

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL  
INSTITUTO DE INFORMÁTICA  
CURSO DE PÓS GRADUAÇÃO EM CIÊNCIAS DA COMPUTAÇÃO

ANÁLISE DE UM MODELADOR DE OBJETOS  
QUE UTILIZA SUPERFÍCIES DE JUNÇÃO  
PARA UNIR SUPERFÍCIES PARAMÉTRICAS  
COMPOSTAS BETA-SPLINE

por

BERNARDO COPSTEIN

Dissertação submetida como requisito parcial para  
a obtenção do grau de Mestre em  
Ciência da Computação

Prof. Anatólio Laschuk  
Orientador



SABi



Porto Alegre, julho de 1992.

UFRGS  
INSTITUTO DE INFORMÁTICA  
BIBLIOTECA

## CIP - CATALOGAÇÃO NA PUBLICAÇÃO

COPSTEIN, Bernardo

Análise de um modelador de objetos que utiliza superfícies de junção para unir superfícies compostas beta-spline. /Bernardo Copstein. - Porto Alegre: CPGCC da UFRGS, 1992.

Dissertação (mestrado) - Universidade Federal do Rio Grande do Sul, Curso de Pós-Graduação em Ciência da Computação, Porto Alegre, 1992. Orientador: Laschuk, Anatólio

Dissertação: Computação Gráfica, Modelagem geométrica, Superfícies paramétricas, Superfícies de Junção

**AGRADECIMENTOS**

Gostaria de agradecer em especial aquelas pessoas cuja contribuição foi fundamental para a conclusão deste trabalho.

Meu orientador, Prof. Anatólio Laschuk, por ter me ensinado a acreditar em minhas próprias idéias.

Meu amigo e colega, João Batista de Oliveira, por ser uma fonte de idéias sempre a disposição.

A professora Beatriz Franciosi pelas valiosas sugestões invariavelmente bem humoradas.

Meu sócio, Márcio S. Pinho, pelas consultas telefônicas.

Meus pais, Raphael e Gisela Copstein, pelo incentivo e compreensão.

A Ana Elisa Neujahr pelo carinho e amor.

## SUMÁRIO

LISTA DE FIGURAS.....	008
RESUMO.....	012
ABSTRACT.....	015
1 INTRODUÇÃO.....	017
1.1 Motivação.....	017
1.2 Objetivos.....	020
1.3 Limitações.....	021
1.4 Organização Geral do trabalho.....	023
2 FUNDAMENTOS GEOMÉTRICOS E MATEMÁTICOS.....	025
2.1 Representações paramétricas.....	025
2.1.1 Equações paramétricas .....	026
2.1.2 Curvas .....	027
2.1.3 Superfícies.....	029
2.1.4 Curvas em superfícies.....	032
2.1.5 Notação.....	035
2.2 Superfícies Hermite.....	036
2.3 Continuidade geométrica.....	039
2.4 Superfícies "beta-spline".....	041
3 O MÉTODO PROPOSTO.....	049
3.1 Uma revisão sobre o assunto.....	049

3.1.1	Determinação de superfícies de junção quando a curva de ancoragem possui uma equação conhecida ou de fácil determinação.....	054
3.1.2	Determinação de superfícies de junção quando as curvas de ancoragem são desconhecidas ou de difícil determinação .....	064
3.1.3	Conclusões .....	068
3.2	O método proposto.....	069
4	DETERMINAÇÃO DA CURVA DE ANCORAGEM.....	073
4.1	O que são curvas de ancoragem.....	073
4.2	Representação de uma curva de ancoragem.....	074
4.3	Classificação dos algoritmos de determinação de de intersecções entre superfícies.....	080
4.4	O método utilizado.....	084
4.4.1	O algoritmo de movimentação sobre a curva.....	085
4.4.2	Simplificações.....	090
4.4.3	Determinação dos pontos de partida para o algoritmo de movimentação.....	091
4.4.4	Cálculo da intersecção entre uma curva e o plano $z=0$ .....	098
4.4.5	Determinação dos vetores tangentes à curva de intersecção.....	102
4.4.6	Movimentação no espaço paramétrico.....	105

4.4.7	Cálculo do passo de movimentação d.....	107
4.4.8	Cálculo dos vetores tangentes no espaço pa- ramétrico.....	110
4.4.9	Determinação da isoparamétrica de refinamento.....	112
4.4.10	Condição de fim de execução do processo.....	113
4.4.11	Generalização para uma superfície composta.....	116
4.4.12	Teste do envelope poliédrico.....	120
4.5	Restrições.....	120
4.6	Simplificação da amostragem da curva.....	123
5	<b>CÁLCULO DOS VETORES DE CONTROLE E GERAÇÃO DA SUPERFÍCIE DE JUNÇÃO.....</b>	<b>124</b>
5.1	Reparametrização das curvas de ancoragem.....	124
5.2	Determinação dos vetores de controle.....	127
5.3	Geração das superfícies de junção.....	132
6.	<b>ALGORITMO DE GERAÇÃO DE OBJETOS USANDO SUPERFÍCIES DE JUNÇÃO.....</b>	<b>137</b>
7	<b>O PROTÓTIPO.....</b>	<b>144</b>
7.1	A arquitetura do sistema.....	144
7.2	O módulo de modelagem de superfícies base.....	149
7.3	O módulo de definição de curvas de ancoragem.....	150
7.4	O módulo de geração de superfícies de junção.....	159
7.5	Observações.....	164

<b>8 ANÁLISE DE DESEMPENHO.....</b>	<b>167</b>
<b>8.1 Identificação das etapas analisadas.....</b>	<b>167</b>
<b>8.2 Análise do módulo de cálculo das superfícies de ancoragem.....</b>	<b>168</b>
<b>8.2.1 Análise do método de determinação de curvas de intersecção entre um plano e uma superfície-base.....</b>	<b>169</b>
<b>8.2.2 Análise do cálculo dos vetores de controle.....</b>	<b>176</b>
<b>8.3 Análise do Módulo de Geração de superfícies de junção.....</b>	<b>178</b>
<b>8.4 Análise comparativa.....</b>	<b>182</b>
<b>9 CONCLUSÃO.....</b>	<b>191</b>
<b>BIBLIOGRAFIA.....</b>	<b>194</b>

## LISTA DE FIGURAS

Figura 1.1 Suavização de objetos usando superfícies de junção.....	019
Figura 2.1 Curvas e vetores de um "patch".....	033
Figura 2.2 Espaço paramétrico de uma curva .....	034
Figura 2.3 Curva Hermite .....	037
Figura 2.4 Superfície Hermite .....	038
Figura 2.5 "Patch Beta-spline" .....	041
Figura 2.6 Segmento de curva "B-spline" .....	043
Figura 3.1 Superfície de junção .....	050
Figura 3.2 Determinação de uma superfície de junção pelo rolar de uma esfera no vale formado pela intersecção deduas superfícies .....	052
Figura 3.3 "Patch" de Coons .....	057
Figura 3.4 Quatro curvas delimitando um "patch" de Coons .....	058
Figura 3.5 Superfície de junção gerada utilizando-se um "patch"de Coons .....	064
Figura 4.1 Curva de ancoragem gerada pelo mapeamento de uma curva definida em um espaço paramétrico em uma superfície paramétrica .....	075
Figura 4.2 Deformação decorrente do mapeamento de	



	uma curva definida no espaço paramétrico sobre uma superfície.....	077
Figura 4.3	Controle do grau de amostragem de uma curva pela diferença de ângulo entre os vetores tangentes .....	086
Figura 4.4	Curva que intersecciona as bordas do "patch" .....	096
Figura 4.5	Curva que não intersecciona as bordas do "patch" .....	096
Figura 4.6	Pontos candidatos a pontos iniciais de segmentos de uma curva descontínua gerada pela intersecção de um "patch" com um plano de corte .....	097
Figura 4.7	Problemas do método de Newton .....	099
Figura 4.8	Obtenção do vetor tangente à curva de ancoragem .....	103
Figura 4.9	Deslocamento no espaço paramétrico .....	106
Figura 4.10	Aproximação dos vetores tangentes no espaço paramétrico .....	111
Figura 4.11	Curva determinada pela fixação de uma das componentes de um ponto aproximado .....	114
Figura 4.12	Situação onde é necessária a troca de curva isoparamétrica durante o deslocamento .....	115
Figura 4.13	Influência da localização do ponto inicial	

sobre o algoritmo de movimentação .....	117
Figura 4.14 Situação onde há necessidade de orientação dos segmentos de curva .....	119
Figura 4.15 Caso em que a intersecção entre o plano e a superfície gera mais de uma curva .....	121
Figura 5.1 Vetores usados no cálculo dos vetores de controle pelo método de Filip .....	129
Figura 5.2 Obtenção dos vetores de controle .....	133
Figura 5.3 Geração da superfície de junção .....	134
Figura 5.4 Controle de forma da superfície de junção ...	135
Figura 5.5 Superfície de junção que se auto-intercepta e seus vetores de controle .....	136
Figura 6.1 Escolha das superfícies-base .....	139
Figura 6.2 Determinação dos planos de corte .....	140
Figura 6.3 Instanciamento das superfícies-base e ajuste dos vetores de controle .....	141
Figura 6.4 Geração da superfície de junção .....	142
Figura 6.5 Conformação da superfície de junção de acordo com o posicionamento das superfícies-base .....	143
Figura 7.1 Arquitetura do sistema .....	145
Figura 7.2 Formato do arquivo de descrição de	

	superfície "beta-spline" .....	151
Figura 7.3	Objeto instanciado na origem do sistema de coordenadas de maneira a facilitar a determinação do plano de corte .....	153
Figura 7.4	Envelope poliédrico .....	154
Figura 7.5	Exibição da curva encontrada sobre a superfície-base .....	156
Figura 7.6	Curva de ancoragem com respectivos vetores de controle .....	157
Figura 7.7	Formato de um arquivo de descrição de curvas de curvas de ancoragem .....	158
Figura 7.8	Seleção do trecho da curva de ancoragem a ser selecionado .....	160
Figura 7.9	Arquivo de descrição de objetos .....	162
Figura 7.10	Arquivo de descrição de objeto e o objeto correspondente .....	166
Figura 8.1	Parametrização de uma superfície de junção ..	180
Figura 8.2	Transição suave entre dois planos .....	182
Figura 8.3	Transição de raio constante .....	185
Figura 8.4	Situação onde é necessária a transição entre três superfícies .....	187
Figura 8.5	Objeto equivalente aos modelados por Warren ..	188
Figura 8.6	Suavização do encontro de duas superfícies ..	189

## RESUMO

Um tópico importante em modelagem geométrica é a definição de objetos que não contenham cantos ou farpas. A geração de superfícies de junção (superfícies que conectam suavemente superfícies distintas) é uma das técnicas utilizadas para solucionar esse tipo de problema.

Pode-se abordar a geração de superfícies de junção sob vários aspectos. Neste trabalho, serão estudadas superfícies de junção que conectam suavemente duas outras superfícies distintas (superfícies-base). Sob este enfoque, a geração das superfícies de junção será dividida em três etapas a saber: modelagem das superfícies-base, determinação das curvas de ancoragem e geração das junções propriamente ditas. Curvas de ancoragem são curvas que determinam a forma do contato entre a junção e cada uma das superfícies base.

As superfícies-base utilizadas são superfícies compostas modeladas utilizando-se "beta-splines" uniformes. Discussões sobre a melhor maneira de se modelar as mesmas foram consideradas fora do contexto deste trabalho.

A determinação das curvas de ancoragem é amplamente discutida. Optou-se por utilizar um plano de

corte contra cada uma das superfícies-base para a determinação das curvas de ancoragem. O cálculo da intersecção entre o plano de corte e cada um dos "patches" das superfícies-base é feito utilizando-se um método numérico híbrido baseado em um algoritmo de movimentação associado a um algoritmo de reticulado.

A geração das superfícies de junção é feita utilizando-se a técnica de Hermite. Para tanto é preciso que se calculem os vetores de controle necessários à técnica de Hermite de forma que se garanta a continuidade desejada ao longo da curva de contato entre a superfície base e a junção. No caso em questão é garantida continuidade geométrica de primeira ordem ( $G^1$ ).

Por fim, apresenta-se protótipo do modelador utilizado para validar o método proposto. São indicados resultados obtidos bem como uma análise comparativa com outras soluções semelhantes encontradas na literatura à disposição.

Os testes com o protótipo foram feitos em um microcomputador com arquitetura compatível com IBM-PC 386 com 640 KBytes de RAM, 20 Mhz, adaptador gráfico VGA e co-processador matemático 80387. O protótipo pode, entretanto, ser utilizado em qualquer microcomputador

compatível com IBM-PC que disponha de no mínimo 640 Kbytes de memória RAM. O uso de co-processador matemático e adaptador gráfico EGA ou de maior capacidade é aconselhado.

**Palavras-chave:** Computação Gráfica, Modelagem geométrica, Superfícies paramétricas Superfícies de junção

Title: Analysis of an Object Modeler that uses Blending Surfaces to connect Parametric Composed Beta Spline Surfaces

### ABSTRACT

An important topic in geometric modeling is the definition of objects without sharp, corners or edges. A blending surface is one that smoothly connects two given surfaces. It has been a very useful way to define objects without sharp, corners or edges.

One can approach the generation of blending surfaces from different points of view. This work deals with blending surfaces that smoothly connect two other surfaces called base-surfaces. According to this point of view the generation of blending surfaces will be divided in 3 steps: base-surfaces modeling, rail curves determination and generation of the blending surfaces. A rail curve is one along which blending and base surfaces meet.

The base-surfaces used are composed-surfaces modeled by uniform beta-splines. The problem of modeling these surfaces is beyond the scope of this work.

The determination of rail-curves is widely discussed. In order to determine each one of the rail curves, an intersection plane is chosen. The computation of the intersection between the intersection plane and each one of the patches of the base-surfaces uses a hybrid numeric algorithm based on a marching method associated with a lattice evaluation algorithm.

The generation of blending surfaces is done using the Hermite technique. To make this possible one has to compute the control vectors needed to generate an Hermite surface and, at the same time, guarantee the desirable continuity along the rail-curves. In this work geometric continuity ( $G^1$ ) is guaranteed.

Finally it is presented the prototype of the modeler used to validate the proposed method. The results obtained are presented as well as a comparative analysis with other solutions found in the literature.

Key-words: Computer Graphics, Geometric modeling, Parametric surfaces, Blending surfaces



## 1. INTRODUÇÃO

### 1.1 Motivação

Segundo Mortenson /MOR 85/ a grande vantagem dos atuais sistemas de modelagem geométrica auxiliada por computador reside na sua capacidade de facilitar a descrição de objetos complexos a partir de elementos mais simples.

Diversas técnicas de descrição de objetos seguem esta orientação. Algumas, como a geometria construtiva, possuem esta idéia inerente na sua própria definição; outras necessitam de recursos ou artifícios adicionais, não sendo, porém, menos poderosas ou eficientes.

O uso de superfícies na descrição de objetos de forma livre, tais como carrocerias de automóveis, fuselagens de aviões, cascos de navios e, mais recentemente suas aplicações na indústria de propaganda e animação têm estimulado as pesquisas na área de modelagem através de superfícies /BAR 87/.

A suavidade nas conexões entre superfícies possui uma variada gama de aplicações. Alhers *et ali* /AHL 90/ tem desenvolvido estudos sobre modelagem de carrocerias de automóveis da Volkswagen. Por sua vez, Warren /WAR 89/ cita,

como exemplo, a modelagem de uma hélice, a qual geralmente requer a definição de uma coleção de superfícies onde cada uma representa apenas uma pequena parte do objeto total. Tem-se ainda aplicação no projeto de objetos mecânicos relativamente simples. Por razões funcionais, objetos com cantos ou farpas não são apropriados. Exemplificando, no projeto de uma válvula hidráulica, cantos ou quinas podem retardar o fluxo do fluido. Da mesma forma existem certos processos industriais incompatíveis com cantos, quinas ou farpas. Produtos fabricados por injeção de material a altas pressões em um molde são um exemplo. No projeto de tais objetos, os projetistas costumam adicionar superfícies cuja única função é prover uma transição suave entre as partes funcionais do mesmo (fig. 1.1).

As superfícies que são usadas para prover uma transição suave entre duas outras superfícies distintas são chamadas neste trabalho de superfícies de junção ("blending surfaces") e a automatização do processo de geração das mesmas é de extrema importância para facilitar o processo de modelagem geométrica de objetos que usam superfícies.

Baseado no exposto anteriormente, este trabalho foi motivado por vários fatores entre os quais pode-se destacar:

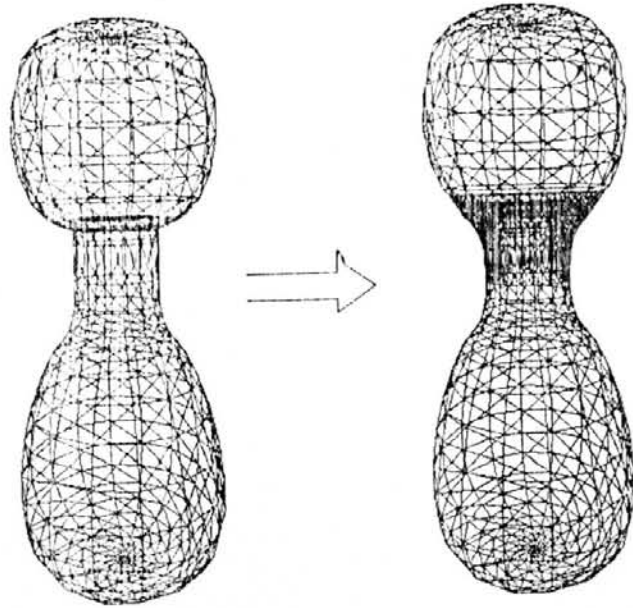


Figura 1.1 - Suavização de objetos usando superfícies de junção

a) A importância do uso de superfícies de junção em diferentes tipos de indústrias e a possibilidade de diversificar esse uso sobretudo na indústria de entretenimento;

b) A atualidade do tema e

c) O conjunto diverso de procedimentos e métodos envolvidos na geração de superfícies de junção e que levam a uma revisão bastante ampla dos conceitos envolvidos em modelagem geométrica.

## 1.2 Objetivos

O objetivo geral deste trabalho é analisar os aspectos relevantes de um modelador geométrico orientado para a definição de objetos de forma livre representados por superfícies paramétricas cúbicas unidas por superfícies de junção.

De acordo com o proposto, um objeto modelado será formado por um conjunto de superfícies e um conjunto de junções de maneira que, para quaisquer duas superfícies pertencentes a um mesmo objeto, ou existe uma junção entre elas ou existe uma continuidade entre as mesmas através de junções e outras superfícies.

As superfícies que compõem o objeto serão formadas por um conjunto de "patches" (setores de superfície) parametricamente definidas por "beta-splines" uniformes. Serão usadas "beta-splines" uniformes devido às características de suavidade, continuidade e controle de forma, além, de serem próprias para a interação em tempo real.

Uma junção será definida por apontadores para as duas superfícies "beta-splines" que serão unidas (superfícies-base) e para as duas curvas de ancoragem. Uma

curva de ancoragem determina a forma da intersecção ou contato de uma superfície-base com a superfície de junção. As curvas de ancoragem serão definidas implicitamente pela intersecção de um plano com a superfície base. As junções serão geradas usando interpolação Hermite. Esta técnica de interpolação exige um ponto de partida, um ponto de chegada e duas tangentes que orientam as direções de partida e chegada da curva. Os pontos de partida e chegada são determinados sobre as curvas de ancoragem e as tangentes, pelas superfícies que serão unidas.

Em resumo, este trabalho se propõe a descrever os procedimentos necessários para a geração das superfícies de junção anteriormente definidas, bem como validar os procedimentos descritos através de um modelador geométrico que permita modelar os objetos com tais características.

### **1.3 Limitações**

Entre os principais fatores que limitam este trabalho, destaca-se o número de enfoques que podem ser dados à questão. Conforme o tipo de aplicação e a maneira pela qual as superfícies a serem

unidas estão definidas, a abordagem do problema poderá ser diferente. Enfoques distintos podem ser encontrados por exemplo nos trabalhos de BARDIS [BARD 89], ROCKWOOD [ROC 89], WARREN [WAR 89] e SY JU [SY 89]. Nestes trabalhos ~~encontram-se~~ procedimentos para a determinação de junções definidas das mais variadas formas: junções de forma livre, junções de forma constante, junções entre mais de duas superfícies, entre outras abordagens. Neste caso, fez-se a seleção e análise de apenas um. No caso, o trabalho limita-se a analisar junções entre duas outras superfícies, ignorando os demais casos

Outra preocupação, que limita sobretudo o potencial do modelador proposto, é o ambiente computacional restrito sobre o qual o mesmo foi projetado. Foi utilizado um IBM-PC compatível com co-processador aritmético e 640 Kbytes de memória.

Finalmente existem limitações decorrentes do fato de se trabalhar com curvas e superfícies compostas que, se por um lado apresentam inúmeras vantagens para se trabalhar com modelagem geométrica, por outro, podem tornar seu tratamento bastante complexo devido à sua generalidade.

## 1.4 Organização Geral do Trabalho

É apresentada a seguir a forma de organização deste trabalho de maneira que se possa ter uma visão geral dos itens que serão apresentados.

No capítulo dois, uma série de conceitos sobre superfícies paramétricas e continuidade entre superfícies, utilizados em capítulos posteriores, são revisados.

No capítulo três é feita uma breve revisão bibliográfica sobre superfícies de junção bem como a identificação das etapas do método proposto.

No capítulo quatro é explicada em detalhes a etapa de determinação de curvas de ancoragem. Inicialmente é feita uma pequena revisão sobre algoritmos de determinação de intersecções entre superfícies. Em seguida um algoritmo para o cálculo de intersecções entre um plano e uma superfície paramétrica cúbica é apresentado. A seguir esse algoritmo é estendido para calcular intersecções entre um plano e uma superfície paramétrica composta.

No capítulo cinco é apresentado o cálculo dos vetores de controle e a geração das superfícies de junção.

No capítulo seis é mostrado como modelar um objeto composto por superfícies-base e junções usando o método proposto.

No capítulo sete são feitas considerações sobre o protótipo implementado.

No capítulo oito são verificadas as potencialidades do método descrito.

Finalmente, no capítulo nove são apresentadas conclusões e as possibilidades de continuidade para este trabalho.



## 2. FUNDAMENTOS GEOMÉTRICOS E MATEMÁTICOS

Este capítulo tem por objetivo revisar conceitos e estabelecer a nomenclatura e a simbologia que será utilizada.

Inicialmente será feita uma revisão sobre a representação paramétrica de curvas e superfícies. Em seguida será apresentada a formulação Hermite utilizada. Na seqüência será revisto o conceito de continuidade geométrica e, por fim, as superfícies "beta-splines" serão introduzidas. O material apresentado no item 2.1 é baseado no livro "Geometric Modeling" /MOR 85/. Os itens 2.2, 2.3 e 2.4 foram escritos a partir do artigo de Daniel Filip /FIL 89/ e do livro "An Introduction to Splines for use in Computer Graphics and Geometric Modeling" /BAR 87/.

### 2.1 Representações paramétricas

Inicialmente será feita uma revisão na justificativa do uso de equações paramétricas. Em seguida serão apresentados os aspectos relevantes para este trabalho da representação paramétrica de curvas, superfícies e curvas em superfícies. Por fim será apresentada a notação que será utilizada.

### 2.1.1 Equações paramétricas

Não é possível expressar as formas necessárias em modelagem geométrica com funções da forma  $y=f(x)$ . Existem diversas razões para tanto. A forma dos objetos que se costuma modelar é, em geral, independente de qualquer sistema de coordenadas. Quando se traçam curvas ou superfícies, rapidamente se percebe que existe uma relação que determina a forma final. Esta relação, porém, não é entre os pontos e os eixos coordenados e sim entre os pontos. De qualquer forma, a maioria das aplicações em modelagem geométrica requer que a escolha do sistema de coordenadas não afete a forma. Além disso, as curvas e superfícies utilizadas em modelagem geométricas são, em geral, não planares e fechadas e não podem, de qualquer maneira, serem representadas por funções não paramétricas.

Por estas razões e outras relacionadas com a facilidade de programação, costuma-se utilizar, em modelagem geométrica, equações paramétricas. Por exemplo, uma curva bidimensional não é expressa por uma função da forma  $y=f(x)$ , mas por um conjunto de duas equações  $x=x(u)$ ,  $y=y(u)$  em função de um parâmetro  $u$ . Um ponto em tal curva é representado pelo vetor

$$\mathbf{p}(u)=[x(u) \ y(u)]$$

Analogamente, um ponto em uma curva no espaço será dado pelo vetor

$$\mathbf{p}(u)=[x(u) \ y(u) \ z(u)]$$

e um ponto em uma superfície é representado pelo vetor

$$\mathbf{p}(u,w)=[x(u,w) \ y(u,w) \ z(u,w)]$$

O ponto é o elemento básico em modelagem geométrica. É definido como um conjunto ordenado de números reais, em geral as coordenadas que indicam sua posição no espaço. Um vetor pode ser definido da mesma maneira. Frequentemente os termos ponto e vetor podem ser usados um no lugar do outro.

Em geral não interessa traçar uma curva paramétrica para todos os valores do parâmetro. Neste trabalho será utilizado sempre o intervalo normalizado [0 1].

### 2.1.2 Curvas

Um segmento de curva é uma coleção de pontos cujas coordenadas são dadas por funções de um parâmetro da forma:

$$x=x(u)$$

$$y=y(u)$$

$$z=z(u)$$

A variável paramétrica  $u$  é limitada ao intervalo  $[0, 1]$  e, intuitivamente, associa-se a idéia de que a curva se desenvolve ou cresce a medida em que o valor de  $u$  aumenta. Diz-se que um segmento de curva é limitado porque ele tem dois pontos limites: um em  $u=0$  e outro em  $u=1$ .

Trata-se as coordenadas de um ponto qualquer como as componentes de um vetor  $\mathbf{p}(u)$ . Outro vetor importante é o vetor tangente. O vetor tangente  $\mathbf{p}'(u)$  é o vetor tangente a curva no ponto  $\mathbf{p}(u)$ . É encontrado derivando-se  $\mathbf{p}(u)$ . Então:

$$\mathbf{p}'(u) = \frac{d\mathbf{p}(u)}{du}$$

e suas componentes são:

$$x'(u) = \frac{dx(u)}{du} \quad y'(u) = \frac{dy(u)}{du} \quad z'(u) = \frac{dz(u)}{du}$$

Estas são derivadas paramétricas. Relações entre elas conduzem a derivadas com significado geométrico no espaço cartesiano onde se encontra a curva:

$$dx(u) = \frac{\frac{dy}{du}}{\frac{dx}{du}}$$

Um vetor tangente assim calculado contém algumas informações interessantes. A direção do vetor é a direção de crescimento ou movimentação da curva e a magnitude do mesmo indica a velocidade com que tal movimento ocorre.

Dois conceitos que serão usados são os conceitos de "espaço objeto" e "espaço paramétrico". Espaço objeto é o espaço tridimensional definido por coordenadas cartesianas  $x, y,$  e  $z$ . É o espaço onde o modelo geométrico é trabalhado. O espaço paramétrico é um conjunto de espaços bidimensionais definidos pelas coordenadas  $(x,u), (y,u)$  e  $(z,u)$ . Como qualquer curva paramétrica é definida por suas três componentes paramétricas, pode-se traçar os gráficos que relacionam o comportamento das componentes cartesianas em relação ao parâmetro.

### 2.1.3 Superfícies

O elemento matemático mais simples que se pode utilizar para modelar uma superfície é um "patch". Um

"patch" é uma coleção de pontos limitados por segmentos de curvas cujas coordenadas são dadas por funções contínuas de dois parâmetros da forma:

$$x=x(u,w) \quad y=y(u,w) \quad z=z(u,w)$$

As variáveis paramétricas  $u,w$  estão limitadas ao intervalo  $[0, 1]$ .

Fixar o valor de uma das variáveis paramétricas resulta em uma curva no "patch" em termos da outra variável que permanece livre. Se este processo for repetido para todos os valores de uma mesma variável e depois para a outra, será formada uma malha de curvas no "patch" onde, para cada ponto  $p(u,w)$ , passam duas curvas: uma de  $u$  constante e outra de  $v$  constante. Cada uma dessas curvas será chamada de curva isoparamétrica em relação ao parâmetro constante. Uma curva de  $u$  constante será dita uma isoparamétrica em  $u$ .

Um "patch" possui sempre quatro isoparamétricas que o limitam e quatro cantos definidos pela intersecção dessas curvas. Esses pontos correspondem aos pontos onde os parâmetros assumem os valores máximos e mínimos:  $p(0,0)$ ,  $p(1,0)$ ,  $p(0,1)$  e  $p(1,1)$ .

Em relação a uma superfície ou "patch" qualquer, é possível determinar um conjunto infinito de planos tangentes que, por sua vez, contém um conjunto de retas tangentes a superfície em um determinado ponto. Derivando a equação do "patch" em relação a cada um dos parâmetros pode-se obter dois vetores tangentes de significado especial. A direção de cada um dos vetores indicará o sentido do crescimento ou movimentação da superfície em relação ao parâmetro. Na verdade, esse será o sentido do crescimento da curva isoparamétrica em relação ao mesmo parâmetro em um dado ponto. Então:

$$\mathbf{p}'_u(u, w) = \frac{\partial \mathbf{p}(u, w)}{\partial u}$$

$$\mathbf{p}'_w(u, w) = \frac{\partial \mathbf{p}(u, w)}{\partial w}$$

Em qualquer ponto  $\mathbf{p}(u, w)$  de um "patch" (não degenerado) é possível determinar um vetor normal ao "patch". Pode-se facilmente encontrar um vetor normal unitário  $\mathbf{n}(u, w)$  calculando-se o produto vetorial dos vetores tangentes  $\mathbf{p}'_u(u, w)$  e  $\mathbf{p}'_w(u, w)$  no ponto.

$$\mathbf{n}(u, w) = \frac{\mathbf{p}'_u(u, w) \times \mathbf{p}'_w(u, w)}{|\mathbf{p}'_u(u, w) \times \mathbf{p}'_w(u, w)|}$$

Em geral costuma-se usar a expressão "normal ao patch" para designar o vetor normal ao "patch" em um dado ponto. A figura 2.1 mostra um "patch" com suas curvas e vetores.

Quanto ao espaço paramétrico da superfície, pode-se dizer que ele difere do de uma curva pelo fato de que existe uma variável paramétrica adicional. Tem-se, então, (ver Mortenson [MOR 85], pg. 171) um conjunto de três espaços dimensionais paramétricos definidos pelas coordenadas  $(u,w,x)$ ,  $(u,w,y)$  e  $(u,w,z)$ .

#### 2.1.4 Curvas em superfícies

Um "patch", como já foi visto, é formado por um conjunto de pontos limitados por quatro curvas. Cada ponto pertencente a um "patch" pode ser associado a um par de parâmetros  $(u,w)$ . Escolhendo-se um subconjunto dos pontos de uma superfície pode-se definir uma curva na mesma. Esta curva pode ser descrita de diversas maneiras. Uma possível representação paramétrica é:

$$m(t) = (u(t),w(t))$$



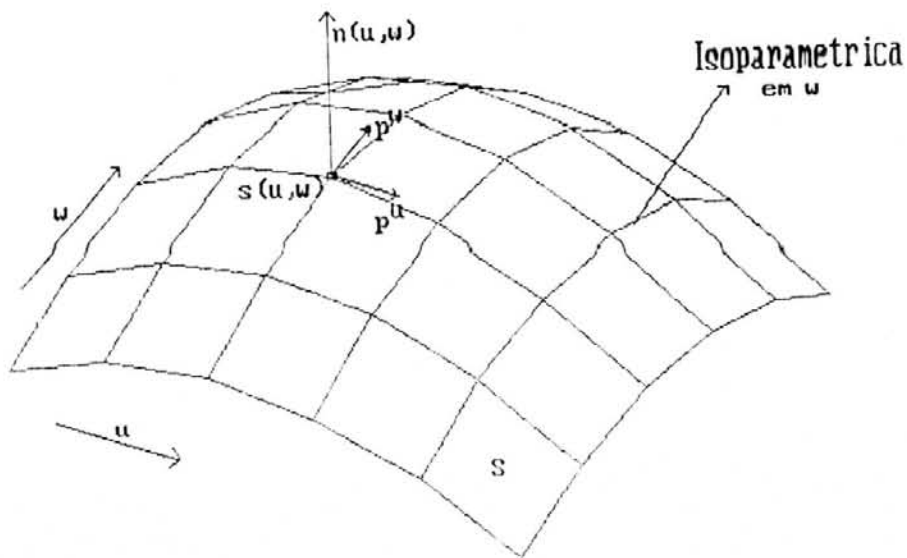


Figura 2.1 - Curvas e vetores de um "patch"

ou seja, a relação que determina o conjunto de pares  $u, w$  que formam a curva pode ser, também, descrita em função de um parâmetro. A partir desta equação paramétrica, pode-se tanto obter os pares  $u, w$  com o objetivo de obter os pontos sobre a superfície quanto para traçar a curva correspondente em um espaço paramétrico  $u, w$  (fig. 2.2). O traçado desta curva pode ser bastante útil como se verá mais adiante.

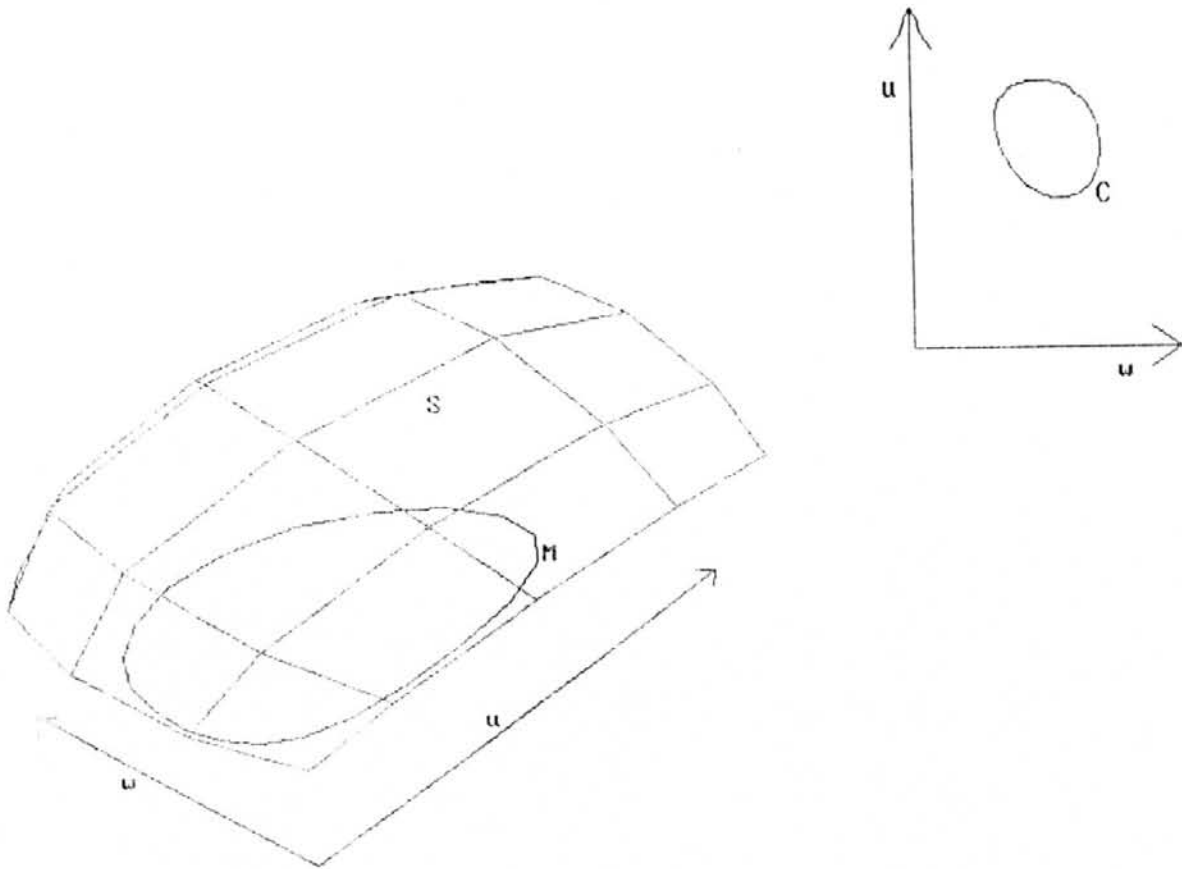


Figura 2.2 - Espaço paramétrico de uma curva.

Neste trabalho será usada a expressão "curva no espaço objeto" quando for feita referência a curva definida no mesmo espaço do "patch" no qual ela se encontra. A expressão "curva no espaço paramétrico" será usada para referenciar a curva que descreve a relação entre os valores dos parâmetros  $u, w$  que determinam a curva no espaço objeto.

A cada ponto de uma curva no espaço objeto é possível determinar vetores tangentes da mesma forma que para qualquer curva paramétrica definida em um espaço tridimensional. A única diferença é que este vetor será também tangente ao "patch" que contém a curva. A cada ponto da curva no espaço paramétrico também é possível determinar vetores tangentes. No caso da curva no espaço objeto o vetor tangente é calculado como demonstrado no item 2.1.2 e utiliza como suporte uma das possíveis retas tangentes àquele ponto. No caso, a reta suporte utilizada é orientada de forma a coincidir com a orientação de um vetor definido pelo ponto de tangência e pelo ponto que se obtém quando se faz um incremento infinitesimal no parâmetro da curva. No caso da curva no espaço paramétrico, o vetor tangente é único (o espaço paramétrico do "patch" é bidimensional), porém, tem o mesmo significado.

#### 2.1.5 Notação

Neste trabalho será usada a simbologia listada a seguir:

- valores escalares, coeficientes, constantes:  
letras minúsculas (a,b,c, ...):

- vetores: letras minúsculas em negrito (**p**, **n**, **r**, ...);

- matrizes: letras maiúsculas em negrito (**V**, **A**, **B**, **C**, ...);

- variáveis paramétricas: *s, t, u, v, w*

- coordenadas cartesianas: *x, y, z* ou *x<sub>1</sub>, y<sub>1</sub>, z<sub>1</sub>*, etc;

- intervalos fechados: [*a b*];

- derivadas:

$$\mathbf{p}'(u) = \frac{d\mathbf{p}(u)}{du}$$

$$\mathbf{p}''(u) = \frac{d^2\mathbf{p}(u)}{du^2}$$

$$\mathbf{p}'_u(u, w) = \frac{d\mathbf{p}(u)}{du}$$

## 2.2 Superfícies Hermite

A formulação de Hermite para geração de curvas paramétricas se utiliza de dois pontos e dois vetores de controle. No caso da figura 2.3, os pontos  $\mathbf{p}(0)$  e  $\mathbf{p}(1)$

correspondem respectivamente aos pontos de partida e chegada da curva, enquanto que os vetores  $\mathbf{t}'_u(0)$  e  $\mathbf{t}'_u(1)$  orientam respectivamente a direção e sentido de partida e chegada da curva.

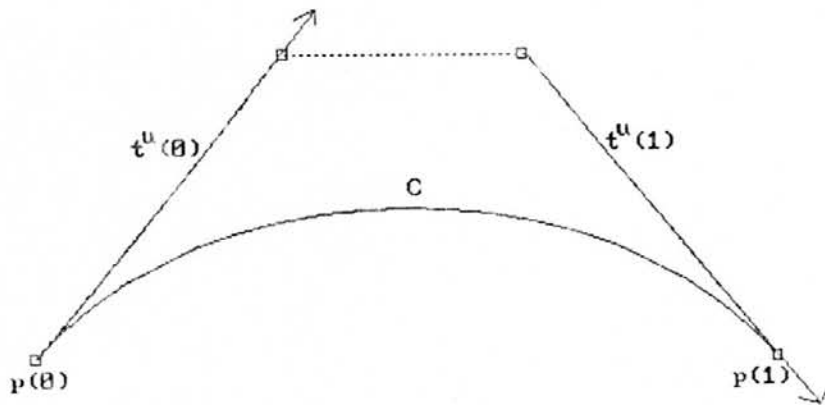


Figura 2.3 - Curva Hermite

Logo, utilizando a formulação Hermite, variando o parâmetro em um intervalo pré-estabelecido (em geral  $[0, 1]$ ), obtém-se uma curva que interpola os pontos  $\mathbf{p}(0)$  e  $\mathbf{p}(1)$  cujo comportamento é influenciado ao longo da variação do

parâmetro pelos vetores  $\mathbf{t}'_u(0)$  e  $\mathbf{t}'_u(1)$ .

Uma superfície Hermite será gerada a partir de duas curvas e dos vetores nos pontos amostrados sobre essas curvas, ou seja, uma superfície Hermite é definida por uma coleção de curvas Hermite ancoradas em duas curvas que se deseja unir (fig 2.4).

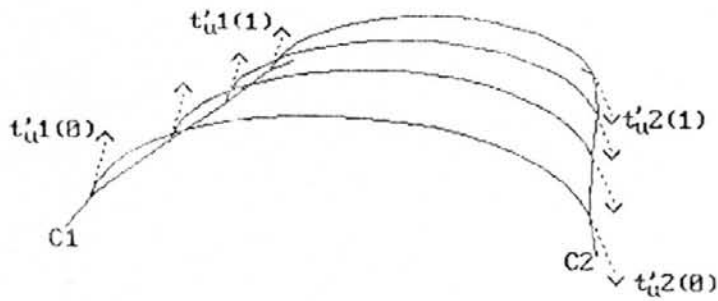


Figura 2.4 - Superfície Hermite.

Dado que  $C_1$  e  $C_2$  são duas curvas que se deseja unir ou que servem de referência para a geração de uma superfície e que  $\mathbf{t}_n(t)$ ,  $n=1,2$ , são os vetores

correspondentes a cada ponto amostrado sobre as respectivas curvas, tem-se a seguinte formulação para uma superfície Hermite:

$$\mathbf{b}(s,t) = H_1(s)\mathbf{c}_1(t) + H_2(s)\mathbf{c}_2(t) + H_3(s)\mathbf{t}_1(t) + H_4(s)\mathbf{t}_2(t)$$

onde:

$$H_1(s) = s^2(2s-3)+1$$

$$H_2(s) = 1 - H_1(s)$$

$$H_3(s) = s(s-1)^2$$

$$H_4(s) = s^2(s-1)$$

são as funções de ponderação para Hermite quando  $0 \leq s \leq 1$  e  $0 \leq t \leq 1$ .

### 2.3 Continuidade geométrica

Diz-se que duas curvas são geometricamente contínuas em relação à primeira derivada paramétrica quando os vetores tangentes no ponto de encontro entre as mesmas possuem a mesma direção.

Como a magnitude dos vetores tangentes em um ponto de uma curva não interfere na direção em que ela se desenvolve, pode-se trabalhar com o vetor tangente

normalizado sem alterar a direção da curva. É o que se costuma chamar de vetor tangente unitário.

Dir-se-á, então, que o encontro de duas curvas se dá com continuidade geométrica com respeito a primeira derivada ( $G^1$ ) se forem respeitadas as seguintes condições:

Dado:

$C_1$ ,  $C_2$  são duas curvas que possuem ponto de encontro em  $c_1(1) = c_2(0)$ .

A curva resultante será  $G^1$  se:

- (i)  $c_1(1) = c_2(0)$
- (ii)  $|c_{1u}'(1)| = |c_{2u}'(0)|$

Raciocínio equivalente pode ser desenvolvido para a segunda derivada. Então diz-se que a curva resultante da junção de  $C_1$  com  $C_2$  terá continuidade geométrica com respeito a segunda derivada ( $G^2$ ) se :

- (i)  $c_1(1) = c_2(0)$
- (ii)  $|c_{1u}'(1)| = |c_{2u}'(0)|$
- (iii)  $|c_{1u}''(1)| = |c_{2u}''(0)|$



## 2.4 Superfícies "beta-splines"

Inicialmente, por questão de simplicidade, será apresentado o conjunto de fórmulas utilizado para representar superfícies "B-splines" uniformes e como a maneira pela qual o mesmo foi organizado facilita a determinação de curvas isoparamétricas sobre as superfícies (isso será importante para o entendimento do capítulo 4). Em seguida será apresentada a idéia que permite generalizar as "b-splines" em "beta-splines".

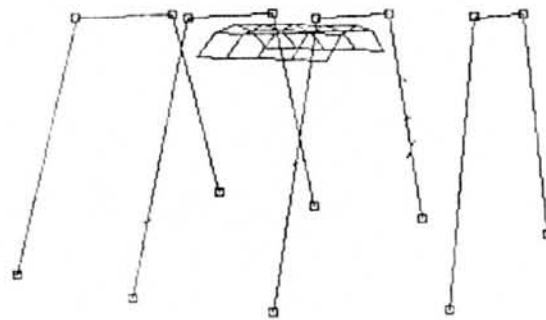


Figura 2.5 - "Patch B-spline"

Segundo Barsky /BAR 87/ uma superfície bicúbica "B-spline" é formada pela soma de funções de ponderação usando como fatores de escala as coordenadas x, y e z de uma matriz retangular de vértices de controle que definem um poliedro chamado poliedro de controle o qual a superfície tenta aproximar (fig 2.5).

Se os vértices do poliedro de controle estiverem organizados em uma matriz  $\mathbf{V}_{ij}$ , a superfície pode ser expressa parametricamente por um duplo somatório:

$$\sum_i \sum_j \mathbf{V}_{ij} B_{ij}(u,v) = \sum_i \sum_j (x_{ij} B_{ij}(u,v), y_{ij} B_{ij}(u,v), z_{ij} B_{ij}(u,v))$$

onde  $B_{ij}$  são as funções de mistura ou as funções de ponderação dos vértices de controle e  $(u,v)$  os parâmetros ( $u = [0 \ 1]$ ,  $v = [0 \ 1]$ ).

Chama-se de região de influência de um vértice de controle a região da superfície que é afetada quando se translada um desses vértices. Uma maneira de delimitar essa região de influência à um retângulo é fazer  $B_{ij}(u,v) = B_i(u)B_j(v)$ , onde  $B_i(u)$  e  $B_j(v)$  são as funções de ponderação para curvas "B-splines" cúbicas. As funções são:

$$B_0(u) = \frac{1}{6} u^3$$

$$B_1(u) = \frac{1}{6} (1 + 3u + 3u^2 - 3u^3)$$

$$B_2(u) = \frac{1}{6} (4 - 6u^2 + 3u^3)$$

$$B_3(u) = \frac{1}{6} (1 - 3u + 3u^2 - u^3)$$

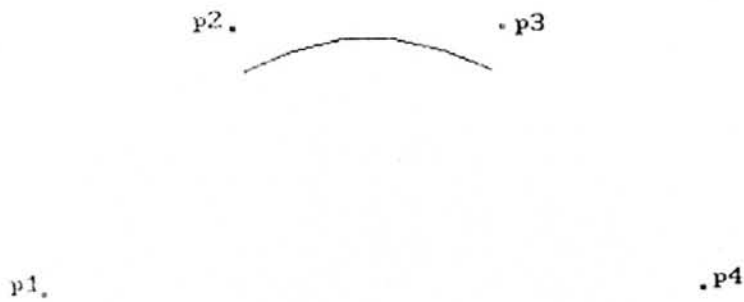


Figura 2.6 - Segmento de curva "B-spline"

Um segmento de curva "B-spline" é aproximado a partir de quatro pontos de controle (fig 2.6) pelo somatório:

$$c(t) = \sum_{i=0}^4 P_i B_i(t)$$

onde  $\mathbf{P}$  é a matriz unidimensional que contém os pontos de controle. Uma curva completa (coleção de segmentos) pode ser representada por:

$$\mathbf{c}_j(t) = \sum_{i=0}^4 \mathbf{P}_{i+j} B_i(t)$$

onde  $j$  indica o índice do primeiro dos quatro pontos de controle que determinam o  $j$ -ésimo segmento de curva.

Como uma superfície, neste caso, é o resultado da multiplicação de funções de ponderação de duas curvas, serão necessários dezesseis vértices de controle para determinar um "patch" de superfície. Então, uma superfície é dada por:

$$\mathbf{s}_{ij}(u,v) = \mathbf{V}_{i+r, j+s} B_r(u) B_s(v)$$

onde  $i, j$  determinam um conjunto de dezesseis vértices necessários ao cálculo de um "patch" dentro da matriz  $\mathbf{V}$  de vértices do poliedro de controle. O desenvolvimento de  $B_{i,j}(u,v)$  em  $B_i(u)$  e  $B_j(v)$  é bastante útil para a determinação das curvas isoparamétricas de uma superfície. A representação expandida de uma superfície é dada por:

$$\begin{aligned}
s_{ij}(u, v) = & \\
& [V_{i+3, j+0} B_3(u) + V_{i+2, j+0} B_2(u) + V_{i+1, j+0} B_1(u) + V_{i+0, j+0} B_0(u)] B_0(v) + \\
& [V_{i+3, j+1} B_3(u) + V_{i+2, j+1} B_2(u) + V_{i+1, j+1} B_1(u) + V_{i+0, j+1} B_0(u)] B_1(v) + \\
& [V_{i+3, j+2} B_3(u) + V_{i+2, j+2} B_2(u) + V_{i+1, j+2} B_1(u) + V_{i+0, j+2} B_0(u)] B_2(v) + \\
& [V_{i+3, j+3} B_3(u) + V_{i+2, j+3} B_2(u) + V_{i+1, j+3} B_1(u) + V_{i+0, j+3} B_0(u)] B_3(v)
\end{aligned}$$

Se, por exemplo, for fixado o parâmetro  $u$  na equação anterior, pode-se escrever:

$$s_{iju}(v) = w_0 B_3(v) + w_1 B_2(v) + w_2 B_1(v) + w_3 B_0(v)$$

onde o aparecimento de  $u$  entre os subscritos indica que seu valor foi fixado e as funções  $w$  podem ser expandidas como segue:

$$\begin{aligned}
w_3 &= V_{i+3, j+0} B_3(u) + V_{i+2, j+0} B_2(u) + V_{i+1, j+0} B_1(u) + V_{i+0, j+0} B_0(u) \\
w_2 &= V_{i+3, j+1} B_3(u) + V_{i+2, j+1} B_2(u) + V_{i+1, j+1} B_1(u) + V_{i+0, j+1} B_0(u) \\
w_1 &= V_{i+3, j+2} B_3(u) + V_{i+2, j+2} B_2(u) + V_{i+1, j+2} B_1(u) + V_{i+0, j+2} B_0(u) \\
w_0 &= V_{i+3, j+3} B_3(u) + V_{i+2, j+3} B_2(u) + V_{i+1, j+3} B_1(u) + V_{i+0, j+3} B_0(u)
\end{aligned}$$

Então,  $s_{iju}(v)$  é um ponto de um segmento de curva  $S$ , "B-spline", definido pelos vértices de controle  $w_0, w_1, w_2, w_3$ .

Consegue-se, dessa forma, determinar facilmente os polígonos de controle das curvas isoparamétricas em  $u$  sobre a superfície. Através de um raciocínio análogo encontra-se

as isoparamétricas em  $v$ .

Mostra-se em seguida a idéia que permite generalizar as "B-splines" em "beta-splines". Será utilizada uma curva em vez de uma superfície por questão de simplicidade.

Suponha-se que  $C_i$  e  $C_{i+1}$  são dois segmentos contínuos de uma curva composta "B-spline". Como as curvas "B-spline" possuem continuidade paramétrica com respeito a segunda derivada ( $C^2$ ) sabe-se que:

$$\mathbf{c}_i(1) = \mathbf{c}_{i+1}(0) \quad (2.3 - 1)$$

$$|\mathbf{c}'_{u_i}(1)| = |\mathbf{c}'_{u_{i+1}}(0)| \quad (2.3 - 2)$$

$$|\mathbf{c}''_{u_i}(1)| = |\mathbf{c}''_{u_{i+1}}(0)| \quad (2.3 - 3)$$

Se os vetores da primeira derivada forem normalizados, as restrições acima continuam valendo. Se, por outro lado, for admissível que a curva mantenha apenas continuidade  $G^1$  pode-se reescrever (2.3 - 2) da seguinte maneira:

$$k_1 \mathbf{c}'_{u_i}(1) = \mathbf{c}'_{u_{i+1}}(0) \quad (2.3 - 4)$$

A restrição acima não garante mais continuidade paramétrica ( $C^1$  ou  $C^2$ ), garantindo apenas continuidade

geométrica de primeira ordem ( $G^1$ ).

Raciocínio semelhante pode ser aplicado à segunda derivada e, então, obtém-se uma curva de continuidade  $G^2$ .

Reescrevendo (2.3 - 2) e (2.3 - 3):

$$k_1 \mathbf{c}'_{u_i}(1) = \mathbf{c}'_{u_{i+1}}(0) \quad (3.3 - 5)$$

$$k_2 \mathbf{c}''_{u_i}(1) = \mathbf{c}''_{u_{i+1}}(0) \quad (3.3 - 6)$$

recalculando as funções de mistura de uma "B-spline" levando em conta as novas restrições obtém-se as funções de ponderação para "Beta-splines".

As "beta-splines" possuem dois graus de liberdade a mais que as "B-splines". Estes são os parâmetros de controle de forma  $\beta_1$  e  $\beta_2$ , que nada mais são do que valores que influenciarão, respectivamente, o módulo dos vetores da primeira e segunda derivadas paramétricas dos segmentos de curva nos pontos de junção, permitindo que se alterem as velocidades de partida e chegada das curvas. As funções de ponderação para "beta-splines" uniformes são as que seguem.

$$b_0(u) = \frac{1}{\delta} \left[ 2u^3 \right]$$

$$b_1(u) = \frac{1}{\delta} \left[ 2 + (6\beta_1)u + (3\beta_2 + 6\beta_1^2)u^2 - (2\beta_2 + 2\beta_1^2 + 2\beta_1 + 2)u^3 \right]$$

$$b_2(u) = \frac{1}{\delta} \left[ (\beta_2 + 4\beta_1^2 + 4\beta_1) + (6\beta_1^3 - 6\beta_1)u - \right. \\ \left. (3\beta_2 + 6\beta_1^3 + 6\beta_1^2)u^2 + (2\beta_2 + 2\beta_1^3 + 2\beta_1^2 + 2\beta_1)u^3 \right]$$

$$b_3(u) = \frac{1}{\delta} \left[ (2\beta_1^3) - (6\beta_1^3)u + (6\beta_1^3)u^2 - (2\beta_1^3)u^3 \right]$$

onde:

$$\delta = \beta_2 + 2\beta_1^3 + 4\beta_1^2 + 4\beta_1 + 2 \neq 0$$

As "beta-splines" são consideradas uma generalização das "B-splines", porque pode-se obter novamente a formulação "B-spline" atribuindo-se os valores zero e um, respectivamente, para os parâmetros  $\beta_1$  e  $\beta_2$ .

A idéia de continuidade geométrica não foi apresentada apenas para que fosse possível entender a generalização da formulação "B-spline" em "beta-spline". Ela será especialmente importante no capítulo cinco, quando será apresentada a geração das superfícies de junção.



### 3. O MÉTODO PROPOSTO

Este capítulo é constituído por uma breve revisão bibliográfica sobre o assunto seguida da apresentação do método proposto para a geração de superfícies de junção.

#### 3.1 Uma revisão sobre o assunto

O tema Superfície de Junção (em inglês "Blending Surface") denota tanto aquelas superfícies que formam uma transição suave entre duas superfícies que se interceptam ou uma superfície que conecta suavemente duas superfícies desconexas /BARD 89/ (fig 3.1).

Na obtenção de superfícies de junção, diferentes idéias têm sido utilizadas, dependendo do propósito e do tipo de superfície envolvida. Existe uma certa concordância entre os autores de que não existe um método genérico para a geração de superfícies de junção /SY 89/.

Sy Ju *et ali* /SY 89/ apresentam um método de geração de superfícies de junção próprio para a suavização de cantos e quinas. O método consiste em definir a junção pelo rastro deixado por uma esfera "rolada" no vale formado

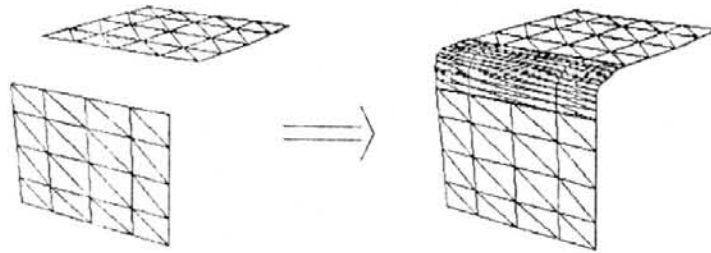


Figura 3.1 - Superfície de junção

por duas superfícies que se tocam ou se interceptam (fig. 3.2).

Joe Warren /WAR 89/ descreve vários métodos para a obtenção de objetos compostos por segmentos de superfícies descritas algebricamente e unidas por superfícies de junção.

Alyn Rockwood /ROC 89/ apresenta um método para a geração de superfícies de junção definidas de maneira implícita entre sólidos descritos pelas suas superfícies limitantes (B-Rep) ou por geometria sólida construtiva

(CSG). Em seu trabalho, classifica as superfícies de junção em parametricamente definidas e implicitamente definidas. Ambas representações possuem suas respectivas vantagens. Superfícies parametricamente definidas são, em geral, mais fáceis de desenhar, subdividir ou aplicar qualquer operação que exija a informação sobre onde determinado elemento se encontra sobre a superfície. São recomendadas para aplicações que requeiram superfícies de forma livre e flexibilidade de modelagem. Superfícies implícitas são usadas em aplicações onde cálculos tais como a distância de um ponto à superfície ou o volume definido pela mesma tornam-se importantes.

Bardis *et ali* /BARD 89/ descrevem um método para obtenção de superfícies de junção entre duas superfícies desconexas geradas por "B-splines" racionais. A junção é determinada a partir de curvas de ancoragem sobre cada uma das superfícies-base. As curvas de ancoragem são geradas por "B-splines" racionais e determinam a forma e o local do contato entre a junção e as superfícies-base. A junção gerada também é descrita por "B-splines" racionais de maneira que se obtém uma representação uniforme tanto para as junções como para as superfícies-base, o que facilita o

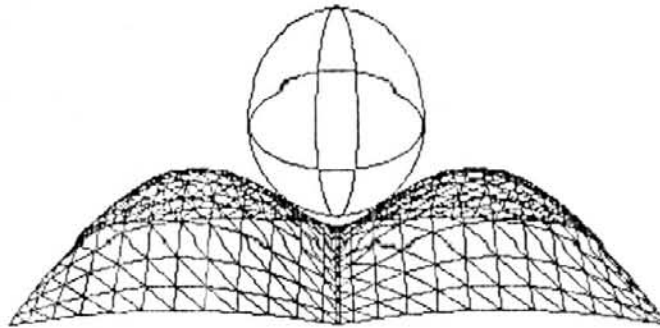


Figura 3.2 - Determinação de uma superfície de junção pelo rolar de uma esfera no vale formado pela intersecção de duas superfícies.

tratamento posterior (visualização, questionamentos, junções entre junções, etc).

Filip /FIL 89/ apresenta um método de geração de superfícies de junção que em muito se assemelha ao método de Bardis. As duas superfícies-base são superfícies parametricamente definidas, sobre as quais é necessário definir uma curva de ancoragem. A grande diferença encontra-se na forma de definir e tratar as curvas de ancoragem.

Finalmente, diversos autores, entre eles Mortenson /MOR 85/, sugerem o uso de curvas isoparamétricas como curvas de ancoragem quando se trabalha com junções entre superfícies-base descritas por "patches" paramétricos. É bastante comum que nestes casos utilizem-se uma das bordas dos "patches" como curva de ancoragem.

Como foi comentado no capítulo um, o problema da geração de superfícies de junção possui diversos enfoques. Um dos fatores que permite a seleção do tipo de enfoque que se deseja adotar é o tipo de superfície que se deseja unir. Como o objetivo deste trabalho é a geração de junções entre superfícies paramétricas compostas "beta-splines" através de curvas quaisquer de ancoragem, a análise das possíveis soluções será restrita as que se ajustam a este caso em particular.

Um ponto a ser analisado com atenção quando se utiliza superfícies paramétricas compostas como superfícies-base é a maneira pela qual serão representadas as curvas de ancoragem. Basicamente pode-se isolar dois tipos de casos:

a) A equação que descreve a curva de ancoragem está disponível ou é de fácil determinação e

b) A equação que descreve a curva de ancoragem é de difícil determinação e, por esta razão, utilizam-se algoritmos que fornecem uma representação para a mesma através de uma lista de pontos. Em geral o espaçamento entre estes pontos é determinado de acordo com um fator de tolerância. O fator de tolerância é um valor que indica a curvatura máxima admitida entre dois pontos consecutivos. Dessa forma, se forem interpolados outros pontos a partir destes sabe-se que o erro na aproximação será no máximo igual ao fator de tolerância.

A escolha da melhor abordagem dependerá fundamentalmente da aplicação em questão bem como do ambiente computacional disponível. Será feita em seguida uma análise dos dois casos.

3.1.1. Determinação de superfícies de junção quando a curva de ancoragem possui uma equação conhecida ou de fácil determinação

Quando as equações das curvas de ancoragem são conhecidas, o problema da geração de superfícies de junção é simplificado por diversas razões:

a) A etapa de determinação das curvas de ancoragem inexistente;

b) O cálculo dos vetores de controle (vetores que orientam a forma da junção) é simplificado e

c) É mais fácil garantir uma determinada continuidade entre a junção e as superfícies-base ao longo de uma curva de ancoragem perfeitamente definida.

Para exemplificar o que foi exposto, será apresentado um método para a geração de superfícies de junção entre dois "patches B-spline".

Dada sua simplicidade, diferentes soluções podem ser obtidas. A solução aqui apresentada será a geração de um "patch" de Coons /CON 67/ como superfície de junção. A vantagem da utilização do "patch" de Coons, neste caso, é que será possível garantir com facilidade continuidade geométrica de primeira ordem ( $G^1$ ) entre a superfície base e a junção ao longo das curvas de ancoragem.

Como foi visto no item 2.1.3, um "patch" cúbico é sempre limitado por quatro curvas. Um ponto pertencente a um "patch", segundo a formulação de Coons, é o resultado da ponderação dos pontos obtidos pela substituição das componentes paramétricas do ponto que se deseja obter nas

equações das curvas que limitam o "patch" (fig 3.3). O equacionamento do "patch" desta forma, porém, leva a que os pontos extremos do mesmo (pontos onde os parâmetros assumem os valores (0,0), (0,1), (1,0) e (1,1)) sejam considerados duas vezes. Por esta razão é necessário subtrair convenientemente os pontos ponderados em excesso. Desta maneira, o "patch" de Coons pode ser equacionado como segue:

$$\begin{aligned}
 p(s,t) = & Fd(t) Ca(s) + Fc(t) Cb(s) \\
 & + Fd(s) Cc(t) + Fc(s) Cd(t) \\
 & - Fd(s) Fd(t) p(0,0) \\
 & - Fd(s) Fc(t) p(1,0) \\
 & - Fc(s) Fc(t) p(1,1) \\
 & - Fc(s) Fd(t) p(0,1).
 \end{aligned}$$

onde:

$p(s,t)$  é um ponto do "patch";  
 $Ca$ ,  $Cb$ ,  $Cc$  e  $Cd$  são os polinômios de grau três (se o "patch" é cúbico) que delimitam o "patch" e  
 $Fc$  e  $Fd$  são as funções de ponderação crescente e decrescente respectivamente.

Por esta formulação pode-se observar como é simples determinar uma superfície de junção utilizando o "patch" de Coons desde que seja possível determinar as equações das curvas de ancoragem. As outras duas curvas





Figura 3.3 - "Patch" de Coons.

necessárias para a delimitação do "patch" podem ser facilmente representadas por Hermite como se verá em seguida (fig. 3.4).

Quando se utilizam "patches beta-spline" ou "B-spline" como superfícies-base e curvas isoparamétricas como curvas de ancoragem, a determinação da equação da curva de ancoragem é extraída da própria equação que descreve o "patch" como foi visto no item 2.4. Tem-se, então:

$$p(t) = \sum_{i=1}^4 p_i b_i(t)$$

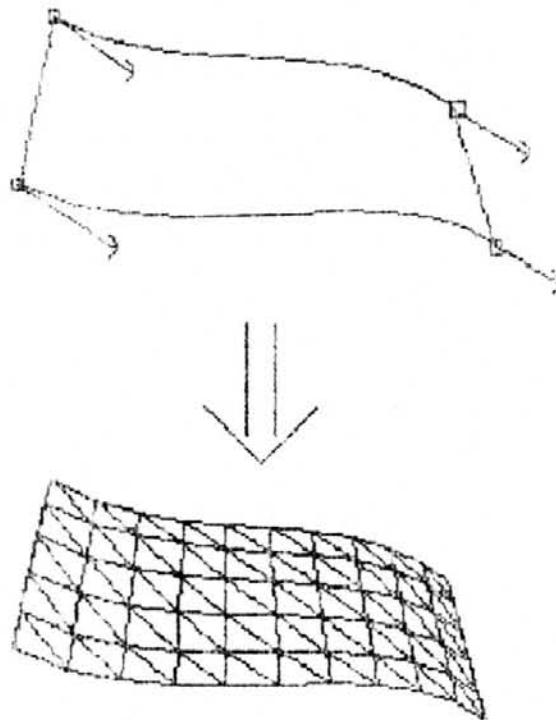


Figura 3.4 - Quatro curvas delimitando um "patch" de Coons.

onde:

$p_i$  são os pontos de controle e

$b_i$  são as funções de ponderação apresentadas no item 2.4;

ou em notação matricial:

$$p(t) = [1 \ t \ t^2 \ t^3] \ \frac{1}{6} \ Gbspl \ [p_0 \ p_1 \ p_2 \ p_3]^T$$

onde:

$$Gbspl = \begin{vmatrix} 1 & 4 & 1 & 0 \\ -3 & 0 & 3 & 0 \\ 3 & -6 & 3 & 0 \\ -1 & 3 & -3 & 1 \end{vmatrix}$$

As duas curvas que faltam para definir o "patch" podem ser facilmente obtidas utilizando-se curvas Hermite. Como visto no item 2.2 uma curva Hermite é definida por dois pontos de ancoragem e dois vetores de controle. Em notação matricial, tem-se a seguinte formulação para uma curva Hermite:

$$H(t) = [1 \ t \ t^2 \ t^3] \ Gherm \ [p_0 \ d_0 \ p_1 \ d_1]^T$$

onde:

$$G_{\text{herm}} = \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ -3 & -2 & 3 & -1 \\ 2 & 1 & -2 & 1 \end{vmatrix}$$

$p_0$  e  $p_1$  são os pontos de ancoragem e  $d_0$  e  $d_1$  são os vetores de controle.

Os dois pontos de ancoragem da curva Hermite coincidirão com os pontos extremos das curvas isoparamétricas usadas como curvas de ancoragem para a junção (pontos onde o parâmetro assume os valores zero e um). Como vetores de controle, é interessante usar vetores que garantam continuidade geométrica entre a curva e a superfície-base. Para tanto pode-se utilizar o vetor obtido pela derivação parcial da superfície-base em relação ao parâmetro que não foi fixado para a determinação da isoparamétrica de ancoragem nos pontos coincidentes com os extremos dessa última.

Por exemplo, supondo que se dispõe de dois "patches" definidos por "B-splines" cúbicos chamados  $S_1$  e  $S_2$  de pontos  $s_i(u,v)$ ,  $i=1,2$  e de curvas isoparamétricas em  $u$  chamadas  $A_i$ ,  $i=1,2$  de pontos  $a_i(t)$  definidas sobre  $S_1$  e  $S_2$ . Um ponto sobre estas curvas pode ser grafado tanto como  $s_i(u(t),v(t))$  como  $a_i(t)$ .

Nesta situação pode-se definir duas curvas  $H_i$ ,  $i=1,2$  de pontos  $h_i(s)$  que completam a descrição do "patch" de Coons pela determinação dos dois pontos de ancoragem e dos dois vetores de controle. Tem-se então para  $H_1$ :

$$p_0 = h_1(0) = s_1(uc, 0)$$

$$p_1 = h_1(1) = s_2(uc, 1)$$

$$d_0 = h'_1(0) = s'_{1v}(uc, 0)$$

$$d_1 = h'_1(1) = s'_{2v}(uc, 1)$$

onde:

$uc$  é o valor constante do parâmetro  $u$  que define a isoparamétrica;

$p_0$  e  $p_1$  são os pontos de ancoragem e

$d_0$  e  $d_1$  são os vetores de controle.

Para  $H_2$  desenvolve-se raciocínio análogo apenas substituindo-se o valor do parâmetro  $v$  na equação da superfície-base de zero para um.

Para simplificar a manipulação das equações das curvas pode-se expandir a notação matricial de forma a obter uma representação uniforme para as quatro curvas que descrevem a junção. Tem-se, então:

$$a_i(s) = \frac{(p_0+4p_1+p_2)}{6} + \frac{(-3p_0+3p_2)}{6}s + \frac{(3p_0-6p_1+3p_2)}{6}s^2$$

$$+ \frac{(-p_0+3p_1-3p_2+p_3)}{6}s^3, \quad i=1,2$$

$$h_i(t) = p_0 + d_0t + (-3p_0-2d_0+3p_1-d_1)t^2$$

$$+ (2p_0+d_0-2p_1+d_1)t^3, \quad i=1,2$$

De posse das quatro equações pode-se escrever a equação que descreve a superfície de junção ("patch" de Coons) como segue:

$$j(t,s) = Fd(t) a_1(s) + Fc(t) a_2(s)$$

$$+ Fd(s) h_1(t) + Fc(s) h_2(t)$$

$$- Fd(s) Fd(t) j(0,0)$$

$$- Fd(s) Fc(t) j(1,0)$$

$$- Fc(s) Fc(t) j(1,1)$$

$$- Fc(s) Fd(t) j(0,1).$$

Para gerar uma superfície de junção com continuidade  $G^1$  ao longo das curvas de ancoragem basta garantir que as funções de ponderação mantenham essa continuidade à medida em que os vetores tangentes dos pontos extremos do "patch" sejam ponderados ao longo da curva de ancoragem. Essas funções são /CON 67/:

$$F_c(w) = -2w^3 + 3w^2 \quad e$$

$$F_d(w) = 2w^3 - 3w^2 + 1 \quad .$$

O método apresentado possui como desvantagem o fato de que o uso de isoparamétricas não permite flexibilidade na escolha das curvas de ancoragem. Por outro lado, o mesmo apresenta algumas vantagens:

- a) A superfície de junção é facilmente gerada;
- b) É de fácil implementação em ambientes computacionais de pequeno porte e
- c) Garante-se a continuidade geométrica ( $G^1$ ) ao longo das curvas de ancoragem.

Na figura 3.5 pode-se ver uma superfície de junção entre dois "patches beta-spline" gerada utilizando-se isoparamétricas como curvas de ancoragem e um "patch" de Coons como superfície de junção.

3.1.2 Determinação de superfícies de junção quando as curvas de ancoragem são desconhecidas ou de difícil determinação:

Quando as equações das curvas de ancoragem são desconhecidas ou de difícil determinação, todas as facilidades citadas no item 3.1.1 desaparecem. Tem-se então que:

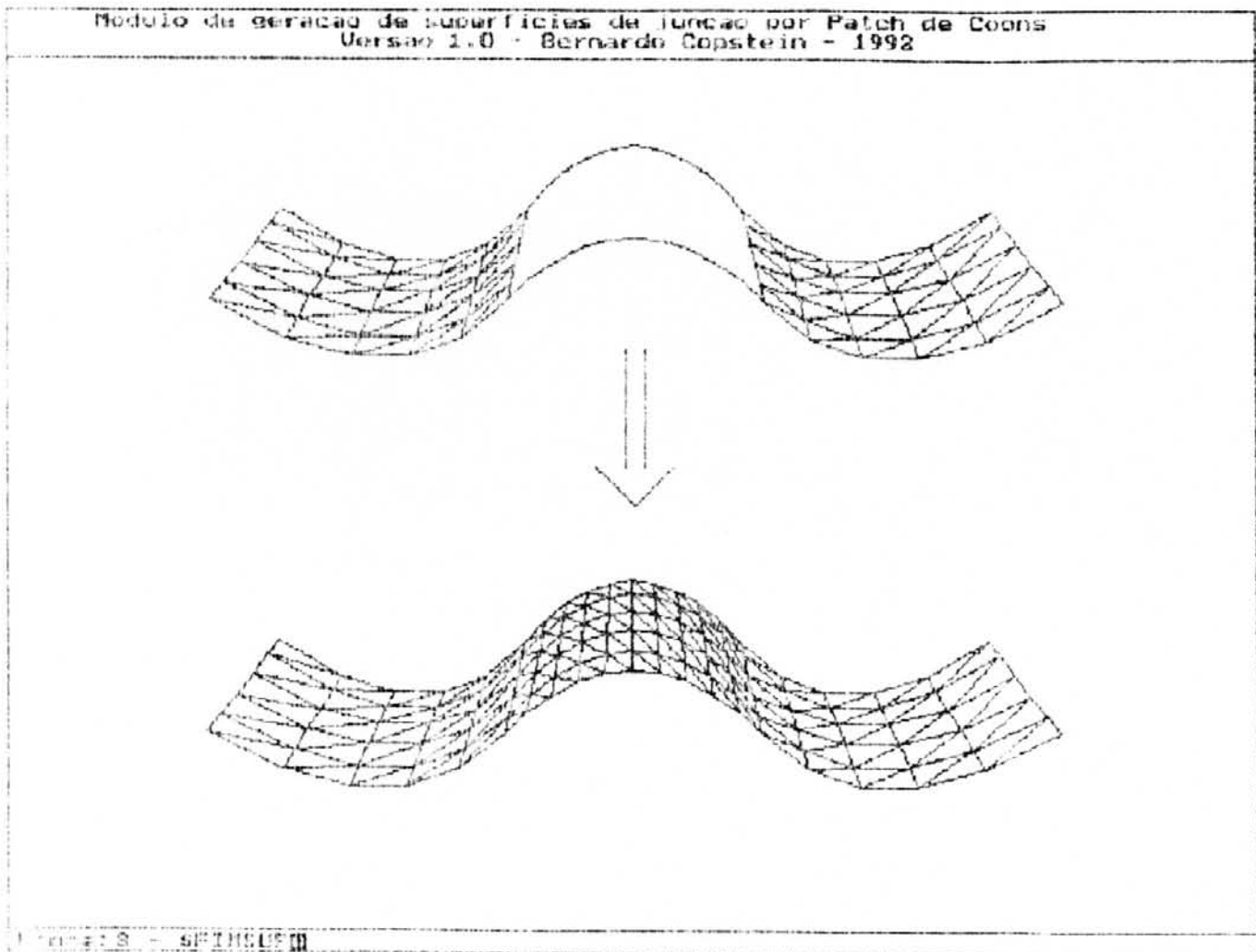


Figura 3.5 - Superfície de junção gerada utilizando-se um "patch" de Coons.



a) passa a existir uma etapa de determinação das curvas de ancoragem. Em geral esta fase é bastante custosa computacionalmente;

b) São necessários artifícios para o cálculo dos vetores de controle e

c) Se as curvas de ancoragem forem representadas por uma seqüência de pontos convenientemente amostrados, ficará difícil garantir algum tipo de continuidade entre as superfícies-base e a junção nos trechos correspondentes ao espaçamento entre os mesmos.

Quando a equação das curvas de ancoragem não está disponível, costuma-se determinar as mesmas a partir da interpolação de um conjunto de pontos amostrados sobre a superfície. Em geral, esses pontos são originários do cálculo de intersecção entre superfícies. Como os pontos interpolados possuem um espaçamento entre si, a curva resultante dessa interpolação não estará inteiramente contida na superfície. Isto gerará o aparecimento de descontinuidades entre a superfície-base e a junção.

Filip e Bardis procuram contornar o problema de descontinuidade interpolando a curva de ancoragem no espaço de parâmetro em vez de no espaço de objeto. Uma curva é

gerada por uma B-spline cúbica em um espaço paramétrico bidimensional. A curva de ancoragem é obtida pelo mapeamento dos pontos da curva descrita no plano paramétrico contra uma determinada superfície base.

Naturalmente esse mapeamento pode ser explicitamente representado como uma "spline" polinomial da seguinte forma:

Dados:

$S(u,v)$  é uma superfície paramétrica qualquer;  
 $C(t)$  é uma curva paramétrica qualquer definida em um espaço paramétrico  $uv$ .

Desta forma tem-se que para cada valor de  $t$  substituído na equação da curva  $C$ , obtém-se um par  $(u(t),v(t))$  que, se substituído na equação da superfície  $S$ , gera um ponto sobre a mesma. Para se obter a equação polinomial que descreve a curva resultante deste mapeamento basta substituir a equação da curva  $C$  em todas as ocorrências dos parâmetros  $u$  e  $v$  na equação da superfície. Desta maneira seria obtida uma equação polinomial paramétrica em função de um único parâmetro extremamente conveniente de se trabalhar.

Ocorre que, se tanto a superfície  $S$  como a curva  $C$

forem geradas por "beta-spline", tal substituição será extremamente trabalhosa devido à quantidade de termos envolvidos no processo.

O tipo de solução que se adota nestes casos, então, é utilizar um método de amostragem da curva de ancoragem que trabalhe no espaço paramétrico. Isso não é suficiente, entretanto, porque como se verá no capítulo quatro, o fato de se ter uma curva convenientemente amostrada em um espaço paramétrico não significa que a curva obtida pelo mapeamento desta no espaço do modelo também o esteja. Por estas razões, o algoritmo de determinação da curva de ancoragem deverá buscar a discretização em um espaço paramétrico, porém, baseado em informações retiradas do espaço do modelo.

No exemplo apresentado no item 3.1.1, a determinação dos vetores de controle é extremamente facilitada. Em outros tipos de situações (onde a curva de ancoragem não é uma isoparamétrica), mesmo que não seja possível obter o vetor de controle diretamente pela derivação parcial da superfície-base (como apresentado em 3.1.1), sempre é possível determiná-lo com auxílio do vetor tangente à curva de ancoragem. Uma maneira, por exemplo, é obtê-lo pelo produto vetorial entre o vetor normal à superfície e o vetor tangente à curva de ancoragem (ver cap.

5). Quando não se dispõe da equação, entretanto, inexistente a entidade matemática a partir da qual pode-se calcular esse vetor auxiliar. Nestes casos, é preciso criar artifícios para permitir o cálculo de tais vetores. Filip /FIL 89/ se utiliza de vetores auxiliares que ligam os pontos que serão unidos. Bardis /BAR 89/ interpola os pontos amostrados obtendo assim a equação de uma aproximação da curva de ancoragem. Neste trabalho o plano de corte usado para a determinação da curva de ancoragem é usado como elemento auxiliar.

### 3.1.3. Conclusões

Ao se analisar o método apresentado em 3.1.1, observa-se que apesar de suas vantagens o mesmo não oferece flexibilidade para o usuário final determinar as curvas de ancoragem. Passa a ser objetivo deste trabalho, então, a busca de uma solução que permita ao usuário final do sistema, uma maior gama de possibilidades no momento da determinação das curvas de ancoragem.

A superfície de junção proposta, então, não necessita que se disponha da descrição da curva de ancoragem no espaço do modelo, porque os valores dos

parâmetros são suficientes para o cálculo dos vetores necessários à definição da junção.

A junção será definida por interpolação de Hermite. A formulação de Hermite necessita dos pontos de partida e chegada em ambas as superfícies-base (pontos mapeados) e de vetores de partida e chegada dependentes das superfícies e calculados nestes pontos.

### 3.2 O método proposto

O método a ser apresentado em seguida é inspirado nos trabalhos de Bardis e Filip sobretudo pela generalidade e flexibilidade por eles descritos. Propõe-se, então, um método para geração de superfícies de junção contínuas (continuidade geométrica) entre duas superfícies paramétricas compostas "beta-spline".

Foram escolhidas superfícies "beta-splines" uniformes bicúbicas para serem usadas como superfícies-base pela sua flexibilidade em modelagem geométrica bem como pela simplicidade de geração das mesmas e sua conseqüente adaptação a ambientes computacionalmente limitados. De qualquer forma, o método não se utiliza do poliedro de controle que define as superfícies "beta-spline" sendo,

portanto, facilmente adaptado a qualquer tipo de superfície paramétrica composta ou não.

Para os propósitos deste trabalho, a modelagem de um objeto que inclui superfícies de junção será dividida em quatro etapas:

- a) Modelagem das superfícies base;
- b) Determinação de curvas de ancoragem sobre cada uma das superfícies base;
- c) Cálculo dos vetores de partida e chegada que orientarão a geração da junção (vetores de controle) e
- d) Geração da superfície de junção propriamente dita.

Esta divisão tomou por base o que se encontra em artigos sobre o assunto, tais como o de Bardis [BARD 89], Filip [FIL 89], Mortenson [MOR 85] e Warren [WAR 89]. A definição explícita das mesmas é importante na medida em que permitirá que, na definição de várias superfícies de junção, compartilhem-se elementos tais como superfícies-base, curvas de ancoragem e vetores de controle. Desta forma, reduz-se o tempo de cálculo destes elementos e economiza-se espaço de armazenamento. Outra

grande vantagem é que esta divisão facilita a construção de ferramentas modulares de "software" para a geração de superfícies de junção uma vez que os diferentes módulos podem corresponder às diferentes etapas.

A primeira etapa, consiste na definição e posicionamento das superfícies a serem unidas. É importante observar que o formato final das junções é dependente da posição em que as superfícies-base se encontram uma em relação a outra. Esta característica pode ser explorada de maneira proveitosa quando se deseja representar objetos flexíveis como se verá no final deste trabalho.

As curvas de ancoragem serão determinadas pela intersecção entre um plano de corte e cada uma das superfícies-base individualmente. Esta técnica permite a definição implícita de uma família muito grande de curvas, incluindo as curvas isoparamétricas tradicionalmente usadas neste tipo de trabalho.

O cálculo dos vetores de controle será feito levando em consideração apenas a superfície-base e o plano de corte que definiu a curva de ancoragem, diferentemente do método proposto por Filip, onde os vetores de partida e chegada são dependentes do posicionamento das duas superfícies-base.

A geração da superfície de junção será feita usando interpolação cúbica por Hermite, como sugerido por Filip. Esse método foi adotado por sua grande simplicidade. A representação por Hermite pode ser posteriormente convertida em uma representação "beta-spline", de forma que se obtenha uma uniformidade na representação das superfícies como foi sugerido por Bardis /BARD 89/. A conversão entre as formulações Hermite e "bspline" não será abordada neste trabalho. Documentação sobre o assunto é encontrada no artigo "An Adaptative Subdivision Method with Crack Prevention for Rendering Beta-Spline Objects" [BAR 87b].



#### 4. DETERMINAÇÃO DAS CURVAS DE ANCORAGEM

Neste capítulo será apresentado o método de determinação de curvas de ancoragem. Será apresentado o que se entende por "curva de ancoragem" no contexto deste trabalho, como as mesmas serão representadas, como se classificam os algoritmos de determinação de intersecção entre superfícies e, finalmente, o método utilizado.

##### 4.1 O que são curvas de ancoragem

No contexto deste trabalho serão chamadas de curvas de ancoragem aquelas curvas perfeitamente ajustadas sobre as superfícies que se pretende unir. Estas curvas determinam o formato da junção no ponto de contato entre a superfície de junção e a superfície-base. Como as superfícies de junção são determinadas a partir de duas superfícies-base, deduz-se que é necessário definir uma curva de ancoragem sobre cada uma das superfícies-base envolvidas.

Considera-se que uma curva paramétrica  $C$  de parâmetro " $t$ " está perfeitamente ajustada em uma superfície paramétrica  $S$  de parâmetros " $u$ " e " $v$ " quando qualquer ponto amostrado da curva é também um ponto da superfície, ou seja,

para qualquer valor do parâmetro "t", usado para amostrar um ponto  $(x_c, y_c, z_c)$  sobre a curva C existe um par de parâmetros  $(u, v)$  correspondente que gerará um ponto  $(x_s, y_s, z_s)$  sobre a superfície onde  $x_s = x_c$ ,  $y_s = y_c$  e  $z_s = z_c$ .

Sejam  $S_1$  e  $S_2$  de parâmetros "u" e "v" duas superfícies-base a serem unidas. Pode-se dizer que

$$\mathbf{c}_i(t) = (u_i(t), v_i(t)), \quad i = 1, 2 \quad (4.1 - 1)$$

é a representação da curva de ancoragem no espaço paramétrico. Ainda

$$\mathbf{m}_i(t) = \mathbf{s}_i(\mathbf{c}_i(t)), \quad i = 1, 2 \quad (4.1 - 2)$$

são as curvas de ancoragem no espaço objeto.

#### 4.2 Representação de uma curva de ancoragem

Expandir a equação (4.1 - 2) não é viável devido à quantidade de termos envolvidos no processo como visto no capítulo 3. Este fato não chega a ser um problema, pois os vetores necessários para a determinação da superfície de junção, a cada ponto amostrado da curva de ancoragem, dependem única e exclusivamente das superfícies-base.

Uma curva de ancoragem pode, então, ser facilmente descrita usando a equação (4.1 - 2) (fig 4.1). Esta representação é vantajosa na medida em que, independente do tipo de interpolação ou aproximação da curva feita no espaço paramétrico, os pontos gerados da superfície estarão sempre

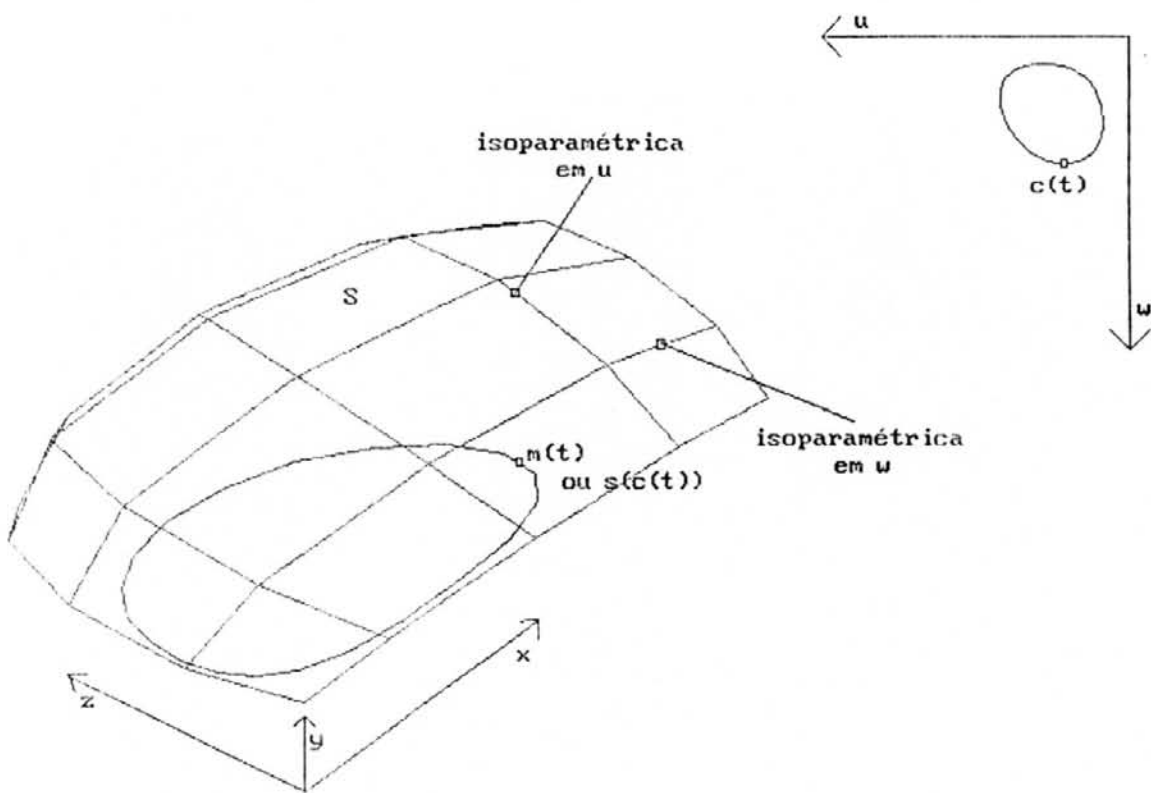


Figura 4.1 - Curva de ancoragem gerada pelo mapeamento de uma curva definida em um espaço paramétrico em uma superfície paramétrica.

inteiramente contidos na mesma. Tal não ocorre se os pontos são interpolados no espaço do modelo. Ao fazê-lo, corre-se o risco da geração de pontos apenas próximos à superfície-base, porém, não contidos na mesma. Esta aproximação gerará falhas ou descontinuidades entre a curva e a superfície.

J. J. Chen *et ali* /Chen 88/ em seu artigo "Predictor - Corrector Type of Intersection Algorithm for  $C^2$  Parametric Surfaces" alerta que se uma curva qualquer for amostrada sobre uma superfície-base apenas a partir de sua descrição no espaço paramétrico, corre-se o risco de perder detalhes importantes da curva final. Ao proceder desta maneira, não se consideram os detalhes resultantes da deformação que ocorre quando do mapeamento da curva definida no espaço paramétrico sobre a superfície (fig 4.2).

A partir dessas observações, conclui-se que a forma de representação escolhida para a curva de ancoragem deve permitir uma amostragem conveniente dos pontos da mesma, ou seja, uma amostragem que permita trabalhar com a curva com o grau de detalhe que se desejar. Será levado em conta também, neste trabalho, que a forma de representação escolhida para a curva de ancoragem deve permitir a construção de uma ferramenta de "software" onde o usuário possa definir interativamente esta curva.

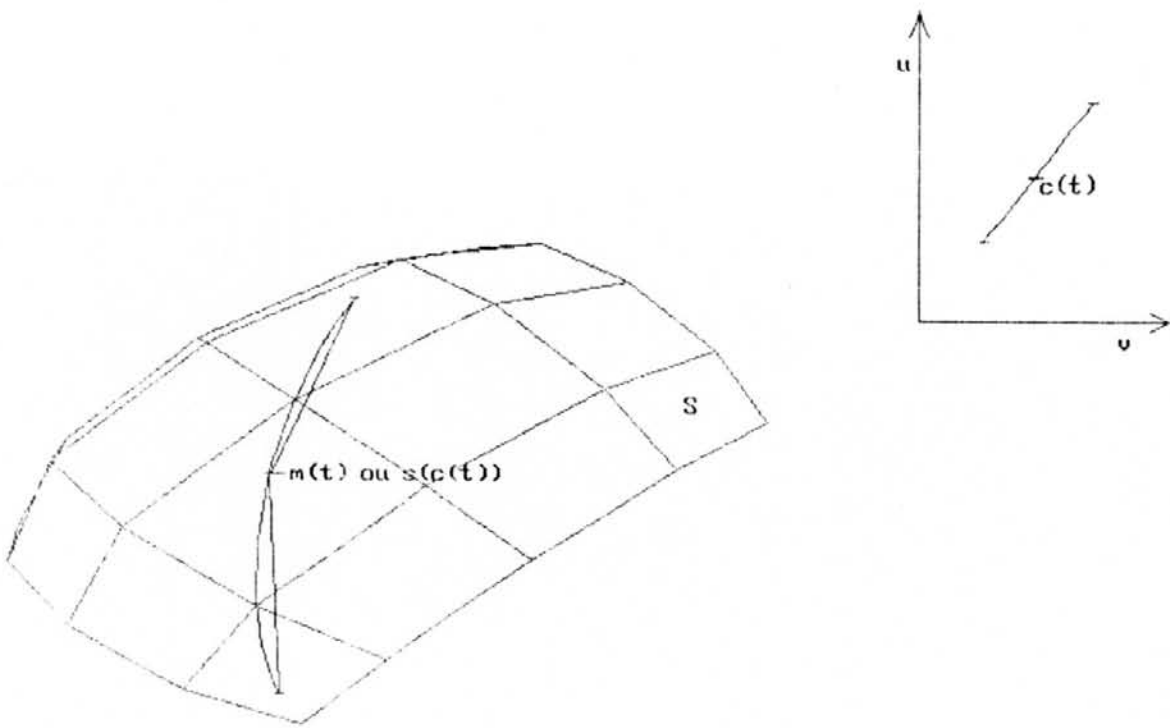


Figura 4.2 - Deformação decorrente do mapeamento de uma curva definida no espaço paramétrico sobre uma superfície.

Mortenson /MOR 85/ cita a utilização de curvas isoparamétricas como uma solução simples para este tipo de problema. Como já foi comentado no capítulo 3 , porém, o conjunto de curvas possível de se determinar sobre um "patch" usando isoparamétricas não oferece um conjunto mínimo razoável para a determinação de curvas de ancoragem.

J. J. Chen *et ali* /CHEN 88/ sugere a determinação de curvas de ancoragem pela intersecção entre duas superfícies quaisquer. O método proposto por Chen apresenta dois inconvenientes se forem considerados os objetivos deste trabalho: construir uma ferramenta interativa sobre um ambiente computacional limitado. O primeiro problema diz respeito a dificuldade de se construir uma interface interativa onde o usuário possa definir com facilidade uma curva a partir da intersecção de duas superfícies. O principal problema de tal tarefa diz respeito a compatibilizar as técnicas de exibição de superfícies com um tempo de resposta razoável. O outro problema do artigo de Chen é que o método apresentado para o cálculo da intersecção terá um tempo de resposta bastante elevado.

Considerou-se, porém, que a idéia apresentada no artigo, ou seja, a determinação de curvas perfeitamente ajustadas a superfícies, implicitamente pela intersecção entre duas superfícies, é bastante atraente por diversas

razões: a família de curvas que se consegue determinar é bastante grande; as curvas definidas estão sempre perfeitamente ajustadas nas superfícies envolvidas; a equação implícita de tais curvas é simples de se obter; existem diversas classes de algoritmos que permitem amostrar pontos sobre essas curvas com o grau de detalhe que se pretende.

A idéia de Chen foi, então, modificada de maneira a simplificar tanto os cálculos envolvidos como a construção de uma interface interativa.

Decidiu-se usar um plano de corte contra a superfície base sobre a qual se deseja determinar a curva de ancoragem. Disto decorre a geração de uma família de curvas onde o número de curvas é evidentemente menor, porém, suficiente para efeito de testes com um protótipo. Por outro lado foi levado em conta, também, o fato de que o nível de abstração exigido para trabalhar-se com um plano de corte é bem menor do que o exigido para se trabalhar com duas superfícies quaisquer.

### 4.3 Classificação dos algoritmos de determinação de intersecções entre superfícies

Gabor Lukacs /LUK 89/ classifica as técnicas de cálculo de intersecções entre superfícies em cinco diferentes categorias:

- Subdivisão recursiva;
- Métodos algébricos;
- Avaliação sobre reticulados;
- Métodos de deslocamentos ("Marching Methods") e
- Métodos topológicos algébricos.

Métodos de subdivisão recursiva se baseiam na mesma idéia dos algoritmos de divisão e conquista, isto é, um problema de difícil solução é subdividido em problemas de menor complexidade. Em geral, as superfícies estão envolvidas por envelopes poliédricos convexos e então a intersecção entre os mesmos é testada. Se os envelopes poliédricos se interseccionam, então as superfícies contidas nos envelopes são subdivididas e as subdivisões envolvidas por novos poliedros menores que serão novamente testados. Os segmentos de superfície cujos envelopes não se interseccionam são descartados; os demais são novamente subdivididos. O processo é repetido, geralmente, até que os



poliedros sejam menores que uma determinada tolerância que permite aceitar os poliedros como pontos da curva. Em geral, essa tolerância é suficientemente pequena para que a curva possa ser, então, refinada por outros métodos tais como o de Newton-Raphson.

Métodos algébricos são usados em casos simples (como intersecção de quádricas) onde a intersecção pode ser obtida analiticamente.

Em métodos que usam avaliação sobre reticulados define-se uma malha de curvas espaciais sobre a primeira superfície. Testa-se, em seguida, os nós do reticulado contra a segunda superfície procurando encontrar pontos que sejam comuns às duas superfícies. Em geral os métodos de reticulado não são usados isoladamente. Em verdade, o que se usa é um método híbrido onde o método de reticulado fornece os dados que possibilitarão a execução de um método de deslocamento.

Métodos de deslocamento são em geral métodos numéricos. Primeiramente se estabelece um ponto sobre a curva de intersecção (em geral por um método de reticulado): o ponto de partida. Em seguida, segundo algum critério, soma-se um incremento ao primeiro ponto procurando uma estimativa do próximo ponto da curva. O ponto estimado é

então refinado usando-se o método de Newton ou o dos mínimos quadrados. O nome dos métodos de deslocamento se deve ao fato de serem os pontos sobre a curva de intersecção gerados na ordem de ocorrência produzindo, então, um "deslocamento" orientado. Uma das grandes vantagens destes métodos é a possibilidade de controlar o grau de detalhe com o qual se deseja obter a intersecção. Para controlar o grau de detalhe basta variar, em cada aproximação, o tamanho do próximo "deslocamento".

Finalmente os métodos topológicos algébricos baseiam-se na teoria que afirma ser possível encontrar todas as  $n$ -soluções para equações polinomiais de  $n$ -variáveis. A idéia básica destes métodos consiste em, a partir do sistema original, encontrar um sistema mais simples onde todas as soluções sejam conhecidas.

Segundo Gabor Lukacs /LUK 89/, nenhuma dessas classes de métodos é consagrada. Os métodos algébricos (topológicos ou não) encontram-se em fase experimental; nos métodos de reticulado se tem grande dificuldade para controlar o grau de detalhe da curva que se deseja; o método de subdivisão recursiva e os métodos de deslocamento são os de maior interesse prático atualmente. O método de subdivisão recursiva se adapta bem à superfícies Bèzier e "B-spline", nenhum detalhe da curva é perdido, mas a fase de

ordenação dos pontos é bastante complexa.

Por outro lado, os métodos de deslocamento são de natureza mais simples, apresentam vantagens tanto na ordenação dos pontos obtidos quanto no controle do grau de detalhe desejado. Os métodos de deslocamento podem ser usados inclusive em superfícies não polinomiais sendo, portanto, uma solução mais genérica.

Independente do método utilizado, o principal problema é a representação final da curva. Uma vez que interpolação dos pontos (como já foi comentado no item 4.2) pode produzir pontos que não estão perfeitamente ajustados na superfície-base, a única solução é armazenar toda a seqüência de pontos obtidos. Tem-se, então, que armazenar, para cada ponto obtido, os valores paramétricos mais as três componentes do sistema Euclidiano. Isto não é nada econômico do ponto de vista de gasto de memória principal, porém, nenhuma outra solução sem perda de informação tem sido apresentada.

#### 4.4 O método utilizado

No contexto deste trabalho, as curvas de ancoragem serão definidas, uma de cada vez, pela intersecção entre as superfícies-base a serem unidas e um plano a ser posicionado interativamente.

Será usado um método de deslocamento aliado a um método de reticulado. Uma vez que as superfícies-base são superfícies compostas ("beta-splines" uniformes de grau três) o algoritmo de caminhamento será aplicado sobre cada "patch" individualmente. Em seguida, os diversos segmentos de curva encontrados sobre os "patches" serão ordenados.

No item 4.4.1, será apresentado o algoritmo de determinação da curva de intersecção entre um plano qualquer e uma superfície paramétrica.

No item 4.4.2, serão listadas algumas idéias apresentadas por Mortenson /MOR 85/ e que foram utilizadas para simplificar etapas do algoritmo.

No item 4.4.4 demonstra-se o cálculo da intersecção entre uma curva paramétrica e o plano  $Z=0$ . Isto ajudará o entendimento do item 4.4.8.

Nos itens de 4.4.5 a 4.4.11, serão detalhados os passos do algoritmo e por isto requerem uma atenção especial.

#### 4.4.1 O Algoritmo de movimentação sobre a curva

Será apresentado a seguir o algoritmo usado para a determinação da curva de ancoragem obtida pela intersecção de um plano de corte e uma superfície paramétrica (no caso um "patch" beta-spline). Para simplificar o método, aplicaram-se transformações geométricas sobre o plano de corte de forma a igualá-lo ao plano  $Z=0$  conforme sugerido em 4.4.2.

Para garantir que os pontos obtidos durante o deslocamento estão inteiramente contidos na superfície, o mesmo será feito no espaço paramétrico, porém, a determinação do passo de avanço a cada novo ponto a ser aproximado será calculado levando-se em conta a diferença de ângulo entre os vetores tangentes à curva no espaço objeto. Garante-se, desta forma, que todos os detalhes da curva no espaço objeto serão levados em conta e que pode-se controlar o grau de detalhe que desejado pela determinação do ângulo máximo que será tolerado entre dois vetores tangentes consecutivos (fig. 4.3).

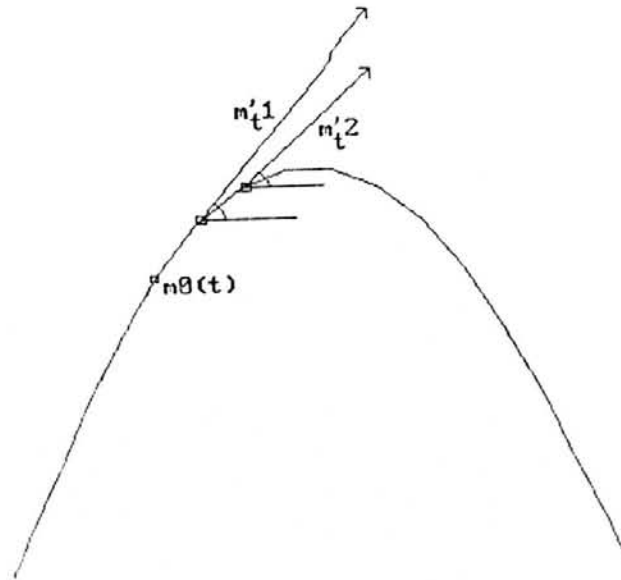


Figura 4.3 - Controle do grau de amostragem de uma curva pela diferença de ângulo entre os vetores tangentes.

Apresenta-se, a seguir, o algoritmo para determinação de uma curva de ancoragem sobre uma superfície composta cúbica Beta-spline pela intersecção da superfície com um plano qualquer:

1. Define-se, iterativamente, o plano de corte que determinará a curva de ancoragem desejada
2. Aplicam-se transformações geométricas sobre a superfície de forma a obter a mesma curva de ancoragem usando o plano  $Z=0$ ;
3. Para cada um dos "patches" da superfície cujo teste do envelope poliédrico indica ser um "patch" candidato a conter um segmento da curva de intersecção:
  - 3.1. Procura-se dois pontos pertencentes à curva de intersecção usando um método de reticulado. Os pontos encontrados são os pontos iniciais; caso a intersecção do "patch" contra o plano origine uma curva descontínua, devem ser encontrados pontos iniciais para cada um dos trechos de curva. Desta forma cada um dos trechos de curva será tratado como uma curva distinta.
  - 3.2. Se foram encontrados os pontos iniciais:

- 3.2.1. Classifica-se a curva que será percorrida em função dos pontos iniciais;
- 3.2.2. Para cada um dos pontos iniciais encontrados
  - 3.2.2.1. Enquanto não ocorrer uma das condições de fim, calcula-se o próximo ponto da seguinte forma:
    - 3.2.2.1.1. Calcula-se uma aproximação ao vetor tangente paramétrico no último ponto obtido sobre a curva definida no espaço de parâmetro. Vetor  $\mathbf{v}$ ;
    - 3.2.2.1.2. Normaliza-se o vetor  $\mathbf{v}$ ;
    - 3.2.2.1.3. Calcula-se o fator de multiplicação do vetor  $\mathbf{v}$  (passo de avanço  $\tau$ ) levando em conta a diferença de ângulo entre os vetores tangentes à curva de intersecção no espaço do objeto;



- 3.2.2.1.4. Aproxima-se o próximo ponto da curva (no espaço paramétrico) por  $v^* \tau$ ;
- 3.2.2.1.5. Refina-se o ponto aproximado usando o método de Newton;
- 3.2.2.2. Armazena-se o segmento de curva encontrado;
- 3.2.2.3. Testam-se os pontos candidatos a pontos iniciais restantes de maneira a verificar se os mesmos não coincidem com o ponto final do último segmento de curva encontrado. Em caso positivo, elimina-se o ponto coincidente da lista de pontos candidatos a pontos iniciais;
- 3.2.3 Elimina-se os pontos em excesso de cada um dos segmentos encontrados sobre os diversos "patches";
4. Ordenam-se os segmentos de curva encontrados sobre os diversos "patches";

#### 4.4.2. Simplificações

Tendo por objetivo otimizar ao máximo o cálculo das curvas de ancoragem, foram feitas algumas simplificações inspiradas nas observações feitas por Mortenson /MOR 85/:

a) A determinação da curva de intersecção entre um plano e uma superfície pode ser obtida pelo cálculo das intersecções entre o plano e um conjunto de curvas isoparamétricas sobre a superfície cuidadosamente selecionadas. A seleção das curvas isoparamétricas deve ser feita de modo a garantir que nenhum detalhe da curva de intersecção seja perdido.

b) O cálculo da intersecção entre uma curva polinomial paramétrica e um plano qualquer pode ser simplificado se forem aplicadas ao plano o conjunto de transformações geométricas necessárias para igualar-se o plano ao plano  $Z=0$ .

#### 4.4.3 Determinação dos pontos de partida para o algoritmo de movimentação

O algoritmo de movimentação necessita de dois pontos de partida (ver item 4.4.8) que pertençam à curva de intersecção definida pelo plano de corte sobre o "patch". Em função desses dois pontos iniciais será definido o início bem como o sentido da movimentação.

A determinação dos pontos iniciais é feita, em linhas gerais, utilizando-se um algoritmo de reticulado. Dada, porém, a importância desta fase para o sucesso da movimentação faz-se necessário o detalhamento de alguns aspectos.

Por estar se tratando com "patches" cúbicos, pode-se considerar que a curva resultante da intersecção de um plano com um "patch" cúbico se enquadra em uma das três seguintes situações:

a) A curva intersecciona as bordas do "patch" (fig. 4.4);

b) A curva tangencia as bordas do "patch" (fig 4.5);

c) A curva não intersecciona nem tangencia as bordas do "patch";

Ignorando-se o caso da letra b, pode-se considerar que a curva resultante da intersecção de um plano com um "patch" cúbico é uma curva fechada caso não existam pontos da mesma pertencentes às bordas do "patch" e, no caso contrário, é uma curva aberta.

Essa classificação será bastante importante para a determinação das condições de fim de execução do processo de movimentação (ver item 4.4.10).

Nas situações em que a intersecção entre um "patch" cúbico e um plano de corte não gera curvas descontínuas, a determinação do ponto inicial para o algoritmo de movimentação pode funcionar como descreve o algoritmo 1.

Algoritmo 1 - Algoritmo para determinação do ponto inicial da curva de intersecção quando a mesma é contínua:

1. Aplica-se um algoritmo de reticulado sobre as bordas do "patch" buscando determinar pontos pertencentes à curva de intersecção entre o

"patch" e o plano de corte sobre as bordas;

2. Caso não existam pontos pertencentes à curva de intersecção sobre bordas do "patch", classifica-se a curva de intersecção como fechada e aplica-se um algoritmo de reticulado sobre todo o "patch", buscando um ponto pertencente à intersecção;

3. Se for encontrado um ponto inicial, então facilmente pesquisa-se a vizinhança deste ponto em busca do segundo ponto pertencente à curva de intersecção necessário ao algoritmo de movimentação. Caso contrário, não existe intersecção entre o plano de corte e o "patch" em questão.

Pelo algoritmo 1, as curvas que têm ponto de tangência com as bordas do "patch" serão tratadas como curvas abertas. Caso existam mais de um ponto de tangência com diferentes bordas do "patch", o algoritmo de movimentação será capaz de determinar apenas um trecho da curva de intersecção devido as condições de fim de execução da movimentação (ver item 4.4.8).

Ocorre que a intersecção entre um "patch" e um plano de corte nem sempre gera uma curva contínua e o algoritmo 1 deve ser alterado.

Neste trabalho, curvas descontínuas determinadas pela intersecção de um "patch" cúbico com um plano de corte terão seus trechos tratados como se fossem curvas distintas. Dessa forma é necessário determinar um ponto de partida para cada um dos trechos de curva. Assim, para um mesmo "patch" existirá mais de um ponto de partida associado e o algoritmo de movimentação terá de ser executado mais de uma vez.

Por serem "patches" cúbicos podem ser feitas algumas simplificações:

a) Caso a curva resultante da intersecção com o plano de corte não contiver pontos de intersecção com as bordas do "patch", então a curva de intersecção é contínua e fechada. Nestes casos o tratamento é idêntico ao exposto no algoritmo 1.

b) Se a curva resultante da intersecção com o plano de corte contiver pontos de intersecção com as bordas

dos "patches", então os diferentes trechos de curva serão curvas abertas. Neste caso todos os pontos pertencentes à curva de intersecção resultante que coincidirem com as bordas do "patch" deverão ser anotados como pontos candidatos a pontos de início para o algoritmo de movimentação.

O fato de se considerar todos os pontos pertencentes à curva de intersecção que se localizarem sobre as bordas do "patch" (fig 4.6) como pontos candidatos a pontos iniciais implicará anotar pontos em excesso pois obrigatoriamente se estará anotando sempre os pontos inicial e final de cada segmento de curva. Por esta razão, após a aplicação do algoritmo de movimentação a partir de um dos pontos anotados como candidatos a pontos iniciais, será necessário varrer a lista dos mesmos com o objetivo de verificar se um deles não coincide com o ponto final da curva recém percorrida. Caso isto ocorra, este ponto deverá ser eliminado da lista evitando que o algoritmo de movimentação seja disparado a partir do mesmo. Desta maneira, evita-se que um mesmo segmento de curva seja gerado duas vezes.

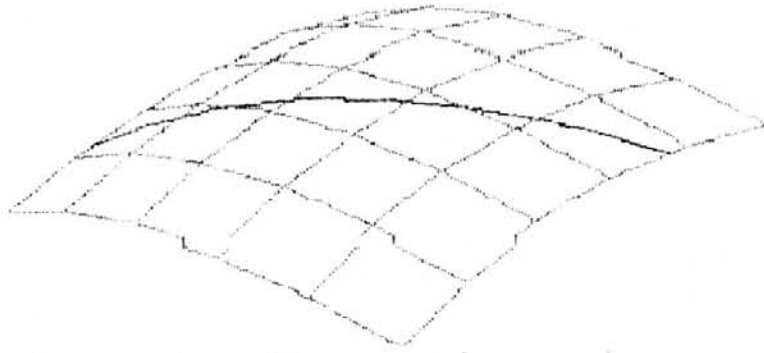


Figura 4.4 - Curva que intersecciona as bordas do "patch"



Figura 4.5 - Curva que não intersecciona as bordas do "patch"



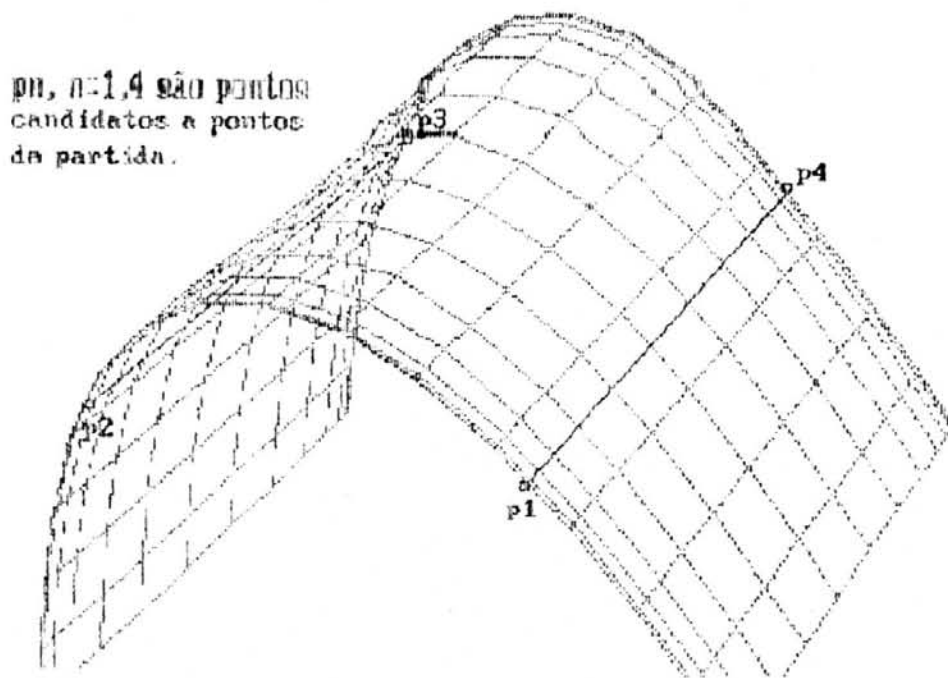


Figura 4.6 - Pontos candidatos a pontos iniciais de segmentos de uma curva descontínua gerada pela intersecção de um "patch" com um plano de corte.

#### 4.4.4 Cálculo da intersecção entre uma curva e o plano $Z = 0$

O cálculo desta intersecção será feito usando o método de Newton, tal como apresentado no "Handbook of Mathematical Formulas" /BART 74/, para refinar uma aproximação inicial.

Pelo método de Newton, tem-se que, uma vez conhecida uma aproximação de uma raiz de uma equação polinomial, pode-se obter uma aproximação mais exata da raiz.

Dada uma função  $f(x)$  contínua em  $[a;b]$  e satisfazendo  $f(a).f(b) < 0$ , o método de Newton consiste em, a partir de uma aproximação inicial  $x_1 \in [a;b]$  da raiz de  $f(x)=0$ , melhorar a exatidão de  $x_1$  através da fórmula iterativa:

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}, \quad f'(x_i) \neq 0$$

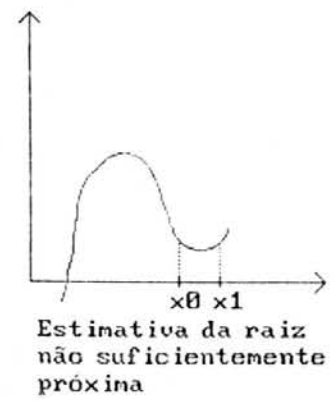
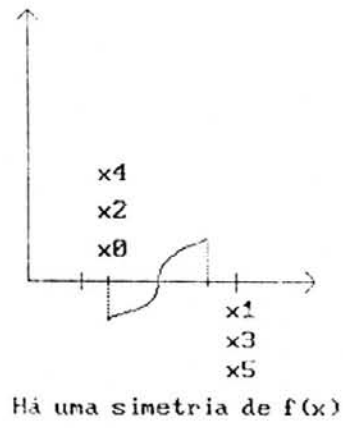
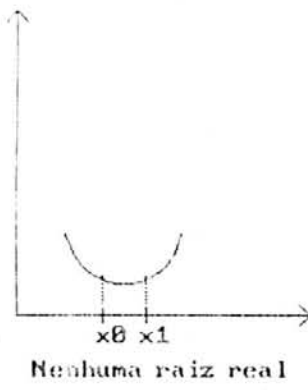


Figura 4.7 - Problemas do método de Newton.

Quando o método de Newton funciona, produz resultados rápidos em poucas iterações. Outras vezes, no entanto, o método de Newton não converge, mas oscila indefinidamente. Isto ocorre se não há raiz real, se há simetria de  $f(x)$  em torno de um ponto  $\bar{x}$  ou se a estimativa inicial estiver tão distante da raiz correta que uma outra parte da função induz um laço na seqüência iterativa (fig. 4.7 ).

No caso em questão, afirma-se que o ponto de intersecção entre a curva e o plano é o ponto onde a distância entre ambos é igual a zero. Busca-se, então, determinar as raízes da função distância entre o plano e a curva. A função distância será

$$d(\mathbf{c}(t), \pi) = 0$$

onde

$d$  é a função distância;

$\mathbf{c}(t)$  é um ponto sobre a curva e

$\pi$  é o plano.

Então, tem-se que a distância do ponto  $\mathbf{c}(t) = (x_0, y_0, z_0)$  ao plano  $\pi = Ax + By + Cz + D$  é

$$d = \frac{Ax_0 + By_0 + Cz_0 + D}{\sqrt{A^2 + B^2 + C^2}} \quad /BART 74/ .$$

Se for fixado o plano  $Z = 0$ , então tem-se que

$$A = 0, B = 0, C = 1, D = 0 .$$

Logo,  $d = z(t)$ .

No caso, o cálculo da distância será dado por

$$d = |z(t)| .$$

Para que seja aplicado o método de Newton, é necessário calcular  $d'_t(t)$ . Para tanto será considerado que:

$$|z(t)| = \text{SGN}(z(t)) \cdot z(t)$$

onde

$\text{SGN}(z(t))$  é a função que devolve o sinal de  $z(t)$ .

Então, tem-se que

$$\text{SGN}(x) = \begin{cases} -1 & \text{se } x < 0 \\ 0 & \text{se } x = 0 \\ 1 & \text{se } x > 0 \end{cases} .$$

Derivando  $\text{SGN}(z(t)) \cdot z(t)$  tem-se

$$\text{SGN}(z(t)) \cdot z'_t(t) \quad .$$

Então,

$$d'_t(t) = z'_t(t), \text{ se } z(t) > 0 ;$$

$$d'_t(t) = -z'_t(t), \text{ se } z(t) < 0 \text{ e}$$

$$d'_t(t) = 0, \text{ se } z(t) = 0 \text{ (caso indesejado).}$$

#### 4.4.4 Determinação dos vetores tangentes à curva de intersecção

Uma vez determinado um ponto pertencente à intersecção de duas superfícies quaisquer e à curva a qual ele pertence, pode-se obter um vetor tangente a esta curva, pelo produto vetorial entre os respectivos vetores normais às superfícies neste mesmo ponto (fig 4.8).

Então, se

$$\mathbf{m}(t) = \mathbf{s}_1(u_1, v_1) = \mathbf{s}_2(u_2, v_2) = (x, y, z)$$

onde

$\mathbf{m}(t)$  é o ponto sobre a curva determinada pela

intersecção das superfícies  $S_1$  e  $S_2$ ,

tem-se que

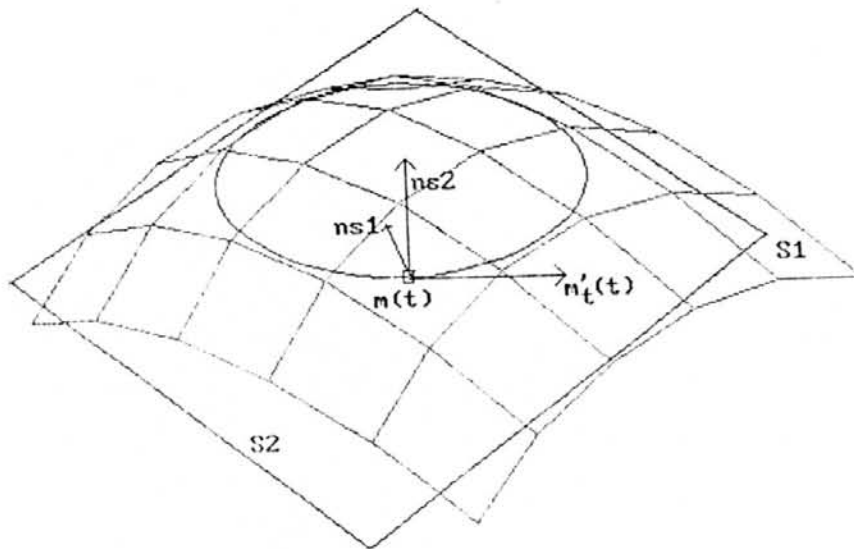


Figura 4.8 - Obtenção do vetor tangente à curva de ancoragem

$$\mathbf{m}'_t(t) = \mathbf{ns}_1(u_1, v_1) \times \mathbf{ns}_2(u_2, v_2)$$

onde

$$\mathbf{ns}_i = \frac{\mathbf{s}_i(u_i, v_i)}{du_i} \times \frac{\mathbf{s}_i(u_i, v_i)}{dv_i}, \quad i = 1, 2.$$

Simplificando para o caso do plano  $Z = 0$ ; tem-se que

se  $S_1$  é uma superfície paramétrica qualquer e  $S_2$  é o plano  $Z = 0$ , então a normal a superfície será

$$\mathbf{ns}_1 = \frac{\mathbf{s}_1(u_1, v_1)}{du_i} \times \frac{\mathbf{s}_1(u_1, v_1)}{dv_i} = \begin{vmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ \frac{sx}{du} & \frac{sy}{du} & \frac{sz}{du} \\ \frac{sx}{dv} & \frac{sy}{dv} & \frac{sz}{dv} \end{vmatrix}$$

onde

$$\mathbf{i} \left( \frac{sy}{du} \frac{sz}{dv} - \frac{sy}{dv} \frac{sz}{du} \right) = \alpha,$$

$$\mathbf{j} \left( \frac{sz}{du} \frac{sx}{dv} - \frac{sx}{du} \frac{sz}{dv} \right) = \beta \text{ e}$$

$$\mathbf{k} \left( \frac{sx}{du} \frac{sy}{dv} - \frac{sy}{du} \frac{sx}{dv} \right) = \delta.$$

A normal ao plano será  $\mathbf{ns}_2 = (0, 0, 1)$  e o vetor tangente procurado será

$$\mathbf{m}'_t(t) = \begin{vmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ \alpha & \beta & \delta \\ 0 & 0 & 1 \end{vmatrix} \text{ ou}$$



$$\mathbf{t}'_t(t) = (\beta, -\alpha, 0).$$

#### 4.4.6 Movimentação no espaço paramétrico

Seja  $m_i$  o  $i$ -ésimo ponto sobre a curva de intersecção de componentes  $(x, y, z)$ , dir-se-á que  $c_i$  é o ponto correspondente no espaço paramétrico de componentes  $(u_1, v_1)$  ou  $(u_2, v_2)$  dependendo de qual superfície seja usada como referência.

Então, se  $c_i$  e  $c_{i+1}$  forem dois pontos consecutivos sobre a curva no espaço paramétrico, pode-se aproximar  $c_{i+1}$  a partir de  $c_i$  da seguinte forma (fig 4.6):

$$\begin{aligned} c_{i+1} &= c_i + |\tau| \cdot \delta \text{ ou } (u_{i+1}, v_{i+1}) \\ &= (u_i, v_i) + (\delta_{\alpha u}, \delta_{\beta v}) \end{aligned} \quad (4.6.2 - 1)$$

onde

$\tau$  é o vetor tangente à curva no espaço paramétrico;

$\alpha u$  e  $\beta v$  são as componentes do vetor  $\tau$  e

$\delta$  é o fator de avanço ou deslocamento sobre a curva.

O fator de avanço  $\delta$ , ou seja, o passo de

deslocamento sobre a curva, pode ser obtido a partir do cálculo da variação do ângulo entre as tangentes de dois pontos consecutivos sobre a curva no espaço do objeto.

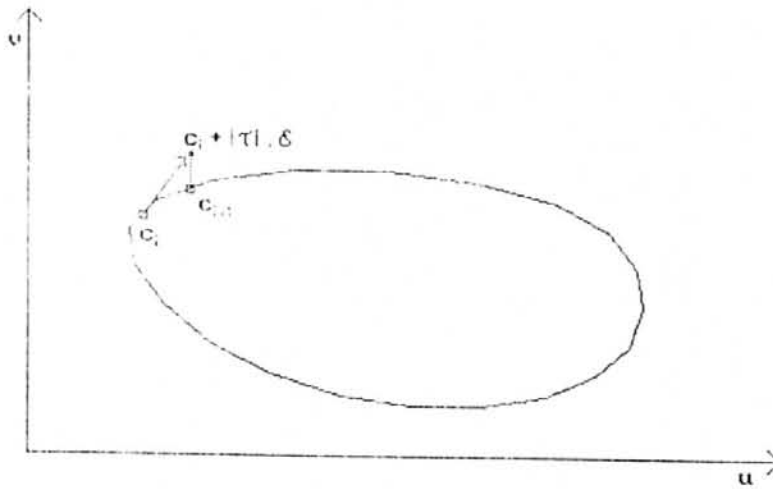


Figura 4.9 Deslocamento no espaço paramétrico

Estima-se inicialmente um valor para  $\delta$  e calcula-se  $c_{i+1}$  por (4.6.2 - 1) e refina-se o valor obtido através do método de Newton. Calcula-se, então, os vetores tangentes no espaço do objeto respectivamente para  $m_i$  e  $m_{i+1}$ . Se a diferença de ângulo entre elas for menor que o estabelecido inicialmente, então o ponto  $c_{i+1}$  é armazenado e o processo é repetido fazendo-se  $c_i = c_{i+1}$ . Caso contrário, o valor inicial de  $\delta$  é reduzido e é feita nova tentativa. Para cada novo ponto, o passo  $\delta$  é recalculado usando o valor do ângulo entre as tangentes do último ponto encontrado e do que se está tentando aproximar.

#### 4.4.7 Cálculo do passo de movimentação $\delta$

O passo de avanço  $\delta$  é calculado de maneira a chegar mais rapidamente a valores que resultem em pontos cuja diferença de ângulo entre as tangentes seja o mais próximo possível do valor que foi especificado, isto é, a fórmula deverá ser tal que o valor de  $\delta$  deve crescer se na última iteração o valor do ângulo entre as tangentes é menor que o determinado e, deve permanecer estável, se o ângulo estiver próximo do valor pré-estabelecido. A redução do valor de  $\delta$ , se necessário, é feita através de um teste como foi mostrado no item anterior.

Foi utilizada a equação

$$\delta = 1 + \cos(90 - \theta_0 + \theta)$$

onde

$\theta_0$  é o ângulo máximo aceitável entre os vetores tangentes no espaço objeto e

$\theta$  é o último ângulo encontrado entre os vetores tangentes.

O valor estimado inicialmente para  $\delta$  é 1. Desta forma, na primeira iteração garante-se que o ponto encontrado estará dentro dos limites (a diferença entre os vetores tangentes será zero). A partir de então o algoritmo se encarrega de incrementar ("acelerar") o valor de  $\delta$ . Por exemplo,

se  $\theta_0 = 90$  e  $\theta = 0$ , então

$$\delta = 1 + \cos(90 - 90 + 0),$$

$$\delta = 1 + \cos(0),$$

$$\delta = 2,$$

ou seja, o valor de  $\delta$  dobrará de uma iteração para outra já que o valor do ângulo entre as duas tangentes no espaço do

objeto está muito menor do que o pretendido. Porém

se  $\theta_0 = 90$  e  $\theta = 90$ , então

$$\delta = 1 + \cos(90 - 90 + 90),$$

$$\delta = 1 + \cos(90),$$

$$\delta = 1,$$

ou seja, o fator de multiplicação será igual ao da iteração anterior.

Como para certos, casos a taxa de "aceleração" do caminhamento é baixa, acrescentou-se uma variável que pode ser ajustada previamente resultando

$$\delta = 1 + ACELERACAO . \cos(90 - \theta_0 + \theta) .$$

Por este método, serão amostrados pontos em excesso, ou seja, durante a fase em que o algoritmo está "acelerando" serão avaliados pontos cuja diferença de ângulo entre as tangentes é menor que a especificada. Desta forma, existe uma etapa de "limpeza" dos pontos em excesso que será vista no item 4.6.9.

Como é mais fácil calcular o  $\sin(\theta)$  ao invés do valor do ângulo, deduziu-se a expressão equivalente

$$\delta = 1 + \text{ACELERACAO} \cdot (\text{sen}(\theta_0)\text{cos}(\theta) - \text{cos}(\theta_0)\text{sen}(\theta)) \cdot$$

#### 4.4.8 Cálculo dos vetores tangentes no espaço paramétrico

Uma vez que não se dispõe de elementos para calcular vetores tangentes à curva no espaço paramétrico, usou-se uma aproximação aos mesmos.

O vetor tangente à curva no ponto  $c_i$  foi aproximado pelo vetor determinado pelos pontos  $c_{i-1}$  e  $c_i$  normalizado (fig 4.10). O erro decorrente desta aproximação não é relevante uma vez que este vetor é usado apenas para se definir uma segunda aproximação (o próximo ponto) que será, então, melhorada.

Para que o vetor possa ser aproximado desde o primeiro passo da movimentação, é necessário que o método disponha de dois pontos iniciais determinados em uma fase de pré-processamento. A determinação dos pontos iniciais é feita testando-se os nós de um reticulado definido sobre a superfície procurando-se identificar dois pontos próximos que pertençam à curva de intersecção. A malha de nós é definida sobre curvas isoparamétricas determinadas sobre a superfície. Desta forma, fica fácil aplicar um algoritmo de subdivisão recursiva sobre a curva determinada pelos dois pontos encontrados, de maneira a tentar encontrar um ponto

próximo suficiente ao primeiro, cuja diferença de ângulo entre as tangentes seja menor ou igual ao especificado.

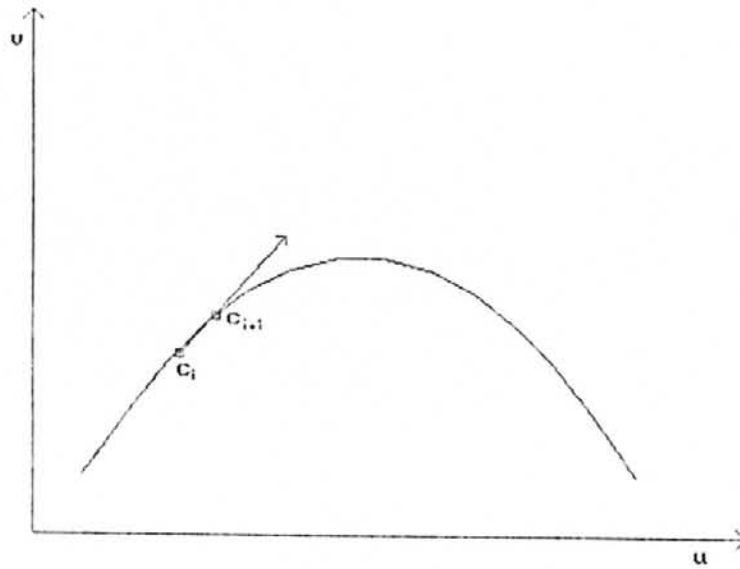


Figura 4.10 - Aproximação dos vetores tangentes no espaço paramétrico

Uma vez determinados os dois pontos de partida fica determinado também o sentido do deslocamento sobre a curva.

#### 4.4.9 Determinação da isoparamétrica de refinamento

Depois de se definir uma aproximação de um ponto pertencente à curva de intersecção é necessário melhorar esta aproximação de forma que o ponto obtido esteja contido na superfície-base. Uma melhor estimativa para este ponto é dada pelo método de Newton o qual toma por base a determinação de uma função envolvendo o ponto e a superfície que o contém. Como já foi visto no item 4.4 pode-se simplificar este problema transformando-o num problema de aproximação de um ponto a uma curva definida sobre a superfície em questão. Neste trabalho, determinaram-se curvas isoparamétricas.

Fixando-se uma das componentes paramétricas do ponto aproximado, determina-se uma curva isoparamétrica sobre a superfície (ver item 2.4). Utiliza-se este ponto como aproximação inicial da intersecção desta curva com o plano  $Z = 0$  (fig 4.11).



Conforme o comportamento da curva, porém, será necessário variar a componente paramétrica que será usada para definir a curva (fig 4.12). Para tanto a cada novo ponto a ser melhorado, sempre que o método de Newton não convergir sobre um determinada isoparamétrica, determina-se uma nova curva usando-se a outra componente paramétrica. Em uma curva de intersecção cuja forma for uma cônica esta troca de isoparamétricas pode ocorrer até quatro vezes.

#### 4.4.10 Condições de fim de execução do processo

Uma condição de fim de execução do processo é uma condição definida *a priori* usada para determinar quando encerrar o deslocamento sobre a curva.

Sabendo-se que o plano de corte é ilimitado e que a superfície é finita ("patch beta-spline") tem-se que, se a curva de intersecção for uma curva aberta, certamente a mesma interseccionará as bordas do "patch" (por bordas do "patch" entende-se as curvas isoparamétricas onde a componente fixada assume os valores zero ou um) e, se a curva for fechada, ocorrerão no máximo pontos de tangência com os limites do "patch".

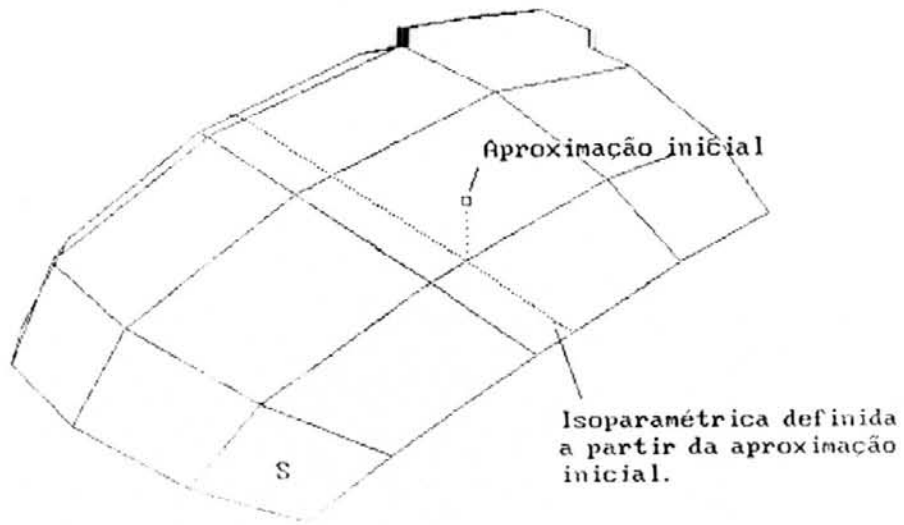


Figura 4.11 - Curva determinada pela fixação de uma das componentes de um ponto aproximado.

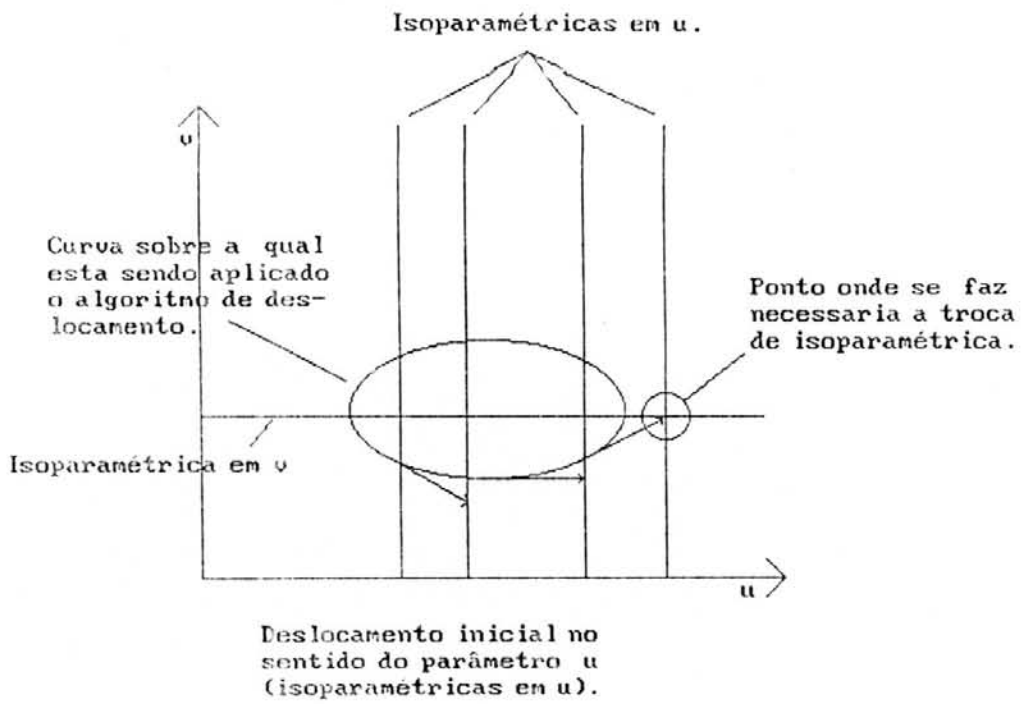


Figura 4.12 - Situação onde é necessária a troca de curva isoparamétrica durante o deslocamento.

Ignorando-se os casos em que a curva de

intersecção tangencia os limites do "patch", pode-se classificar uma curva de "aberta" quando existem pontos pertencentes à curva sobre os limites do "patch" e de "fechada" quando tal não ocorrer,

A classificação da curva de intersecção é feita na fase inicial quando da determinação dos pontos de partida. A busca é feita inicialmente nas bordas do "patch". Desta forma pode-se classificar a curva facilitando o teste do "ponto final". Além disso, garante-se que ao se encontrar um dos limites do "patch" durante o deslocamento, encontrou-se o fim do segmento de curva sobre aquele "patch" (fig. 4.10).

A determinação do "ponto final" para curvas abertas é feita quando se encontra um dos limites do "patch" e, para as curva fechadas, quando se encontra um ponto que se localiza entre os dois primeiros pontos encontrados durante o caminhamento.

#### 4.4.11 Generalização para uma superfície composta

Até agora foi visto como determinar a curva de ancoragem sobre um "patch" cúbico. Como as superfícies utilizadas neste trabalho são superfícies compostas "beta-spline" uniformes de grau três, é necessário

generalizar o método para superfícies compostas.

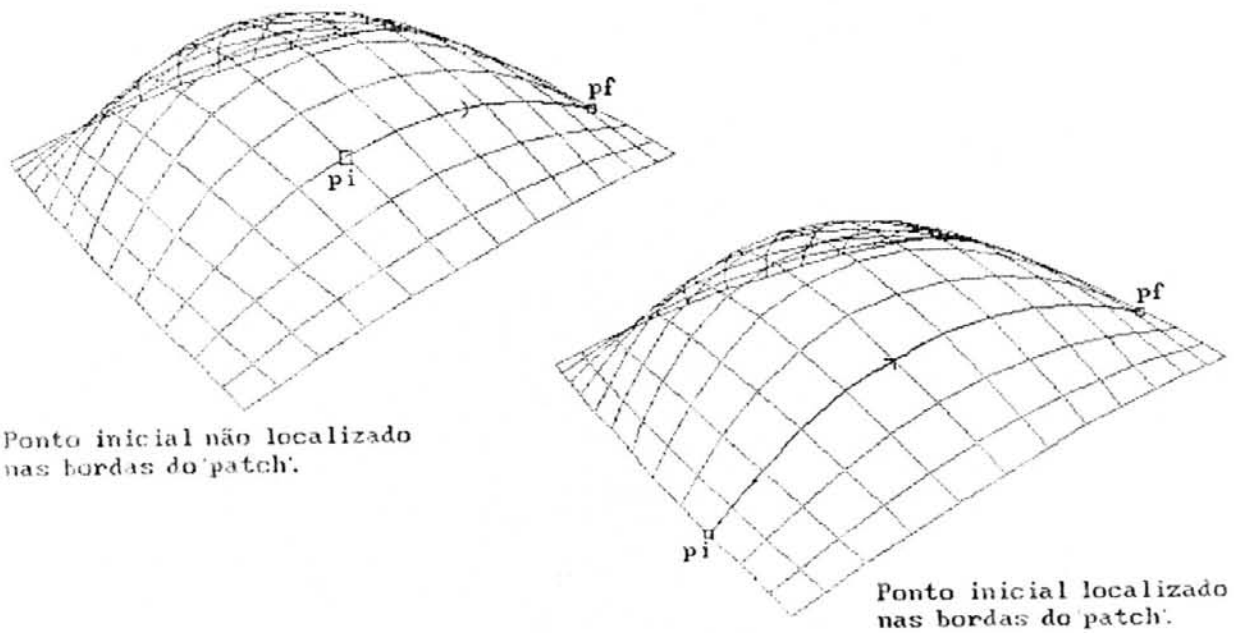


Figura 4.13 - Influência da localização do ponto inicial sobre o algoritmo de movimentação.

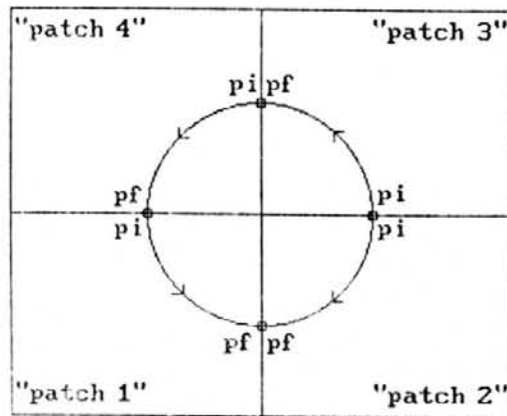
Basicamente o que se fará é encontrar os segmentos.

de curva sobre cada um dos "patches" e, em seguida, ordená-los de maneira a obter uma curva contínua.

Para ordenar os segmentos testa-se os pontos inicial e final de todos os segmentos encontrados procurando por pontos coincidentes. Uma vez que os segmentos são calculados sempre até as bordas dos "patches", inclusive, todos deverão possuir pontos comuns. Em geral devem ser encontrados pontos finais coincidentes com pontos iniciais. Quando os pontos coincidentes forem ambos pontos finais ou iniciais um dos segmentos tem de ser convenientemente orientado (fig. 4.14).

Pelo fato de calcularem-se os pontos sobre as bordas dos "patches", deve-se ter um cuidado especial quando ocorrerem curvas de ancoragem coincidentes com as mesmas. Neste caso, como os "patches" tem bordas comuns, cada segmento de curva será calculado duas vezes, tendo-se então, que eliminar os segmentos duplicados.

A intersecção entre uma superfície composta e um plano nem sempre será uma curva contínua (fig. 4.15). Nestes casos considerar-se-á que a intersecção gerou mais de uma curva e a seleção de qual das duas deverá ser utilizada como curva de ancoragem deverá ser feita interativamente.



Espaços paranétricos de uma superfície composta.  
 O segmento de curva do "patch 2" deve ser reorientado.

Figura 4.14 - Situação onde há necessidade de orientação dos segmentos de curva.

#### 4.4.11 Teste do envelope poliédrico

Uma maneira de evitar cálculos desnecessários é selecionar os "patches" que são candidatos a conterem um trecho da curva de intersecção entre o plano e o "patch" antes de ativar o algoritmo de busca desta curva sobre o "patch". Essa seleção se torna especialmente importante porque a fase de reticulado que procura buscar os pontos iniciais pertencentes à curva de intersecção terá de testar todos os nós do reticulado antes de poder chegar à conclusão de que não existe trecho da intersecção sobre o mesmo.

O teste do envelope poliédrico consiste em verificar se existe diferença de sinal entre o valores máximo e mínimo encontrados entre as cotas dos pontos amostrados sobre o "patch". Sempre que houver diferença de sinal o "patch" é candidato a conter um trecho da curva de intersecção.

#### 4.5 Restrições

Um dos principais problemas deste método é decorrente de problemas de convergência do método de Newton. Nem sempre, ao tentar-se melhorar um ponto aproximado sobre a curva, o método converge de maneira esperada. Pode-se dizer que os problemas ocorrem em dois casos:



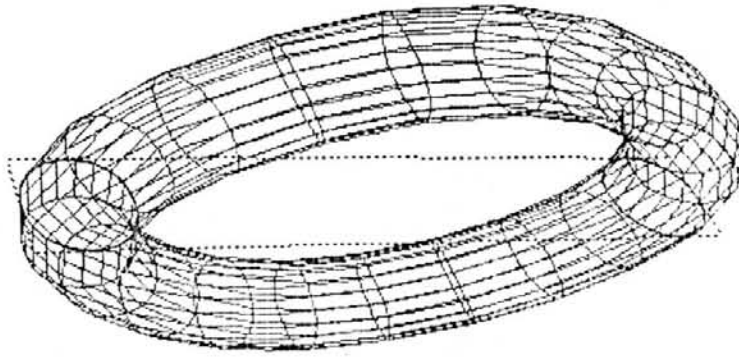


Figura 4.15 - Caso em que a intersecção entre o plano e a superfície gera mais de uma curva.

a) Existe a necessidade de se trocar a variável que define a isoparamétrica. Às vezes, o sistema não converge para nenhuma das duas isoparamétricas e o método entra em um caso de indefinição. A solução, em geral, nestes casos é alterar a variável "ACELERACAO " do cálculo do passo de deslocamento. Outro problema é quando se fazem necessárias duas trocas de isoparamétrica em dois pontos consecutivos. Nestes casos pode ocorrer uma inversão no sentido de deslocamento fazendo com que o algoritmo percorra novamente os pontos já visitados e

b) A curva possui dois trechos muito próximos. Quando a curva possui uma concavidade muito fechada ou quando a intersecção entre a superfície e o plano dá origem a duas curvas muito próximas, dependendo do tamanho do passo de caminhamento, pode ocorrer do algoritmo "pular" de uma curva para outra. Em geral, a solução é a mesma do caso anterior.

Outro caso particular que o sistema não trata corretamente é quando ocorrem pontos de tangência de uma curva de intersecção com as bordas de um "patch". Neste caso o algoritmo considera a curva aberta e os pontos de tangência como pontos iniciais ou finais. Desta forma só será detectado um trecho da curva sobre o "patch".

#### 4.6 Simplificação da amostragem da curva

Pela maneira como o método foi implementado são encontrados pontos em excesso, ou seja, a curva é amostrada com mais detalhe do que foi especificado.

Para eliminar os pontos excedentes, percorre-se a lista somando as diferenças de ângulo entre as tangentes no espaço de objeto de cada dois pontos consecutivos. Cada vez que o valor acumulado alcançar o limite pré-definido acrescenta-se o último ponto computado à lista final. O processo é, então, reiniciado a partir do próximo ponto ainda não visitado até que toda a lista tenha sido percorrida. O primeiro e o último pontos da lista são sempre mantidos.

Após a etapa de simplificação, tem-se a curva de ancoragem amostrada com o grau de detalhe que se especificou (fig. 4.15).

## 5. CÁLCULO DOS VETORES DE CONTROLE E GERAÇÃO DA SUPERFÍCIE DE JUNÇÃO

Este capítulo aborda a reparametrização das curvas de ancoragem, o cálculo dos vetores de controle, bem como a geração da superfície de junção a partir dos mesmos. São analisadas, também, as possibilidades de controle do formato da junção.

### 5.1 Reparametrização das curvas de ancoragem

Como foi visto no capítulo três (item 3.2), uma superfície Hermite é gerada por uma coleção de curvas Hermite ancoradas nas duas curvas que se deseja unir. No caso deste trabalho, as superfícies de junção serão definidas por uma coleção de curvas Hermite ancoradas nas duas curvas de ancoragem definidas sobre cada uma das superfícies-base.

Dadas duas curvas  $C_1$  e  $C_2$  a partir das quais se deseja gerar uma superfície Hermite (fig. 3.2) é necessário que, para cada ponto amostrado sobre a curva  $C_1$ , exista um ponto correspondente amostrado sobre a curva  $C_2$ .

Quando C1 e C2 são curvas paramétricas simples (não são curvas compostas), a determinação de pontos correspondentes sobre ambas as curvas é bastante simples. No caso em questão, não se dispõe da equação paramétrica das curvas de ancoragem, mas apenas de um conjunto de pontos amostrados sobre cada uma das curvas. Nada garante também que ambas as curvas foram amostradas com igual número de pontos, ou seja, para cada ponto disponível sobre a curva C1 não existe, obrigatoriamente, um ponto correspondente sobre a curva C2.

Para contornar esse problema optou-se pela reparametrização das curvas em função de seus comprimentos. Uma aproximação do comprimento das curvas pode ser obtida pelo somatório dos segmentos de reta que unem os pontos amostrados sobre a mesma. A obtenção de pontos intermediários entre os já calculados pode ser feita por interpolação linear de dois pontos consecutivos.

Esse tipo de solução, entretanto, trouxe certas preocupações. O comprimento calculado será apenas uma aproximação do comprimento original da curva e os pontos calculados por interpolação linear podem não estar contidos sobre a superfície-base, ou seja, a aproximação pode ser por demais grosseira.

Essas preocupações foram contornadas a partir do seguinte raciocínio:

a) O erro no cálculo do comprimento das curvas é desprezível, uma vez que este valor será usado apenas como parâmetro de reparametrização. É portanto um valor relativo  $\varrho$

b) As curvas de ancoragem são amostradas com o grau de detalhe desejado, especificado de acordo com a necessidade. Logo, qualquer ponto obtido por interpolação linear entre dois pontos assim calculados terá um erro menor que o tolerado. Para que isso seja verdade, entretanto, deve-se garantir que a superfície de junção será gerada utilizando como pontos de ancoragem todos os pontos originalmente calculados sobre as curvas de ancoragem.

Procede-se, assim, da seguinte maneira. Calcula-se o valor do parâmetro em relação ao comprimento aproximado da curva para cada um dos pontos originalmente amostrados sobre as mesmas. Gera-se então uma lista de valores paramétricos resultante da união dos valores paramétricos encontrados sobre ambas as curvas. Calcula-se, a seguir, sobre cada uma das curvas, os pontos correspondentes aos valores paramétricos amostrados. Desta forma garante-se que todos os

pontos amostrados sobre a primeira das curvas tem correspondente na segunda e que todos os pontos originalmente amostrados quando da determinação das curvas de ancoragem estão sendo utilizados.

## 5.2 Determinação dos vetores de controle

Para gerar uma curva Hermite necessita-se de dois pontos e dois vetores de controle associados aos mesmos. Esses vetores são denominados vetores de controle, porque determinarão a direção e a velocidade de saída da curva, como visto no item 3.2. Se um determinado vetor for considerado "de partida" ou "de chegada", isso é consequência da sua posição no espaço e de como o algoritmo de geração da curva os recebe.

Para gerar uma superfície de junção usando a formulação de Hermite, será necessário associar, para cada ponto amostrado sobre as curvas de ancoragem, um vetor de controle. Esses vetores deverão ser calculados de tal maneira que garantam a continuidade desejada entre as superfícies-base e a superfícies de junção. No caso deste trabalho será suficiente garantir continuidade geométrica.

Diversas soluções são apresentadas na literatura

para o cálculo dos vetores de controle. Filip /FIL 89/, propõe que o vetor de controle  $V$ , associado a um ponto  $c_1(t)$  amostrado sobre a curva  $C_1$  contida na superfície base  $S_1$ , seja um vetor contido no plano tangente à superfície  $S_1$  no ponto  $c_1(t)$ . Para calcular tal vetor, Filip se utiliza do vetor  $n$  normal à superfície  $S_1$  no ponto  $c_1(t)$  (ou  $s_1(u(t),v(t))$ ) e de um vetor  $k=C_2(t)-C_1(t)$  (fig 5.1). O vetor  $V$  assim determinado é ainda multiplicado por um fator  $F$  calculado a partir do vetor  $k$  e que procura aproximar a curvatura da superfície de junção de um arco de círculo.

O método proposto por Filip, porém, apresenta alguns inconvenientes. Em primeiro lugar, existem casos onde o método não funciona. Por exemplo, quando o vetor  $k$  é ortogonal ao vetor  $n$ . Nesta situação, Filip sugere outras formas de cálculo. Em segundo lugar, os vetores de controle assim calculados tornam-se dependentes não apenas do par de superfícies-base que está sendo unido, mas também da posição relativa das mesmas. Desta forma, qualquer alteração na posição das superfícies base implicará em recálculo dos vetores de controle.

Considerou-se, então, que a determinação dos vetores pelo método sugerido por Filip não se prestava a construção de um "software" interativo onde o usuário pode ter a necessidade de fazer experiências com a posição das



superfícies-base. Da mesma forma, seria interessante poder supor que um dado conjunto de vetores de controle, calculados para uma determinado curva de ancoragem associada a uma superfície base específica, pode ser usado para a

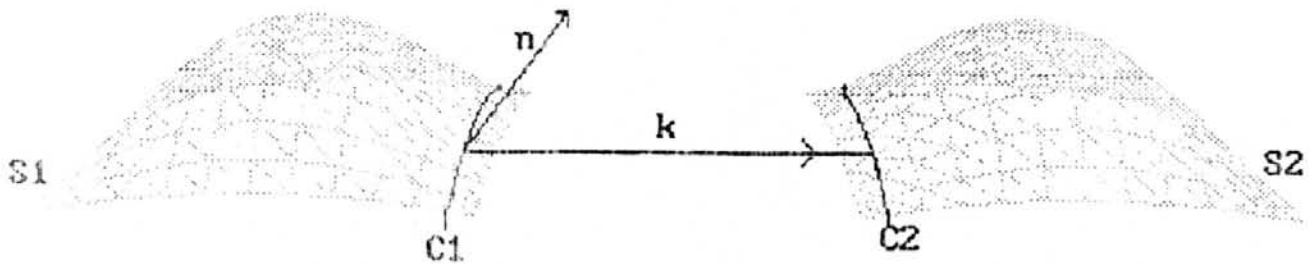


Figura 5.1 - Vetores usados no cálculo dos vetores de controle pelo método de Filip.

junção de tal superfície com qualquer outra superfície-base distinta.

Considerando-se estes fatos buscou-se uma maneira alternativa para a determinação dos vetores de controle aproveitando-se a maneira pela qual as curvas de ancoragem são definidas. Os vetores de controle são, então, calculados como segue.

Dado que uma curva de ancoragem  $M$  é determinada pela intersecção de um plano  $\pi$  e uma superfície  $S$ , considera-se que os vetores  $v(t)$  associados a cada ponto  $m(t)$  da curva de ancoragem devem ser simultaneamente ortogonais à curva  $M$  e tangentes à superfície  $S$ . Desta maneira garante-se continuidade geométrica entre a superfície-base e a junção gerada com esses vetores. Garante-se ainda que os mesmos terão uma direção que corresponde à direção que se pretende ao determinar a curva de ancoragem. Tais vetores podem ser obtidos pelo produto vetorial entre o vetor tangente à curva  $M$  no ponto em questão ( $m(t)$ ) e o vetor normal à superfície  $S$  no mesmo ponto (fig. 5.2).

Então:

$$v = m'_t \times n$$

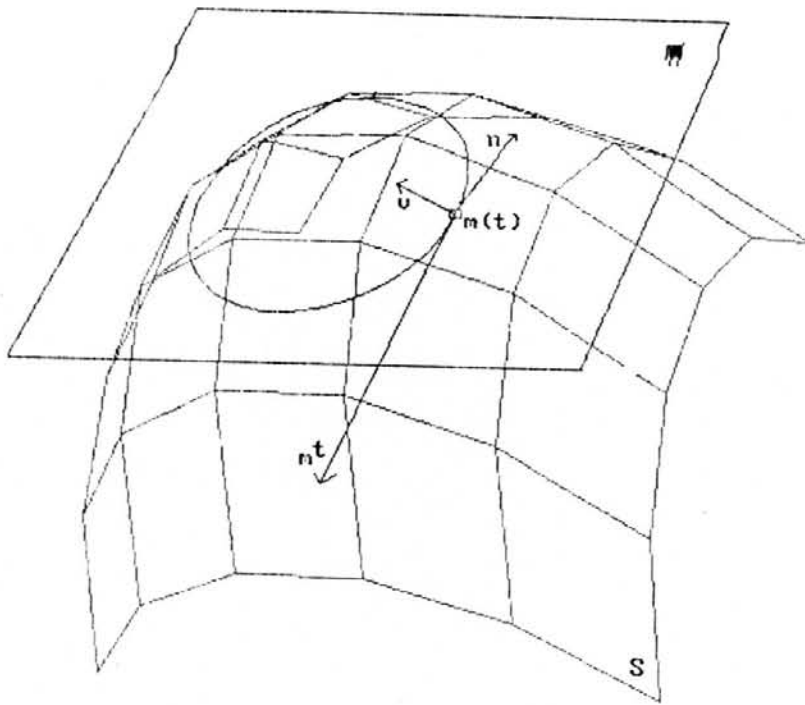


Figura 5.2 - Obtenção dos vetores de controle.

onde:

$\mathbf{v}$  é o vetor de controle;

$\mathbf{n}$  é o vetor normal ao plano  $\pi$ .  $\mathbf{n} = (0,0,1)$  se  $\pi$  é o plano  $Z = 0$  e

$\mathbf{m}$  é o vetor tangente à curva  $C$ .  $\mathbf{m} = (\alpha, \beta, \delta)$ , onde  $\alpha, \beta, \delta$  são calculados como visto no item 4.6.1.

Já que o cálculo dos vetores de controle é dependente do plano de corte que define a curva de ancoragem, os mesmos tornam-se dependentes desta e não do posicionamento das superfícies a serem unidas no espaço. Este fato traz grandes vantagens na medida em que permite que se associe a uma determinada superfície-base, uma ou mais curvas de ancoragem e seus respectivos vetores de controle. Desta forma, estas curvas podem ser usadas para a determinação de junções com quaisquer outras superfícies-base sem a necessidade de novos cálculos. Em resumo, as curvas de ancoragem e os vetores de controle tornam-se dependentes de uma superfície-base e não de uma instância da mesma ou de um par de superfícies a serem unidas.

### 5.3 Geração das superfícies de junção

Dispondo de duas superfícies-base com suas respectivas curvas de ancoragem e vetores de controle, a

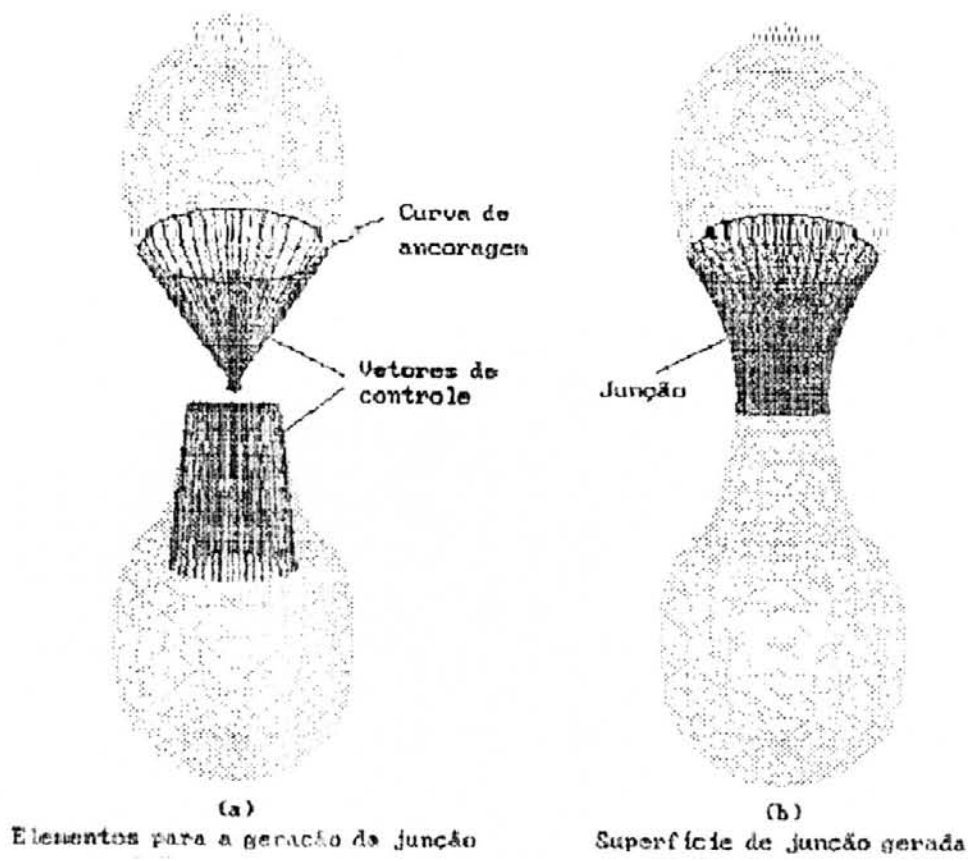


Figura 5.3 - Geração da superfície de junção.

geração de uma junção entre as mesmas torna-se mera aplicação do método de aproximação de superfícies Hermite apresentado no item 3.2 (fig. 5.3).

Restam alguns fatores, porém, que podem ser ajustados. O sentido dos vetores de controle deve ser convenientemente orientado conforme desempenhem o papel de vetores de chegada ou de partida. Pode-se, também, optar por diferentes combinações no sentido dos vetores de forma a obter diferentes superfícies de junção.

O tamanho dos vetores também é bastante significativo. Em geral, os vetores de controle são armazenados normalizados. Aplicando-se um fator de escala sobre os mesmos resulta um mecanismo de controle bastante eficiente sobre a forma de junção (fig. 5.4).

Se o fator de multiplicação dos vetores for exagerado, corre-se o risco de gerar uma superfície que se auto-intercepta o que pode ser desejável ou não dependendo da aplicação (fig. 5.5). O fator máximo de multiplicação que se pode aplicar sobre um vetor de controle, de maneira que a superfície não se auto-intercepte depende da distância das superfícies-base e dos ângulos relativos dos mesmos.

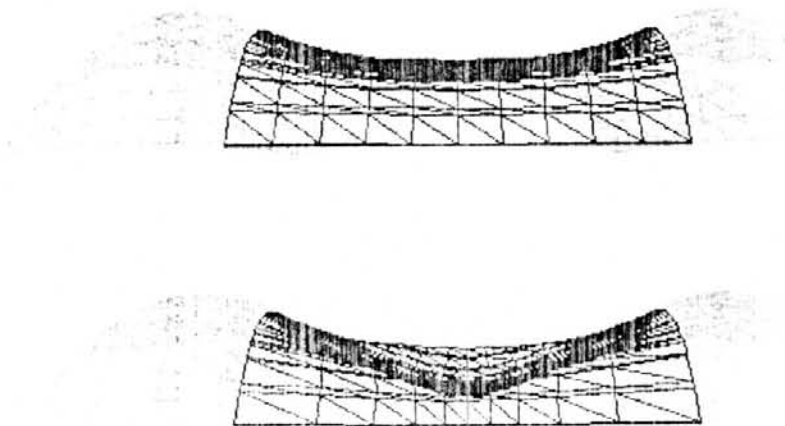


Figura 5.4 - Controle de forma da superfície de junção.

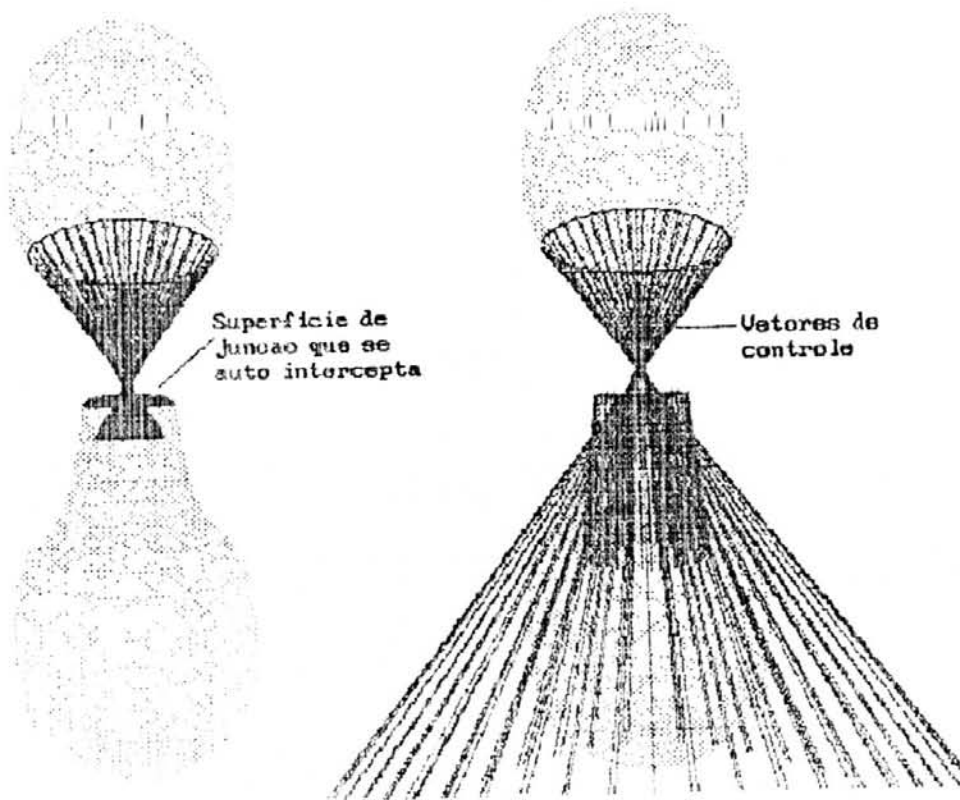


Figura 5.5 - Superfície de junção que se auto-intercepta e seus vetores de controle.



## 6. ALGORITMO DE GERAÇÃO DE OBJETOS USANDO SUPERFÍCIES DE JUNÇÃO

O objetivo deste capítulo é demonstrar, através de um exemplo, todos os passos necessários para a modelagem de um objeto usando uma junção.

Será utilizado como exemplo um objeto simples, formado pela união de uma esfera com um cilindro (fig. 6.1) usando-se para tanto uma superfície de junção.

Como os objetos a serem unidos (ou superfícies-base), foram selecionados, o primeiro passo é determinar sobre os mesmos as curvas de ancoragem. Conforme proposto no capítulo quatro, cada uma das curvas de ancoragem é determinada pela intersecção de um plano com uma das superfícies base. Deve-se, então, definir os planos de corte. Os planos de corte utilizados neste exemplo podem ser vistos na figura 6.2.

Dispondo-se das curvas de ancoragem passa-se para o cálculo dos vetores de controle. Para cada ponto amostrado sobre as curvas de ancoragem, deve existir um vetor de controle correspondente como foi visto no capítulo cinco. Destaca-se aqui, mais uma vez, que os vetores de controle, pela forma que são calculados, são independentes da junção

que será gerada.

O passo seguinte é instanciar as duas superfícies-base e ajustar convenientemente os vetores de controle (fig. 6.3). Conforme se viu no capítulo cinco, o tamanho e o sentido dos vetores influenciam no formato final da junção. As superfícies base são instanciadas através de transformações geométricas simples (translação, escalamento e rotação), enquanto que os vetores de controle são ajustados através de um fator de multiplicação.

Finalmente tem-se todas as condições para gerar a superfície de junção aplicando-se a formulação Hermite vista no capítulo três. O resultado pode ser visto na figura 6.4.

O modo como este método foi idealizado se presta sobretudo para a simulação de efeitos de corpos elásticos. Alterando-se as posições das superfícies-base, a superfície de junção ajusta-se ao novo posicionamento. Pode-se aproveitar este fato para simulações de objetos elásticos ou simplesmente para usar como um efeito especial em animação (fig. 6.5).

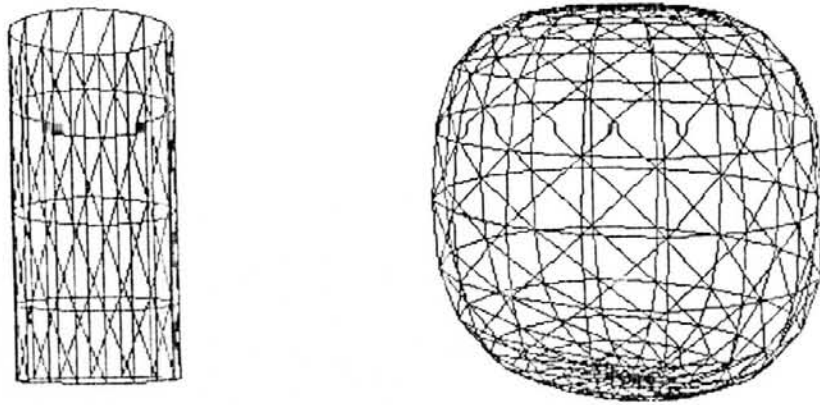


Figura 6.1 - Escolha das superfícies-base.

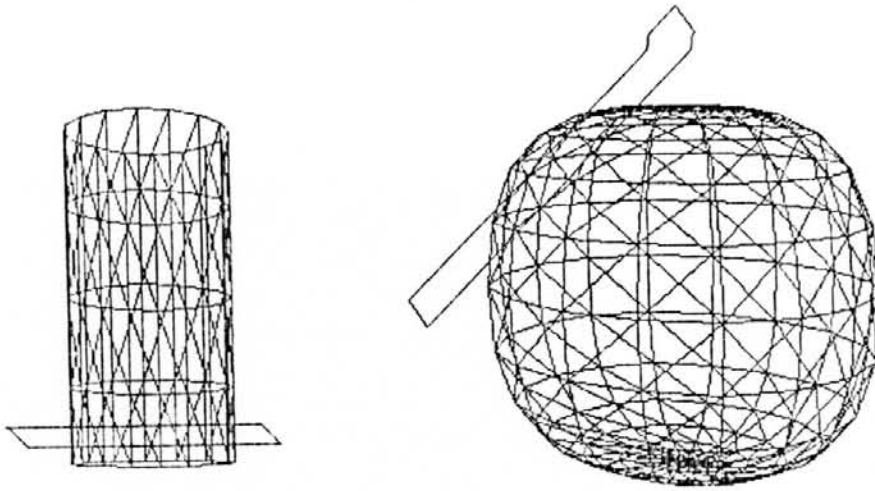


Figura 6.2 - Determinação dos planos de corte.

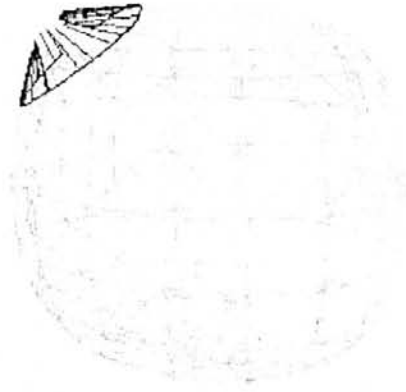


Figura 6.3 - Instanciamento das superfícies-base e ajuste dos vetores de controle.

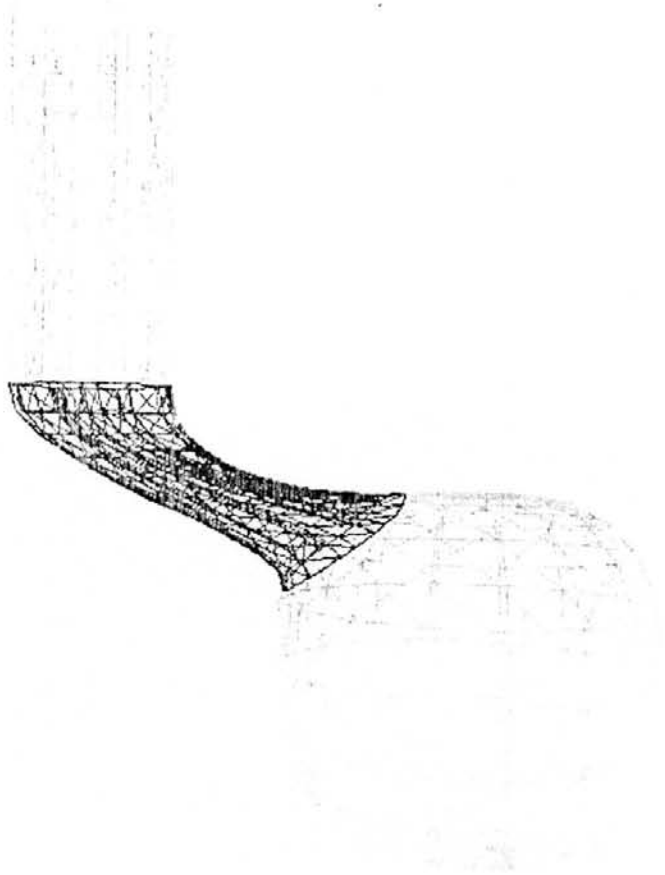


Figura 6.4 - Geração da superfície de junção.

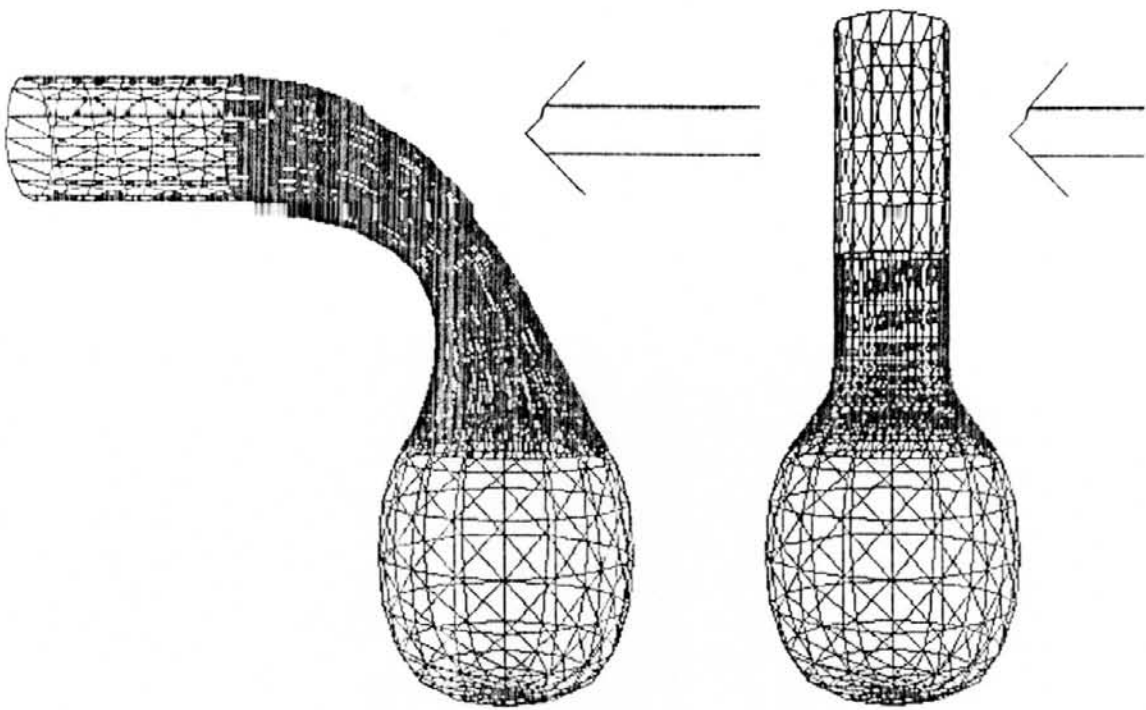


Figura 6.5 - Conformação da superfície de junção de acordo com o posicionamento das superfícies base.

## 7. O PROTÓTIPO

Neste capítulo descrevê-se, de uma maneira geral, o protótipo implementado com o objetivo de validar as idéias apresentadas neste trabalho. O sistema implementado possibilita a modelagem de objetos que incluam superfícies de junção, ou seja, é um conjunto de ferramentas de "software" que permite que se definam objetos compostos por superfícies-base e junções entre as mesmas.

Inicialmente será apresentada a arquitetura do sistema de modelagem. Em seguida, cada um dos módulos será detalhado.

### 7.1 Arquitetura do sistema

A arquitetura do sistema pode ser vista na figura 7.1 . O sistema é completamente modular e a comunicação entre os diversos módulos é feita através de arquivos. Esta arquitetura permite que os diferentes módulos possam ser desenvolvidos e testados de maneira independente e o acréscimo de novos módulos fica bastante facilitado. Além disso, um sistema composto por módulos independentes e autônomos se adapta bem a ambientes computacionais pequenos onde os problemas com gasto de memória principal são



constantes.

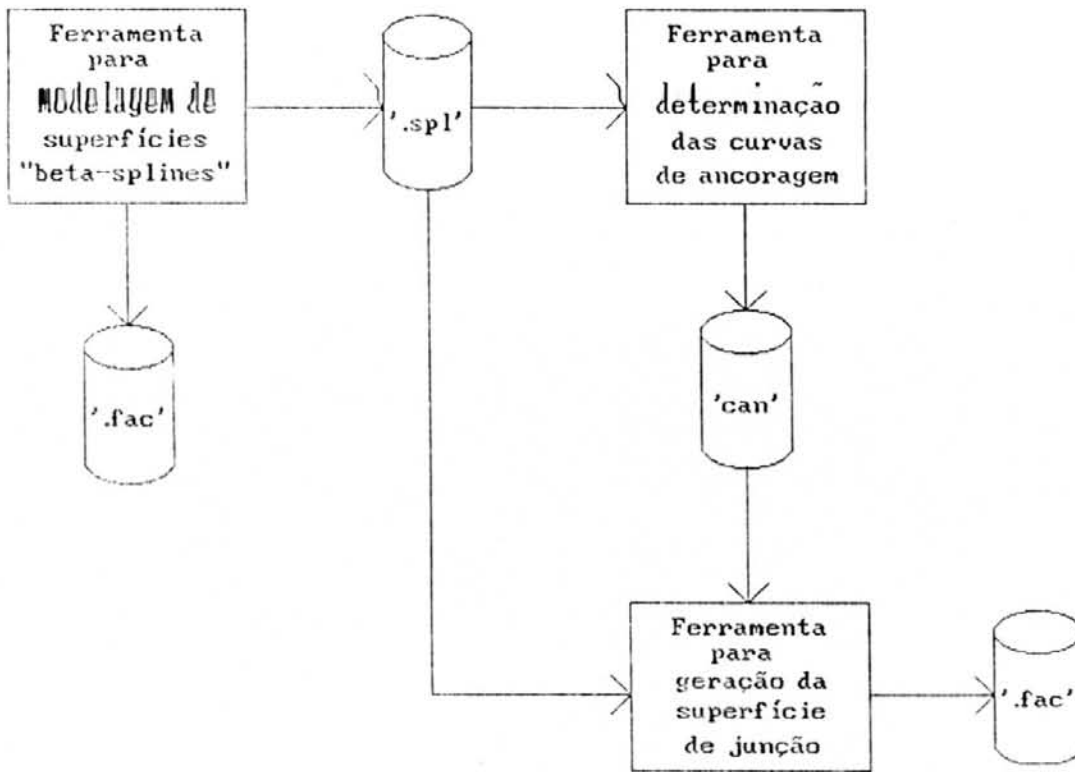


Figura 7.1 - Arquitetura do sistema.

Os arquivos de comunicação entre os módulos são todos "tipo texto" (ASCII) o que permite sua geração, também, por processadores de texto. Essa característica torna-se relevante em um ambiente experimental pois permite que se confirmem os resultados intermediários além de possibilitar alterações manuais. Outros aspectos considerados foram a portabilidade dos arquivos entre ambientes computacionais diversos e a facilidade de compatibilizar o sistema com outros sistemas de modelagem ou visualização existentes. Esta última característica foi aproveitada na integração com o sistema ANIMA /COP 91/. O sistema ANIMA é um sistema acadêmico e experimental em desenvolvimento no Laboratório de Imagens da PUCRS e que permite a modelagem, visualização e animação de objetos 3D. A integração permite a utilização do módulo de síntese de imagens realísticas do ANIMA para a visualização dos objetos modelados com superfícies de junção.

Cada um dos módulos é responsável por uma das etapas de geração de uma superfície de junção descritas nos capítulos quatro e cinco. A forma de organização dos arquivos permite que resultados intermediários sejam reaproveitados, reduzindo o tempo de processamento.

O primeiro módulo permite a definição de superfícies "beta-spline". É a ferramenta onde se definem as

superfícies-base. É capaz de gerar dois tipos de arquivos de saída.

A primeira das possibilidades é a geração de um arquivo que contém a descrição da superfície "beta-spline" modelada em termos de "poliedro de controle" e "parâmetros de controle de forma". É a descrição que será usada quando do uso destas superfícies como base para a geração de superfícies de junção. Os arquivos que contém a descrição de superfícies "beta-spline" possuem a terminação ".spl" no nome.

O segundo tipo de arquivo que pode ser gerado é o que contém uma aproximação à superfície por polígonos. Esse arquivo destina-se à integração com o sistema ANIMA que prevê apenas objetos formados por polígonos. Arquivos deste tipo contém a lista de faces (polígonos) que aproximam a superfície e possuem a terminação ".fac" no nome.

O módulo seguinte é o módulo de definição de curvas de ancoragem. Ele permite que se defina um plano de corte sobre a superfície, determinando uma curva de ancoragem sobre a mesma pela intersecção entre o plano e a superfície. Uma vez que a definição dos vetores de controle é dependente do plano de corte, os mesmos são calculados juntamente com a curva de ancoragem. O arquivo gerado por

este contém a descrição da curva de ancoragem e de seus respectivos vetores de controle. Um identificador da superfície-base correspondente também é armazenado. Desta maneira é possível criar uma biblioteca de curvas de ancoragem definidas sobre uma determinada superfície que podem ser usadas, posteriormente, conforme a necessidade. Arquivos de descrição de curvas de ancoragem possuem a terminação ".can" no nome.

O módulo de definição de superfícies de junção é o responsável pela geração do objeto que se deseja modelar. Recebe como entrada um arquivo que contém a descrição do objeto desejado. Nesta descrição, determina-se as instâncias das superfícies-base no espaço, indica-se quais superfícies devem ser unidas e quais curvas de ancoragem devem ser usadas. Esta descrição foi chamada de cena por se assemelhar a descrição de uma cena em um sistema de animação. Neste último também são descritas as instâncias dos "atores" envolvidos.

O arquivo de descrição de cenas é editado em um processador de textos comum. O módulo reduz-se, então, a um interpretador deste arquivo. Depois de interpretado o arquivo, o sistema calcula as junções e as exibe juntamente com as superfícies-base e as curvas de ancoragem. O módulo pode gerar um arquivo de terminação ".rnd" que contém as

informações necessárias à exibição do objeto gerado no módulo de visualização com realismo do sistema ANIMA. O sistema tem condições, ainda, de gerar um arquivo com a descrição das superfícies de junção.

O protótipo poderia conter, também, um módulo de conversão da descrição Hermite das junções para uma descrição "beta-spline". Desta forma, seria possível realimentar o sistema com as superfícies geradas por ele mesmo e gerar, então, junções entre junções. Este módulo não foi implementado, porém, por questões de tempo e fica sugerido como extensão do sistema.

## 7.2 O módulo de modelagem de superfícies base

O módulo de definição das superfícies-base é um sistema de modelagem de superfícies "beta-spline" onde se modela um poliedro de controle que, por sua vez, define uma superfície "beta-spline". O modelador usado foi desenvolvido anteriormente por este autor e seus detalhes estão descritos no trabalho "Um modelador de Sólidos Baseado em Superfícies Beta-Spline" /COP 90/. Para tanto se utilizam técnicas tais como "sweep" rotacional, "sweep" translacional e modelagem por secções laterais fartamente abordadas na literatura. A

ferramenta permite ainda que os vértices do poliedro definido por uma destas técnicas possam ser editados individualmente.

Depois de definida a superfície, o módulo gera um arquivo de terminação ".spl" com a descrição da mesma. O formato deste arquivo pode ser visto na figura 7.2 . O arquivo contém os valores  $\beta_1$  e  $\beta_2$  dos parâmetros de controle de forma, as dimensões da matriz que armazena o poliedro de controle e os vértices do mesmo. Como todos os arquivos do sistema, o arquivo de descrição de superfícies "beta-spline" é modo texto (ASCII) e cada informação é antecedida por um identificador significativo.

### 7.3 O módulo de definição de curvas de ancoragem

O módulo de definição de curvas de ancoragem é um módulo interativo onde o usuário pode definir uma curva de ancoragem sobre uma superfície de maneira implícita, através da intersecção da mesma com um plano de corte.

Inicialmente deve ser informado ao sistema o nome do arquivo de descrição da superfície "beta-spline" sobre a qual será definida a curva de ancoragem. O arquivo é carregado e é calculado o envelope poliédrico que envolve a

superfície. Isso é necessário porque, devido ao ambiente computacionalmente limitado, não é possível interagir com a superfície diretamente. O usuário trabalha, então, com o

```
$BETA_SPLINE plano.spl
$ORIGEM TRANSLACIONAL
$LINHAS_MAT 4
$COLUNAS_MAT 4
$BETA1 1.000000
$BETA2 0.000000
$MATRIZ
2.0 0.0 -12.0      2.0 0.0 -12.0      0.0 1.0 -12.0      15.0 0.0 -12.0
2.0 0.0 -12.0      2.0 0.0 -12.0      0.0 1.0 -12.0      15.0 0.0 -12.0
2.0 0.0  0.0       2.0 0.0  0.0       0.0 1.0  0.0       15.0 0.0  0.0
2.0 0.0  0.0       2.0 0.0  0.0       0.0 1.0  0.0       15.0 0.0  0.0
$FIM
```

Figura 7.2 - Formato do arquivo de descrição de superfície  
"beta-spline"

envelope, podendo solicitar a visualização da superfície sempre que desejar.

Como descrito no capítulo quatro, o sistema é capaz de calcular apenas a intersecção da superfície com o plano  $Z=0$ . Diferentes cortes são obtidos então pelo instanciamento da superfície em diferentes posições em relação a este plano. Para que isso seja transparente ao usuário o sistema instancia, inicialmente, a superfície centralizada na origem do espaço Euclidiano (fig. 7.3). O sistema permite, então, que o usuário aplique operações de rotação (ao redor de qualquer eixo) e de translação (ao longo do eixo Z) sobre a superfície correspondendo, do ponto de vista do usuário, a transformações aplicadas sobre o plano de corte.

A interação é feita usando-se o envelope poliédrico (devido à necessidade de tempo de resposta baixo). Para facilitar o trabalho, a intersecção do envelope que envolve a superfície com o plano  $Z=0$  é mostrada continuamente (fig. 7.4).

O sistema oferece ainda a possibilidade do usuário girar o sistema de eixos, de maneira que se possa obter diferentes vistas da região de trabalho.



A qualquer momento, o sistema está pronto para

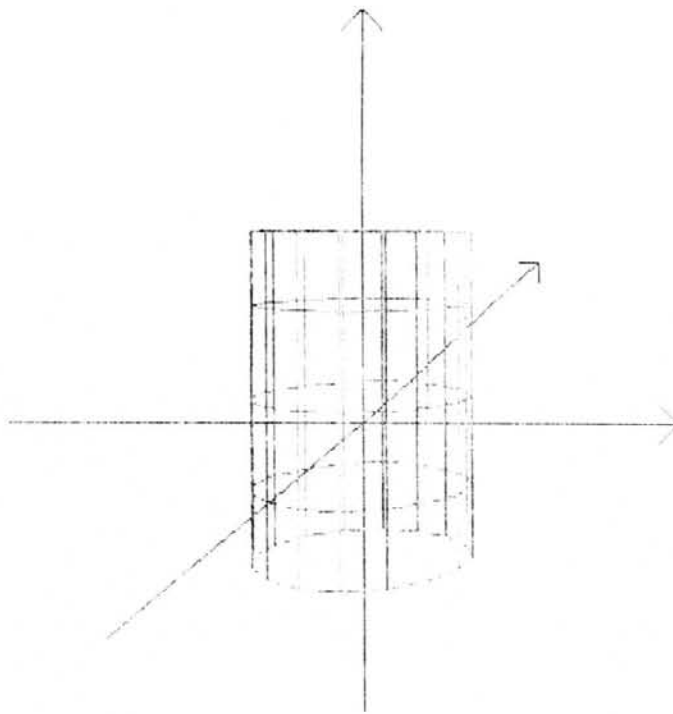


Figura 7.3 - Objeto instanciado na origem do sistema de coordenadas de maneira a facilitar a determinação do plano de corte.

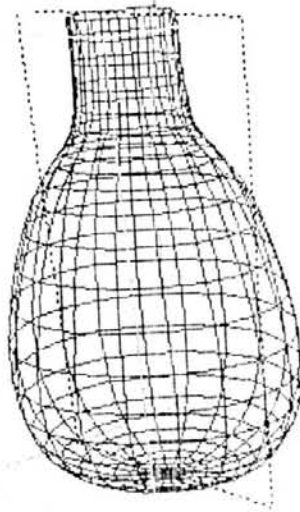


Figura 7.4 - Envelope poliédrico.

calcular a curva de ancoragem. Pode-se solicitar apenas a curva ou a curva e os vetores de controle. Em ambos os casos, o sistema exibe o algoritmo de deslocamento "procurando" pela intersecção e em seguida exibe a curva encontrada sobre a superfície-base (fig. 7.5). No segundo caso, além da exibição conjunta dos vetores de controle (fig. 7.6), o sistema se encarrega de gravar a descrição dos elementos calculados em um arquivo de terminação ".can".

O formato de um arquivo ".can" pode ser visto na figura 7.7. O arquivo contém basicamente a descrição da curva de ancoragem e dos vetores de controle. A descrição da curva de ancoragem contém para cada ponto amostrado da curva as seguintes informações: as componentes  $X, Y, Z$  dos pontos no espaço Euclidiano; os valores paramétricos  $(u, v)$  que correspondem aos pontos sobre os "patches" da superfície-base e um par de valores inteiros que permite identificar o "patch" da superfície a que correspondem os valores paramétricos. As componentes normalizadas dos vetores de controle associados a cada um destes pontos, segue estas informações.

O grau de detalhe com que o usuário deseja obter a curva pode ser definido a qualquer momento.

Quando a intersecção entre o plano e a

superfície-base gera mais de uma curva, o sistema pede

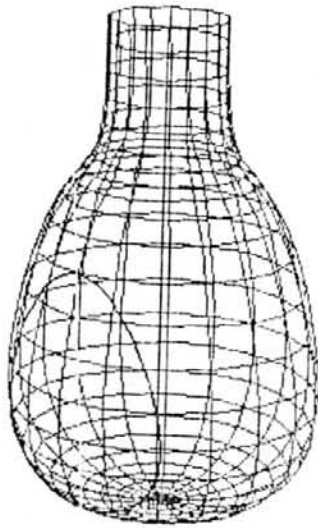


Figura 7.5 - Exibição da curva encontrada sobre a superfície-base.

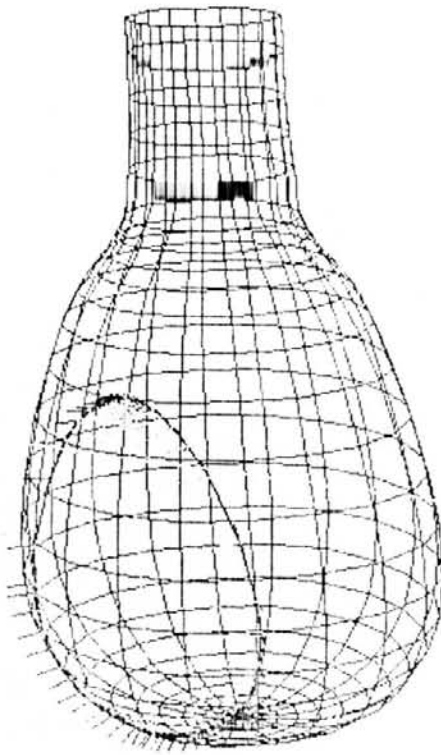


Figura 7.6 - Curva de ancoragem com respectivos vetores de controle.

```
$CURVA DE ANCORAGEM ss.can
$SUPERFICIE patch.spl
$ROTACAO -5.000000 10.000000 0.000000
$NUM PONTOS 10
0.745419 1.000000 3 3 0.211955 -0.140260 -0.967162
0.699084 0.846760 3 3 0.221549 -0.213056 -0.951598
0.656385 0.685134 3 3 0.178101 -0.308786 -0.934308
0.633244 0.585139 3 3 0.108721 -0.359803 -0.926672
0.614282 0.493446 3 3 0.020847 -0.378973 -0.925173
0.598854 0.409517 3 3 -0.061749 -0.363893 -0.929392
0.582282 0.305074 3 3 -0.140681 -0.312078 -0.939583
0.564140 0.159275 3 3 -0.193876 -0.226798 -0.954458
0.552484 0.020990 3 3 -0.207765 -0.164373 -0.964269
0.551199 0.000000 3 3 -0.208466 -0.156817 -0.965376
$ENDCURVA
```

Figura 7.7 Formato de um arquivo de descrição de curvas de ancoragem.

primeiro que o usuário indique qual dos trechos deve ser considerado e em seguida elimina os demais. Isso pode ocorrer se a superfície-base for, por exemplo, um toro (fig. 7.8).

Se a curva de ancoragem intersecciona muitos "patches", o sistema pode apresentar problemas de falta de memória durante o processamento. Problemas de convergência do método de Newton podem ocorrer conforme descrito no item 4.6.8, e são indicados por mensagens.

#### **7.4 O módulo de geração de superfícies de junção**

Devido às limitações do equipamento, este não é um módulo interativo. A definição de um objeto usando superfícies de junção é feita através da descrição de uma pequena cena. Nesta descrição, posiciona-se as superfícies-base, associa-se a cada uma as respectivas curvas de ancoragem e indica-se quais superfícies-base devem ser unidas entre si. O módulo, então, interpreta esta descrição e gera o objeto desejado. São exibidas as arestas do objeto ("wire-frame") e é gerado um arquivo que permite a visualização do mesmo através do sistema ANIMA. A descrição para visualização é composta por vários arquivos: um arquivo

principal que contém a descrição da cena completa e uma

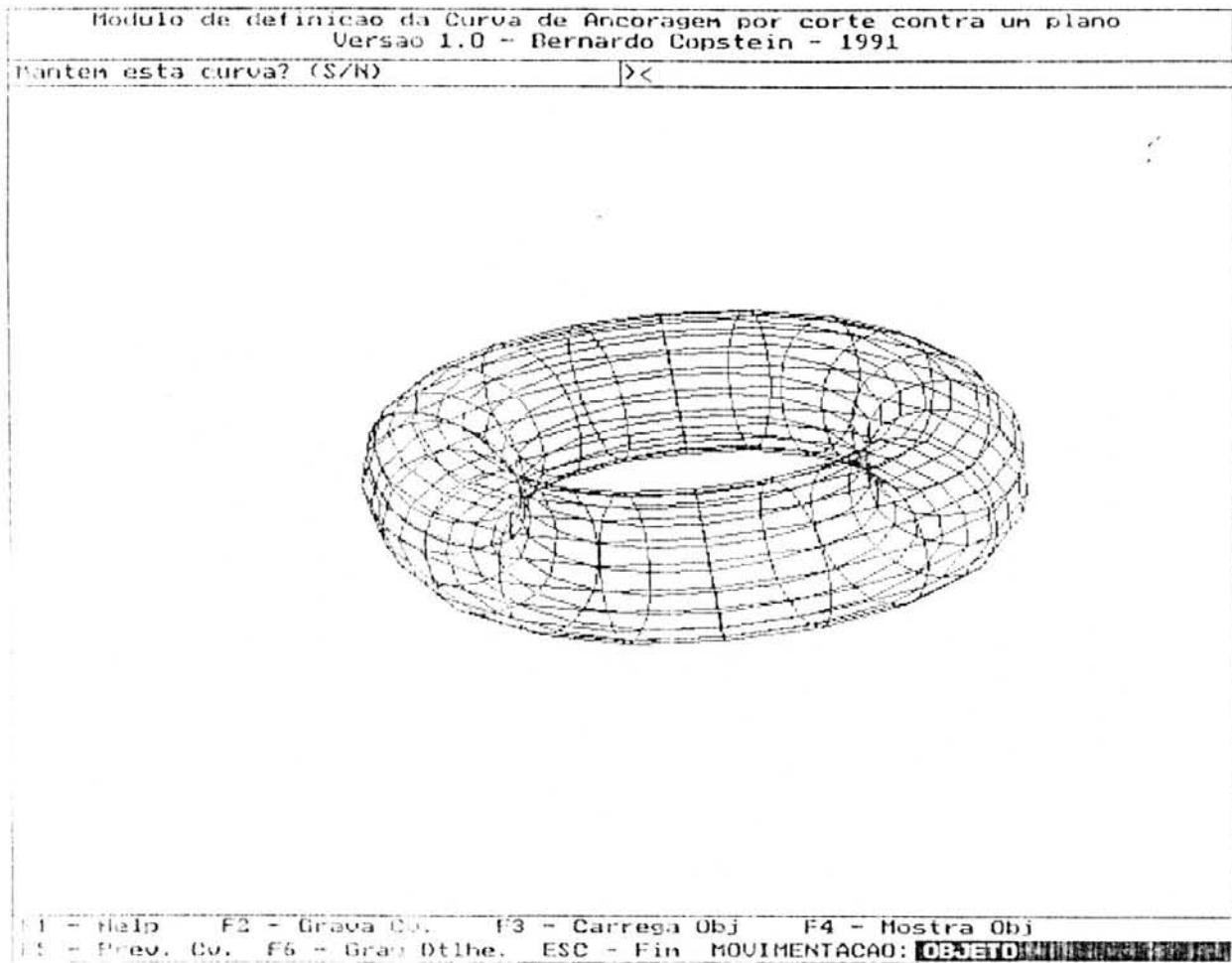


Figura 7.8 - Seleção do trecho do trecho da curva de ancoragem a ser selecionado.



série de arquivos menores que correspondem a aproximações por polígonos dos elementos que compõem a cena.

O módulo pode ser usado de maneira interativa se utilizado em conjunto com um editor de textos capaz de permanecer residente em memória (tipo "sidekick"). O fundamental para a perfeita interação com o módulo é o conhecimento do arquivo que define um objeto.

Um objeto pode ser composto por várias superfícies-base que conectar-se-ão entre si através de diversas superfícies de junção. Na figura 7.9, pode-se ver exemplos de arquivos de descrição de objetos.

Inicialmente especifica-se os parâmetros da câmera sintética que permitirá observar o resultado do trabalho. Para tanto devem ser indicados a posição do observador, alvo e ângulo de visão. Na seqüência, é especificado se o interpretador deverá apenas gerar uma imagem "wire-frame" ou também gravar a descrição do objeto final, para posterior visualização pelo sistema ANIMA. Em seguida, instanciam-se as superfícies-base. Deve-se indicar o nome do arquivo ".spl" que contém a superfície, um identificador numérico através do qual seja possível se referenciar a instância e os valores para as transformações geométricas de translação, escala e rotação em cada um dos eixos. O fato de cada

superfície-base corresponder a uma instância de uma

```

$CAMERA
ALVO          10.0  8.0  0.0
OBSERVADOR   -15.0 21.0 -30.0
ANG_ABERTURA 55.0
$MODO WIRE
$CARREGA_SUP
pl.spl 0      0.0  0.0  0.0  180.0  0.0 -90.0  1.0 1.0 1.0
pl.spl 1      10.0 10.0  0.0   0.0   0.0  0.0  1.0 1.0 1.0
pl.spl 2      10.0  0.0 -10.0  90.0   0.0  90.0  1.0 1.0 1.0
$FIMSUP
$GBLEND HERM
c3pl.can 0 INUC -10.0  c3pl.can 1 NORMC 10.0
c4pl.can 0 INUC -10.0  c1pl.can 2 NORMC -10.0
c1pl.can 1 INUC 10.0   c2pl.can 2 NORMC -10.0
$FIMBLEND
$ESPERA TECLA 13
$FIM

```

Figura 7.9 - Arquivo de descrição de objetos.

superfície "beta-spline" permite que uma mesma descrição de superfície possa ser usada diversas vezes na descrição de um mesmo objeto.

Todas as superfícies "beta-spline" utilizadas na "montagem" de um objeto são inicialmente centradas na origem do sistema de coordenadas, de forma que o usuário tenha um ponto de referência fixo para o instanciamento das superfícies-base.

A definição das junções segue a das definição das instâncias. Indica-se, então, o nome do arquivo que contém a descrição da curva de ancoragem que servirá de curva de partida para a junção e o número da instância (ou seja a superfície-base) sobre a qual será mapeada. Isso possibilita que uma mesma descrição de curva de ancoragem seja associada a diversas instâncias de superfícies "beta-spline" sobre a qual a curva foi definida. Por fim, deve-se indicar, também, um fator de multiplicação que será aplicado sobre os vetores de controle, influenciando seu sentido e seu tamanho. Para que a descrição da junção fique completa, essas informações devem ser repetidas descrevendo a superfície-base onde será feita a ancoragem da chegada da junção.

O sistema não faz verificação de consistência em relação a sobreposição ou cruzamentos de superfícies-base,

junções ou curvas de ancoragem.

Na figura 7.10 pode-se ver um exemplo completo, onde o arquivo de descrição do objeto é mostrado junto com o objeto final.

### 7.5 Observações

A forma como foi pensada a arquitetura do sistema permite grande flexibilidade na criação de objetos usando superfícies de junções na medida em que permite o aproveitamento de etapas do processo.

O fato dos módulos serem autônomos e independentes traz as vantagens citadas anteriormente, porém, o usuário que tiver de interagir diversas vezes com os diferentes módulos durante uma secção de trabalho pode se ressentir da falta de um ambiente integrado. Tal ambiente não existe, porém, devido as limitações do equipamento utilizado.

Finalmente, o módulo de geração de superfícies pode ter sua interface melhorada. Apesar do ambiente

computacionalmente restrito, poderá trabalhar com vistas bidimensionais e envelopes poliédricos de maneira a permitir o instanciamento interativo das superfícies-base. Acredita-se mesmo que o ideal seria que as instâncias e as junções pudessem ser definidas tanto textualmente como interativamente. Neste caso, a definição interativa serviria para um ajuste fino de uma definição inicialmente textual.

Apesar destas considerações, o sistema implementado cumpre seu objetivo de servir de meio de validação para o método apresentado. Atualmente, com as novas disponibilidades de equipamento (estações SUN) todo este protótipo poderia ser repensado e, provavelmente, muitas limitações impostas no próprio método não seriam mais consideradas.

```

$CAMERA
ALVO          10.0  8.0  0.0
OBSERVADOR   -15.0 20.0 -30.0
ANG_ABERTURA 110.0
$MODO WIRE
$CARREGA_SUP
  corpo.spl  0    0.0  0.0  0.0  90.0  0.0  0.0  1.0 1.0 1.0
  cilt.spl   1    0.0 20.0  0.0 -90.0  0.0  0.0  1.0 1.0 1.0
$FIMSUP
$GBLEND HERM
  c1corpo.gan 0 NORMC 12.0  c1cilt.gan 1 NORMC -10.0
$FIMBLEND
$ESPERA TECLA 13
$FIM

```

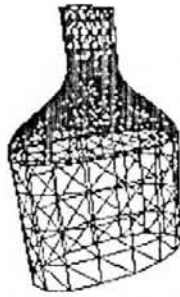


Figura 7.10 - Arquivo de descrição de objeto e o objeto correspondente.

## 8. ANÁLISE DE DESEMPENHO

Neste capítulo busca-se analisar os resultados obtidos através do uso do "software" implementado. Pretende-se assim, identificar as situações onde o mesmo apresenta um desempenho satisfatório. Desta maneira, espera-se fornecer subsídios que permitam nortear o uso ou não do método apresentado.

Inicialmente será feita a identificação das etapas do processo de modelagem que serão analisadas. Em seguida cada uma destas etapas será revista em separado. Por fim, é feita uma análise comparativa entre os objetos possíveis de se modelar com as técnicas apresentadas por vários autores e o presente trabalho.

### 8.1. Identificação das etapas analisadas

O método é composto por quatro etapas, a saber: modelagem das superfícies-base, cálculo das curvas de ancoragem (que no protótipo abrange também o cálculo dos vetores de controle) e geração das junções.

A etapa de modelagem das superfícies-base não foi considerada porque é uma etapa basicamente interativa, onde

os resultados dependem diretamente da habilidade do usuário em operar o sistema. Considerações sobre a influência do tipo de interface implementada na modelagem das superfícies, foram consideradas fora do contexto deste trabalho.

A etapa de cálculo das curvas de ancoragem mereceu maior atenção neste capítulo. Foram identificadas diversas variáveis que influenciaram nos resultados bem como determinados intervalos de valores e situações, onde o método de determinação de curvas de ancoragem funciona satisfatoriamente. São feitas considerações, também, sobre a geração dos vetores de controle.

Finalmente a etapa de geração das superfícies de junção mereceu, também, atenção. É abordada a questão da amostragem das superfícies de junção para efeito de visualização, bem como o cálculo dos vetores de controle.

## 8.2 Análise do módulo de cálculo das superfícies de ancoragem

Como já se viu anteriormente, o cálculo das curvas de ancoragem divide-se em três etapas: definição do plano de



corte, cálculo da curva de intersecção entre o plano e a superfície e cálculo dos vetores de controle.

A definição do plano de corte é uma etapa interativa e, portanto, não serão feitas considerações sobre a mesma. Já as etapas de cálculo da curva de intersecção e cálculo dos vetores de controle serão analisadas em maiores detalhes separadamente.

#### 8.2.1 Análise do método de determinação de curvas de intersecção entre um plano e uma superfície-base

O cálculo da curva de intersecção entre um plano e uma superfície-base é extremamente importante no contexto deste trabalho, uma vez que esta curva corresponde a curva de ancoragem utilizada na geração das superfícies de junção.

As curvas de ancoragem são descritas por uma lista de pontos. Em função desta forma de representação, existe uma preocupação muito grande com a quantidade de pontos que devem ser amostrados na mesma, de maneira que não se perca nenhum detalhe da curva. Dependendo da situação, porém, alguns detalhes podem ser considerados desprezíveis. Por esta razão criou-se o que se chamou de "grau de detalhe". O "grau de detalhe" é um parâmetro relativo que permite indicar se a curva de intersecção entre o plano de corte e a

superfície-base (curva de ancoragem) deve ser amostrado com maior ou menor riqueza de detalhe. Esse "grau de detalhe" corresponderá a um valor em graus que especifica o ângulo máximo tolerado entre os vetores tangentes à curva de intersecção calculados em dois pontos consecutivos.

Um dos principais problemas da implementação do protótipo é decorrente do fato de se ter trabalhado com um fator de tolerância bastante elevado. Devido a problemas com o armazenamento de números em ponto flutuante, foram considerados iguais valores com uma diferença menor do que  $10^{-5}$ . Sendo assim, quando se usam valores pequenos para o "grau de detalhe", o algoritmo de movimentação tem dificuldade de distinguir o avanço sobre a curva de intersecção já que a distância entre os pontos ou o valor do ângulo entre as tangentes podem situar-se neste intervalo.

Um problema mais grave, ocorre quando o algoritmo de Newton-Raphson não converge (por não ser a estimativa inicial suficientemente próxima ou por haver uma simetria ao redor do ponto) forçando a troca de isoparamétrica (ver capítulo 4). Essa troca inoportuna pode trazer como consequência a inversão do sentido da movimentação fazendo com que se percorram novamente pontos já visitados da curva. Podem ocorrer problemas, também, quando os valores do "grau de detalhe" forem muito altos. Nesta situação a "fórmula de

aceleração" da movimentação (ver capítulo 4) pode gerar estimativas iniciais muito grosseiras para que o método de Newton possa convergir.

Finalmente a implementação do protótipo impõe certos limites (por razões de gerenciamento de memória) ao número máximo de pontos que podem ser amostrados sobre uma curva de ancoragem. Quanto menor o valor indicado para o "grau de detalhe", maior a probabilidade de se alcançarem estes limites.

Procurou-se analisar, também, o tempo de resposta do método de determinação das intersecções em relação ao "grau de detalhe" especificado e ao número de "patches" da superfície. Essa análise é importante porque, por ser um "software" interativo, convém ao usuário do sistema ter conhecimento de que valores devem ser usados para o "grau de detalhe" nos diferentes momentos do processo de modelagem.

Através de diversas medições (ver gráficos 1,2,3 e 4) concluiu-se que o algoritmo implementado possui um comportamento exponencial tanto do tempo de resposta quanto do número de pontos obtidos em relação ao "grau de detalhe". A variação do número de "patches" da superfície a ser interseccionada não implicou em uma variação linear do tempo de resposta, se verá em seguida, como seria intuitivo

esperar.

Todas as medições foram feitas em um equipamento IBM-PC AT 386 compatível com relógio de 20 Mhz e co-processador matemático 80387.

Superfície: esfera.spl  
Número de "patches": 16

Grau de Detalhe	Num. Pts.	Tempo	"Patches" Intersec.
3	ERRO	(memória)	8
5	67	8.97	8
10	42	6.89	8
15	33	4.72	8
20	29	4.12	8
25	24	4.86	8

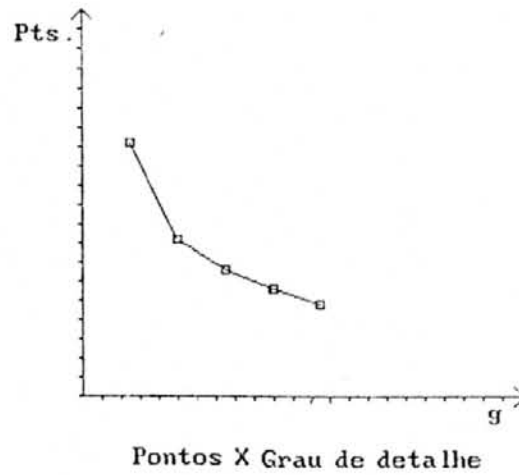
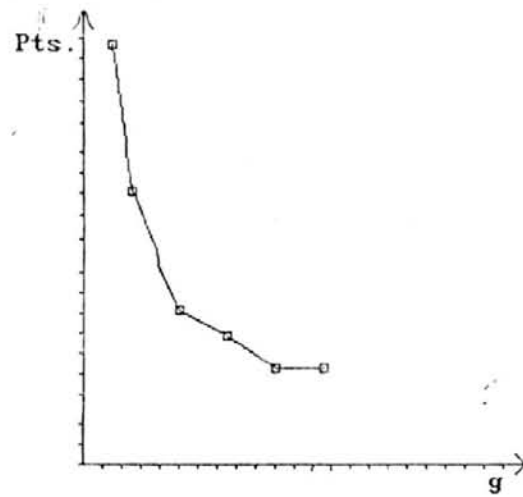


Gráfico 1.

Superfície: esfera.spl  
 Número de "patches": 16

Grau de Detalhe	Nun. Pts.	Tempo	"Patches" Intersec.
3	97	8.79	6
5	61	5.7	6
10	37	3.6	6
15	29	3.07	6
20	23	2.69	6
25	23	2.57	6

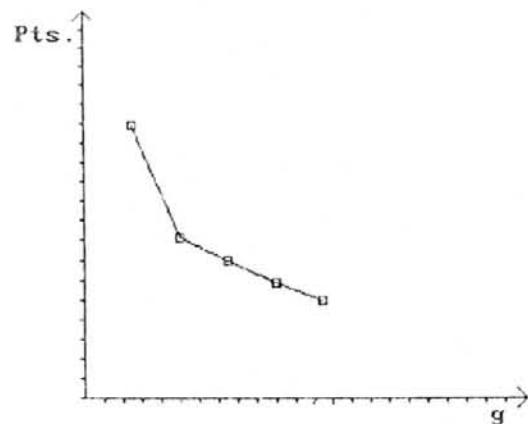


Pontos X Grau de detalhe

Gráfico 2.

Superfície: gota.spl  
 Número de "patches": 16

Grau de Detalhe	Nun. Pts.	Tempo	"Patches" Intersec.
3	ERRO	(inversao)	4
5	69	9.2	4
10	41	6.15	4
15	35	5.16	4
20	29	4.39	4
25	25	4.0	4



Pontos X Grau de detalhe

Gráfico 3.

Superfície:  
Número de "patches":

Grau de Detalhe	Num. Pts.	Tempo	"Patches" Intersec.
3	92	7.47	4
5	58	4.94	4
10	34	3.18	4
15	22	2.36	4
20	20	2.14	4
25	16	1.7	4

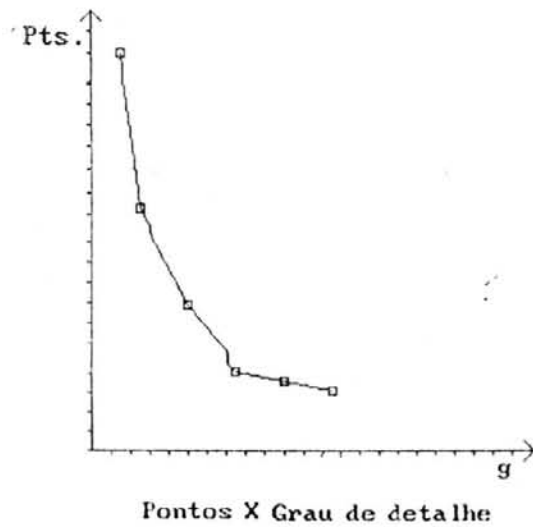


Gráfico 4.

Pela análise dos gráficos, pode-se observar que valores maiores que 15 para o "grau de detalhe" passam a ser injustificados, pois tanto o tempo de resposta quanto o número de pontos obtidos tende a se estabilizar. Da mesma forma, valores menores que 5 implicam em uma taxa de insucesso (situações onde o protótipo entrou em condição de erro) elevada.

A utilização de diferentes superfícies a serem interseccionadas mostrou que o fator de influência no tempo

de resposta não é o número de "patches" da superfície mas sim o número de "patches" cortados pela curva de intersecção e a posição do plano de corte. Esses dois fatores influenciam na medida em que a etapa de reticulado do método de determinação da curva de intersecção (ver capítulo 4) só será aplicada a cada um dos "patches" da superfícies após ter sido feito o teste do envelope poliédrico sobre o "patch" em questão. O teste só é conclusivo em afirmar que um dado "patch" não é candidato a conter parte da curva de intersecção. Dependendo da posição do plano de corte pode ser necessário aplicar a etapa do reticulado sobre "patches" onde não existiam trechos da intersecção.

A partir de todas essas observações foi possível concluir que o método apresentado obtém resultados satisfatórios quando trabalha-se com os valores para o grau de detalhe contidos no intervalo [5;15].

Cabe ressaltar, também, que apesar do "grau de detalhe" não ser um parâmetro que permita calcular com exatidão o número de pontos que serão amostrados sobre a curva (devido a imprecisões e arredondamentos inseridos nas diversas etapas do método), o usuário do sistema com pouco tempo de uso já é capaz de prever qual o "grau de detalhe" que corresponde a suas expectativas.

### 8.2.2 Análise do cálculo dos vetores de controle

O fato de se dispor do plano de corte facilita bastante o cálculo dos vetores de controle. Na maioria dos artigos consultados os vetores de controle são dependentes do posicionamento das superfícies a serem unidas. O conhecimento do plano de corte que gerou a curva de ancoragem permite a definição de vetores de controle independentes do par de superfícies que serão unidas capazes de garantir continuidade geométrica  $G^1$  entre a superfície base e qualquer superfície definida a partir da mesma.

Para que se possa afirmar que se obteve continuidade geométrica  $G^1$  no ponto de contato entre a superfície-base e a superfície de junção deve-se mostrar que os vetores tangentes calculados sobre ambas superfícies no mesmo ponto de contato tem a mesma direção (ver cap. 3). Como a superfície de junção é uma superfície Hermite cúbica, o próprio vetor de controle é um vetor tangente à mesma no ponto de contato. Resta mostrar, então, que esse mesmo vetor de controle é também tangente à superfície-base. Tal pode ser feito se for analisada a maneira pela qual o vetor de controle foi determinado.

Pelo método escolhido, como se viu no capítulo 5, precisa-se em primeiro lugar de um vetor  $\mathbf{n}$  normal à



superfície  $S$  no ponto  $m(t)$  da curva de ancoragem  $M$ . Pode-se obter o vetor  $n$  pelo produto vetorial das derivadas parciais de  $S$  em relação a seus parâmetros  $u$  e  $v$ :

$$\mathbf{n} = \frac{\partial S}{\partial u} \times \frac{\partial S}{\partial v}$$

como  $S = S(u(t), v(t))$ , tem-se:

$$\mathbf{n} = \frac{\partial S}{\partial u} \frac{\partial u}{\partial t} \times \frac{\partial S}{\partial v} \frac{\partial v}{\partial t}$$

Além do vetor  $n$ , é necessário um segundo vetor chamado  $f$  tangente à curva  $M$  no ponto  $m(t)$ . Então:

$$\mathbf{f} = m'_t(t)$$

Sobre os vetores  $n$  e  $f$  pode-se compor o seguinte raciocínio:

a) O vetor resultante do produto vetorial  $n \times f$  é ortogonal a  $n$  e, portanto, tangente à superfície  $S$  no ponto  $m(t)$  e

b) O vetor resultante do produto vetorial  $n \times f$  é ortogonal ao vetor  $f$  e por isso ortogonal à curva  $M$  no ponto  $m(t)$ .

Assim, o produto  $n \times f$  fornece um vetor que

tangencia a superfície  $S$ . Desta forma pode-se concluir que este vetor aponta na mesma direção que a borda da superfície  $S$  no ponto  $m(t)$ , servindo como vetor de orientação para a superfície de junção.

Existe uma situação onde o cálculo dos vetores de controle, tal como foi apresentado, pode não funcionar a contento. Se a curva de intersecção passar por um ponto de acumulação da superfície "beta-spline" (um ponto gerado por um vértice de controle duplo ou triplo onde o vetor tangente tem módulo zero), haverá uma descontinuidade na função da derivada primeira da superfície neste ponto. A solução, neste caso, é trabalhar com um valor para a derivada primeira estimado com base nos valores das derivadas nos pontos vizinhos ao mesmo.

### **8.3. Análise do módulo de geração de superfícies de junção**

O módulo de geração de superfícies de junção possui uma etapa interativa bastante significativa onde são determinados os pares de superfícies a serem unidos, a direção dos vetores de controle e o fator de multiplicação

dos mesmos. Nesta etapa, o único cuidado necessário é com o fator de multiplicação dos vetores de controle. Se este for exagerado, a superfície de junção resultante poderá apresentar auto-interceptações. O valor máximo que se pode indicar para o fator de multiplicação variará conforme o caso em função da distância entre as superfícies base e do fato de se aplicarem valores diferenciados nos vetores de controle de partida e chegada.

Se for considerado que, em uma superfície de junção de parâmetros  $s$  e  $t$  (fig. 8.1), o parâmetro  $s$  corresponde a parametrização da superfície na direção das curvas de ancoragem e que o parâmetro  $t$  corresponde a parametrização das curvas ortogonais as curvas de ancoragem pode-se observar que:

a) Como foi mostrado no capítulo 5, para que seja possível gerar uma junção utilizando duas curvas de ancoragem distintas, é necessário que os pontos amostrados sobre as duas curvas de ancoragem correspondam a união dos pontos originalmente amostrados sobre cada uma. Isso implica que o sistema não tem condições de saber, até o momento da definição do par de superfícies que serão unidas, qual será o grau de detalhe efetivo com que as curvas de ancoragem serão amostradas no momento da geração da junção. Sabe-se

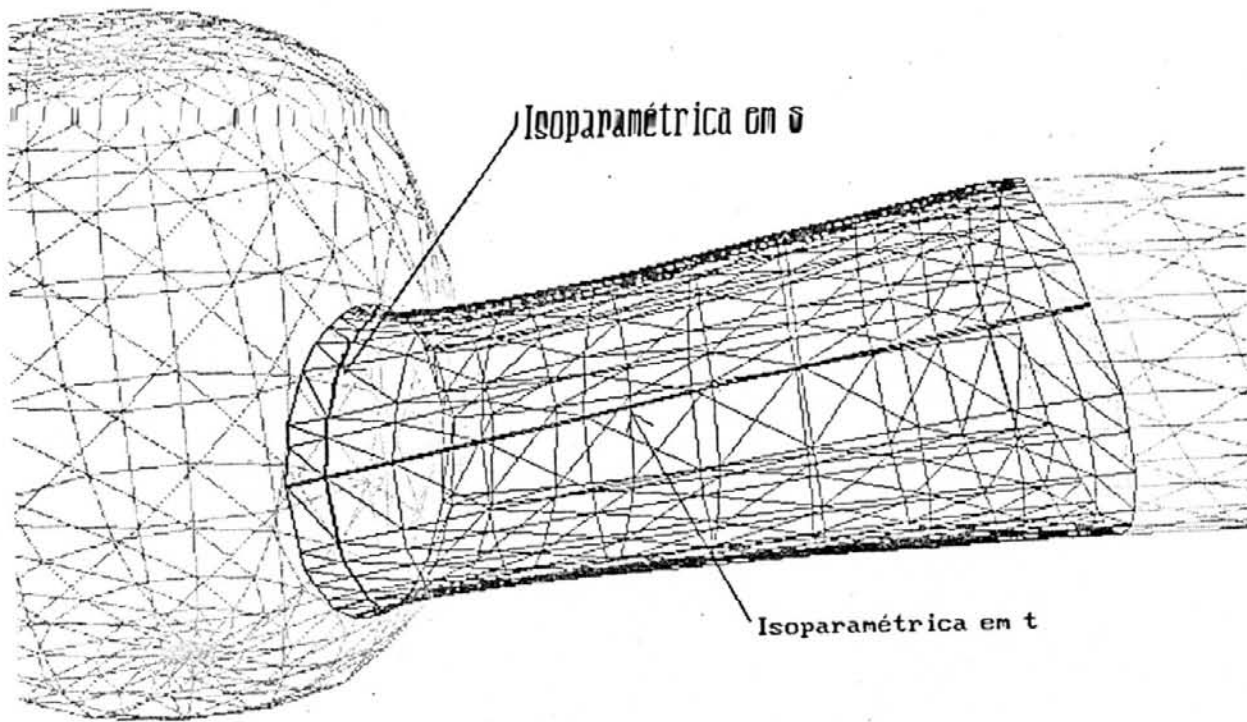


Figura 8.1 - Parametrização de uma superfície de junção.

apenas que a curva será amostrada com um grau de detalhe menor (curva mais detalhada) do que aquele que foi especificado quando da geração das curvas de ancoragem. Esse

fato pode ser problemático em ambientes onde existem problemas de espaço de armazenamento para a descrição das superfícies de junção. Por outro lado, pode ser uma vantagem quando houver necessidade da maior quantidade de detalhe possível.

b) Na versão atual do sistema, as curvas isoparamétricas em  $t$  são amostradas a intervalos constantes para efeito de visualização. Desta maneira, é possível que a amostragem regular acabe por omitir detalhes importantes destas curvas. Uma solução para este problema é aplicar a mesma idéia usada para as isoparamétricas em  $s$ , ou seja, analisar o comportamento de cada uma das isoparamétricas em  $t$ , determinar os pontos de amostragem segundo um parâmetro que indique o grau de detalhe desejado e, finalmente, amostrar todas as curvas pelo conjunto união dos pontos amostrados sobre todas as isoparamétricas individualmente. Essa solução, entretanto, foi possível na direção do parâmetro  $s$  porque nesta direção o comportamento da superfície pode ser descrito por apenas duas curvas. No sentido do parâmetro  $t$ , o número de curvas a ser analisado é muito elevado (igual ao número de pontos amostrados sobre as curvas em  $s$ ) e outro tipo de solução deve ser pesquisado.

#### 8.4 Análise comparativa

De maneira que se possa avaliar objetivamente os resultados apresentados, será feita uma análise comparativa entre os resultados obtidos pelos diversos autores dos artigos que inspiraram este trabalho, o método proposto no capítulo três e as possibilidades do protótipo.

Quando se compara o método proposto com o método apresentado no capítulo 3, verifica-se que a grande vantagem deste é o fato de que o mesmo garante continuidade geométrica de primeira ordem ( $G^1$ ) ao longo das curvas de ancoragem. O método proposto, por outro lado, garante esta continuidade apenas nos pontos amostrados sobre as curvas de ancoragem. Isso se deve ao fato de se utilizar interpolação linear entre os pontos amostrados gerando possíveis falhas entre a superfície de junção e as superfícies-base. Foi discutido no item 5.1, entretanto, que essa descontinuidade sempre terá uma magnitude inferior ao "grau de detalhe" especificado. De qualquer forma, este fato é compensado pela maior flexibilidade oferecida decorrente do fato das curvas de ancoragem serem definidas através de um plano de corte ao invés de isoparamétricas.

No artigo de Filip /FIL 89/ destacam-se dois tipos de superfícies de junção. Primeiramente citam-se superfícies

utilizadas na suavização de quinas ou cantos formados pelo encontro de duas superfícies quaisquer. Um exemplo seria a tentativa de se prover uma transição suave entre dois planos dispostos ortogonalmente como pode ser visto na figura 8.2. Esse tipo de problema é facilmente resolvível pelo sistema apresentado, uma vez que se pode determinar com facilidade as curvas de ancoragem utilizando um plano de corte.

Um segundo tipo de junção comentado por Filip são as chamadas transições de raio constante. São situações onde as curvas de ancoragem assemelham-se a círculos e a junção tem a forma aproximada de um cilindro. Como pode-se ver pela figura 8.3 o sistema proposto não tem dificuldade para gerar tais junções desde que seja possível determinar as curvas de ancoragem utilizando um plano de corte.

A única situação apresentada no artigo de Filip que não é possível de se resolver com o sistema proposto é o tipo de situação que se forma quando se constrói um objeto que se assemelha a um cubo, unindo-se as faces de um cubo duas a duas por superfícies de junção. Nos cantos de tal objeto, formar-se-ão regiões abertas cuja superfície de transição necessária para cobri-los deve ser capaz de unir três superfícies ao mesmo tempo (fig. 8.4).

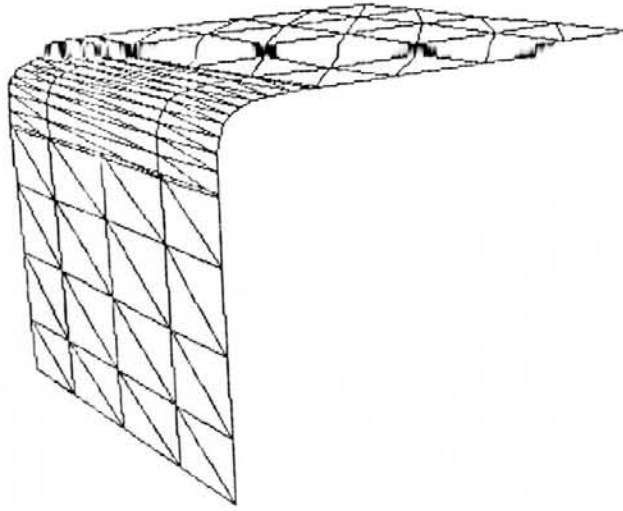


Figura 8.2 - Transição suave entre dois planos.



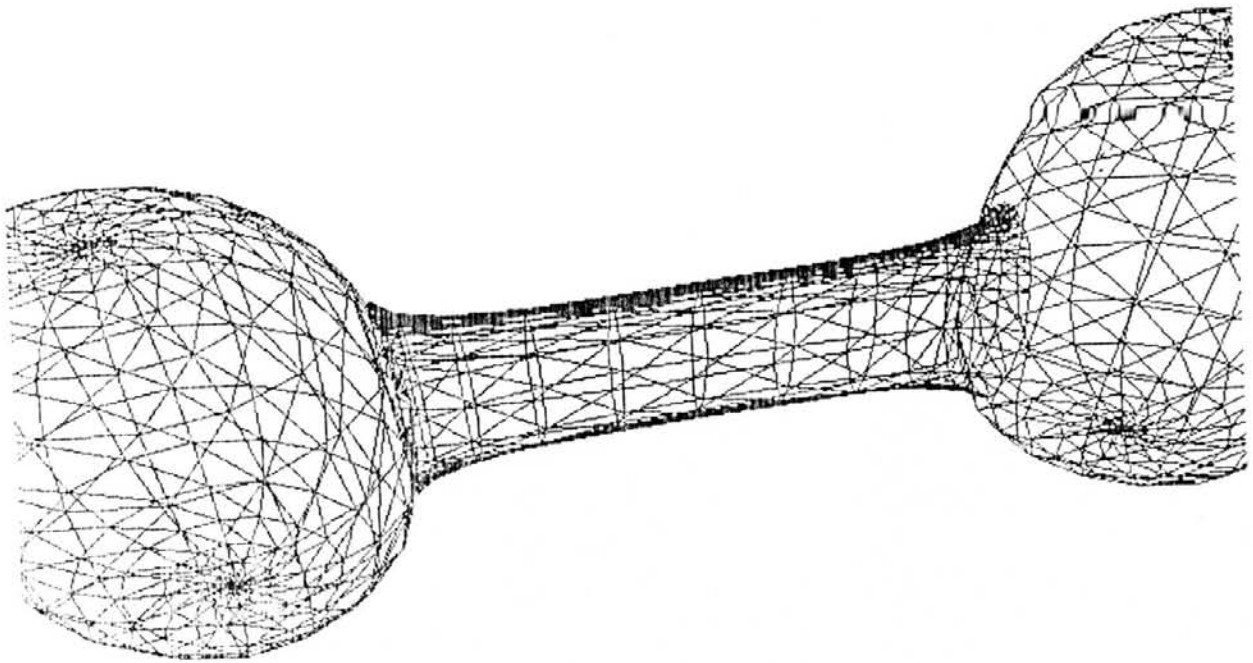


Figura 8.3 - Transição de raio constante.

Já os objetos modelados com superfícies de transição entre superfícies algebricamente definidas descritas no artigo de Warren /WAR 89/, podem ser perfeitamente modelados pelo sistema proposto desde que as superfícies algebricamente definidas sejam modeladas como superfícies "beta-splines". Na figura 8.5, tem-se um exemplo de um dos objetos propostos por Warren modelados no sistema apresentado neste trabalho.

O método de eliminação de cantos e quinas para a suavização do encontro de duas superfícies que se interceptam apresentado por Choi e Sy Ju /SY 89/, também pode ter seus resultados simulados pelo sistema em questão (fig. 8.6). Só não serão possíveis de serem simulados os casos que envolvem mais de duas superfícies ao mesmo tempo.

Finalmente Bardis em seu artigo "Blending Rational Surfaces" /BARD 89/ apresenta um método cuja abordagem é bastante semelhante à do método apresentado. A única vantagem do método apresentado por Bardis é que ele gera uma descrição única para superfícies base e junções, não fazendo distinção entre as mesmas após estas terem sido geradas.

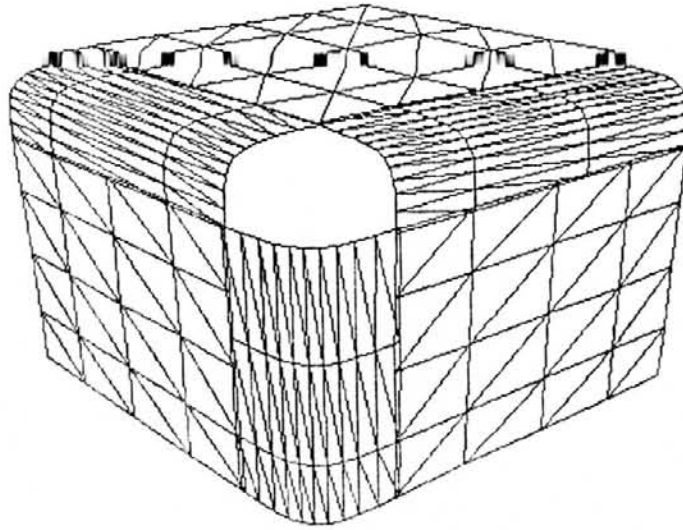


Figura 8.4 - Situação onde é necessária a transição entre três superfícies.

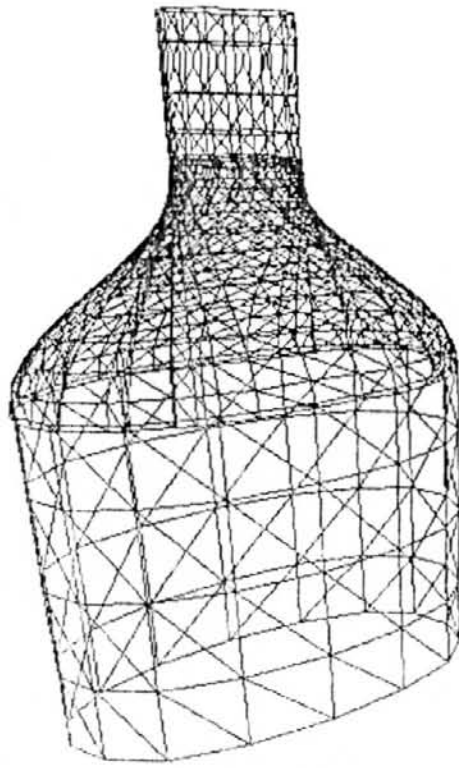


Figura 8.5 - Objeto equivalente aos modelados por Warren.

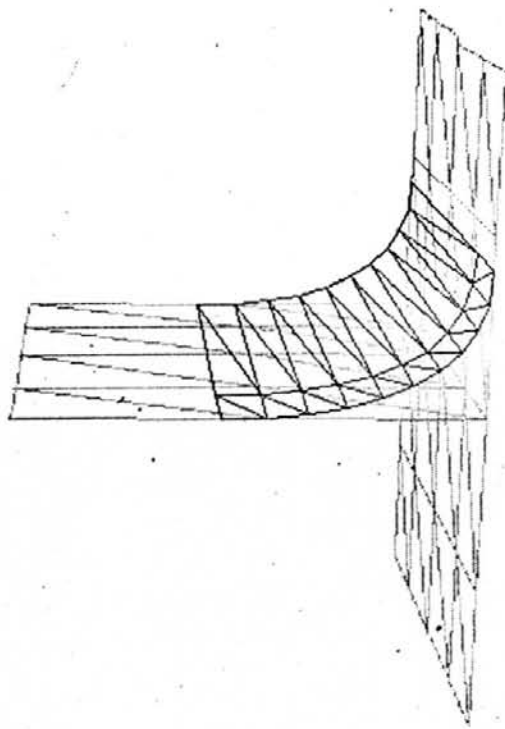


Figura 8.6 - Suavização do encontro de duas superfícies.

De uma maneira geral pode-se dizer que o método

apresentado embora não acrescente nenhuma solução inovadora, procura reunir todas as qualidades das diversas soluções estudadas em um ambiente flexível e compacto, capaz de oferecer um tempo de resposta razoável em um ambiente de pequeno porte. Desta forma, desde que se respeite o fato de que o sistema só trata junções entre duas outras superfícies, pode-se afirmar que o mesmo se apresenta genérico suficiente para abranger a maioria das situações mencionadas nos artigos verificados.

Por fim, pode-se dizer que o método apresentado se destaca em relação aos demais devido à grande velocidade no cálculo das junções propriamente ditas. Este fato se deve, sobretudo, à total independência das curvas de ancoragem e vetores de controle da junção a ser gerada, bem como da simplicidade do cálculo das junções propriamente ditas. Esta característica é aproveitada sobretudo para permitir a definição interativa das junções. Outra grande aplicação é a facilidade para se gerarem seqüências animadas onde as junções modelam-se de acordo com tranformações geométricas aplicadas sobre as superfícies base ou mesmo pela substituição das mesmas.

## 9. Conclusão

No decorrer deste trabalho pode-se verificar que os aspectos relevantes na implementação de um sistema de modelagem baseado em superfícies de junção dizem respeito sobretudo a determinação das curvas de ancoragem e dos vetores de controle. As curvas de ancoragem são, porém, a etapa mais delicada devido a inexistência de algoritmos consagrados.

Dentro de certas limitações, o protótipo implementado permite a modelagem de objetos compostos utilizando superfícies de junção, provando assim a funcionalidade do método proposto.

Se este trabalho fosse iniciado hoje, certamente muitas das soluções e simplificações adotadas teriam de ser revistas. Este fato se explica por dois motivos:

a) O tipo de equipamento atualmente disponível no CPGCC/UFRGS é mais avançado do que aqueles que existiam na época do início deste trabalho e muitas das considerações que foram feitas a respeito da capacidade do equipamento não seriam relevantes e

b) Grande parte dos artigos citados na

bibliografia tornaram-se disponíveis depois de certas etapas já terem sido definidas ou implementadas.

De qualquer forma, acredita-se que estas considerações apenas destacam a atualidade do problema estudado e sugerem a continuidade deste trabalho.

Propõem-se, então, alguns itens como sugestão para futuras extensões deste trabalho.

a) A utilização de outros métodos numéricos nas situações onde o método de Newton não funciona bem;

b) A busca de alternativas para a determinação de curvas de ancoragem de maneira a ampliar a família de curvas possíveis de se definir. A utilização de uma superfície qualquer ao invés de um plano de corte é uma possibilidade;

c) A utilização de uma terceira curva de ancoragem, intermediária, que obrigue a superfície de junção a passar por uma determinada região aumentando o controle sobre sua forma final.

d) A conversão da representação Hermite utilizada com as superfícies de junção para uma representação "beta-spline" equivalente a usada com as superfícies-base. A



uniformidade das representações permitiria uma uniformidade nos algoritmos de tratamento das superfícies que descrevem os objetos bem como a determinação de junções entre junções;

e) A característica das superfícies de junção de se ajustarem às transformações geométricas aplicadas sobre as superfícies base pode ser melhor explorada em um ambiente onde interesse trabalhar com superfícies elásticas e

f) A modularidade do sistema e a possibilidade do reaproveitamento das superfícies base e curvas de ancoragem podem ser explorados de forma a definir um ambiente de projeto cooperativo.

## BIBLIOGRAFIA

- [AHL 90] AHLERS, Martin; HOCHFELD, Hans Jurgen    Role of Bezier Curves and Surfaces in the Volkswagen CAD approach for 1967 to today    Computer Aided Design, v. 22, n. 9, nov. 1990.
- [BAR 89] BARDIS, L.; PATRIKALAKIS, N. M.    Blending Rational B-spline Surfaces - In: European Computer Graphics (EUROGRAPHICS'89), Sept. 4-8, Hamburg, Frg.    Proceedings... Amsterdam: North Holland, 1989.
- \*[BAR 89a] BARSKY, Brian A.; BARTELS, Richard H.; BEATY, John C.    An introduction to Splines for use in Computer Graphics and Geometric Modeling    Los Altos, California: Morgan Kaufmann, 1987.
- [BAR 87b] BARSKY, Brian A.; DEROSE, Tony D.; DIPPE, Mark D.    An adaptative Subdivision Method with Crack Prevention for Rendering Beta-Spline Objects    University of California, Computer Science Division, Report n0. UCB/CSD 87/348, Berkeley, California, 1987.
- [BAR 74] BARTCH, Hans-Jochen    Handbook of Mathematical Formulas    New York: Academic Press, 1974.

- [CHE 88] CHEN, J. J.; OZSOY, T.M. Predictor Corrector type of Intersection Algorithm for  $C^2$  Parametric Surfaces Computer Aided Design, v. 20, n. 6, 1988.
- [CON 67] COONS, S. A. Surfaces for Computer Aided Design of Space Forms. Cambridge, MA:M.I.T., June 1967 (Project Mac, TR41).
- [COP 90] COPSTEIN, Bernardo. MSS-Modelador de Sólidos baseado em Beta Splines. Porto Alegre: CPGCC da UFRGS, jul. 1990 (Trabalho Individual, 181).
- [FAU 79] FAUX, I. D.; Pratt, M. J. Computational Geometry for Design and Manufacture, New York: John Wiley & Sons, 1979.
- \*[FIL 89] FILIP, Daniel J Blending Parametric Surfaces ACM Transactions on Graphics, v. 8, n. 3, July 1989.
- [LUK 89] LUKACS, Gabor The Generalized Inverse Matrix and the Surface Surface Intersection Problem. In: STABER, W.; SEIDEL, H. P. Teory and Practice of Geometric Modeling. New York: Springer Verlag, 1989.
- \*[MOR 85] MORTENSON, Michael Geometric Modeling. New York: John Wiley & Sons, 1985.


- [ROC 89] ROCKWOOD, Alyn P. The Displacement Method for Implicit Blending Surfaces in Solid Models ACM Transactions on Graphics, v. 8, n. 4, Oct. 1989.
- [SYJ 89] SY JU; CHOI B. K. Constant Radius Blending in Surface Modeling. Computer Aided design, v. 21, n. 4, May 1989.
- [WAR 89] WARREN, Joe Blending Algebraic Surfaces. ACM Transactions on Graphics , v. 8, n. 4, Oct. 1989.




**Informática**  
UFRGS

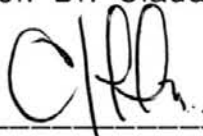
"Análise de um modelador de objetos que utiliza superfícies de junção para unir superfícies paramétricas compostas Beta-Spline".

Dissertação apresentada aos Srs.:

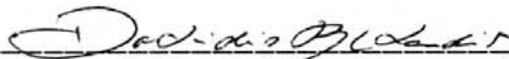
  
-----  
Prof. Anatólio Laschuk

  
-----

Prof. Dr. Cláudio Walter

  
-----

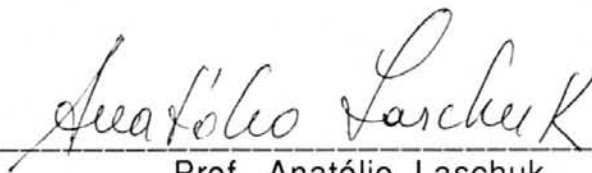
Prof. Dr. Clesio Luis Tozzi (UNICAMP)

  
-----

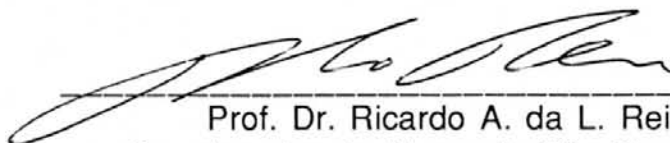
Prof. Dr. Dalcidio Moraes Claudio

Vista e permitida a impressão.

Porto Alegre, 28 / 10- / 92.

  
-----

Prof. Anatólio Laschuk  
Orientador.

  
-----

Prof. Dr. Ricardo A. da L. Reis,  
Coordenador do Curso de Pós-Graduação  
em Ciência da Computação.