalhe d_s^{j-1} é calculada no processo de decomposição, sendo armazenados os seus componentes no mesmo espaço ocupado por d_s^{j-1}. O elemento que caracteriza um detalhe é o par $(r_s^{j-1}, \theta_s^{j-1})$, onde r_s^{j-1} é seu módulo e θ_s^{j-1} é o ângulo que este forma com o eixo x de seu sistema local, que são computados por um processo de transformação (mudança de coordenadas), depois de cada cálculo de análise (expressão (37)). Desta forma, a operação de escalamento será feita com o elemento r_s^{j-1} e a orientação adicionando sobre θ_s^{j-1}. Os valores próprios de d_s^{j-1} são recuperados realizando a transformada inversa, antes de aplicação da fórmula de composição (expressão (38)). Este processo é linear no número de pontos de controle, não incrementando assim a complexidade computacional do algoritmo.

Um outro efeito do detalhe expressado em forma polar é manter a coerência de detalhe do objeto quando se deformam segmentos vizinhos a este, tal como sugerido por Stollnitz et al. [StDeSa96]. Em cada deformação(reposicionamento dos coeficientes de escala respectivas) serão atualizadas as orientações dos detalhes afetados.

5.1.1 Manipulação direta em multiresolução

Dado um objeto limitado por $f^n(t)$ de ordem k, o reposicionamento de um número maior do que k de seus pontos de controle vai permitir a deformação de uma porção maior de sua superfície, combinando a manipulação direta descrita na seção 3.1 e a representação do objeto em multiresolução.

Uma deformação local pode acontecer somente com a aplicação da técnica descrita da seção 3.1 sobre $f^n(t)$, que permitirá uma pequena deformação sobre ela; uma deformação um pouco maior pode se conseguir aplicando a deformação local sobre a superfície f^{n-1}, já que ao se reconstruir a superfície f^n um maior número de pontos de controle serão reposicionados. Pode-se aplicar a deformação local em qualquer nível da superície (f^j) e os resultados da deformação serão exibidos nos níveis superiores no momento de reconstrução, tal como apresentado pela expressão (40).

A melhor forma de deformar a superfície do objeto com manipulação direta em múltiplos níveis da curva é aplicar a deformação local no processo de decomposição da curva. Isto é, dado um ponto $f^n(t_i)$ aplicar a deformação local neste nível sendo os pontos de controle obtidos através de $\tilde{C}^n = C^n + \Delta C^n$. O processo de decomposição é aplicado sobre $\tilde{C}^n$, obtendo-se os coeficientes C^{n-1} e D^{n-1}. Para o nível seguinte, deve-se aplicar a deformação local com os coeficientes C^{n-1} obtendo-se $\tilde{C}^{n-1} = C^{n-1} + \Delta C^{n-1}$, decompondo depois em C^{n-2} e D^{n-2}. O processo é repetido recursivamente até o nível j permitido. Este processo permite manter os coeficientes de detalhe respectivos com seus valores própios depois de cada deformação. O objeto reconstruído mostrará os efeitos deste processo na superfície refinada, sendo $\tilde{f}^n(t)$ o resultado final.

$$\begin{array}{rcl} \tilde{f}^n(t) & = & \Phi^n \tilde{C}^n \\ & = & \Phi^n(C^n + \Delta C^n + P^n(\Delta C^{n-1} + P^{n-1}(\Delta C^{n-2} + ... + ...P^{n-j+1}\Delta C^{n-j}...))) \\ & = & f^n + \Delta f^n + \Delta f_n^{n-1} + \Delta f_n^{n-2} + \Delta f_n^{n-3} + ... + \Delta f_n^{n-j} \\ & = & f^n + \Delta \tilde{f}^n \end{array}.$$

A variação final da superfície f^n, influenciada pelas variações dos $j + 1$ níveis de resoluções inferiores, será dada por

$$\Delta \tilde{f}^n = \Delta f^n + \Delta f_n^{n-1} + \Delta f_n^{n-2} + ... + \Delta f_n^{n-j},$$

onde Δf_n^{n-i}, $(i = 1, ..., j)$, é a variação da superfície f^{n-i} projetada no espaço de resolução n (mais refinada).

A Figura 12 mostra a seqüência de estados de deformação de um objeto em seus diversos níveis de representação. Nas imagens (a) e (b) desta figura, observa-se que a deformação total (sem propagar nas múltiplas resoluções), a curva próxima a superfície original, afeta menor segmento que os outros que se são deformações com propagação em resoluções inferiores. Cada linha que se observa na figura, além da primeira linha próxima à curva original, são resultados com coeficientes ($w = 1, 2, 3, 4$ e 5) da função de propagação de deformação, tal como se explicará na seção 5.1.3. Na imagem (c) desta figura observa-se cinco deformações , para coeficientes de função de propagação (w) 1, 2, 3, 4 e 5, de esquerda superior em sentido anti-horário, respectivamente.

5.1.2 Propagação de pontos através dos segmentos de curvas em múltiplas resoluções

As variações ΔC^{n-j} em cada nível de resolução são calculadas em uma vizinhança de pontos de c_h^{n-j}, de forma que exista a relação de propagação de segmentos f_s^n através dos j níveis de resoluções inferiores. Nesta seção, trata-se de relacionar os coeficientes afetados de um nível j para o outro $j - 1$.

Dada a superfície de um objeto definida pela função f^n com 2^n pontos de controle $C^n = \{c_0^n, c_1^n, ..., c_{2^n-1}^n\}$, qualquer ponto p sobre o segmento f_s^n influenciado pelos pontos de controle da matriz $B_s^n = [c_{s-1}^n\, c_s^n\, c_{s+1}^n\, c_{s+2}^n]^T$, converge, segundo a decomposição, para pontos do segmento f_h^{n-1}, com $h = \lfloor s/2 \rfloor$. Este critério vai permitir relacionar que dois segmentos adjacentes, par e ímpar, do nível n possam convergir a um segmento do nível $n-1$. O segmento f_h^{n-1}, como qualquer segmento de curva estabelecido neste trabalho, é influenciado pela matriz de pontos de controle (coeficientes de escala) relativos a este, $B_h^{n-1} = [c_{h-1}^{n-1}\, c_h^{n-1}\, c_{h+1}^{n-1}\, c_{h+2}^{n-1}]^T$. O fecho convexo formado pelos elementos de B_h^{n-1} inclui o f_h^{n-1}.

Como o segmento f_s^j tem como ponto de controle de referência c_s^j, então o segmento de nível inferior $(j-1)$ é f_h^{j-1}, com ponto de controle de referência dado por c_h^{j-1}, onde $h = \lfloor s/2 \rfloor$. Com este critério pode-se propagar a correspondência de segmentos e pontos de controle, recursivamente, através das multiplas resoluções de curvas.

Assim, teria-se, vide Figura 13, que o segmento f_s^{j-1}, com ponto de referência c_s^{j-1}, é o segmento de convergência dos segmentos adjacentes f_{2s}^j e f_{2s+1}^j, cujos pontos de referência são c_{2s}^j e c_{2s+1}^j, respectivamente. A relação entre os pontos de controle de C^j e C^{j-1} foi estabelicida nas seções de decomposição e composição de curvas, onde c_{2s}^j e vizinhança geram c_s^{j-1}.

Para relacionar os coeficientes de detalhe de um nível j com os do nível $j-1$, pode-se deduzir uma expressão análoga à obtida para os coeficientes de escala; qualquer ação que afeta nos pontos dos segmentos f_{2s}^j e f_{2s+1}^j, é transmitido a uma vizinhaça do coeficiente de detalhe d_s^{j-1}, sendo o mais representativo este coeficiente. Como observou-se na dedução de cálculo de coeficientes de detalhe, estes geram um detalhe d_s^{j-1} relacionado ao coeficiente de escala c_{2s+1}^j (ímpar). Em geral, a relação de segmento f_s^j com o coeficiente de detalhe d_h^{j-1} é dada pela expressão $h = \lfloor (s+2)/2 \rfloor$.

Como determinar um ponto p_{s_i} no segmento de curva f_s^{j+1} que converge a um ponto p_{h_m} do segmento de curva f_h^j?. Pode-se obter uma resposta aproximada a esta pergunta seguindo o critério abordado na propagação de segmentos em múltiplos níveis. Como dois segmentos de curvas adjacentes do nível j convergem a um segmento de curva do nível $j-1$, pode-se deduzir que o ponto $p_{h_m} = f_h^j(t_{h_m})$ e vizinhança convergem para um ponto $p_{s_i} = f_s^{j-1}(t_{s_i})$ e vizinhança, sendo estes dois pontos, p_{h_m} e p_{s_i}, representativos de propagação de um nível j para um nível $j-1$, respectivamente.

A relação entre h e s é dada, como explicado na propagação de segmentos, por $s = \lfloor h/2 \rfloor$. O parâmetro t_{h_m} é conhecido, enquanto o parâmetro t_{s_i} será calculado analisando a relação de convergência de segmentos de um nível j para $j-1$. Uma forma de calcular t_{s_i}

é usando a proporção de propagação, já que dois segmentos (par e ímpar) de f^j convergem a um segmento de f^{j-1}. Assim, o parâmetro t_{s_i} será

$$t_{s_i} = t_{h_m}/2 + (h/2 - s), \text{ com } s = \lfloor h/2 \rfloor,$$

porque, em qualquer segmento de curva, o parâmetro t varía de 0 a 1, (isto é, $0 \leq t_{s_i} < 1$). Outra forma de calcular o ponto p_{s_i} seria aplicando um processo de decomposição com os pontos vizinhos de p_{h_m} sobre o segmento f_h^{j+1}, com centro neste ponto. Este último critério é uma consequência de pontos b-splines interpolantes. Já que a vizinhança de pontos usados para este cálculo estão sobre a mesma curva, eles gerarão coeficientes de detalhe nulos correspondentes a essa vizinhança, portanto, o ponto p_{s_i} calculado estará sobre a curva de nível inferior. Com este critério, o parâmetro t_{s_i} será determinado pela projeção de p_{s_i} sobre a curva, usando-se algúm método de iteração (seja o método de Newton, partição binária adequada, ou outros) sobre o segmento f_h^j e seus pontos de controle relativos, segundo os passos de determinação de um ponto sobre a curva (seção 3.2).

5.1.3 Função de propagação

Um vetor de deslocamento $\Delta\mathbf{d}$, que indica o deslocamento de um ponto da curva de sua posição inicial até o final, pode ser distribuido ou propagado nos níveis inferiores da curva em porções determinadas pela função de distribuição. Esta função de distribuição, também conhecida por filtro, terá uma forma de função exponencial, que vai determinar uma área igual a 1 no intervalo 0 a 1. Vide Figura 14, onde a área sombreada é o sector de filtro, que divididos em m segmentos verticais da mesma base, formam o filtro para determinar a porção de $\Delta\mathbf{d}$ a se aplicar em um determinado nível de resolução da curva.

Os m segmentos verticais da área somam 1, de forma que a soma de todas as porções distribuidas em todos os níveis de resolução da curva somam o total ($\Delta\mathbf{d}$). O valor m é o número de níveis que a curva $f(t)$ se descompõe para se aplicar a manipulação direta em cada nível.

Se a_i é a área do segmento vertical do filtro, o deslocamento aplicado no nível i da curva será dado por

$$\Delta\mathbf{d_i} = a_i \Delta\mathbf{d}.$$

Seja $g(t)$ a função de distribuição do vetor deslocamento, tal que

$$\int_0^1 g(t)dt = 1.$$

Então, a função $g(t)$ tem a forma de uma função normalizada $h(t)$, como

$$g(t) = \frac{h(t)}{\int_0^1 h(t)dt}.$$

A função $h(t)$ é da forma exponencial negativa $h(t) = e^{-wt}$, sendo a constante w que determina o grau de decaimento da curva. Assim, para $w = 1$ o segmento de área correspondente ao último nível de resolução da curva será um pouco maior que para $w > 1$.

A função $g(t)$, con estas restrições, será

$$g(t) = \frac{we^{-wt}}{1 - e^{-w}}, \tag{41}$$

porque

$$\int_0^1 h(t)dt = \int_0^1 e^{-wt}dt = \frac{1 - e^{-w}}{w}.$$

Para determinar o valor da porção vertical a_i, será calculada

$$a_i = \int_{t_i}^{t_{i+1}} g(t)dt,$$

onde o intervalo $[0, 1]$ é subdividida em m subintervalos iguais, já que cada subintervalo será asinalada para um nível de resolução inteira da curva. Com este critério, se considera que a_0 será a porcão correspondente ao nível de resolução n (mais refinada) da curva, a_1 corresponde ao nível de resolução $n - 1$, e assim sucessivamente.

6 Conclusões e recomendações

Foi abordada, em uma primeira etapa, a modelagem de curvas fechadas, para serem consideradas como superfícies de corpos de formas arbitrárias, baseada nas bases B-splines de ordem k. Especificamente foram usadas B-splines cúbicas, com características contínuas e periódicas, que permitem uma maior flexibilidade para a geração da curvas fechadas.

A distribuição de pontos de controle, que algumas vezes chamou-se de vértices de polígono de controle da curva fechada, determina a característica complexa do objeto. Estes pontos, na etapa de modelagem, são reposicionados mediante a manipulação direta de segmentos de curva, gerando desta forma um objeto ainda mais complexo. O reposicionamento de quatro pontos de controle, no caso de B-spline cúbico, afeta somente a deformação local do objeto. Para deformações maiores, foi necessário manipular o objeto em multiresolução.

O agente externo que produz a deformação da superfície do corpo, no caso de modelagem, poderia ser o *mouse*, que tem um ponto inicial de aplicação e uma magnitude com uma orientação (pode ser uma força, na deformação automática), que deve produzir um efeito de deformação real e restringida da superfície. Este efeito é conseguido com a ajuda dos coeficientes de detalhe do processo de multiresolução do objeto. A orientação dos coeficientes de detalhe permite a restrição de uma deformação orientada na superfície do objeto, e também permite controlar a deformação local e global do corpo, ao se aplicar a manipulação direta da superfície em um nível de resolução dado.

As técnicas de modelagem e deformação, descritas neste trabalho, podem ser aplicadas diretamente na análise de deformação automática dos corpos como conseqüência de choques em uma animação destes. Para isto, se teria que adicionar a cada corpo a parte dinâmica, tais como a massa, momentos de inércia, etc. Desta forma, seria possível simular a deformação dos corpos, que na realidade não seriam efeitos puramentes físicos, senão quase físicos. Esta linha é o objetivo a ser alcançado futuramente.

Anexo A

As Semicolunas das matrizes de filtro, para multiresolução dos objetos, obtida pela análise de multiresolução biortogonal para B-splines cúbicas, se apresentam nesta seção. A relação de Primal e dual denota-se por um par $(primal, dual)$.

Para primal 4,

$$Pq = \begin{bmatrix} 0.375 \\ 0.25 \\ 0.0625 \end{bmatrix} \quad \text{e} \quad Bb = \begin{bmatrix} 0.375 \\ -0.25 \\ 0.0625 \end{bmatrix}.$$

Para dual 2,

$$Qq = \begin{bmatrix} 1.25 \\ -0.15625 \\ -0.375 \\ -0.09375 \end{bmatrix} \quad \text{e} \quad Aa = \begin{bmatrix} 1.25 \\ 0.15625 \\ -0.375 \\ 0.09375 \end{bmatrix}.$$

Para dual 4,

$$Qq = \begin{bmatrix} 1.09375 \\ -0.273437 \\ -0.375 \\ 0.00390625 \\ 0.078125 \\ 0.0195312 \end{bmatrix} \quad \text{e} \quad Aa = \begin{bmatrix} 1.09375 \\ 0.273437 \\ -0.375 \\ -0.00390625 \\ 0.078125 \\ -0.0195312 \end{bmatrix}.$$

Para dual 6,

$$Qq = \begin{bmatrix} 1.02539 \\ -0.320435 \\ -0.35791 \\ 0.0679932 \\ 0.112305 \\ 0.00671387 \\ -0.0170898 \\ -0.00427246 \end{bmatrix} \quad \text{e} \quad Aa = \begin{bmatrix} 1.02539 \\ 0.320435 \\ -0.35791 \\ -0.0679932 \\ 0.112305 \\ -0.00671387 \\ -0.0170898 \\ 0.00427246 \end{bmatrix}.$$

Para dual 8,

$$Qq = \begin{bmatrix} 0.986938 \\ -0.345428 \\ -0.342529 \\ 0.110291 \\ 0.127686 \\ -0.0125122 \\ -0.0324707 \\ -0.00331116 \\ 0.00384521 \\ 0.000961304 \end{bmatrix} \quad \text{e} \quad Aa = \begin{bmatrix} 0.986938 \\ 0.345428 \\ -0.342529 \\ -0.110291 \\ 0.127686 \\ 0.0125122 \\ -0.0324707 \\ 0.00331116 \\ 0.00384521 \\ -0.000961304 \end{bmatrix}.$$

Para dual 12,

$$Qq = \begin{bmatrix} 0.945082 \\ -0.371282 \\ -0.320374 \\ 0.161443 \\ 0.137804 \\ -0.0478584 \\ -0.0521088 \\ 0.00775886 \\ 0.014451 \\ 0.000261068 \\ -0.0025177 \\ -0.000373721 \\ 0.000204563 \\ 0.0000511408 \end{bmatrix} \quad \text{e} \quad Aa = \begin{bmatrix} 0.945082 \\ 0.371282 \\ -0.320374 \\ -0.161443 \\ 0.137804 \\ 0.0478584 \\ -0.0521088 \\ -0.00775886 \\ 0.014451 \\ -0.000261068 \\ -0.0025177 \\ 0.000373721 \\ 0.000204563 \\ -0.0000511408 \end{bmatrix}.$$

Nos processos de testes, na implementação, foram usados filtros (4, 8).

Referências

[BarBea89] R. Bartels e J. Beatty. A Technique for the Direct Manipulation of Spline Curves. *Graphics Interface*, pp. 33-39, 1989.

[BarFow93] R. Bartels e B. Fowler. Constraint Based Curve Manipulation. *IEEE Computer Graphics and Applications*, pp. 43-49, sept. 1993.

[Farin90] G. Farin. Curves and Surfaces for Computer Aided Geometric Design: A prectical guide. *Academic Press, inc., San Diego*, 2da ed., 1990, 444 pgs.

[FinSal94] A. Finkelstein e D. Salesin. Multiresolution Curves. *Computer Graphics Proceedings*, SIGGRAPH 1994.

[ForBar88] D. Forsey e R. Bartels. Hierarchical B-spline refinement. *Computer Graphics*, 22(4):205-212, 1988.

[GomVel98] J. Gomes e L. Velho. From Fourier Analysis to Wavalets. *Course Notes (próximo)*, SIGGRAPH'98.

[GuLiTa+97] Z. Guan, J. Ling, N. Tao, X. Ping e T. Rongxi. Study and Application of Physics-Based Deformable Curves and Surfaces. *Computer Graphics*, vol 21, num 3, pp. 305-313, 1997.

[HsHuKa92] W. Hsu, J. Hughes e H. Kaufman. Direct Manipulation of Free-Form Deformations. *Computer Graphics Proceedings*, SIGGRAPH 1992.

[KazElb97] R. Kazinnik e G. Elber. Orthogonal Decomposition of Non-Uniform Bspline Space using Wavelets. *Eurographics'97*, pp. c26-c38, vol 16, num 3, 1997.

[LycMor87] T. Lyche e K. Morken. Knot removal for parametric B-spline curves and surfaces. *Computer Aided Geometric Design*, 4(1987) 217-231.

[Piegl89] L. Piegl e N. Fog. Constrained B-spline curve and surfece fitting. *Computer Aided design (CAD)*, v. 21, No. 10, pp. 641-648, 1989.

[PieTil95] L. Piegl e W. Tiller. The NURBS Book. *Springer Editions, Berlin, New York*, 1995, 646 pgs.

[RogAda90] D. Rogers e J. Adams. Mathematical Elements for Computer Graphics. *McGraw-Hill International Editions*, 2da ed., 1990, 611 pgs.

[RogFog89] D. Rogers. Modifying the shape of rational B-splines. Part 1: curves. *Computer Aided design (CAD)*, v. 21, No. 8, pp. 509-518, 1989.

[StDeSa96] E. Stollnitz, T. Derose e D. Salesin. Wavelets for Computer Graphics: Theory and applications. *Morgan Kauffmann Publishers, inc.*, 1996, 245 pgs.

[SweSch95] W. Sweldens e P. Schroder. Building Yor Own Wavelets at Home. *Technical Report 1995:5*, Department od Mathematics, Univ. of South Carolina, 1995.

[ThuWut97] G. Thurmer e A. Wuthrich. Normal Computation for Discrete Surfaces in 3D Space. *Eurographics'97*, vol 16, num 3, 1997.

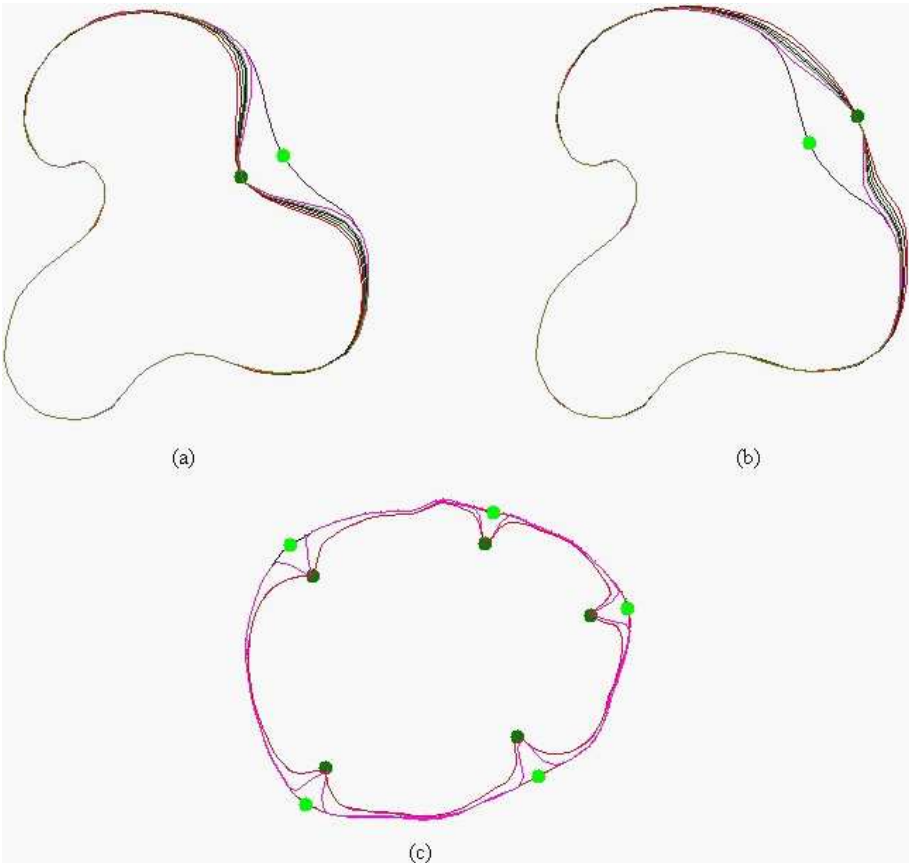


Figura 12: Um objeto deformado em vários níveis de resolução em diferentes graus: (a) propagação da deformação do exterior para o interior, a camada mais interior com coeficiente $w = 1$; (b) propagação de deformação de interior para o exterior, a camada mais exterior com coeficiente $w = 1$; (c) deformação do objeto com 5 tipos de distribuição: esquerda superior $w = 1$, esquerda inferior $w = 2$, seguiente nesta direção $w = 3$, seguiente $w = 4$, e superior direita $w = 5$.

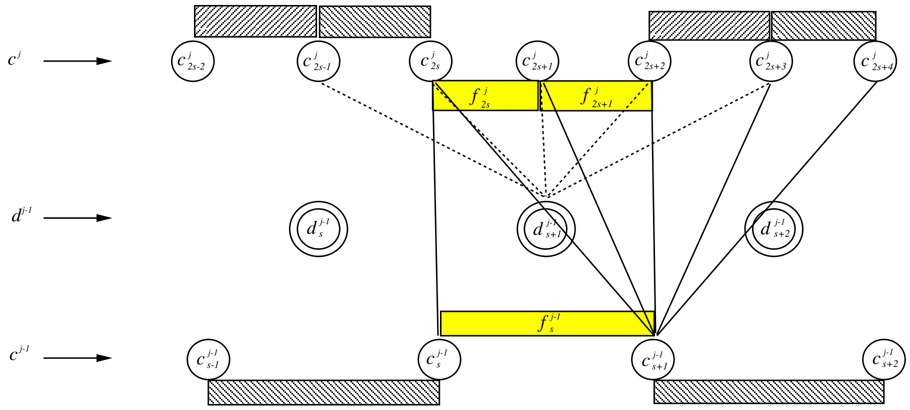


Figura 13: Convergência de segmentos de curva de resolução j para resolução $j-1$, e a relação de seus coeficientes.

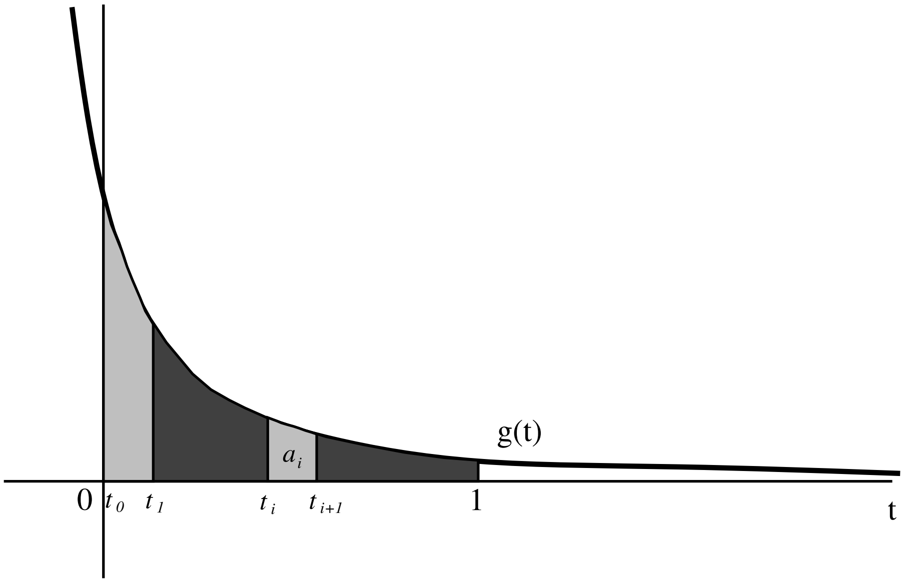


Figura 14: Função de distribuição do vetor de desfase nos níveis de resoluções inferiores da curva.