

## 6

### Reconstrução Pseudo-Tricúbica

ADFs podem ser implementadas com base em uma variedade de funções de distância, estruturas espaciais e métodos de reconstrução (Seção 2.2.2) — além da combinação clássica de *octree* e interpolação trilinear. No curso deste trabalho, foram investigadas diferentes estruturas e métodos de reconstrução que pudessem reparar algumas das fraquezas das ADFs clássicas. Por falta de tempo, este estudo não pôde ser tão amadurecido quanto desejava-se. No entanto, este capítulo inclui um resumo dos resultados obtidos, por serem considerados relevantes.

Em relação à estrutura espacial das ADFs, as *octrees* oferecem uma boa combinação de simplicidade e adaptabilidade, e são difíceis de serem substituídas. Em algumas situações, uma árvore  $k$ -dimensional poderia oferecer vantagens sobre a *octree*; no entanto é possível estimar que, mesmo que este ganho exista, não seria grande o suficiente para justificar a mudança.

A pesquisa de um método de reconstrução mais poderoso mostrou-se bem mais promissora. De fato, trabalhos anteriores já haviam explorado métodos de reconstrução mais poderosos, embora menos eficientes que a interpolação trilinear. Uma das pesquisas mais relevantes, empreendida por Bajaj *et al.* [3], explorou *interpolação tricúbica* nas células de uma *octree*. Os campos tricúbicos podem aproximar melhor superfícies curvas e, ao mesmo tempo, garantir continuidade  $C^1$ ; mas por outro lado, têm custos de memória e reconstrução muito altos quando comparados aos campos trilineares.

O objetivo desta pesquisa foi desenvolver um método de reconstrução com desempenho comparável ao da interpolação trilinear, mas capaz de representar melhor superfícies curvas. O método desenvolvido consiste em uma simplificação da interpolação tricúbica, denominada *interpolação pseudo-tricúbica*. As seções seguintes descrevem a motivação para o desenvolvimento do método, como ele foi implementado, e os resultados obtidos até o momento.

## 6.1 Motivação

O principal problema dos campos trilineares é que eles não aproximam bem os campos de distância de superfícies curvas. Um campo trilinear possui quatro termos não-lineares  $(xy, xz, yz, xyz)$  que conferem alguma curvatura às iso-superfícies. Porém, a curvatura máxima que se pode obter na prática é muito limitada, como pode ser observado na Figura 6.2(a).

Em diversas apresentações sobre ADFs, Frisken *et al.* mostraram a Figura 6.1 como argumento de que os campos trilineares podem conter iso-superfícies muito curvas. Infelizmente, o exemplo mostrado é totalmente artificial, e na prática os campos trilineares das células de ADFs não têm mais curvatura do que a superfície ilustrada na Figura 6.2(a). Este comportamento é o esperado, já que as oito amostras de distância de um campo trilinear contêm pouquíssima informação para descrever uma superfície curva.

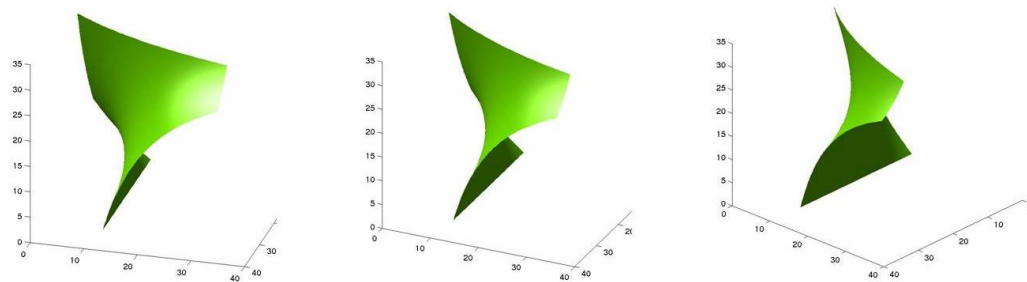


Figura 6.1: Iso-superfície de um campo trilinear não-Euclidiano. Imagens divulgadas por Frisken *et al.* [10].

Para avaliar a capacidade de representação de superfícies curvas dos campos trilinear, tricúbico e pseudo-tricúbico, o seguinte experimento foi realizado. Uma célula foi configurada com amostras do campo de distância correspondente a  $\frac{1}{8}$  de uma esfera de raio unitário. Em seguida, a superfície contida na célula foi plotada utilizando os três métodos de interpolação citados. Os diferentes resultados podem ser observados na Figura 6.2. Como esperado, a iso-superfície do campo trilinear não representa bem a curvatura do pedaço de esfera; o campo tricúbico aproxima o pedaço de esfera com grande precisão; e o campo gerado pela interpolação pseudo-tricúbica proposta aproxima-se bastante do campo tricúbico.

O maior obstáculo para a utilização de interpolação tricúbica em uma ADF é o custo de memória. Enquanto que um campo trilinear requer uma única amostra da função de distância ( $f$ ) por ponto, um campo tricúbico requer oito

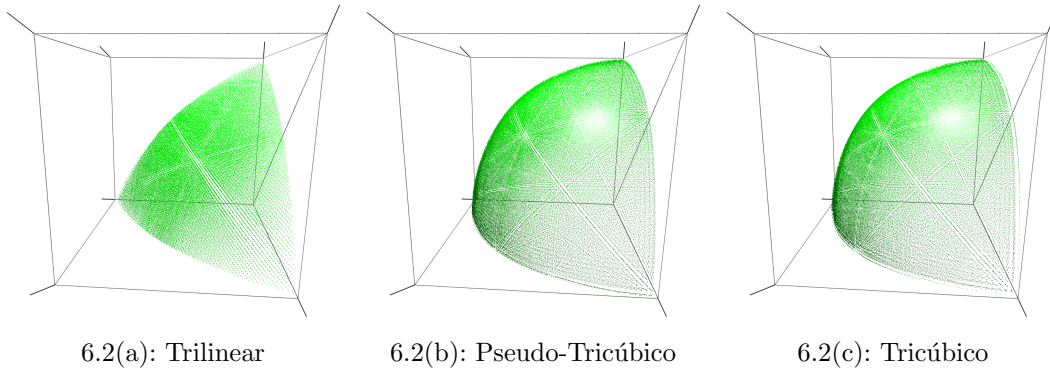


Figura 6.2: Célula de ADF, configurada com amostras do campo de distância de  $\frac{1}{8}$  de uma esfera, plotada por três interpoladores diferentes.

valores [3]:

$$f, \frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z}, \frac{\partial^2 f}{\partial x \partial y}, \frac{\partial^2 f}{\partial x \partial z}, \frac{\partial^2 f}{\partial y \partial z}, \frac{\partial^3 f}{\partial x \partial y \partial z}$$

Infelizmente, o custo de oito escalares por *voxel* é muito alto, e dificilmente compensaria a substituição da reconstrução trilinear.

A proposta da interpolação pseudo-tricúbica é ser um híbrido entre a interpolação trilinear e a tricúbica — uma interpolação mais poderosa que a trilinear e mais eficiente que a tricúbica. Ao invés de oito escalares por ponto, a interpolação pseudo-tricúbica requer quatro: uma distância e um gradiente. Ambos os valores já são calculados para as ADFs desde a introdução da textura de normais (Seção 4.4.2). A Tabela 6.1 mostra um comparativo dos custos dos três métodos de interpolação mencionados.

	Trilinear	Pseudo-Tricúbica	Tricúbica
Valores por Ponto	$f$	$f, \frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z}$	$f, \frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z}, \frac{\partial^2 f}{\partial x \partial y}, \frac{\partial^2 f}{\partial x \partial z}, \frac{\partial^2 f}{\partial y \partial z}, \frac{\partial^3 f}{\partial x \partial y \partial z}$
Operações	7 interpolações lineares.	7 interp. cúbicas e 10 lineares.	21 interp. cúbicas.

Tabela 6.1: Comparativo de custos dos métodos de interpolação.

## 6.2 Implementação

As ADFs pseudo-tricúbicas contam com as seguintes mudanças em relação às ADFs trilineares dos capítulos anteriores:

1. Não existe textura de normais. As normais usadas na iluminação são calculadas pelo gradiente do campo de distância local de uma célula.

2. A textura de *voxels* possui quatro canais ao invés de um. Cada *texel* armazena uma distância e um gradiente.
3. Tanto na construção da ADF quanto na reconstrução do campo de distância, utiliza-se a interpolação pseudo-tricúbica ao invés da trilinear.

Assim como outras interpolações 3-D, a interpolação pseudo-tricúbica é formada por uma seqüência de interpolações unidimensionais. Esta seqüência é dividida em sete passos, como ilustrado na Figura 6.3: os primeiros quatro passos interpolam os *voxels* da célula ao longo do eixo  $x$ ; os dois passos seguintes interpolam os valores dos quatro passos anteriores ao longo do eixo  $y$ , e o último passo interpola os valores dos dois passos anteriores ao longo do eixo  $z$ . Na interpolação trilinear, cada um destes passos realiza somente uma interpolação linear entre dois pontos. Já na interpolação tricúbica, cada passo realiza uma *interpolação cúbica de Hermite*, que requer tanto os valores de distância quanto as derivadas parciais ao longo do eixo de interpolação.

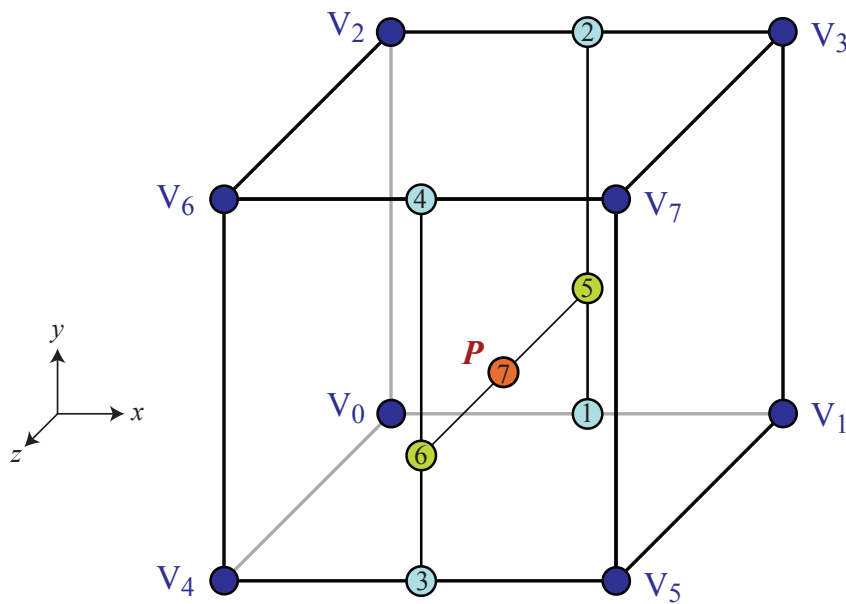


Figura 6.3: Decomposição de uma interpolação 3-D em sub-interpolações unidimensionais.

Para os quatro primeiros passos da interpolação tricúbica, as derivadas  $\frac{\partial f}{\partial x}$  são fornecidas diretamente pelos gradientes armazenados em cada *voxel*. Nos dois passos seguintes, as derivadas  $\frac{\partial f}{\partial y}$  precisam ser estimadas por interpolação cúbica, e para isso são necessárias derivadas parciais de segunda ordem  $\frac{\partial^2 f}{\partial x \partial y}$  em cada *voxel*. Finalmente, no último passo a derivada  $\frac{\partial f}{\partial z}$  precisa ser estimada por interpolação bi-cúbica, e para isso são necessárias as derivadas de segunda e terceira ordem  $\frac{\partial^2 f}{\partial x \partial z}$ ,  $\frac{\partial^2 f}{\partial y \partial z}$ ,  $\frac{\partial^3 f}{\partial x \partial y \partial z}$ .

Na interpolação pseudo-tricúbica, os quatro primeiros passos realizam interpolações cúbicas de Hermite, de forma idêntica à interpolação tricúbica.

Porém, nos passos seguintes, as derivadas são estimadas por interpolação linear, e não por interpolação cúbica. Ou seja, os valores de  $\frac{\partial f}{\partial y}$  são estimados por interpolação linear, enquanto  $\frac{\partial f}{\partial z}$  é estimado por interpolação bi-linear. Assim, as derivadas parciais de segunda e terceira ordem não são necessárias, e cada *voxel* pode armazenar apenas uma amostra de distância e um gradiente.

### 6.3 Resultados

A principal expectativa com as ADFs pseudo-tricúbicas era de que elas utilizassem menos células para representar modelos com curvas. De fato, este resultado se confirmou: em média, o número de células das ADFs pseudo-tricúbicas é 50% menor, como pode ser observado na Tabela 6.2.

Modelo	Nível Máximo	Erro Máximo	ADF Trilinear	ADF Pseudo-Tricúbica
Eros	8	$1e^{-3}$	40.299	16.162
Ramses	8	$7e^{-4}$	30.979	13.882

Tabela 6.2: Comparativo entre o número de células das ADFs trilineares e pseudo-tricúbicas. Neste exemplo, todas as ADFs são superficiais.

Por outro lado, atualmente cada *voxel* da ADF pseudo-tricúbica ocupa o dobro de memória de um *voxel* da ADF trilinear com normais suaves, o que faz com que ambas as ADFs tenham aproximadamente o mesmo tamanho.

No quesito desempenho, as ADFs pseudo-tricúbicas também se saem bem. Em média, durante a renderização, a taxa de quadros por segundo ficou apenas 30% abaixo das ADFs trilineares. Parte dessa diferença pode ser atribuída a uma implementação menos cuidadosa; e à reconstrução de normais, que nas ADFs trilineares é feita de forma barata, pela textura de normais.

A Figura 6.4 ilustra o salto de qualidade obtido para a representação de superfícies curvas. Com poucos níveis de subdivisão, uma esfera pode ser representada sem parecer facetada. Observe que estas imagens foram iluminadas sem normais suaves, apenas com o gradiente do campo de distância (calculado por diferenças centrais). No caso das ADFs trilineares, as discontinuidades são facilmente visíveis; mas, nas ADFs pseudo-tricúbicas, elas são bastante suaves, o que indica um aumento de qualidade significativo no campo de distância.

O mesmo ganho de qualidade pode ser observado na Figura 6.5. Desta vez, a ADF pseudo-tricúbica é comparada com uma ADF trilinear que utiliza normais suaves. É possível notar que a ADF pseudo-tricúbica representa melhor os pequenos detalhes do modelo. Além disso, mesmo não utilizando nenhuma técnica de suavização de normais, não é possível enxergar qualquer discontinuidade sem que se utilize muito *zoom*.

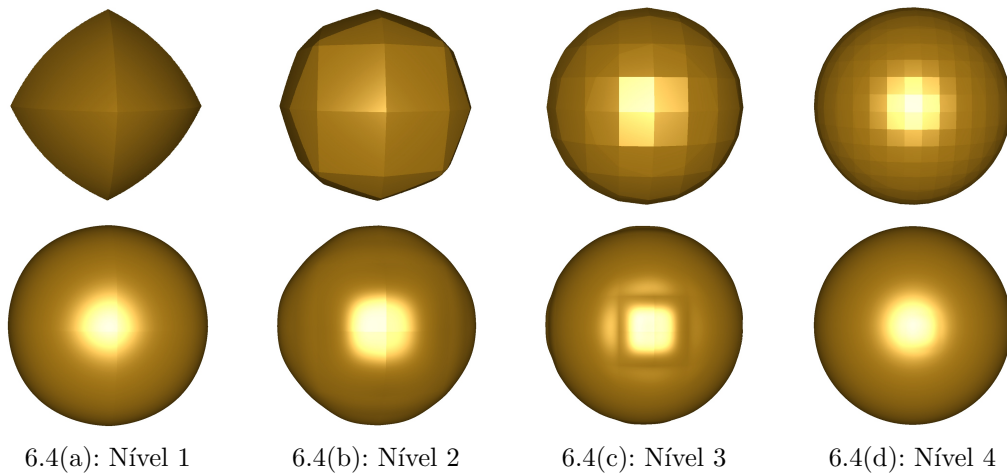
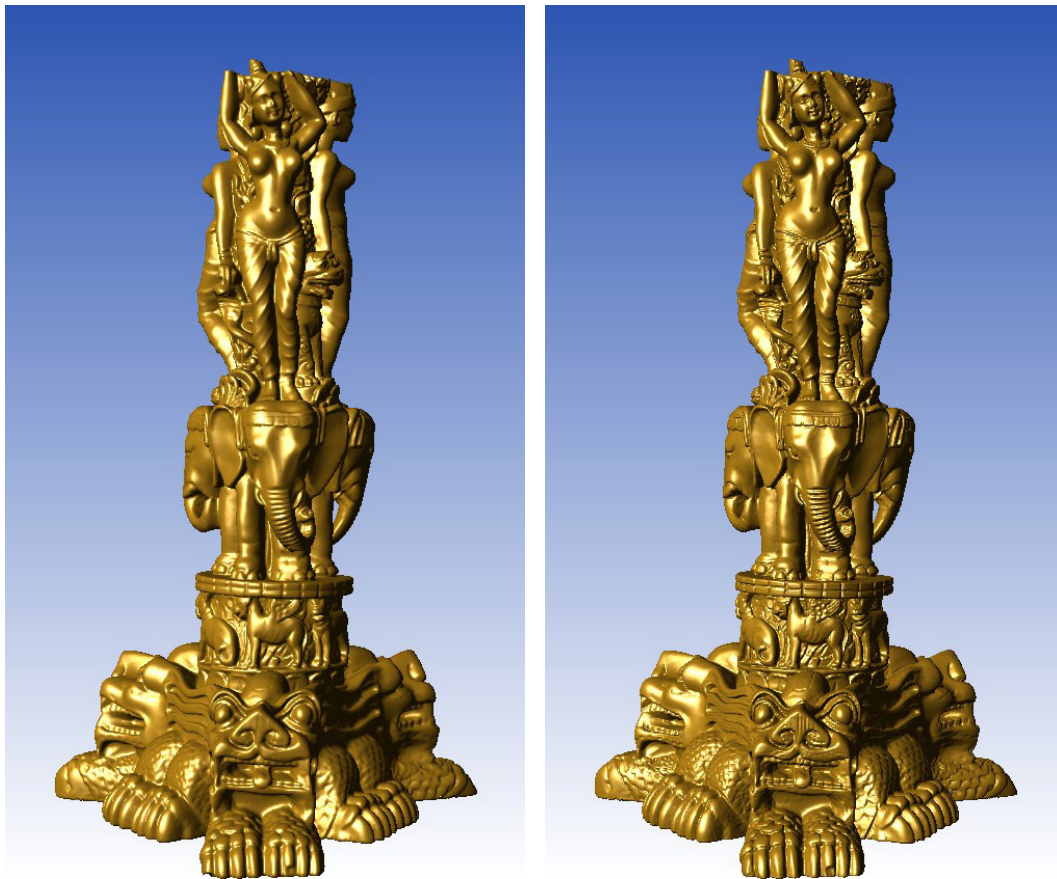


Figura 6.4: Comparativo entre ADFs trilineares (acima) e pseudo-tricúbicas (abaixo), utilizando a ADF de uma esfera em diferentes níveis de subdivisão. Neste exemplo, as ADFs trilineares foram renderizadas sem normais suaves.



6.5(a): ADF trilinear, com normais suaves.

6.5(b): ADF pseudo-tricúbica.

Figura 6.5: Comparativo entre duas ADFs de nível 9. A ADF pseudo-tricúbica consegue representar mais detalhes, e a iluminação é naturalmente mais suave.