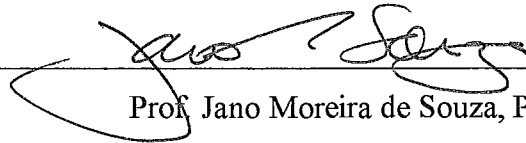


# JUNÇÕES DE POLILINHAS E POLÍGONOS COM USO DE FILTROS RASTER

Rodrigo Salvador Monteiro

TESE SUBMETIDA AO CORPO DOCENTE DA COORDENAÇÃO DOS PROGRAMAS DE PÓS-GRADUAÇÃO DE ENGENHARIA DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS EM ENGENHARIA DE SISTEMAS E COMPUTAÇÃO.

Aprovada por:



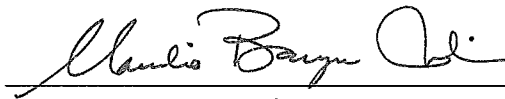
---

Prof. Jano Moreira de Souza, Ph.D.



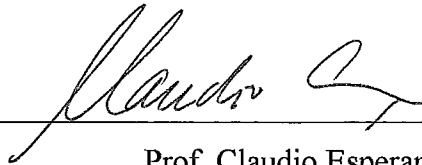
---

Prof. Geraldo Zimbrão da Silva, D.Sc.



---

Prof. Claudia Bauzer Medeiros, Ph.D.



---

Prof. Claudio Esperança, Ph.D.

RIO DE JANEIRO, RJ - BRASIL

AGOSTO DE 2001

MONTEIRO, RODRIGO SALVADOR

Junções de Polilinhas e Polígonos com Uso  
de Filtros Raster [Rio de Janeiro] 2001.

VI, 80 p. 29,7 cm (COPPE/UFRJ, M.Sc.,  
Engenharia de Sistemas e Computação, 2001)

Tese - Universidade Federal do Rio de  
Janeiro, COPPE

1. Banco de Dados Espaciais

I. COPPE/UFRJ II. Título ( série )

Resumo da Tese apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

## JUNÇÕES DE POLILINHAS E POLÍGONOS COM USO DE FILTROS RASTER

Rodrigo Salvador Monteiro

Agosto/2001

Orientadores: Jano Moreira de Souza

Geraldo Zimbrão da Silva

Programa: Engenharia de Sistemas e Computação

Este trabalho trata do processamento eficiente de junções espaciais. Os conceitos relacionados são apresentados assim como diversas estruturas capazes de desempenhar diferentes etapas do processo. Em especial, avaliamos o uso de filtros raster na junção de conjuntos de polígonos e polilinhas. As aproximações 4CRS e 5CDRS foram usadas, correspondendo a última a uma nova proposta parte de um trabalho conjunto com AZEVEDO (2001). Os resultados experimentais com conjuntos de dados reais foram extremamente favoráveis comprovando a eficiência da abordagem.

Abstract of Thesis presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

## POLYLINE AND POLYGON JOINS USING RASTER FILTERS

Rodrigo Salvador Monteiro

August/2001

Advisors: Jano Moreira de Souza

Geraldo Zimbrão da Silva

Department: Systems and Computing Engineering

This work addresses the efficient processing of spatial joins. The related concepts are presented as far as several structures to perform different steps of the process. Mainly, we evaluated the use of raster filters for polygon and polyline joins. The 4CRS and 5CDRS approximations were used, the last one corresponding to a new proposal which was conceived together with AZEVEDO (2001). The experimental results with real data sets were promising confirming the efficiency of the approach.

## Índice

<b>1</b>	<b><i>Introdução</i></b> .....	<b>1</b>
	1.1 Definindo o Problema .....	2
	1.2 Contribuições desta Tese .....	2
	1.3 Estrutura da Tese.....	3
<b>2</b>	<b><i>Processamento de consultas espaciais</i></b> .....	<b>4</b>
	2.1 Definição de conceitos básicos .....	5
	2.2 Junção espacial.....	8
	2.3 Arquitetura para processamento de junção espacial .....	9
	2.4 Métodos de acesso espacial .....	11
	2.4.1 Quadtree .....	11
	2.4.2 Árvore R .....	18
	2.5 Filtro geométrico.....	23
	2.6 Teste de geometria exata.....	24
<b>3</b>	<b><i>Filtros Geométricos</i></b> .....	<b>26</b>
	3.1 Tipos de Filtro Geométrico .....	26
	3.2 Vantagens da utilização de Filtros Raster .....	29
	3.3 Divisão do espaço em Células .....	30
<b>4</b>	<b><i>Aproximações Raster para polígonos e polilinhas</i></b> .....	<b>34</b>
	4.1 Assinatura Raster de Quatro Cores .....	34
	4.1.1 Geração da Assinatura.....	35
	4.1.2 Algoritmo de Comparação .....	36
	4.1.3 Mudança de Escala.....	38
	4.1.4 Compactação e outras variantes .....	39

4.1.5	Resultados Experimentais (4CRS) .....	40
4.2	Assinatura Raster Direcional de Cinco Cores.....	41
4.2.1	Geração da Assinatura.....	41
4.2.2	Algoritmo de Comparação .....	47
4.2.3	Mudança de Escala.....	49
4.2.4	Variantes.....	51
4.2.5	Resultados Experimentais (5CDRS) .....	52
4.3	Algoritmo de Comparação entre as Aproximações 4CRS e 5CDRS .....	53
<b>5</b>	<b><i>Resultados Experimentais</i></b> .....	<b>55</b>
5.1	Ambiente de teste.....	55
5.2	Dados de teste .....	56
5.3	Aproximações e Árvores.....	57
5.4	Resultados da execução de junções .....	59
<b>6</b>	<b><i>Conclusões</i></b> .....	<b>66</b>
6.1	Contribuições da tese .....	66
6.2	Trabalho futuros.....	66
<b>7</b>	<b><i>Bibliografia</i></b> .....	<b>68</b>
	<b><i>Anexo A</i></b> .....	<b>73</b>
	<b><i>Anexo B</i></b> .....	<b>77</b>
	<b><i>Anexo C</i></b> .....	<b>80</b>

# 1 Introdução

Apesar de não se tratar de uma área nova de estudo, banco de dados espaciais ainda carece de esforços, não somente em suas diversas vertentes mas também em operações básicas para o processamento de consultas. Tal fato pode ser comprovado pelo grande número e variedade de trabalhos e pesquisas realizados na área nos últimos anos.

Novas aplicações surgem necessitando de dados representados de forma cada vez mais confiável. Como exemplo podemos citar a agricultura de precisão e as rodovias inteligentes (GORDON et al., 1994). No primeiro caso, a informação precisa sobre a distribuição de nutrientes no solo é fundamental, e no segundo o posicionamento e deslocamento devem ser extremamente realistas.

Com o aumento na complexidade dos dados, torna-se de fundamental importância o processamento eficiente de consultas espaciais. Este merece ainda mais atenção se lembrarmos a natureza interativa de algumas aplicações, como por exemplo, o processo de tomada de decisão apoiado por um sistema de informações geográficas.

Existindo o consenso de que um banco de dados espacial deve oferecer para os dados espaciais as mesmas funcionalidades que um banco de dados relacional oferece para os dados convencionais, um banco de dados espacial deve fornecer suporte para consultas *ad-hoc* sobre os atributos espaciais armazenados. Assim, como nos banco de dados relacionais, tais consultas devem ser decompostas em consultas menores, mais simples, que possam ser implementadas através de um conjunto pequeno de operadores espaciais. Nesse cenário, um algoritmo eficiente para a realização de junções espaciais é de fundamental importância para a avaliação efetiva de consultas.

Uma questão ortogonal à complexidade dos dados diz respeito à forma de representação do objeto do mundo real. Dependendo da sua natureza, este pode ser constituído por polígonos, linhas ou pontos. Os objetos representados por polígonos são de longe os mais estudados devido ao seu amplo uso e complexidade. Os objetos representados por pontos podem muitas vezes utilizar estruturas análogas às existente para os dados escalares, e por este motivo são pouco estudados. Já os objetos representados por linhas ou polilinhas utilizam muitas vezes soluções desenvolvidas para polígonos, mesmo

quando estas não são adequadas, o que é extremamente contraditório quando estamos falando de execução eficiente de consultas.

## 1.1 Definindo o Problema

Conforme apontado por BRINKHOFF et al. (1994), entende-se a junção espacial como um subconjunto do produto cartesiano de dois conjuntos, A e B, não necessariamente distintos, contendo respectivamente  $m$  e  $n$  elementos. Esse subconjunto é formado pelos elementos do produto cartesiano que atendam a um determinado predicado espacial. De especial interesse para aplicações práticas é a sobreposição de objetos espaciais.

BRINKHOFF et al. (1994) apresentam um processador de junções espaciais em múltiplos passos visando uma melhor organização e compatibilidade dos trabalhos desenvolvidos na área. Em particular, o segundo passo deste processador é realizado com o uso de filtros geométricos e aproximações.

As aproximações existentes, voltadas para polígonos, não representam polilinhas de forma adequada. Com o intuito de preencher esta lacuna, foi desenvolvida, em conjunto com AZEVEDO (2001), uma aproximação raster para polilinhas. Dentre os tipos de filtros geométricos optamos por um filtro raster pela sua capacidade de ser conclusivo tanto na presença quanto na ausência de interseção.

## 1.2 Contribuições desta Tese

Um novo filtro geométrico para polilinhas, o 5CDRS, desenvolvido em conjunto com AZEVEDO (2001) é proposto. Em AZEVEDO (2001) maiores detalhes sobre o seu desenvolvimento e resultados experimentais de junções de conjuntos de polilinhas podem ser encontrados.

O presente trabalho tem por objetivo apresentar e avaliar a eficiência de filtros raster para a junção de polígonos e polilinhas. As aplicações deste tipo de consulta espacial são as mais diversas, indo desde estudos ambientais envolvendo áreas florestais e zonas de desmatamento ou “espinhas de peixe” (representada por polilinhas) até planejamento e



controle de ampliação de redes elétricas, esgoto, dutos, e outros em relação a áreas de demanda.

Finalmente, este trabalho também destina-se a ser usado como um guia para o processamento de junções espaciais, uma vez que apresenta todo o processo citando e exemplificando estruturas a serem usadas em cada uma das etapas.

### **1.3 Estrutura da Tese**

Este trabalho está organizado da seguinte forma. O capítulo 2 apresenta os principais conceitos relacionados ao processamento de junções espaciais. O capítulo 3 é dedicado aos filtros geométricos. No capítulo 4 as aproximações raster 4CRS e 5CDRS são apresentadas. O capítulo 5 destina-se aos resultados experimentais de junções de conjuntos de polilinhas e polígonos. Finalmente, o capítulo 6 apresenta as conclusões, descrevendo as principais contribuições do presente trabalho assim como as direções futuras.

## 2 Processamento de consultas espaciais

Este capítulo destina-se a apresentar conceitos e estruturas relacionadas ao processamento de consultas espaciais. Para tanto uma revisão da literatura é apresentada enfatizando as principais estruturas e métodos referenciados por trabalhos da área. O ambiente e o escopo da tese são aqui elucidados. É importante ressaltar que o presente capítulo faz parte de um trabalho conjunto com AZEVEDO (2001) constituindo também um capítulo do mesmo.

Caso o leitor já tenha conhecimento sobre processamento de consultas em um banco de dados espaciais e da arquitetura de múltiplos passos (BRINKHOFF et al., 1994), poderá se dirigir ao capítulo 3. Por outro lado, para a leitura do presente capítulo é pressuposto o conhecimento dos conceitos relacionados ao processamento de consultas em banco de dados tradicionais.

Um banco de dados espacial deve oferecer as mesmas funcionalidades para os dados espaciais que um banco de dados relacional oferece aos dados convencionais. Logo, um banco de dados espacial deve ser capaz de responder consultas *ad-hoc* sobre os atributos espaciais armazenados. Consultas devem ser decompostas em consultas menores, mais simples, que possam ser implementadas através de um conjunto pequeno de operadores espaciais.

Considere, como exemplo, um banco de dados que possua uma relação de todos os estados dos Estados Unidos e outra com todos os rios deste país (figura 2.1). Poderíamos executar uma consulta neste banco solicitando uma relação por estado de todos os rios que cruzam o seu território. Poderíamos também querer saber os rios que passam por pelo menos três estados diferentes, a indicação do estado interceptado pelo maior número de rios, além de diversas outras possibilidades.

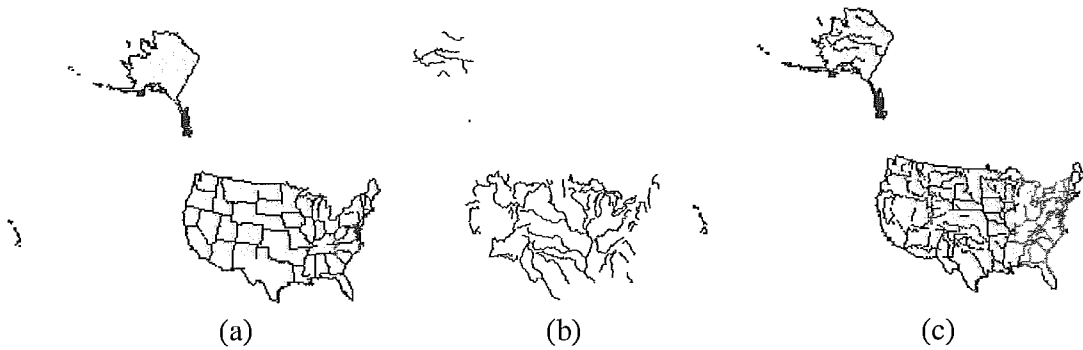


Figura 2.1 – Mapa político (a) e hidrográfico (b) dos Estados Unidos e sua sobreposição (c).

As consultas apresentadas acima combinam objetos espaciais das relações de acordo com seus atributos geométricos caracterizando consultas espaciais.

Este capítulo está estruturado da seguinte forma: na seção 2.1 são definidos conceitos relevantes para o melhor entendimento do presente trabalho; a seção 2.2 apresenta com maiores detalhes o conceito de junção espacial; na seção 2.3 é apresentada uma arquitetura modular para o processamento de consultas espaciais. As seções seguintes enfatizam cada passo desta arquitetura. Na seção 2.4 serão estudados alguns métodos de acesso espacial, destacando *Quadtree* e *Árvore R*. A seção 2.5 apresenta filtros geométricos; e, finalmente, a seção 2.6 o teste de geometria exata.

## 2.1 Definição de conceitos básicos

Esta seção tem por finalidade definir formalmente alguns termos e conceitos relevantes para o entendimento do presente trabalho.

- **Dado espacial** é um termo usado para descrever dados que contém informações sobre o espaço geométrico ocupado pelos objetos numa base de dados. Estes dados são diversificados. Consistem de pontos, linhas, retângulos, polígonos, superfícies, volumes e até mesmo dados de dimensões superiores. Dados espaciais são normalmente acompanhados por atributos ou dados não espaciais, como por exemplo, o nome de um rio ou tipo de solo de uma região.

- **Banco de dados espaciais** é um sistema de banco de dados que oferece suporte a tipos de dados espaciais tanto em seu modelo de dados como em sua linguagem de consulta, implementando tais tipos e oferecendo pelo menos índices espaciais e algoritmos eficientes para operações espaciais. (ZIMBRÃO, 1999)
- **Predicado espacial** é uma operação de restrição sobre os atributos espaciais de um conjunto a fim de extrair os elementos que satisfaçam a uma condição.
- **Imagem Raster** é um reticulado regular de células retangulares ordenadas por linha, o que torna desnecessário o armazenamento das coordenadas de cada célula. Cada célula de uma estrutura raster possui um e apenas um valor em uma camada temática. (ZIMBRÃO, 1999)
- **Retângulo envolvente mínimo** ou **MBR** (*Minimum Bounding Rectangle*) refere-se ao menor retângulo alinhado com os eixos cartesianos que contém inteiramente o objeto representado. Também comumente referenciado como caixa envolvente e caixa limitante constitui uma aproximação simples e por tal motivo amplamente usada na literatura. A figura 2.2 apresenta o mapa do Rio de Janeiro e o seu MBR.

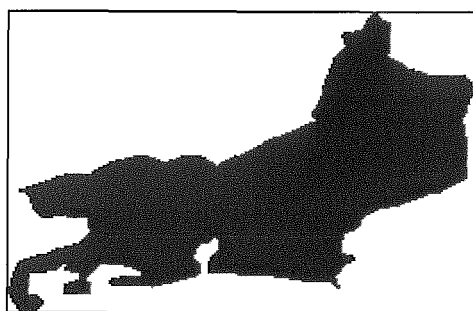


Figura 2.2 – Retângulo envolvente mínimo (MBR)

- Uma **polilinha** pode ser definida como um conjunto conexo de linhas poligonais, sendo estas últimas compostas por segmentos consecutivos. Cada linha poligonal é definida como uma sequência de pontos onde cada ponto define com seu antecessor um segmento. Cada linha poligonal é chamada de parte. Portanto, uma polilinha é descrita através de suas partes e dos pontos que as definem. A figura 2.3 apresenta a polilinha correspondente ao Rio Amazonas e seus afluentes.

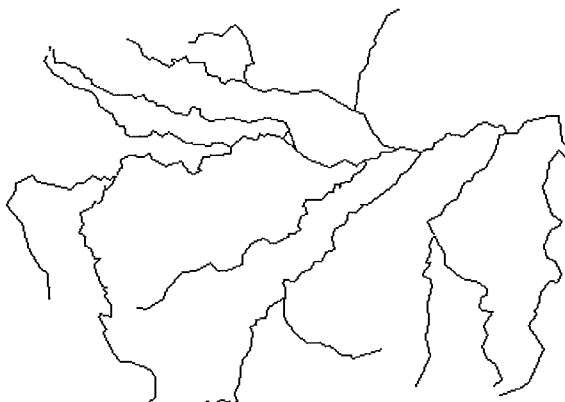


Figura 2.3 – Exemplo de polilinha

- Entende-se por **regiões ou imagens binárias** aquelas cujos elementos básicos podem assumir apenas dois valores distintos, como, por exemplo, preto ou branco. Uma imagem binária de uma região pode ser obtida considerando os pontos internos como pretos ou 1 e os externos como brancos ou 0, como exemplificado na figura 2.4.

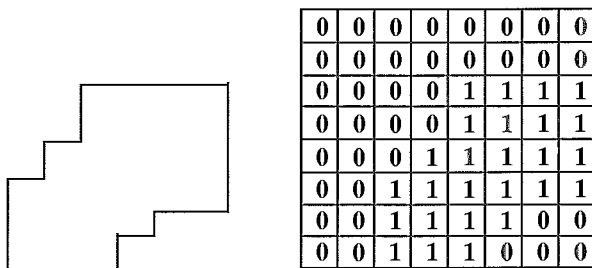


Figura 2.4 – Objeto e sua imagem binária

## 2.2 Junção espacial

Conforme apontado por BRINKHOFF et al. (1994), entende-se a junção espacial como um subconjunto do produto cartesiano de dois conjuntos, A e B, não necessariamente distintos, contendo respectivamente  $m$  e  $n$  elementos. Esse subconjunto é formado pelos elementos do produto cartesiano que atendam a um determinado predicado espacial.

Junções espaciais podem ser mais eficientemente avaliadas através da construção de índices sobre os conjuntos de dados, de forma que simultaneamente se percorra estes índices buscando por pares de objetos espaciais que se interceptam. Esta abordagem corresponderia à abordagem *sort-merge* dos bancos de dados relacionais, e é apontada por BRINKHOFF et al. (1993a) como sendo bastante eficiente, especialmente no caso de preexistência dos índices na base. Sem a utilização de índices no processamento da consulta teremos o tempo de processamento crescendo exponencialmente com o número de objetos envolvidos, já que para cada elemento de um conjunto todos os elementos do outro deverão ser testados. Além disso, o tamanho dos objetos muitas vezes pode impedir que todos os elementos necessários para a comparação sejam trazidos para a memória, implicando em grandes volumes de dados transferidos do disco para a memória e vice-versa (*swap*).

Tipicamente, um índice é composto de duas partes: a estrutura de índice que armazena apenas chaves para os dados, e a estrutura que armazena os dados em si. Desta forma, um índice espacial deve armazenar os objetos espaciais em uma estrutura espacial de acordo com uma chave geométrica (BRINKHOFF et al., 1994).

Devido à sua simplicidade, o MBR (*minimum bounding rectangle* – menor retângulo envolvente) é a chave geométrica de uso mais difundido. Quando utilizamos MBRs, a complexidade de um objeto espacial é reduzida a quatro parâmetros que detêm as mais importantes características do objeto: sua posição e sua extensão. Apesar disso, conforme apontado por BRINKHOFF et al. (1994) e ZIMBRÃO et al. (1998), os objetos do mundo real são muito mal aproximados por MBRs – em particular, polilinhas são pessimamente aproximadas por MBRs.

Nesse cenário, é de fundamental importância para a avaliação efetiva de consultas um algoritmo eficiente para a realização de junções espaciais. BRINKHOFF et al. (1994)

definem um processador modular de junções espaciais. Tal modelo tem sido muito citado em trabalhos subsequentes, e por isso o trabalho aqui apresentado foi desenvolvido de forma a ser compatível com tal processador.

## 2.3 Arquitetura para processamento de junção espacial

Visando uma melhor eficiência, organização e estudo do processo de desenvolvimento de índices e filtros para dados espaciais, BRINKHOFF et al. (1994) propuseram uma arquitetura de múltiplos passos ou MSQP (*Multi-Step Query Processor*) para o processamento de junções espaciais (figura 2.5), cujos passos serão detalhados mais à frente. O principal objetivo desta arquitetura é acelerar os passos mais custosos através da redução do número de objetos espaciais a serem comparados. Tal redução é feita utilizando filtros em passos anteriores. Neste trabalho são propostas e avaliadas aproximações a serem empregadas como filtros geométricos no segundo passo.

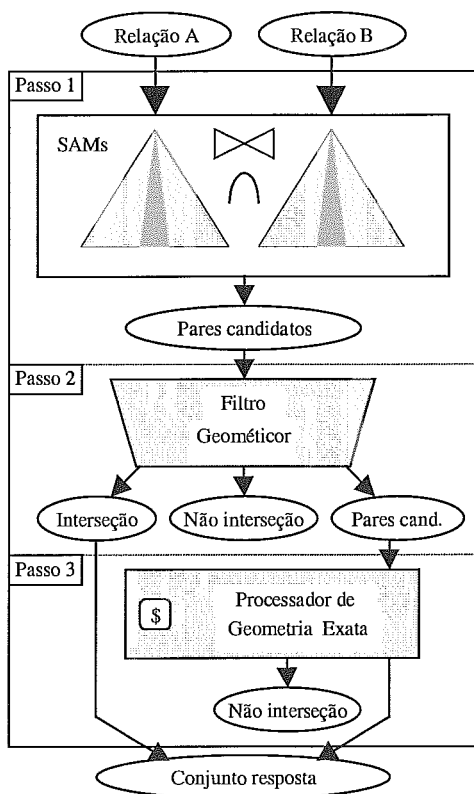


Figura 2.5 - Processador de consulta em múltiplos passos (MSQP).

A principal contribuição da arquitetura apresentada na figura 2.5 é estabelecer uma modularização do processamento de junção espacial. O único ponto em comum entre módulos diferentes é o conjunto de objetos transferido de um para o outro, não importando a forma como este é computado. Este fato permite que diferentes algoritmos e estruturas de dados possam ser propostas de forma independente para cada etapa do processo.

O primeiro passo dessa arquitetura corresponde a um método de acesso espacial ou SAM (*Spatial Access Method*) que tem como objetivo restringir o espaço de busca. Os métodos de acesso tradicionalmente empregados usam MBRs dos objetos e por este motivo este passo não tem como saída o resultado da operação de junção, mas sim um conjunto de pares candidatos que corresponde a um super-conjunto da solução. Esse passo é conhecido como junção de MBRs, e cabe ressaltar que qualquer que seja o método utilizado para implementá-lo o produto final deverá ser o mesmo conjunto, pois é composto pelos MBRs que se interceptam.

O segundo passo consiste em comparar os pares de candidatos resultantes do passo anterior através de um filtro geométrico. Um filtro geométrico usa uma representação aproximada e compacta do objeto que procura guardar as principais características do mesmo. Alguns exemplos são apresentados no capítulo 3. Como resultado da comparação dos filtros dos dois objetos candidatos temos três possibilidades: os dois objetos pertencem à solução (*hit*); os dois objetos não pertencem à solução (*false hit*); e inconclusivo.

O terceiro passo tem como entrada os pares de objetos que tiveram a comparação de seus filtros geométricos inconclusiva. Este passo consiste em compará-los através da sua real representação. Este é o mais custoso requerendo tempo de CPU para computar a interseção exata e tempo de I/O para ler os objetos espaciais do disco, conforme resultados obtidos por BRINKHOFF et al. (1994). A quantidade de tempo gasta neste passo pode ser consideravelmente reduzida com melhores aproximações nos passos anteriores.

Assim, invariavelmente todos os algoritmos de junção espacial que têm sido propostos nos últimos anos realizam uma pré-junção sobre os MBRs dos polígonos, e somente então passam a investigar quais os pares desse conjunto resposta que farão parte do resultado da junção original. Muitos desses trabalhos param nesse passo, não chegando a realizar os passos 2 e 3 do MSQP: realizam apenas a junção de MBRs. Por outro lado,



alguns métodos não implementam o passo 2, passando o conjunto produzido pela junção de MBRs diretamente para o passo 3.

## 2.4 Métodos de acesso espacial

Esta seção apresenta métodos de acesso espacial candidatos a desempenhar o primeiro passo do MSQP. Duas classes de métodos recebem atenção especial devido a seu amplo uso na literatura: a Quadtree, apresentada na seção 2.4.1; e, a Árvore R, apresentada na seção 2.4.2. No final da seção, outros métodos são referenciados.

### 2.4.1 Quadtree

Segundo SAMET (1990) o termo quadtree é usado para descrever uma classe de estruturas de dados hierárquica cuja propriedade comum é o princípio da decomposição recursiva do espaço. Estas estruturas, propostas pelo mesmo autor, podem se diferenciar nas seguintes características:

- Tipo de dado a ser representado
- Critério usado para o processo de decomposição
- A resolução (variável ou não)

As Quadtrees são usadas para os mais diversos tipos de dados como pontos, áreas, curvas, superfícies e volumes. A decomposição pode ser regular (divisão em partes iguais a cada nível) ou determinada por características do novo dado. Como exemplos temos a Quadtree de região, para o primeiro caso, e Quadtree de pontos, para o segundo. Detalhes destas duas estruturas serão apresentados mais à frente. Na área de computação gráfica, esta diferenciação de decomposição do espaço é referenciada respectivamente como hierarquias no espaço da imagem e hierarquias no espaço do objeto (SUTHERLAND et al., 1974). O último item, resolução, refere-se ao número de vezes que o processo de decomposição é aplicado, podendo ser pré-determinada ou obtida em função de características do novo

Um dado. Uma quarta característica relevante para algumas aplicações pode ser acrescentada referindo-se a representação das fronteiras do objeto (curvas e superfícies) ou organização de seu interior (áreas e volumes).

Para exemplificar algumas das características apresentadas acima e melhor entendimento do propósito da Quadtree são apresentadas a seguir duas estruturas dessa classe: Quadtree de região e Quadtree de pontos.

A Quadtree de região destina-se à representação de regiões binárias bidimensionais. Uma vez definidos os limites da região com base em sua borda esta estrutura baseia-se na subdivisão sucessiva do espaço em quatro quadrantes de mesmo tamanho. O critério usado para decomposição do espaço consiste em efetuar uma nova subdivisão em um quadrante enquanto este não for homogêneo, ou seja, possuir apenas elementos iguais. Esta Quadtree pode ser classificada como uma estrutura de resolução variável.

Como exemplo de uma Quadtree de região considere a região apresentada na figura 2.6.a e a sua imagem binária  $2^3 \times 2^3$  na figura 2.6.b, onde 1 representa elementos dentro da região e 0 elementos fora da região. Os quadrantes resultantes do processo de decomposição são apresentados na figura 2.6.c e a Quadtree correspondentes na figura 2.6.d.

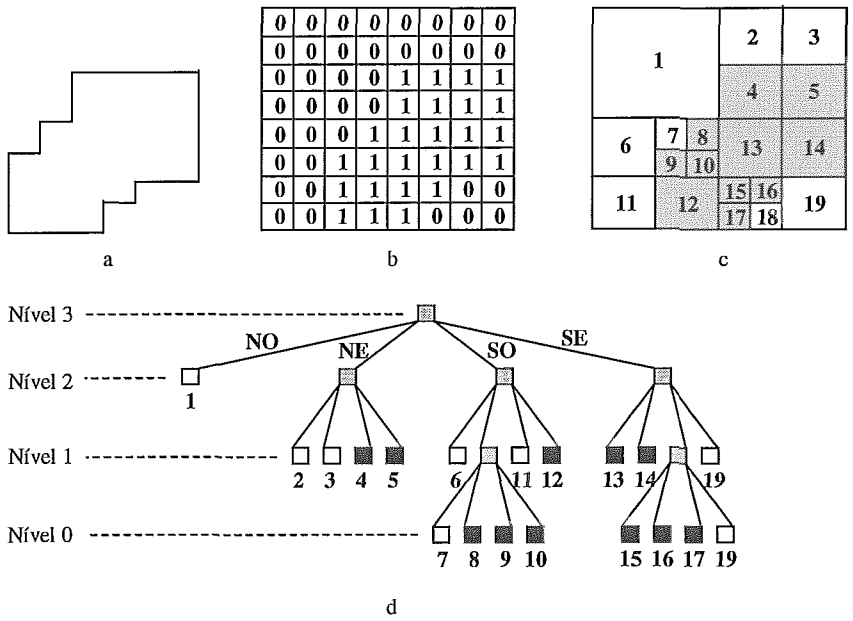


Figura 2.6 – Quadtree de região

Na representação da figura 2.6.d, o nó raiz corresponde a toda a imagem binária da figura 2.6.b. Cada um dos quatro filhos representa um quadrante resultante da subdivisão da região representada pelo nó, rotulados com NO, NE, SO e SE. Os nós folhas da árvore correspondem a regiões homogêneas onde, portanto, a subdivisão não foi necessária. Um nó folha é rotulado como preto ou branco dependendo se este representa uma porção dentro ou fora da região representada. Os nós internos são rotulados como cinza. O nó raiz se encontra no nível N enquanto o nível 0 corresponde a um único elemento da imagem.

A Quadtree de pontos destina-se à representação de um conjunto de pontos num espaço bidimensional. A principal diferença em relação a Quadtree de região fica por conta da decomposição do espaço. Ao invés de regular, a decomposição ocorre em função da entrada, permitindo a formação de quadrantes retangulares. Esta estrutura é tida como uma adaptação da árvore binária de busca (KNUTH, 1973), onde cada ponto, com coordenadas  $X_p$  e  $Y_p$ , divide o seu quadrante em quatro novos quadrantes da seguinte forma:

- Quadrante NO:  $X < X_p$  e  $Y > Y_p$
- Quadrante NE:  $X > X_p$  e  $Y > Y_p$
- Quadrante SO:  $X < X_p$  e  $Y < Y_p$
- Quadrante SE:  $X > X_p$  e  $Y < Y_p$

A inserção de um novo ponto consiste em percorrer a árvore a partir da raiz até alcançar uma folha, a qual corresponde ao quadrante sem subdivisão que contém ponto, onde este será inserido. A descida na árvore é efetuada comparando, segundo as regras acima, as coordenadas do novo ponto com as do ponto do ramo sendo percorrido. Um novo ponto adicionado à estrutura sempre leva a criação de quatro novos quadrantes.

Como exemplo de uma Quadtree de pontos considere a figura 2.7, onde os pontos foram inseridos na seguinte ordem: Teresópolis (1), Niterói (2), Rio de Janeiro (3), Friburgo (4), Búzios (5), Cabo Frio (6) e Petrópolis (7).

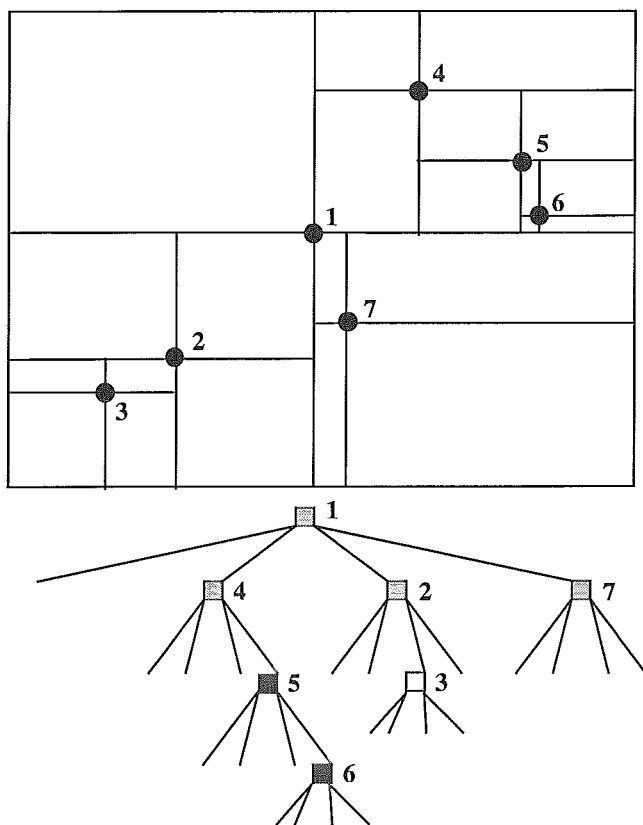


Figura 2.7 - Quadtree de pontos

O formato da árvore é altamente dependente da ordem de inserção dos pontos assim como ocorre na árvore binária de busca.

Devido ao seu amplo e difundido uso ao longo dos anos<sup>1</sup>, estruturas pertencentes à classe das Quadtrees surgiram cada qual com adaptações visando solucionar problemas específicos. Como exemplos de estruturas candidatas para armazenar os MBRs no primeiro passo do MSQP temos MX-CIF Quadtree<sup>2</sup>, MX-CIF Quadtree expandida e RR Quadtree as quais são apresentadas a seguir.

A MX-CIF Quadtree, pertencente à família das CIF Quadtrees, foi criada independentemente por KEDEM (1982) e por ABEL et al. (1983), associa cada retângulo

---

<sup>1</sup> uma Quadtree já era usada como método de acesso espacial por MORTON (1966).

<sup>2</sup> MX-CIF abreviação em inglês de Matrix – Caltech Intermediate Form.

com um nó da Quadtree correspondendo ao menor bloco que o contém inteiramente. Desta forma, podem existir retângulos associados a nós internos. O processo de decomposição pára quando um bloco não contém inteiramente nenhum retângulo. Um critério alternativo consiste em estabelecer um tamanho mínimo, de forma que o bloco só poderá ser subdividido se seu tamanho for maior que o limite. KEDEM (1982) sugere que o limite seja igual ao tamanho esperado dos retângulos.

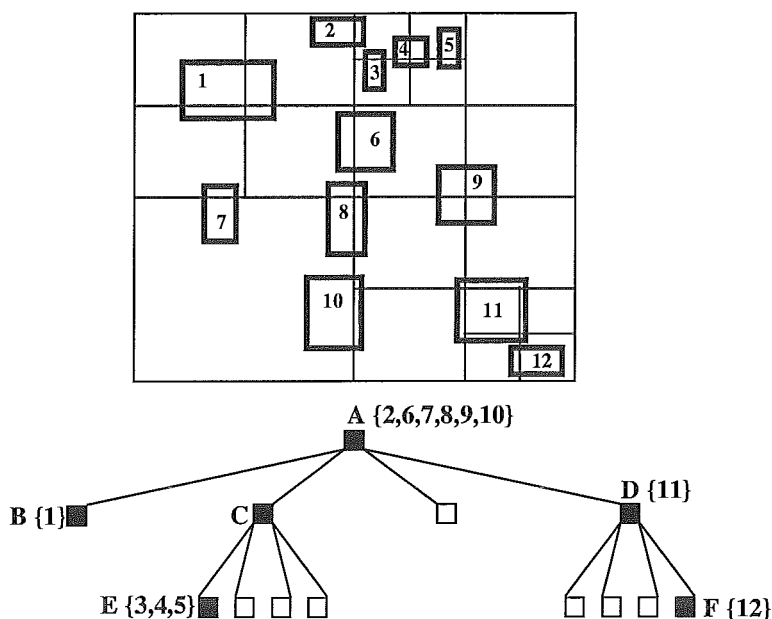


Figura 2.8 - MX-CIF Quadtree

A figura 2.8 apresenta um exemplo da MX-CIF Quadtree. Note que um retângulo é associado a apenas um nó e não é considerado membro dos filhos deste. Como exemplo o retângulo 11 possui interseção com os nós D e F porém só está associado ao nó D. Por outro lado, mais de um retângulo pode ser associado a um nó, existindo diversas formas de organizá-los. Esta organização pode variar desde nenhuma ordenação (ABEL et al., 1983) até formas mais elaboradas, como uma abordagem usando uma árvore binária para cada eixo descrita por KEDEM (1982).

A MX-CIF Quadtree associa a cada retângulo apenas um nó da árvore. Outra abordagem possível é a representação através de múltiplos blocos ou nós. A MX-CIF Quadtree expandida e RR Quadtree se enquadram nessa categoria.

A MX-CIF Quadtree expandida procura representar cada retângulo de uma forma mais compacta no que diz respeito à área. Proposta por ABEL (1985), esta representação segue os mesmos passos de uma MX-CIF Quadtree diferenciado-se quando o menor bloco contendo o retângulo inteiramente é encontrado. Neste ponto um número pré-fixado de subdivisões é efetuado no bloco. O retângulo é representado pelo conjunto de blocos sem subdivisão da Quadtree que contém alguma porção do objeto. Em outras palavras, cada retângulo é representado pelo conjunto de nós folhas que contém alguma porção do objeto. Note que na MX-CIF Quadtree um retângulo pode ser associado a um nó interno. Um exemplo de MX-CIF Quadtree expandida é apresentado na figura 2.9 juntamente com a MX-CIF Quadtree para o mesmo conjunto de retângulos. Nesse exemplo o número pré-fixado de subdivisões é igual a 1, ou seja, uma nova subdivisão é efetuada no bloco contendo o retângulo inteiramente.

A RR Quadtree criada por SHAFFER (1986) possui duas variantes denominadas  $RR_1$  e  $RR_2$  Quadtree. O que as diferencia é o critério de decomposição. Na  $RR_1$  Quadtree um bloco é subdividido até que contenha apenas um retângulo ou todos os retângulos possuam interseção entre si. Na  $RR_2$  Quadtree um bloco é subdividido até que contenha apenas um retângulo ou todos os retângulos façam parte de uma cadeia de interseção. A figura 2.10 apresenta exemplos dessas representações.

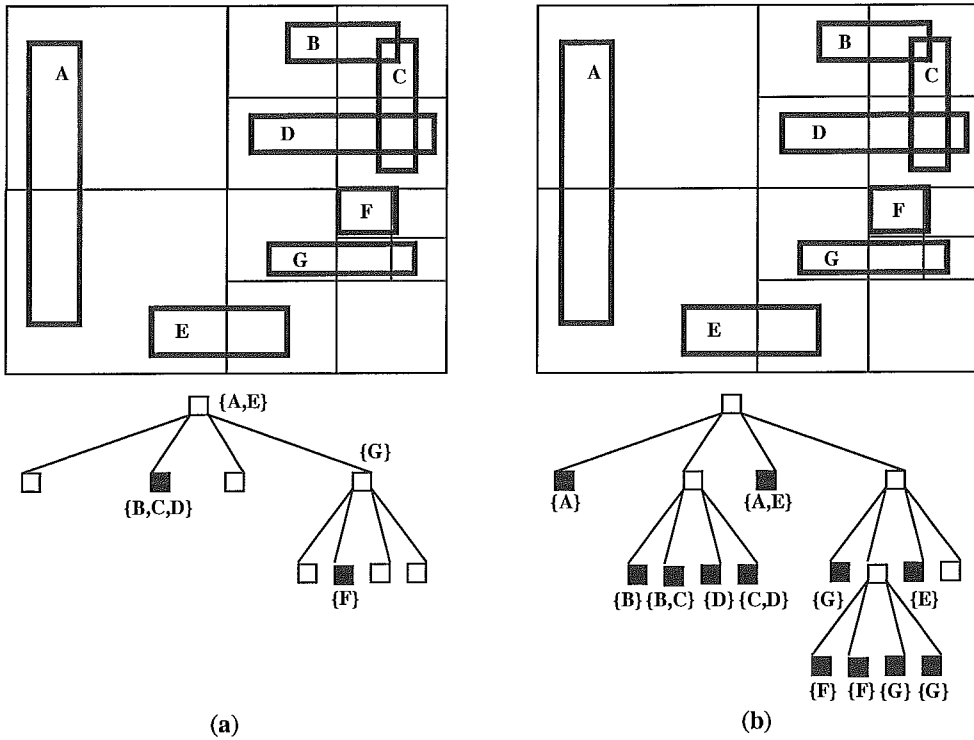


Figura 2.9 - (a) MX-CIF quadtree (b) MX-CIF quadtree expandida

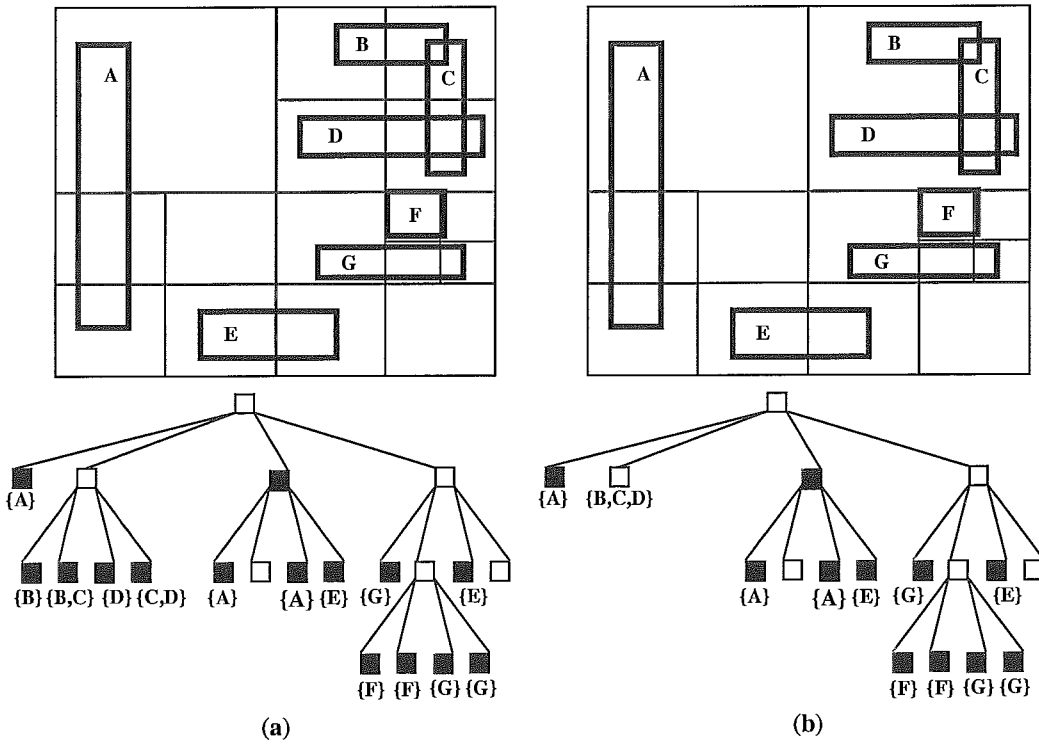


Figura 2.10 - (a) RR<sub>1</sub> quadtree (b) RR<sub>2</sub> quadtree

## 2.4.2 Árvore R

A Árvore R (GUTTMAN, 1984) é uma estrutura de dados hierárquica balanceada, nos moldes das árvores B, sendo amplamente utilizada na representação de coleções de dados geométricos através de retângulos em  $n$  dimensões. As regras básicas para a formação de uma árvore R são semelhantes às da árvore B como se segue:

- Todos os nós folha aparecem no mesmo nível;
- Um nó folha contém coleções de pares  $(R,O)$  na qual  $R$  é o menor retângulo que envolve o objeto apontado por  $O$ ;
- Um nó interno contém coleções de pares  $(R,P)$  no qual  $R$  é o retângulo envolvente mínimo dos objetos que estão no nó filho apontado por  $P$ .
- Diz-se que uma árvore R tem ordem  $(m,M)$  quando cada nó contém entre  $m$  e  $M$  entradas, onde  $m \leq \lceil M/2 \rceil$ , excetuando-se a raiz que tem pelo menos duas entradas a não ser que esta seja nó folha.

Freqüentemente nós correspondem a páginas do disco, e os parâmetros que definem a árvore são determinados de forma que o número de nós visitados durante a busca seja o menor possível. Uma árvore R não é uma divisão do espaço e sim um agrupamento de objetos o que permite que diferentes nós se sobreponham. Assim, um mesmo retângulo pode estar associado a um ou a vários nós o que leva a possibilidade da busca requerer a visita a vários nós antes de averiguar a presença ou ausência de um retângulo específico.

Como exemplo, considere os objetos e seus respectivos retângulos envolventes mínimos da figura 2.11. Para  $M = 3$  e  $m = 1$  temos uma possível árvore R para a coleção. Nós folha são representados com letras indicando os retângulos envolventes mínimos dos objetos que eles representam, em tonalidade mais escura. Linhas pontilhadas e números assinalam nós internos. Observe que a árvore R não é única; sua estrutura é dependente da ordem em que os nós foram inseridos e possivelmente removidos da árvore.



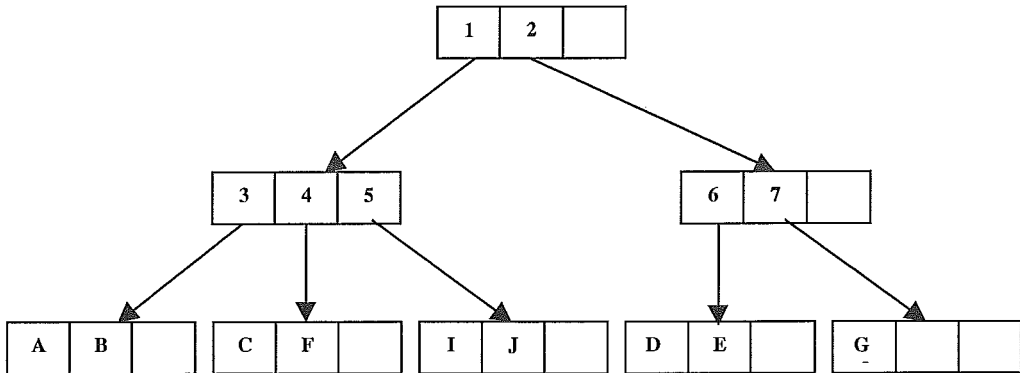
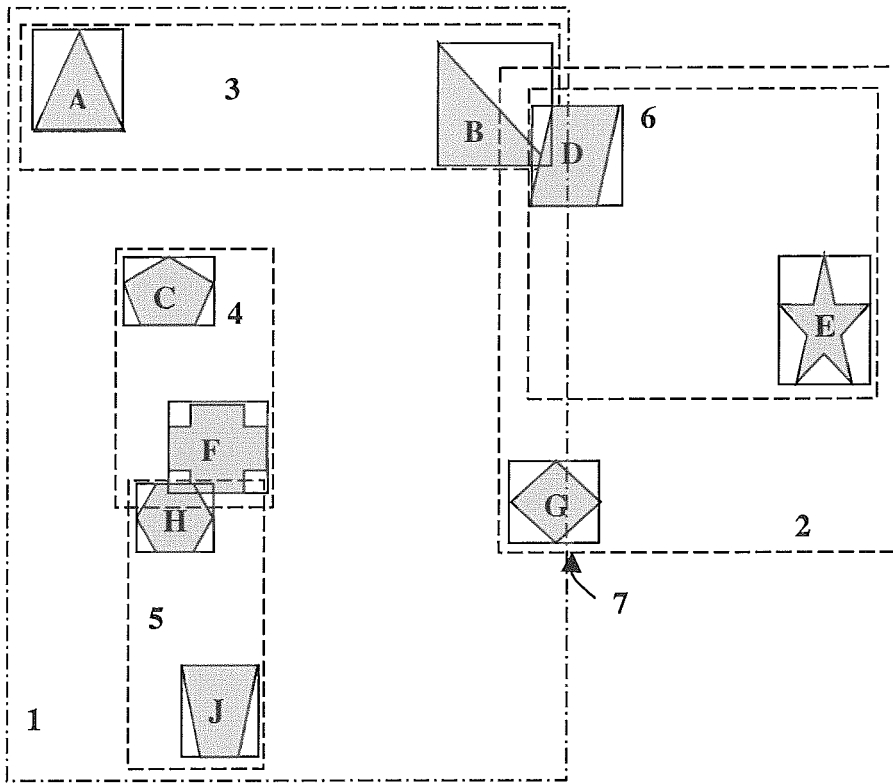


Figura 2.11 – Árvore R

A inserção de um novo objeto é semelhante a das árvores B. Novos retângulos são adicionados em nós folha. A determinação da folha apropriada é feita descendo na árvore a partir da raiz e a cada passo escolhendo a sub-árvore cujo aumento de área do retângulo correspondente é mínimo. Uma vez determinado o nó folha, verifica-se se a inserção causará *overflow*. Se isto ocorrer, o nó é então dividido em dois, e as  $M+1$  entradas são redistribuídas. A divisão é propagada para cima atualizando-se os MBRs das entradas do

caminho da folha até a raiz. A árvore  $R$  utiliza como heurística para divisão de nós obter dois novos nós que possuam a menor área desperdiçada<sup>1</sup> possível.

A remoção de um retângulo  $R$  de uma árvore  $R$  ocorre através da localização do nó folha  $L$  contendo  $R$  e removendo  $R$  de  $L$ . Em seguida, é feito o ajuste dos retângulos das entradas dos nós no caminho de  $L$  até a raiz. Os nós que sofrem *underflow* são removidos e adicionados a um conjunto auxiliar, digamos  $U$ . Se a raiz sofre *underflow* e tem um filho, seu filho torna-se a nova raiz. As entradas do conjunto  $U$  são reinseridas em suas alturas originais.

A remoção pode também ser realizada analogamente à árvore  $B$ . Nós com *underflow* são redistribuídos em nós irmãos, podendo, além disso, também considerar para reinserção o irmão que sofrerá menor acréscimo de área. Entretanto, a reinserção é mais conveniente considerando o fato de que ela possibilita a árvore  $R$  refletir dinamicamente as mudanças na estrutura espacial ao invés de gradualmente degradar a árvore mantendo os retângulos sempre em suas mesmas posições.

Durante a busca numa árvore  $R$  um grande número de nós pode ser visitado pois o objeto buscado pode ter interseção com vários nós. Como exemplo, considere a busca do retângulo  $B$  (figura 2.11). Começando pela raiz observamos que ambas as entradas 1 e 2 têm interseção com este retângulo, logo é necessário descer para as sub-árvores apontadas pelas mesmas. No nível seguinte, o nó apontado por 1 possui apenas a entrada 3 interceptando o objeto. O nó apontado por 2 possui a entrada 6 com interseção com o retângulo em questão sendo necessário também descer por este nó. A partir da entrada 3 vamos para um nó folha e neste alcançamos  $B$ . A partir da entrada 6 caminhamos para um nó folha que não contém  $B$ .

Existem outras maneiras para executar a divisão de nós. Uma delas é distribuir os MBRs entre os nós visando a redução do número de árvores visitadas durante a busca. Tal fato é obtido minimizando a área total de cobertura dos retângulos dos nós, ou seja, reduzindo a área de desperdício entre o MBR do nó e o conjunto de retângulos incluídos no mesmo. Uma outra alternativa é diminuir o número de nós visitados em buscas

---

<sup>1</sup> Área desperdiçada de um nó é a área não coberta por MBR.

subseqüentes, o que pode ser alcançado minimizando a área comum entre ambos. A área de interseção entre retângulos em cada nível da árvore deve ser mínima. Muitas vezes os dois princípios são contraditórios como pode ser observado na figura 2.12. A primeira alternativa aparece na figura 2.12.b e a segunda na figura 2.12.c.

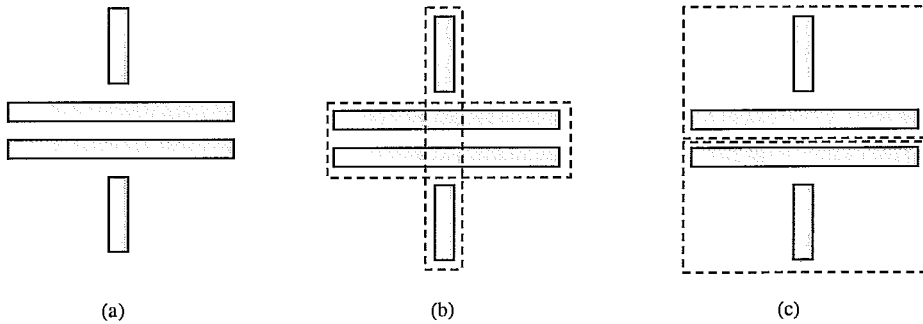


Figura 2.12 - (a) Quatro retângulos envolventes mínimos; (b) divisão minimizando a área de cobertura dos retângulos; (c) divisão minimizando a área de sobreposição entre os nós.

Outra heurística para divisão é minimizar o perímetro de cada MBR aproximando-os a quadrados. Quadrados são mais fáceis de agrupar do que retângulos. Pode-se também utilizar como heurística obter taxa máxima de utilização dos nós já que nós cheios diminuem o número de nós acessados durante a busca.

Uma variante das árvores R é a árvore R+, baseada no critério de evitar sobreposição entre os retângulos limitantes. Em particular, todos os retângulos limitantes, exceto os da folha, não possuem interseção. Cada retângulo envolvente mínimo dos objetos está associado com todos os retângulos limitantes que o interceptam. Como resultado existe mais de um caminho para alcançar um objeto. Consequentemente a altura da árvore é aumentada enquanto que o tempo de busca é acelerado.

A figura 2.13 rerepresenta os objetos da figura 2.11 considerando retângulos limitantes seguindo o critério da árvore R+. Observe que os retângulos B e F aparecem em dois nós diferentes e que não existe sobreposição entre retângulos limitantes de nós internos. Existem outras possibilidades para a representação da árvore R+ ilustrada.

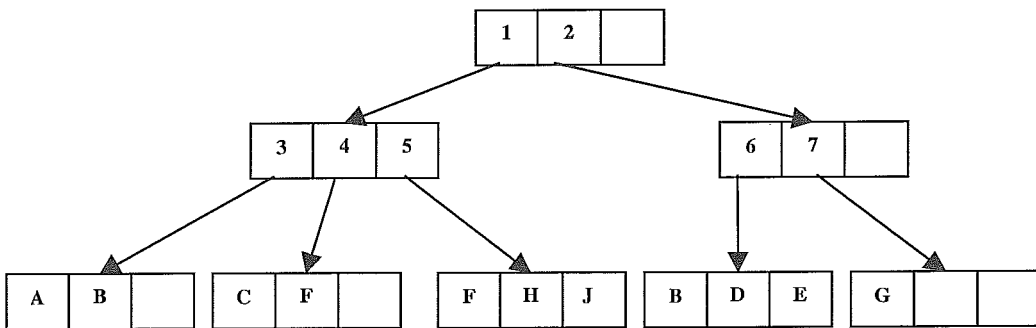
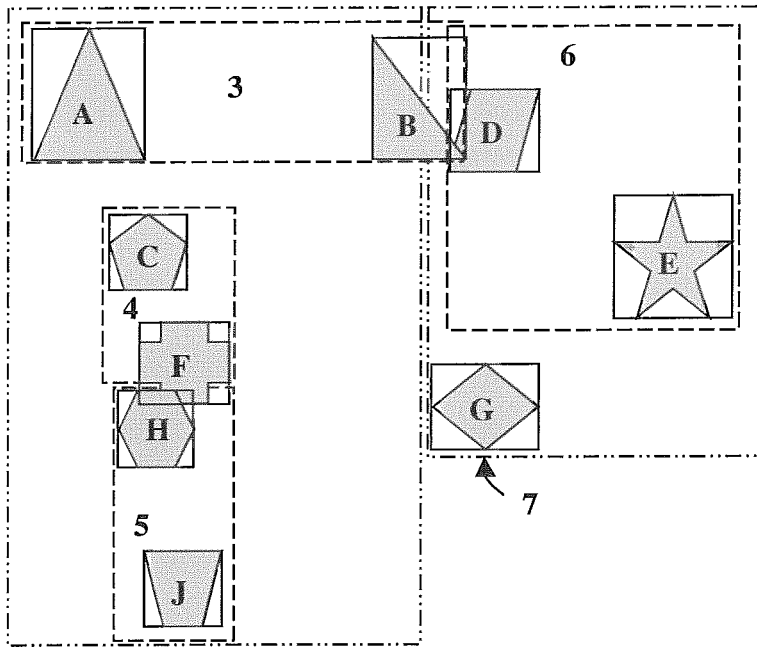


Figura 2.13 - Árvore R+

Outra variante da Árvore R é a Árvore R\*. A principal diferença entre estas duas estruturas se encontra no algoritmo de inserção de nós. Durante a inserção quando se atinge o penúltimo nível escolhe-se para inserção o nó folha que resultar em menor acréscimo de área total após a inserção. Se ocorrer empates, escolhe-se a folha com menor incremento de área. Se empatar novamente, escolher a folha com menor área.

Todas as representações de conjuntos de retângulos apresentadas são estruturas candidatas ao primeiro passo do MSQP, além de muitas outras presentes na literatura tais como: abordagens baseadas em hash (DEWITT et al., 1996, MING et al., 1995, 1996, ZIMBRÃO et al., 1997); algoritmos paralelos usando Árvores R\* (BRINKHOFF et al.,

1996); X-Tree (BERCHTOLD et al., 1996); separação por tamanho dos MBRs em uma árvore especial chamada de árvore de filtro (KOUDAS et al., 1997); algoritmo baseado em *plane-sweep* para realizar a junção quando nenhum índice existe sobre os conjuntos (ARGE et al., 1998); algoritmo para ser utilizado quando apenas um dos conjuntos possui índice, situação típica que ocorre quando uma consulta está encadeada com outra (MAMOULIS et al., 1999).

Como já mencionado na seção 2.3, um dos benefícios da modularização do MSQP é a possibilidade de testar e propor novas estruturas para cada um dos passos de forma independente. Assim, escolhemos, sem perda de generalidade, uma variante da Árvore R\* para a execução do primeiro passo, cujas características peculiares serão apresentadas mais à frente.

## 2.5 Filtro geométrico

Um filtro geométrico usa uma representação aproximada e compacta do objeto a qual guarda suas principais características. A utilização do filtro geométrico durante o processamento de consultas espaciais tem como objetivo reduzir o tempo de processamento obtendo uma resposta conclusiva através da comparação entre objetos mais simples (as aproximações dos objetos) sem ter que acessar o objeto real, o que pode ser custoso de acordo com a complexidade do mesmo.

Como exemplo, considere o filtro geométrico Retângulo Máximo Incluso ou MER (*Maximum Enclosed Rectangle*) que guarda informações do interior do objeto sendo aproximado através do maior retângulo que ele contém. A figura 2.14 ilustra o MER aproximando uma das ilhas do Havaí. Utilizando o MER, no segundo passo da arquitetura, podemos concluir que dois objetos se interceptam se as aproximações dos mesmos se sobrepõem.

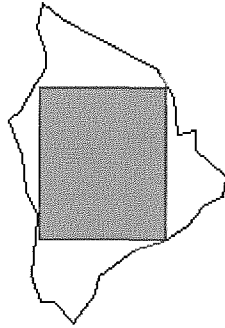


Figura 2.14 - MER para uma ilha do Havaí.

O capítulo 3 é dedicado a maiores detalhes sobre filtros geométricos, uma vez que estes se referem ao principal assunto desta tese.

## 2.6 Teste de geometria exata

O terceiro passo da arquitetura de múltiplos passos é o mais custoso pois as representações reais dos objetos são comparadas, requerendo tempo de E/S para lê-las do disco e tempo de CPU para processar a interseção exata. Com o objetivo de otimizar sua execução, muitos algoritmos e estruturas espaciais foram desenvolvidos, entre os quais podemos destacar um algoritmo baseado em *scan line* (MONTEIRO et al., 2001).

Este algoritmo foi desenvolvido com o intuito de verificar se duas polilinhas se interceptam. Possui três etapas.:

- **Primeira etapa:**

- 1) Para cada parte da polilinha faça:

- i. Ordenar os (n-1) segmentos da parte considerando o primeiro ponto de cada segmento e o sentido crescente de um dos eixos cartesianos;
- ii. Inserir os segmentos ordenados numa nova lista (formação de seqüências monotônicas);

- **Segunda etapa:**

- 2) Para cada seqüência monotônica faça:

- i. Inserir todos os segmentos num *heap* (KNUTH, 1973, SWARCFITER et al., 1994), agregando a cada um o indicador da polilinha e da sequência monotônica a qual pertence;

• **Terceira etapa:**

- 3) Criar lista vazia para armazenar elementos do *heap*, chamada de lista de comparações.
- 4) Enquanto existir elemento no *heap* faça
  - i. Retirar raiz do *heap* atribuindo a elemento atual;
  - ii. Para cada elemento da lista de comparações faça:
    1. Se identificador de polilinha do elemento da lista é diferente do identificador de polilinha do elemento atual
      - a. Se existe interseção entre os segmentos destes elementos
        - i. **Retornar interseção como resultado** finalizando o processamento
    - iii. Se existir elemento na lista de comparações com mesmo identificador de sequência monotônica do elemento atual
      1. Substituir este elemento por elemento atual.
    - iv. Senão
      1. inserir elemento atual na lista de comparações
- 5) **Retornar não interseção como resultado**

O *heap*, utilizado na segunda parte do algoritmo, tem como objetivo obter ordenação com complexidade  $O(n \log n)$ . Assim, outra estrutura ou algoritmo pode ser empregado para executar a ordenação desde que seja garantida a complexidade de pior caso.

## 3 Filtros Geométricos

O presente capítulo tem por finalidade apresentar e elucidar a função dos filtros geométricos, responsáveis por desempenhar o segundo passo do MSQP.

Filtros geométricos correspondem ao uso de aproximações dos objetos para, sem acessar a sua representação exata, definir se um par deve fazer parte ou não de um conjunto solução. Para tanto, as aproximações procuram representar as características relevantes dos objetos de forma concisa e compacta.

Este capítulo apresenta na seção 3.1 uma classificação de aproximações e filtros geométricos. A seção 3.2 demonstra as vantagens dos filtros raster e, finalmente, a seção 3.3 apresenta um esquema de divisão do espaço em células, responsável por viabilizar o uso de aproximações raster.

### 3.1 Tipos de Filtro Geométrico

Os filtros geométricos são classificados de acordo com o tipo de aproximação utilizada. As diferentes aproximações geométricas, por sua vez, são comumente classificadas de acordo com características que determinam a sua função. BRINKHOFF et al. (1993b) classificam as aproximações geométricas como conservadoras, progressistas e generalizadoras.

As aproximações conservadoras são caracterizadas por conter todo o objeto aproximado. Desta forma, os filtros que as utilizam permitem o reconhecimento de pares de objetos que não se interceptam. Este reconhecimento se dá através da constatação de que as aproximações conservadoras não se interceptam. Alguns exemplos de aproximações que desempenham o papel de filtros conservadores são apresentados na figura 3.1 (BRINKHOFF et al., 1993b).

Observando mais detalhadamente a figura 3.1 temos algumas aproximações que se utilizam de formas geométricas básicas para representar o objeto. Entre estas podemos citar as aproximações (a), (f) e (g) onde são usados, respectivamente, o retângulo, o círculo e a elipse mínimos que envolvem o objeto. A aproximação (b) corresponde a um refinamento



da aproximação (a) onde um ângulo de rotação é adicionado. As demais aproximações usam um polígono convexo como representação. Diferem entre si pelo número de pontos usados na aproximação. A aproximação (c) (*Convex Hull*) corresponde ao menor polígono convexo que contém o objeto, não estabelecendo limite no número de pontos. Já as aproximações n-C, como as apresentadas em (d) e (e), correspondem ao menor polígono convexo com n pontos que contém o objeto.

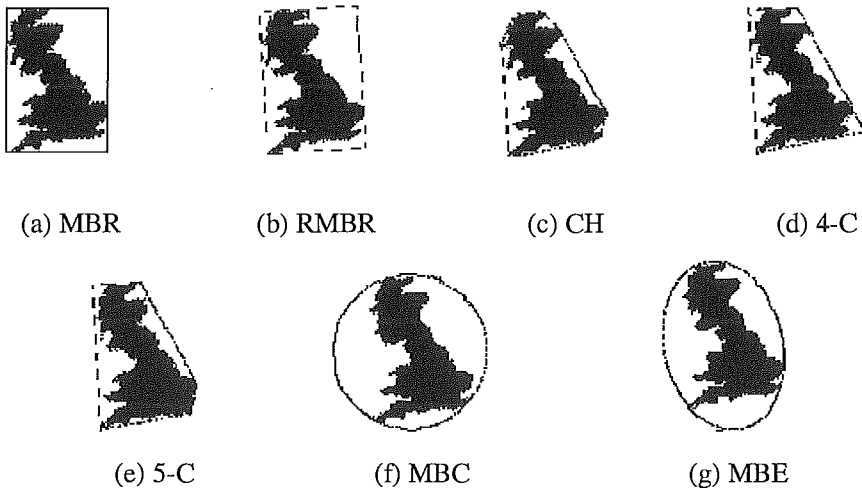


Figura 3.1 – Aproximações conservadoras (BRINKHOFF et al., 1993b).

As aproximações progressistas são caracterizadas por estarem totalmente contidas no objeto aproximado. Assim, os filtros dessa classe permitem o reconhecimento de pares de objetos que se interceptam através da constatação de que suas aproximações progressistas se interceptam. Alguns exemplos de aproximações que desempenham o papel de filtros progressistas são apresentados na figura 3.2 (BRINKHOFF et al., 1994). São apresentados o maior retângulo e maior círculo contido no objeto respectivamente em (a) e (b).



Figura 3.2 – Aproximações progressistas (BRINKHOFF et al., 1994)

As aproximações generalizadoras procuram simplificar o contorno do objeto aproximado, o que pode ser feito, por exemplo, através da redução do número de vértices. De um modo geral, não existe uma relação topológica entre o objeto original e a aproximação generalizadora, ou seja, o objeto não está contido na aproximação nem a aproximação está contida no objeto. São muito úteis, por exemplo, na apresentação de mapas.

Uma classe de filtros geométricos generalizadores em especial corresponde ao uso de aproximações raster. Esta classe de filtros permite a identificação tanto de pares que se interceptam quanto de pares que não se interceptam. Baseia-se na divisão do espaço do objeto em células, cada qual procurando sintetizar alguma característica de interesse da área coberta através do armazenamento de um número pequeno de bits. Alguns exemplos de aproximações que desempenham o papel de filtros raster são apresentados na figura 3.3.

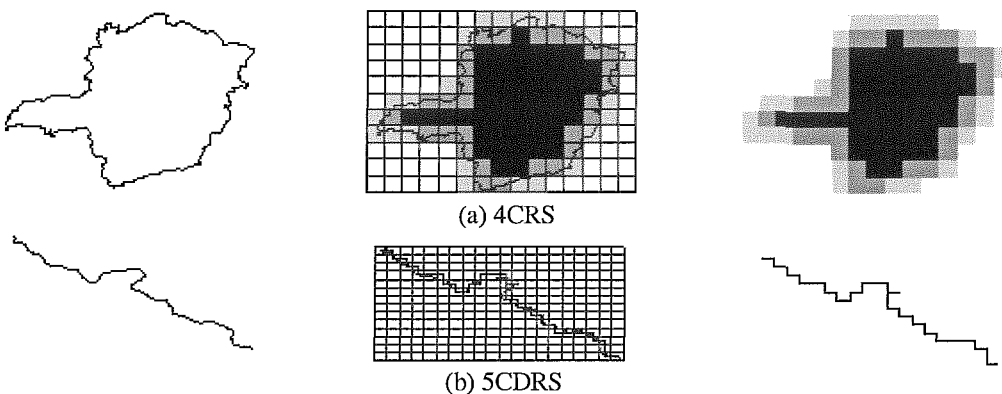


Figura 3.3 – Aproximações raster

## 3.2 Vantagens da utilização de Filtros Raster

Uma das vantagens dos filtros raster, conforme já mencionado, é permitir o reconhecimento tanto de pares que se interceptam como de pares que não se interceptam. Assim sendo, o uso de uma aproximação raster pode ser comparado a uma combinação de duas aproximações: uma conservadora e outra progressista.

ZIMBRÃO (1999) usa o conceito de área de indecisão como medida de qualidade de uma aproximação para polígonos. A área de indecisão é definida como a área contida na aproximação conservadora e não contida na aproximação progressista. É chamada de área de indecisão porque se as aproximações de dois polígonos possuírem interseção apenas nesta área não podemos garantir um resultado conclusivo por parte do filtro. De fato, apenas o filtro raster permite obter em certos casos conclusões sobre interseções na área de indecisão. Os demais filtros são sempre inconclusivos. A figura 3.4 ilustra esta medida de qualidade, evidenciando a superioridade da aproximação raster.

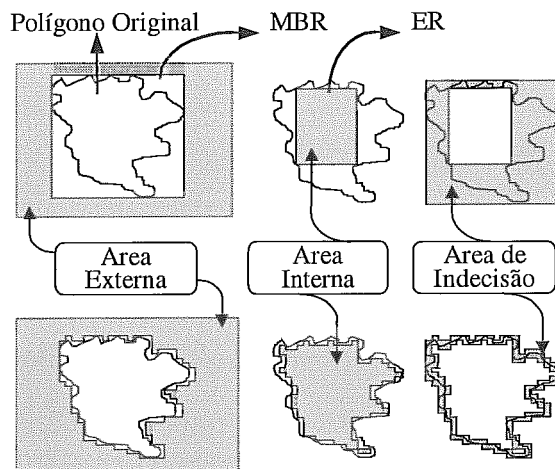


Figura 3.4 – Área de Indecisão (ZIMBRÃO et al., 1998)

As demais aproximações não apresentam um desempenho satisfatório no que diz respeito a polilinhas, conforme apontado em AZEVEDO (2001), uma vez que foram concebidas para a representação de polígonos. Já a aproximação 5CDRS, descrita no

próximo capítulo, apresentou resultados compatíveis com os níveis apresentados por BRINKHOFF et al. (1994).

As aproximações raster possuem a possibilidade de regular através de um parâmetro (apresentado na seção seguinte) o nível de abstração desejado. Toda a porção do objeto contida dentro de uma célula é representada por poucos bits, não importando a complexidade da mesma. O número de células usado possui reflexo direto no tamanho de cada célula. Assim, a parametrização do número máximo de células, apresentada na seção seguinte, permite uma *sintonia fina* da proporção qualidade x tamanho da aproximação. A seção a seguir apresenta em detalhes o procedimento para divisão do espaço em células.

### 3.3 Divisão do espaço em Células

As aproximações raster necessitam que o espaço do objeto seja decomposto em células para que as características ou tipos das mesmas sejam extraídas e armazenadas. Porém não basta decompor o espaço de cada objeto de forma relativa ao mesmo. Embora tal abordagem permita gerar uma aproximação, se o fizermos sem considerar uma decomposição absoluta em relação às coordenadas georeferenciáveis<sup>1</sup>, nunca poderemos comparar duas aproximações. Isto ocorre porque não será possível estabelecer uma relação de correspondência entre as células de duas aproximações diferentes, onde ambas representem exatamente a mesma região do espaço, conforme ilustrado na figura 3.5. Assim, é necessário estabelecer uma divisão do espaço em células independente do posicionamento do objeto sendo aproximado.

---

<sup>1</sup> Georeferência: estabelecer uma relação entre coordenadas de um mapa e coordenadas do mundo real.

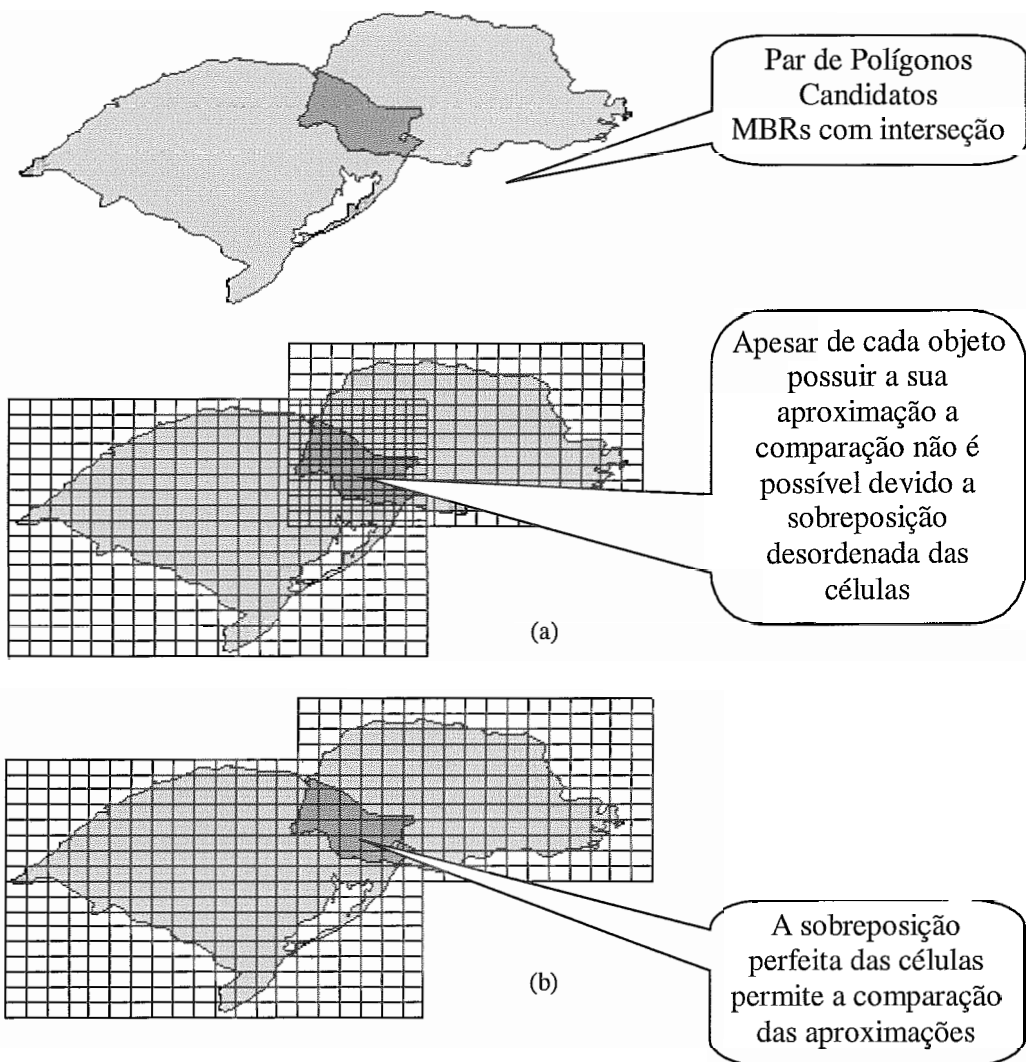


Figura 3.5 – Decomposição do espaço em células. (a) Decomposição relativa ao objeto.  
(b) Decomposição absoluta.

Numa primeira abordagem, podemos pensar em uma grade de tamanho fixo  $n$ , onde as coordenadas múltiplas de  $n$  correspondem às arestas das células. O problema dessa abordagem ocorre quando um objeto com dimensões grandes em relação a  $n$  é aproximado. O número de células pode ser proibitivamente grande. Por outro lado, um objeto pequeno em relação a  $n$  será pessimamente aproximado, podendo sua assinatura, no pior caso, degenerar em uma única célula. A impossibilidade de garantir um número máximo de células por aproximação inviabiliza esta abordagem.

Numa segunda abordagem, podemos fixar um número máximo de células para cada aproximação. Para tanto, devemos aumentar o tamanho da célula à medida que as dimensões do objeto crescem. Porém, se aumentarmos o tamanho da célula de forma arbitrária, retornaremos ao problema de não poder estabelecer uma relação entre células de duas aproximações de forma que estas correspondam à mesma região do espaço.

Finalmente, combinando características das duas abordagens acima, podemos estabelecer o tamanho de cada célula como sendo  $2^n$ , onde  $n$  é inteiro e pode variar de acordo com as dimensões do objeto. Ao mesmo tempo, as coordenadas múltiplas de  $2^n$  correspondem às arestas das células. Com esta abordagem podemos garantir um número máximo de células para a aproximação de um objeto, ao mesmo tempo que permitimos estabelecer uma correspondência entre células de duas aproximações. A figura 3.6 ilustra essas características.

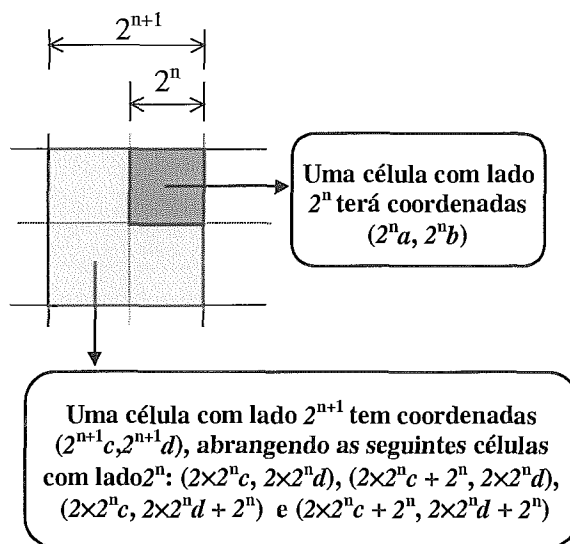


Figura 3.6 – Alinhamento dos cantos das células

As  $m \times n$  células decompõem o espaço correspondente ao MBR- $2^k$  do objeto, o qual corresponde a um MBR cujos vértices são da forma  $(2^ka_0, 2^kb_0)$  e  $(2^ka_m, 2^kb_n)$ , onde  $a_0, a_m, b_0, b_n$  e  $k$  são inteiros. Além disso,  $k$  é escolhido de tal modo que  $(a_m - a_0)(b_n - b_0) < K + 1$ , onde  $K$  é o número máximo de células fixado no momento da geração. O MBR- $2^k$  é calculado com base no MBR do polígono da seguinte forma: inicialmente o maior  $k$  é escolhido de forma que  $2^k$  seja menor que o lado do MBR original. A partir daí, em

repetições sucessivas,  $k$  é diminuído de 1 enquanto as condições anteriores valerem. Por fim,  $a_0$  e  $b_0$  são os maiores inteiros que satisfazem  $2^k a_0 \leq x$  mínimo do MBR e  $2^k b_0 \leq y$  mínimo do MBR respectivamente. Analogamente,  $a_m$  e  $b_n$  são os menores inteiros que satisfazem  $2^k a_m \geq x$  máximo do MBR e  $2^k b_n \geq y$  máximo do MBR respectivamente.

Os pontos  $2^k a_0, 2^k a_1, \dots, 2^k a_m$  definem um conjunto de linhas paralelas ao eixo vertical que chamaremos de  $X_0, X_1, \dots, X_m$ . De modo semelhante, definimos as linhas horizontais  $Y_0, Y_1, \dots, Y_n$ .

## 4 Aproximações Raster para polígonos e polilinhas

Este capítulo apresenta em detalhes as duas estruturas alvo deste trabalho: as aproximações 4CRS (Four Color Raster Signature) e 5CDRS (Five Color Directional Raster Signature). A 4CRS ou Assinatura Raster de Quatro Cores apresentada em ZIMBRÃO et al. (1998) constitui uma das contribuições da tese de doutorado do mesmo autor. A 5CDRS ou Assinatura Raster Direcional de Cinco Cores deu origem ao artigo ZIMBRÃO et al. (2000) submetido, aprovado e apresentado no XV Simpósio Brasileiro de Banco de Dados e faz parte de um trabalho desenvolvido em conjunto com AZEVEDO (2001).

Podemos definir uma aproximação raster como a representação de um objeto através da sobreposição de uma grade de células, cada qual armazenando as características de interesse do mesmo. A resolução da grade pode ser ajustada de forma a se obter uma maior precisão ou maior compactação. Quanto maior o número de células, menor será a área do objeto por ela aproximada e portanto mais precisa a representação. Por outro lado, quanto menor o número de células maior o nível de abstração da complexidade do objeto e portanto maior a compactação.

Apenas um pequeno número de bits é armazenado para cada célula, não importando a complexidade (número de vértices) da porção do objeto contido na mesma. Além disso, as aproximações raster possuem a característica de combinar propriedades conservadoras e progressistas. Assim, são excelentes candidatas a desempenhar o papel de filtros geométricos no processamento de junções espaciais. Estas e outras características são abordadas em detalhes na seção 4.1 para a 4CRS e na seção 4.2 para a 5CDRS. A seção 4.3 apresenta o algoritmo de comparação entre as duas aproximações.

### 4.1 Assinatura Raster de Quatro Cores

Nesta seção a aproximação 4CRS é apresentada, descrevendo algoritmos para sua computação, comparação para detecção de interseção de polígonos e mudança de escala. A seguir, o uso de compactação e outras variantes são relatadas. Alguns resultados obtidos por ZIMBRÃO et al. (1998) são apresentados ao final da seção.



A aproximação 4CRS destina-se a representação de polígonos e corresponde a um mapa de bits do mesmo contendo quatro cores distintas, a saber. Por se tratar de uma assinatura, pode ser computada uma única vez e armazenada para uso futuro. Para o processamento de consultas espaciais, um índice espacial constitui o local mais adequado para o armazenamento das assinaturas. Porém requisitos de espaço devem ser observados. A assinatura 4CRS mostrou-se compacta o suficiente para se adequar a estes requisitos.

### 4.1.1 Geração da Assinatura

Conforme mencionado no capítulo 2, baseado nos resultados e conclusões obtidos por BRINKHOFF et al. (1994) apenas melhorias nos filtros geométricos empregados no segundo passo do MSQP são capazes de reduzir o número de pares enviados ao terceiro passo, quando a junção de MBR é empregada no primeiro passo. Como o terceiro passo é o mais dispendioso, qualquer melhoria no segundo passo pode resultar em melhorias significativas no tempo total de execução.

Nesse contexto, a aproximação 4CRS foi criada com o intuito de substituir, com vantagens, as aproximações 5C (conservadora) e EL&EC (progressista) usadas por BRINKHOFF et al. (1994).

A aproximação 4CRS é composta por uma grade de  $m \times n$  células, cada uma contendo dois bits de informação indicando uma das quatro cores presentes na tabela 4.1.

Tabela 4.1 – Tipos de célula 4CRS

Valor dos bits	Tipo da Célula	Cor	Descrição
00	Vazia	Branco	A célula não intercepta o polígono
01	Pouco	Cinza Claro	A célula contém uma interseção de 50% ou menos com o polígono
10	Muito	Cinza Escuro	A célula contém uma interseção de mais de 50% com o polígono
11	Cheia	Preto	A célula está totalmente ocupada pelo polígono

As  $m \times n$  células decompõem o espaço correspondente ao  $MBR-2^k$  conforme descrito na seção 3.3 do capítulo anterior. Para cada par de linhas verticais ( $X_i, X_{i+1}$ ) a

interseção com o polígono original é computada. ZIMBRAO et al. (1998) usa uma versão simplificada do algoritmo de Sutherland-Hodgman (ROGER, 1986) para executar esta operação. Para cada polígono resultante  $P_i$ , a interseção com os pares de linhas horizontais  $(Y_i, Y_{i+1})$  é computada. O resultado para cada par de linhas é a interseção do polígono original com a célula  $(i, j)$ . Por último, a área de interseção de cada célula é calculada e sua classificação é obtida de acordo com a tabela 4.1. Um exemplo da aproximação 4CRS é apresentado na figura 4.1.

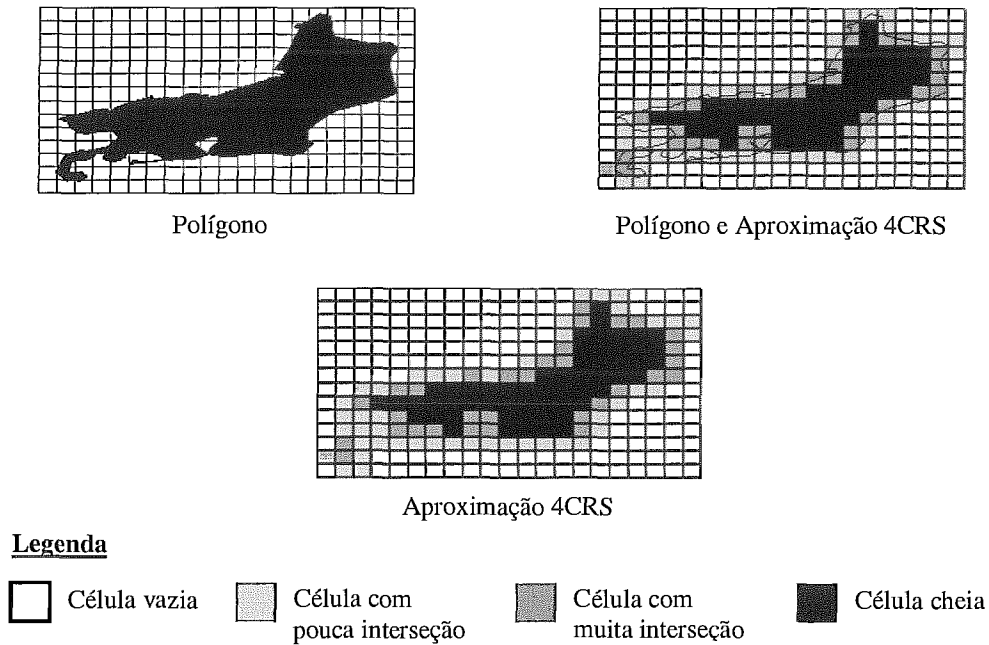


Figura 4.1 – Exemplo de uma aproximação 4CRS

## 4.1.2 Algoritmo de Comparação

Para um par de polígonos candidatos vindo da junção de MBRs as suas aproximações devem ser comparadas na etapa do filtro geométrico. No caso da 4CRS, esta comparação é efetuada através da sobreposição das aproximações na área onde ocorre a interseção dos MBRs. Para que esta comparação seja possível, dois requisitos devem ser observados: as células dos polígonos em questão devem possuir o mesmo tamanho, e para tanto pode ser necessária uma mudança de escala, a qual é apresentada na próxima seção; e

as células devem se sobrepor perfeitamente, o que é garantido através dos algoritmo de geração do MBR-2<sup>k</sup> e divisão do espaço em células apresentados em detalhes na seção 3.3. Observados estes requisitos, para cada par de células sobrepostas na área de interseção dos MBRs teremos as possibilidades ilustradas na tabela 4.2.

Tabela 4.2 – Possíveis resultados da comparação entre duas células da aproximação 4CRS

Tipo de célula × Interseção	Vazia	Pouca Interseção	Muita Interseção	Cheia
Vazia	Rejeitado	Rejeitado	Rejeitado	Rejeitado
Pouca Interseção	Rejeitado	<i>Inconclusivo</i>	<i>Inconclusivo</i>	Aceito
Muita Interseção	Rejeitado	<i>Inconclusivo</i>	Aceito	Aceito
Cheia	Rejeitado	Aceito	Aceito	Aceito

Observe que se ocorre a sobreposição de duas células com muita interseção, chegamos à conclusão de que os polígonos se interceptam. Isto se explica pelo fato de que uma célula com muita interseção possui mais da metade de sua área ocupada pelo polígono. Sendo assim, dois polígonos que ocupam mais da metade da área da mesma célula obrigatoriamente se interceptam em alguma região no interior da mesma.

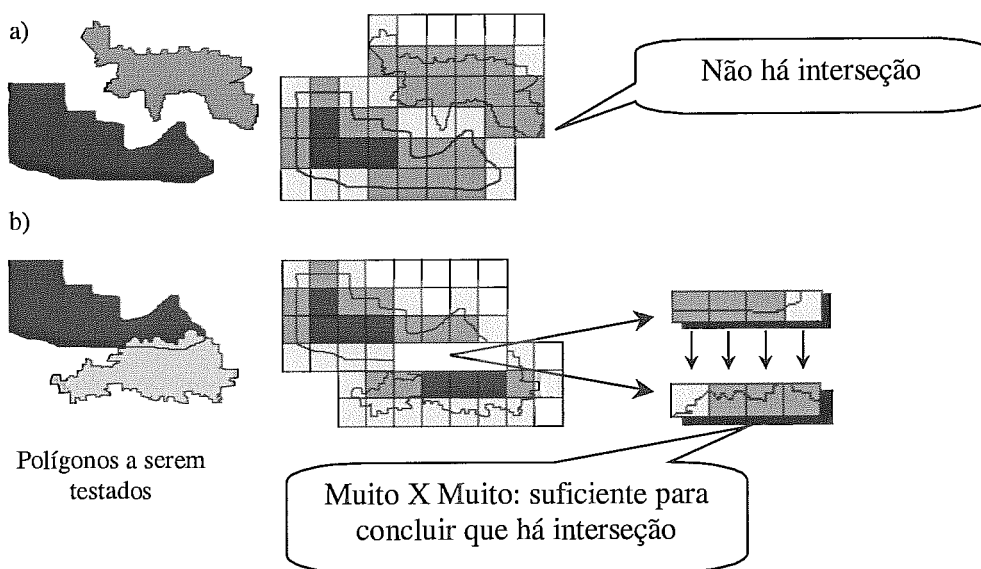


Figura 4.2 – (a) polígonos sem interseção, (b) polígonos com interseção

Outro ponto a ser ressaltado é que basta uma comparação resultar em *Aceito* para que a interseção seja detectada. Por outro lado, todas as células a serem comparadas devem resultar em *Rejeitado* para que o par candidato seja descartado. Dois exemplos de comparação de duas assinaturas 4CRS podem ser observados na figura 4.2 apresentada em ZIMBRÃO et al. (1998).

### 4.1.3 Mudança de Escala

Uma vez garantida a sobreposição perfeita das células através dos procedimentos apresentados na seção 3.3, basta, para tornar sempre possível a comparação entre duas aproximações, um procedimento que permita alterar a resolução de uma das aproximações. Esta mudança de escala tem como objetivo obter células de mesmo tamanho nas duas aproximações.

Note que apesar de podermos subdividir as células do tipo *Vazia* e *Cheia* sem cometer erros, o mesmo não é possível para as células do tipo *Muito* e *Pouco*. Assim a mudança de escala sempre será feita agrupando células da aproximação de maior resolução. Cada célula da nova resolução será formada por agrupamentos de 2<sup>m</sup> células, o que é sempre possível graças a divisão do espaço adotada. A mudança de escala é ilustrada na figura 4.3.

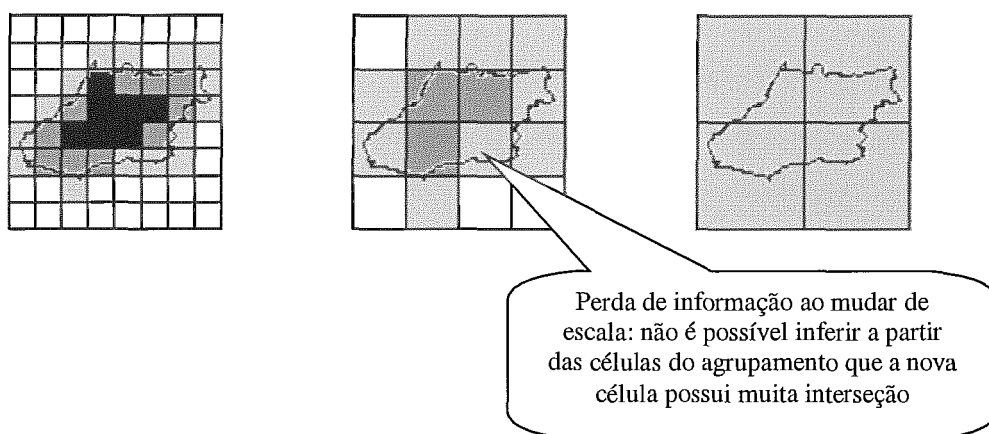


Figura 4.3 – Mudança de resolução da aproximação 4CRS do estado de Goiás

Feito o agrupamento de  $2^m$  células devemos descobrir qual o tipo da nova célula. Para tal, uma abordagem pessimista deve ser adotada para não incorrer em erros. Por este motivo, devemos fazer a mudança de escala diretamente da escala original para a escala desejada, sem passar por escalas intermediárias, de forma a perder o mínimo de informação possível. O algoritmo para computar o tipo da nova célula em função dos tipos das células do agrupamento é apresentado a seguir:

1. se todas as células forem *Vazias*, o resultado é célula *Vazia*;
2. se todas as células forem *Cheias*, o resultado é célula *Cheia*;
3. caso contrário, devemos realizar um somatório onde cada célula *Vazia* ou *Pouca* contará como valendo 0, *Muita* contará como valendo 0.5 e *Cheia* contará como valendo 1; se a média for menor do que 0.5 o resultado é célula com *Pouca* interseção, se não o resultado é célula com *Muita* interseção.

#### 4.1.4 Compactação e outras variantes

Através de estudos estatísticos realizados por ZIMBRÃO (1999), foi constatado que, com um limite superior de 750 células, 80% das mesmas são vazias ou cheias. Além disso, foi observado que com o aumento da resolução este percentual tende a aumentar ainda mais. Este fato sugeriu a utilização de algum tipo de compactação. Fazendo uso de uma adaptação do algoritmo de Huffman para representar padrões de 3x3 células, foram obtidas assinaturas compactadas possuindo apenas 40% do tamanho original. O principal ganho diz respeito a obtenção de uma maior densidade de informação, através do uso de uma maior quantidade de células, o que reflete em uma melhor capacidade de decisão e filtragem no passo 2.

Maiores de detalhes sobre o algoritmo de compactação podem ser obtidos em ZIMBRÃO (1999). No mesmo trabalho outras variantes da assinatura são exploradas, como por exemplo o uso de um número maior de cores e uso de células especiais. Para o presente

trabalho usamos apenas as características já apresentadas, uma vez que entendemos que tais variações não teriam efeitos na junção com polilinhas fazendo uso da aproximação 5CDRS.

### 4.1.5 Resultados Experimentais (4CRS)

Esta seção apresenta alguns resultados experimentais obtidos por ZIMBRÃO et al. (1998) na execução de junções de conjuntos de polígonos com o intuito de ilustrar a eficiência da aproximação 4CRS. Os percentuais de identificação são apresentados e comparados com outras aproximações concorrentes na tabela 4.3. Não são apresentados porém, detalhes sobre o ambiente de teste e características dos dados, os quais podem ser obtidos na referência supracitada.

Tabela 4.3 – Desempenho dos filtros (a) 4CRS-350 células, (b) 4CRS-750 células e (c) 5-C/ER&EL (os melhores resultados estão em negrito) Fonte: (ZIMBRÃO et al., 1998)

	Aceitos	Rejeitados	Candidatos (menor é melhor)	Rejeitados Identificados	Aceitos Identificados
4CRS (350 pontos)	59.1%	21.6%	19.2%	66.2%	87.8%
4CRS (750 pontos)	<b>61.5%</b>	<b>24.6%</b>	<b>13.7%</b>	<b>75.2%</b>	<b>91.53%</b>
5-C & ER [ZIMB98]	23.5%	23.6%	52.8%	71.4%	35.2%
5-C & ER-EL [ZIMB98]	40.0%	23.6%	36.4%	71.4%	60.2%
5-C & ER (Europa A) [BRI94]	23%	23%	54%	66%	30%
5-C & ER-EL (Europa A) [BRI93b,BRI94]	40.5%	20.8%	40.84%	66.3%	59.2%

Os ganhos obtidos com o uso da aproximação 4CRS são muito significativos, principalmente no que diz respeito a identificação de pares de polígonos que se interceptam. Como consequência, temos uma grande redução do número de pares candidatos que passam ao teste de interseção exata, refletindo numa significativa redução do tempo total de execução.

## 4.2 Assinatura Raster Direcional de Cinco Cores

Nesta seção a aproximação 5CDRS é apresentada juntamente com seus algoritmos para geração, comparação para detecção de interseção de polilinhas e mudança de escala. Alguns resultados obtidos em ZIMBRÃO et al. (2000) são apresentados ao final da seção.

A aproximação 5CDRS destina-se a representação de polilinhas e corresponde a um código encadeado (*chain code*) (WILF, 1981), contendo vetores de deslocamento da mesma em relação a uma grade de células. Se tratando de uma assinatura, da mesma forma que a 4CRS, pode ser computada uma única vez e armazenada para uso futuro. Para o processamento de consultas espaciais, um índice espacial constitui o local mais adequado para o armazenamento das assinaturas. Porém requisitos de espaço devem ser observados. A assinatura 5CDRS mostrou-se compacta o suficiente para se adequar a estes requisitos.

### 4.2.1 Geração da Assinatura

A aproximação 5CDRS apresentada em ZIMBRÃO et al. (2000) foi criada com o intuito de obter uma representação compacta para polilinhas capaz de desempenhar o passo 2 do MSQP, requisitos estes que, como já mencionado na seção 3.2, não são atendidos de forma satisfatória por nenhuma estrutura existente atualmente.

A aproximação 5CDRS, da mesma forma que a 4CRS, é composta por uma grade de  $m \times n$  células. Porém a 5CDRS não armazena informação para todas as células de sua grade. Ao invés disso, esta se utiliza de vetores de deslocamento para indicar o caminho percorrido pela polilinha na grade de células. Apesar de se tratar de uma aproximação raster, a representação armazenada pode ser melhor classificada como vetorial.

O algoritmo de geração da aproximação 5CDRS consiste em percorrer cada uma das partes da polilinha identificando a célula inicial e os sucessivos deslocamentos. Duas abordagens são apresentadas em AZEVEDO (2001): a primeira gerando uma única assinatura para todas as partes; a segunda gerando uma assinatura para cada parte da polilinha. Como resultados experimentais descritos em ZIMBRÃO et al. (2000) e MONTEIRO et al. (2001) demonstraram vantagens no uso da segunda abordagem, esta será utilizada no presente trabalho. Os passos do algoritmo de geração, para cada parte da

polilinha, são apresentados a seguir. Alguns detalhes foram omitidos propositalmente com a intenção de facilitar o entendimento geral do algoritmo. Tais detalhes são abordados na sequência.

- 1) Obter o  $MBR-2^k$  da parte da polilinha com base no MBR da mesma conforme descrito na seção 3.3;
- 2) Obter a célula do  $MBR-2^k$  contendo o ponto inicial da parte. Armazenar as coordenadas desta célula na grade;
- 3) Marcar a célula inicial como a célula atual, o ponto inicial como o ponto atual e o ponto de entrada como inválido;
- 4) Obter a célula do  $MBR-2^k$  contendo o próximo ponto da parte e marcá-la como a célula destino;
- 5) Enquanto a célula atual for diferente da célula destino faça:
  - i. Computar a interseção do segmento correspondente ao ponto atual e ao próximo ponto com a célula atual. São sempre obtidos um ou dois pontos;
  - ii. Se forem obtidos dois pontos, o ponto correspondente ao ponto de entrada é descartado;
  - iii. Usar o ponto não descartado para computar a próxima célula;
  - iv. Computar e armazenar os vetores de deslocamento correspondentes;
  - v. A próxima célula passa a ser a atual;
  - vi. O ponto não descartado passa a ser o ponto de entrada;
- 6) O próximo ponto passa a ser o ponto atual;
- 7) Se o ponto atual for o último da parte, termina o procedimento;
- 8) Retornar ao item 4.



Um exemplo da assinatura 5CDRS é apresentado na figura 4.4. A figura 4.5 (a) apresenta em detalhe um rio dos Estados Unidos e sua aproximação e (b) apresenta os rios dos Estados Unidos e suas aproximações.

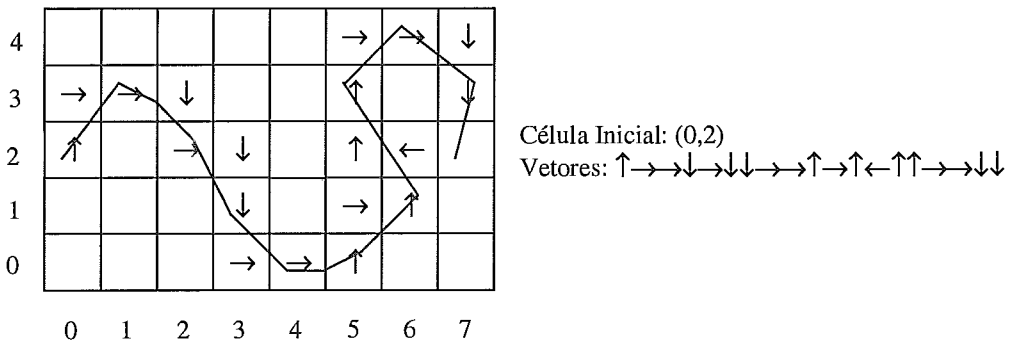
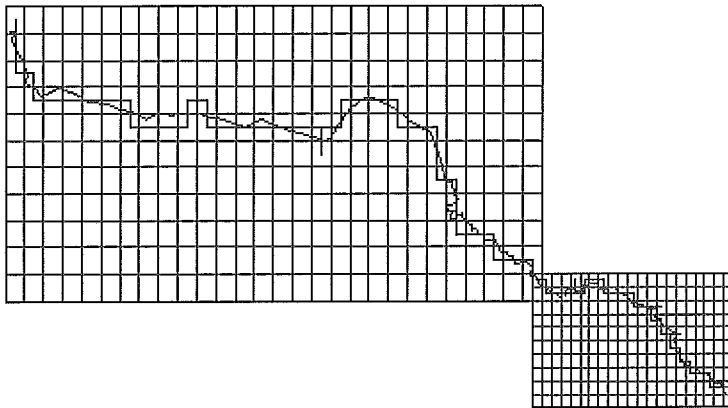
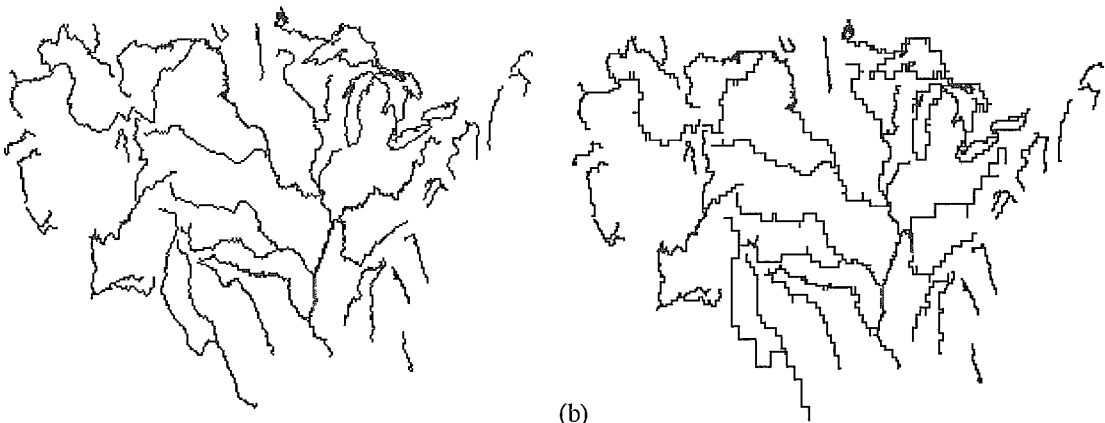


Figura 4.4 – Exemplo aproximação 5CDRS



(a)



(b)

Figura 4.5 – (a) Polilinha com duas partes e suas respectivas aproximações. (b) Rios dos Estados Unidos a esquerda e suas aproximações a direita.

As coordenadas do centro da célula são usadas para desenhar as aproximações

Cada vetor de deslocamento é representado por dois bits conforme indicado na tabela 4.4.

Tabela 4.4 – Tipos de vetores 5CDRS

Valor dos bits	Vetor	Descrição
00	Cima	A continuação da polilinha segue para a célula de cima
01	Direita	A continuação da polilinha segue para a célula da direita
10	Baixo	A continuação da polilinha segue para a célula de baixo
11	Esquerda	A continuação da polilinha segue para a célula da esquerda

A partir de agora, podemos tratar com mais detalhes de três pontos descritos de forma superficial no algoritmo acima transcrito: a forma de representação das coordenadas da célula na grade, a convenção para obtenção da célula que contém um ponto e o cálculo dos vetores de deslocamento.

As células do  $MBR-2^k$  são numeradas, começando do zero, no sentido crescente dos eixos determinando as coordenadas das células na grade. Assim sendo, a coordenada da grade para a célula inicial é obtida e armazenada, e não a coordenada real da célula. A figura 4.6 apresenta alguns exemplos.

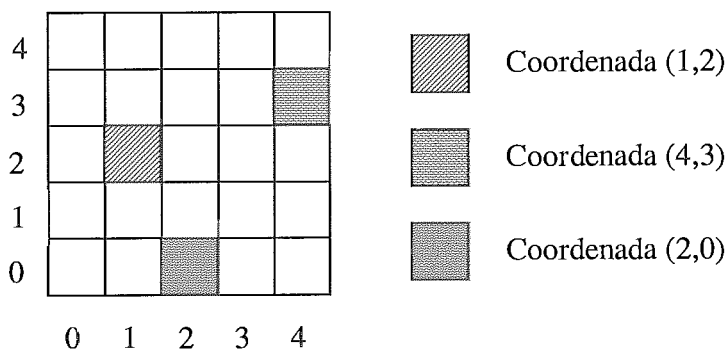


Figura 4.6 – Coordenadas das células na grade

Quanto a obtenção da célula da grade que contém um determinado ponto, devemos adotar algum tipo de convenção para bordas das células. A convenção escolhida foi a de que os pontos sobre as arestas superior e direita não pertencem a célula, enquanto os pontos sobre as arestas inferior e esquerda, excluindo os cantos superior esquerdo e inferior direito, pertencem a célula. A figura 4.7 ilustra esta convenção.

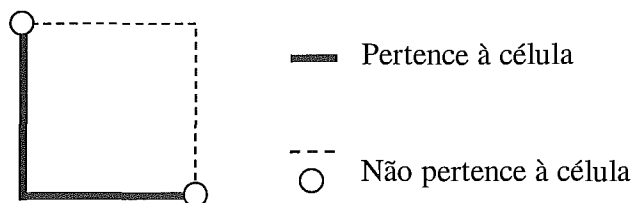


Figura 4.7 – Convenção de interseção com as bordas da célula

O cálculo dos vetores de deslocamento é efetuado em função das arestas interceptadas pelo ponto não descartado. Quando apenas uma aresta é interceptada, o deslocamento correspondente é trivial. Porém, no caso de interseção com os cantos da células, também torna-se necessária a adoção de uma convenção. As tabelas 4.5 e 4.6 apresentam todas as possibilidades de arestas interceptadas e os deslocamentos correspondentes. Note que a interseção com o canto inferior esquerdo recebe um tratamento especial. Isto ocorre porque, dos quatro cantos, este é o único que pertence a célula atual (observe a convenção na figura 4.7) e portanto não indica qual deve ser a próxima célula no deslocamento. Assim, a única forma de descobrir qual o deslocamento adequado é testar o próximo ponto da parte. Devemos garantir que o deslocamento nos leve a uma célula que possui pelo menos um ponto de interseção com o segmento atual de acordo com a convenção da figura 4.7.

Tabela 4.5 – Vetores de deslocamento correspondentes as interseções com arestas

Localização do ponto de interseção (não descartado)	Deslocamento
Aresta direita	Direita
Aresta esquerda	Esquerda
Aresta superior	Acima
Aresta inferior	Abaixo
Canto superior direito	Acima e Direita
Canto superior esquerdo	Acima
Canto inferior direito	Direita
Canto inferior esquerdo	tabela 4.6 (*)

(\*) A coordenada do próximo ponto é analisada segundo a tabela 4.6

Tabela 4.6 – Deslocamentos para interseção com o canto inferior esquerdo. As comparações são apresentadas em ordem de prioridade.

Coordenada do próximo ponto	Deslocamento ou Ação
$Y \geq Y$ mínimo da célula atual e $X \geq X$ mínimo da célula atual	Ponto Entrada = (X min, Y min) da célula atual
$Y \geq Y$ mínimo da célula atual	Esquerda
$X \geq X$ mínimo da célula atual	Abaixo
Outros casos	Esquerda e Abaixo

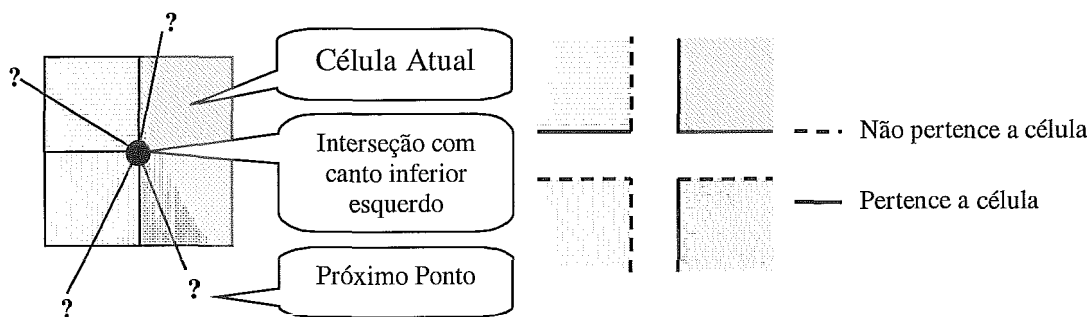


Figura 4.8 – Deslocamentos para interseção com o canto inferior esquerdo.

Esta é a função dos testes apresentados na tabela 4.6 e ilustrados na figura 4.8. A ação de atribuição do ponto de entrada tem a finalidade de manter a célula atual e passar a considerar a próxima saída da célula, de outra forma o algoritmo passaria a executar um ciclo infinito. Dois exemplos de interseção com um canto de célula podem ser observados na figura 4.4 nas passagens da célula (2,2) para a (3,1) e da célula (5,0) para a (6,1).

Um fato importante a ser observado é que, ao contrário do que acontece com aproximação 4CRS, limitar o número de células da grade não determina um tamanho máximo para a aproximação 5CDRS. Isto ocorre porque os vetores de deslocamento podem descrever qualquer percurso na grade podendo passar diversas vezes pela mesma célula. Em outras palavras, um limite superior para o número de células não corresponde a um limite superior para o número de vetores. Como as aproximações serão armazenadas num índice espacial, onde requisitos de espaço de armazenamento devem ser observados, devemos estabelecer um limite superior para o tamanho da assinatura. Nos nossos experimentos usamos, como índice espacial, uma adaptação da Árvore  $R^*$  que permite o armazenamento de entradas de tamanho variável. As características peculiares a esta estrutura são apresentadas no capítulo 5.

Como não é possível saber de antemão o tamanho da aproximação 5CDRS, após a sua geração verificamos se este excedeu o limite preestabelecido. Caso o limite seja excedido, a resolução da grade é diminuída, aumentando em uma unidade o expoente da potência de 2 que define o tamanho da célula (maiores detalhes na seção 3.3). A assinatura é gerada novamente para a grade com a resolução reduzida (passo 2 em diante). Ao final, um novo teste é efetuado repetindo o processo quantas vezes for necessário.

## 4.2.2 Algoritmo de Comparação

Para um par de polilinhas candidatas vindo da junção de MBRs as suas aproximações devem ser comparadas na etapa do filtro geométrico. Como ocorre com a 4CRS, na 5CDRS esta comparação é efetuada através da sobreposição das aproximações na área onde ocorre a interseção dos MBRs. Da mesma forma, os requisitos de mesmo tamanho de células e sobreposição perfeita devem ser satisfeitos. Obter células de mesmo tamanho nas duas aproximações é possível com o algoritmo de mudança de escala apresentado na próxima seção. A sobreposição perfeita das células é, mais uma vez, garantida através dos algoritmos de geração do  $MBR-2^k$  e divisão do espaço em células apresentados em detalhes na seção 3.3.

Como a aproximação 5CDRS armazena vetores de deslocamento e não o tipo de cada célula na grade, devemos, antes da comparação, descobrir o tipo das mesmas. De fato,

o tipo das células é descoberto no momento da mudança de escala, abordado na próxima seção. Por enquanto é suficiente saber que a partir dos vetores de deslocamento podem ser obtidos cinco tipos de células, os quais são apresentados na tabela 4.7.

Tabela 4.7 – Tipos de células 5CDRS

Tipo de Célula	Descrição
Horizontal	Polilinha intercepta a célula horizontalmente
Vertical	Polilinha intercepta a célula verticalmente
Horizontal e Vertical	Polilinha intercepta a célula horizontalmente e verticalmente
Vazia	Polilinha não intercepta a célula
Inconclusiva	Outros tipos de interseção

Caso ocorram diferentes tipos de interseção na mesma célula, o tipo mais conclusivo será armazenado. Ordenando os tipos de célula com interseção do mais conclusivo para o menos conclusivo temos: Horizontal e Vertical; Horizontal, Vertical; e Inconclusivo. Exemplos de interseções e tipos de célula correspondentes podem ser observados na figura 4.9 (a). A figura 4.9 (b) apresenta células com mais de um tipo de interseção, prevalecendo o mais conclusivo.

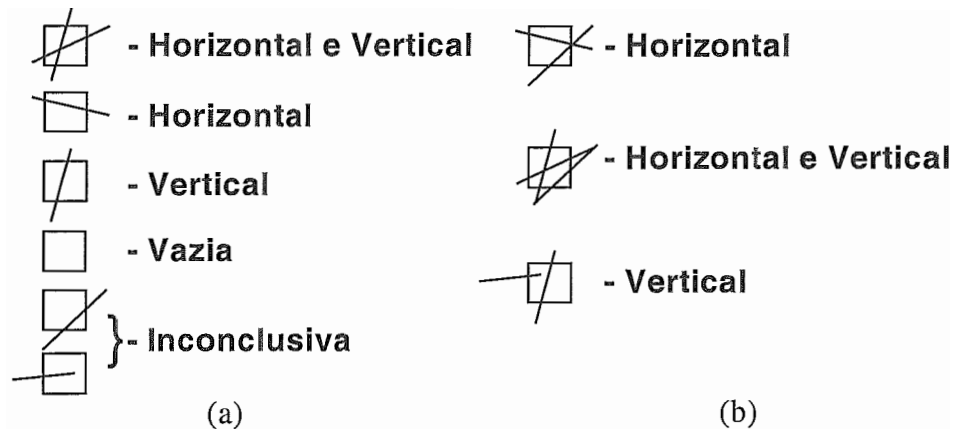


Figura 4.9 – (a) Exemplos de células e seus tipos correspondentes (b) Células com mais de um tipo de interseção prevalecendo o mais conclusivo.

Uma vez obtidos os tipos de célula podemos comparar as células correspondentes pertencentes a área de interseção dos MBRs. Para cada par de células sobrepostas teremos as possibilidades ilustradas na tabela 4.8.

Tabela 4.8 – Possíveis resultados da comparação entre duas células da aproximação 5CDRS

Tipo de célula × Interseção	Vazia	Inconclusiva	Vertical	Horizontal	Horizontal e Vertical
Vazia	Rejeitado	Rejeitado	Rejeitado	Rejeitado	Rejeitado
Inconclusiva	Rejeitado	<i>Inconclusivo</i>	<i>Inconclusivo</i>	<i>Inconclusivo</i>	<i>Inconclusivo</i>
Vertical	Rejeitado	<i>Inconclusivo</i>	<i>Inconclusivo</i>	<b>Aceito</b>	<b>Aceito</b>
Horizontal	Rejeitado	<i>Inconclusivo</i>	<b>Aceito</b>	<i>Inconclusivo</i>	<b>Aceito</b>
Horizontal e Vertical	Rejeitado	<i>Inconclusivo</i>	<b>Aceito</b>	<b>Aceito</b>	<b>Aceito</b>

Observe que se ocorre a sobreposição de uma célula horizontal com uma vertical chegamos a conclusão de que as polilinhas se interceptam. Isto ocorre porque uma das polilinhas entra na célula pela aresta superior e sai pela inferior (ou vice-versa) e a outra entra pela aresta da direita e sai pela da esquerda (ou vice-versa). Sendo assim, as polilinhas obrigatoriamente se interceptam em algum ponto no interior da célula.

Da mesma forma que a aproximação 4CRS, basta uma comparação resultar em *Aceito* para que a interseção seja detectada. Por outro lado, todas as células a serem comparadas devem resultar em *Rejeitado* para que o par candidato seja descartado.

### 4.2.3 Mudança de Escala

Esta seção apresenta o algoritmo de mudança de escala para a aproximação 5CDRS. Além de fazer com que as células passem a ter o mesmo tamanho, tornando possível a comparação (a sobreposição é garantida pelos algoritmos da seção 3.3), este algoritmo é responsável por determinar o tipo de interseção que cada célula possui.

Como não é possível subdividir as células com interseção sem cometer erros, a mudança de escala sempre será feita agrupando células da aproximação de maior resolução.

Cada célula da nova resolução será formada por agrupamentos de  $2^m$  células, o que é sempre possível graças a divisão do espaço adotada.

Os passos do algoritmo de mudança de escala são apresentados a seguir:

- 1) Obter o MBR- $2^k$  com o tamanho de célula desejado com base no MBR da polilinha, dando origem à nova grade;
- 2) Criar uma matriz com dimensões iguais a grade obtida. Esta será responsável por armazenar o tipo de cada célula;
- 3) Iniciar a matriz atribuindo o tipo Vazia para cada elemento;
- 4) Calcular a coordenada do canto inferior esquerdo da célula inicial na grade original. A coordenada obtida passa a ser o ponto atual;
- 5) Marcar a posição de entrada como inválida;
- 6) Descobrir a célula da nova grade que contém o ponto atual e marcá-la como a célula atual;
- 7) Para cada vetor de deslocamento da assinatura
  - i. Obter o próximo ponto através do deslocamento fornecido pelo vetor atual, o qual fornece a direção na qual deve ser acrescido ou subtraído o tamanho de uma célula da grade original ao ponto atual;
  - ii. Descobrir a célula da nova grade que contém o próximo ponto;
  - iii. Se a célula obtida for diferente da célula atual;
    1. Avaliar a situação da célula atual;
    2. A célula obtida passa a ser a célula atual;
    3. A posição de entrada é armazenada (vetor que causou a entrada na célula);
- 8) Avaliar a situação da célula atual.



Ao avaliar a situação de uma célula são levados em conta a sua situação atual, a posição de entrada e a posição de saída. A posição de saída é dada pelo vetor que causou a saída da célula. Se as posições de entrada e saída forem direita e esquerda ou vice-versa, uma interseção horizontal é assinalada. Se as posições de entrada e saída forem acima e abaixo, uma interseção vertical é assinalada. Em todos os outros casos uma interseção inconclusiva é assinalada. A interseção assinalada e a situação atual da célula são avaliadas e a nova situação é armazenada. As possíveis combinações são apresentadas na tabela 4.9.

Tabela 4.9 – Avaliação da situação atual de uma célula e um novo tipo de interseção

Situação atual	Interseção assinalada		
	Inconclusiva	Horizontal	Vertical
Vazia	Inconclusiva	Horizontal	Vertical
Inconclusiva	Inconclusiva	Horizontal	Vertical
Horizontal	Horizontal	Horizontal	Horizontal e Vertical
Vertical	Vertical	Horizontal e Vertical	Vertical
Horizontal e Vertical	Horizontal e Vertical	Horizontal e Vertical	Horizontal e Vertical

Note que o algoritmo apresentado também pode ser usado quando não há mudança de escala com o intuito de computar os tipos de células a partir da assinatura vetorial. Para tal, basta considerar a nova grade igual a grade original.

#### 4.2.4 Variantes

Uma variante da assinatura 5CDRS, a qual na verdade foi a sua precursora, é a aproximação RIOS<sup>1</sup> (ZIMBRÃO et al., 2001) ou Assinatura Raster com Posições de Entrada e Saída. A aproximação RIOS possui a possibilidade de subdividir cada aresta de uma célula em 2<sup>n</sup> partes, totalizando 4 x 2<sup>n</sup> posições possíveis para a entrada/saída da polilinha na célula. Este refinamento permite um reconhecimento mais eficiente de interseções entre polilinhas, as custas, porém, do armazenamento de mais informação.

---

<sup>1</sup> do inglês Raster Input / Output Signature

Podemos dizer que a aproximação 5CDRS corresponde à aproximação RIOS quando  $n = 0$ . Um exemplo de subdivisão de uma célula é ilustrado pela figura 4.10.

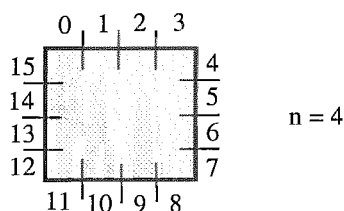


Figura 4.10 – Subdivisão de uma célula em  $2^4$  posições de Entrada/Saída.

Para o presente trabalho, não usamos a aproximação RIOS uma vez que as informações a respeito do ponto de entrada da polilinha em relação a uma das arestas da célula não é relevante para a comparação com a 4CRS. Estaríamos, portanto, obtendo os mesmos percentuais de reconhecimento de interseção que a 5CDRS e usando uma aproximação menos compacta. Maiores detalhes sobre a aproximação RIOS podem ser obtidos em AZEVEDO (2001).

#### 4.2.5 Resultados Experimentais (5CDRS)

Alguns resultados experimentais obtidos em AZEVEDO (2001) na execução de junções de conjuntos de polilinhas são apresentados a seguir. A tabela 4.10 apresenta os conjuntos de dados usados e as junções correspondentes. A tabela 4.11 apresenta os resultados.

Tabela 4.10 – Conjuntos e junções. AZEVEDO (2001)

Conjuntos		Junção
EUA	Rios x Estradas	A
EUA Nova Jersey	Ruas x Ferrovias	B
	Rios x Ferrovias	C
	Rios x Ruas	D
Canada	Rios x Estradas	E
Amazônia	Rios x Rodovias	F

Tabela 4.11 – Desempenho do filtro 5CDRS - 350 células. AZEVEDO (2001)

Junção	Aceitos (%)	Rejeitados (%)	Candidatos (%)	Aceitos Identificados (%)	Rejeitados Identificados (%)	Capacidade total de decisão (%)
A	1,94	47,14	50,91	9,58	59,15	49,09
B	0,00	75,87	24,13	0,00	78,49	75,87
C	1,53	59,49	38,98	8,21	73,12	61,02
D	0,00	77,17	22,83	0,00	82,26	77,17
E	2,07	44,83	53,10	9,09	58,04	46,90
F	3,64	65,90	30,45	19,13	81,41	69,55
Média	1,53	61,73	36,74	7,67	72,08	63,26

### 4.3 Algoritmo de Comparação entre as Aproximações 4CRS e 5CDRS

Se considerarmos que em uma base de dados espaciais as aproximações 4CRS e 5CDRS estão sendo utilizadas para representar respectivamente os polígonos e polilinhas, torna-se extremamente interessante aproveitar a existência destas aproximações para consultas que relacionem atributos de polígonos com polilinhas.

Como ambas as aproximações utilizam o mesmo esquema de divisão do espaço em células, podemos estabelecer uma tabela de comparação entre os tipos de célula de cada uma. A mudança de escala será feita com o algoritmo correspondente de cada aproximação, sendo sempre alterada a aproximação de maior resolução de forma a obter células de mesmo tamanho que a aproximação de menor resolução. Para cada par de células sobrepostas na área de interseção dos MBRs teremos as possibilidades ilustradas na tabela 4.12.

Tabela 4.12 – Possíveis resultados da comparação entre células das aproximações 4CRS e 5CDRS

Tipo de célula × Interseção	Vazia	Pouca Interseção	Muita Interseção	Cheia
<b>Vazia</b>	<b>Rejeitado</b>	<b>Rejeitado</b>	<b>Rejeitado</b>	<b>Rejeitado</b>
<b>Inconclusiva</b>	<b>Rejeitado</b>	<i>Inconclusivo</i>	<i>Inconclusivo</i>	<b>Aceito</b>
<b>Horizontal</b>	<b>Rejeitado</b>	<i>Inconclusivo</i>	<i>Inconclusivo</i>	<b>Aceito</b>
<b>Vertical</b>	<b>Rejeitado</b>	<i>Inconclusivo</i>	<i>Inconclusivo</i>	<b>Aceito</b>
<b>Horizontal e Vertical</b>	<b>Rejeitado</b>	<i>Inconclusivo</i>	<i>Inconclusivo</i>	<b>Aceito</b>

## 5 Resultados Experimentais

Apresentamos aqui os resultados obtidos nos testes de junção entre polígonos e polilinhas usando filtros raster. As aproximações raster usadas para desempenhar o passo de filtragem geométrica foram a 5CDRS e 4CRS, apresentadas no capítulo anterior, para polilinhas e polígonos respectivamente.

Este capítulo está estruturado da seguinte forma: a seção 5.1 descreve o ambiente de teste usado. A seção 5.2 apresenta os dados e suas características. A seção 5.3 apresenta características das aproximações e das árvores geradas. Finalmente, os resultados obtidos a partir da execução de junções são apresentados na seção 5.4.

### 5.1 Ambiente de teste

Para a execução das junções e obtenção dos resultados apresentados neste capítulo foi implementado em C++, visando portabilidade, um módulo com os três passos do MSQP. A máquina de teste usada foi um Pentium II 350 Mhz com 128 Mb de memória.

No primeiro passo do MSQP implementamos uma variante da *Árvore R\** a fim de permitir o armazenamento de aproximações de tamanho variável. Os nós internos dessa variante não diferem em nada da *R\*-Tree* original. A adaptação ocorre apenas nas folhas. Ao invés de termos um número máximo de entradas passamos a ter um tamanho máximo de armazenamento por nó. O split, portanto, passa a ser executado em função da soma do tamanho das aproximações armazenadas e não mais em função da quantidade das mesmas. Na criação da árvore, o parâmetro de número de entradas por nó é substituído pelos tamanhos máximo e mínimo da aproximação (os quais devem ser respeitados no momento da geração da mesma). Os números máximo e mínimo de entradas, usados pelos algoritmos da *Árvore R\**, são calculados em função desses tamanhos. Para armazenar entradas de tamanho fixo, basta configurar os tamanhos máximo e mínimo como iguais. Desta forma, voltamos as características da *Árvore R\** original.

No segundo passo usamos as aproximações 5CDRS e 4CRS armazenadas na árvore, onde a primeira possui tamanho variável e a segunda tamanho fixo.

O terceiro passo foi implementado através de dois algoritmos executados em sequência. O primeiro é um *scan line*, apresentado no capítulo 2, para detecção de interseção da polilinha com o contorno do polígono. O segundo efetua um teste para verificar se a polilinha está contida no polígono. Este último consiste em três etapas: primeiro, um segmento unindo um ponto qualquer da polilinha a um ponto qualquer do MBR do polígono é traçado; em seguida, o número de interseções do segmento com o contorno do polígono é computado; por último, se o número de interseções for ímpar, o polígono contém a polilinha, caso contrário não contém.

O módulo de execução de junções permite indicar se os passos 2 e 3 serão executados ou não. Esta flexibilidade dá origem a diversas variantes na execução da junção, o que permite tirar algumas conclusões, apresentadas na seção 5.4.

Um módulo para visualização dos dados e suas assinaturas também foi implementado, a fim de facilitar a validação e entendimento dos resultados obtidos.

## 5.2 Dados de teste

Nossos dados de teste correspondem a vários conjuntos de polígonos e polilinhas. Todos os conjuntos correspondem a dados reais representando rios, estradas, municípios, condados, áreas de desflorestamento e vegetação. Os anexos A, B e C apresentam telas do módulo de visualização exibindo estes dados. A complexidade e volume dos dados dos conjuntos é bem variada. Algumas características desses conjuntos são apresentadas nas tabelas 5.1 e 5.2.

Tabela 5.1 - Características dos conjuntos de dados de polígonos

Conjuntos de dados		Número de polígonos	Número de vértices	Média de vértices
EUA	Condados	3272	80573	25
Amazônia	Municípios	857	404699	472
	Vegetação	7602	493404	65
	Desflorestamento	6033	168982	28

Os dados da Amazônia foram cedidos pelo IBAMA enquanto que os dos EUA foram fornecidos pela ESRI (empresa responsável pelo desenvolvimento de softwares como o ArcView e o ArcInfo). Os dados foram originalmente obtidos em formato *shape* (de propriedade da ESRI) e convertidos para um arquivo contendo os objetos e as coordenadas de seus pontos. O arquivo obtido é usado para a geração das assinaturas e como repositório da representação real dos objetos necessária a execução do terceiro passo do MSQP.

Tabela 5.2 - Características dos conjuntos de dados de polilinhas

Conjuntos de dados		Número de polilinhas	Número de vértices	Média de vértices
EUA	Rios	123	6373	52
	Estradas	634	11334	18
Amazônia	Rios	12827	312858	24
	Estradas	1601	12481	8

Um fato importante a ser observado é que alguns tipos de dados apresentam uma certa correlação. Muitas fronteiras de municípios são definidas pelo curso de rios. Este fato é desfavorável para as aproximações empregadas uma vez que a interseção entre os objetos ocorre justamente na área de indecisão. Apesar disso, os resultados obtidos foram bons até mesmo para dados com estas características como pode ser constatado na seção 5.4.

### 5.3 Aproximações e Árvores

A partir de cada um dos conjuntos de dados apresentados na seção anterior foi gerado um arquivo de aproximações. O tamanho deste arquivo em relação ao respectivo conjunto nos fornece uma noção de quão compacta é a assinatura. Os tamanhos desses arquivos são apresentados nas tabelas 5.3 e 5.4. Para a aproximação 5CDRS o tempo de geração também é apresentado. O tempo de geração da 4CRS não foi medido uma vez que em ZIMBRÃO (1999) a sua eficácia foi avaliada obtendo resultados positivos.

Tabela 5.3 - Tamanho dos arquivos de aproximações e representação real dos conjuntos de polígonos

Conjuntos de dados		Base (Kb)	4CRS (Kb)
EUA	Condados	1535	358
	Municípios	7448	94
Amazônia	Vegetação	9318	832
	Desflorestamento	3315	660

Tabela 5.4 - Tamanho dos arquivos de aproximações e representação real dos conjuntos de polilinhas acompanhado do tempo de geração das aproximações

Conjuntos de dados		Base (Kb)	5CDRS	
			(Kb)	(Seg)
EUA	Rios	115	6	0,15
	Estradas	213	26	0,37
Amazônia	Rios	6202	501	9,3
	Estradas	287	64	0,6

Para cada arquivo de aproximações gerado, foram criadas duas árvores R\*: uma com as aproximações e outra sem. A árvore com as aproximações é usada quando desejamos executar o segundo passo do MSQP. Não executando o segundo passo, a árvore sem aproximações, contendo apenas a identificação e MBR do objeto, é usada. Com esta medida estamos evitando acessos a disco desnecessários na execução sem o segundo passo, o que seria favorável a nossa abordagem. Os tamanhos das árvores são apresentados na tabela 5.5.

Tabela 5.5 - Tamanho das árvores R\* com e sem aproximação. O valor percentual representa a razão dos tamanhos sem aproximação e com aproximação

Conjuntos de dados		Árvore com aproximação (Kb)	Árvore sem aproximação (Kb)	%
EUA	Condados	534	144	27
	Rios	12	10	83
	Estradas	44	32	73
Amazônia	Municípios	136	40	29
	Vegetação	1318	352	27
	Desflorestamento	1006	280	28
	Rios	790	568	72
	Estradas	102	76	74



## 5.4 Resultados da execução de junções

Com o intuito de avaliar a eficiência das aproximações raster ao desempenhar o papel de filtros geométricos no segundo passo do MSQP, foram executadas uma série de junções envolvendo os dados e as estruturas descritas nas seções anteriores. A tabela 5.6 apresenta os conjuntos relacionados a cada uma das junções executadas.

Tabela 5.6 - Conjuntos e junções avaliados

	Conjuntos	Junção
EUA	Condados X Rios	A
	Condados X Estradas	B
Amazônia	Municípios X Rios	C
	Municípios X Estradas	D
	Vegetação X Rios	E
	Vegetação X Estradas	F
	Desflorestamento X Rios	G
	Desflorestamento X Estradas	H

Os objetos cujos MBRs se interceptam não podem ser descartados no primeiro passo e, portanto, seguem para o passo seguinte do MSQP. Este conjunto deve ser particionado em dois subconjuntos: um com os pares contendo interseção e outro com os pares que não se interceptam. O perfil das junções executadas é apresentado na tabela 5.7, onde a parcela de pares contendo interseções é apresentada. O restante corresponde à parcela de pares que não se interceptam.

Tabela 5.7 - Perfil das junções avaliadas

Junção	Número de pares candidatos provenientes do passo 1	Pares com interseção	
		Nº	%
A	3896	1189	30,52
B	4768	2345	49,18
C	34782	19746	56,77
D	5470	2847	52,05
E	55163	20205	36,63
F	9279	4236	45,65
G	15485	5498	35,51
H	6119	2724	44,52

As aproximações usadas nesses experimentos foram geradas com uma grade de no máximo 350 células. Optamos por este valor por apresentar em ZIMBRÃO (1999) um resultado extremamente positivo e ao mesmo tempo originar aproximações compactas. O tamanho máximo da aproximação 5CDRS foi fixado em 95 bytes, o que permite um número máximo de 256 vetores de deslocamento. A variação desses limites corresponde a um trabalho futuro desta tese.

A tabela 5.8 apresenta os resultados da aplicação dos filtros raster nos conjuntos de pares candidatos da tabela 5.7, os quais correspondem aos conjuntos de entrada para o passo 2. A coluna *Aceitos* indica a parcela dos pares de entrada que tem a sua interseção reconhecida pelo filtro. Os *Rejeitados* correspondem à parcela dos pares de entrada cuja não interseção é descoberta pelo filtro. O restante, que obteve um teste inconclusivo, é representado na coluna *Candidatos* e corresponde aos pares que terão as suas representações reais comparadas. As duas últimas colunas correspondem aos percentuais de interseções e não interseções reconhecidos pelo filtro em relação ao total existente no conjunto de entrada.

Tabela 5.8 - Desempenhos dos filtros raster

Junção	Aceitos (%)	Rejeitados (%)	Candidatos (%)	Aceitos Identificados (%)	Rejeitados Identificados (%)
A	5,56	56,44	38,00	18,25	81,23
B	38,42	38,65	22,93	78,12	76,06
C	10,27	22,72	67,01	18,10	52,56
D	37,71	32,76	29,53	72,46	68,32
E	13,04	38,57	48,39	35,62	60,87
F	21,53	38,04	40,43	47,17	70,00
G	13,67	41,65	44,68	38,52	64,59
H	21,55	40,03	38,42	48,42	72,16
Média	20,22	38,61	41,17	44,58	68,22

Os resultados acima indicam que, em média, a aplicação de filtros raster reduz em torno de 58% (20,22 + 38,61) o número de testes de interseção exata, mesmo, como mencionado anteriormente, na presença de conjuntos de dados com características desfavoráveis. Esta média está acima dos 46% de redução correspondente à média dos filtros para polígonos propostos por BRINKHOFF et al. (1993b).

Para melhor quantificar o benefício do uso do filtro raster na junção de polilinhas e polígonos, comparamos a junção completa, ou seja, os três passos do MSQP, com a junção sem o filtro raster (não executando o passo 2). Primeiramente, medimos a distribuição do tempo total de execução entre os passos. As tabelas 5.9 e 5.10 apresentam esta distribuição para as diversas junções. A figuras 5.1 ilustra a distribuição média entre os passos. Estes resultados confirmam que a execução do primeiro e segundo passos são irrisórias comparando com o tempo gasto na execução do terceiro, como apontado em BRINKHOFF et al. (1994).

Tabela 5.9 - Distribuição do tempo de processamento na execução das junções executando os passos 1, 2 e 3 do MSQP

Junção	Passo 1 (%)	Passo 2 (%)	Passo 3 (%)
A	4,26	2,19	93,54
B	11,92	5,12	82,96
C	0,66	0,27	99,06
D	2,45	0,55	97,00
E	2,24	0,53	97,23
F	5,47	0,95	93,59
G	3,06	0,17	96,77
H	3,00	2,58	94,42
Média	4,13	1,55	94,32

Tabela 5.10 - Distribuição do tempo de processamento na execução das junções executando os passos 1 e 3 do MSQP

Junção	Passo 1 (%)	Passo 3 (%)
A	0,89	99,11
B	1,96	98,04
C	0,19	99,81
D	0,33	99,67
E	0,85	99,15
F	1,52	98,48
G	0,95	99,05
H	1,64	98,36
Média	1,04	98,96

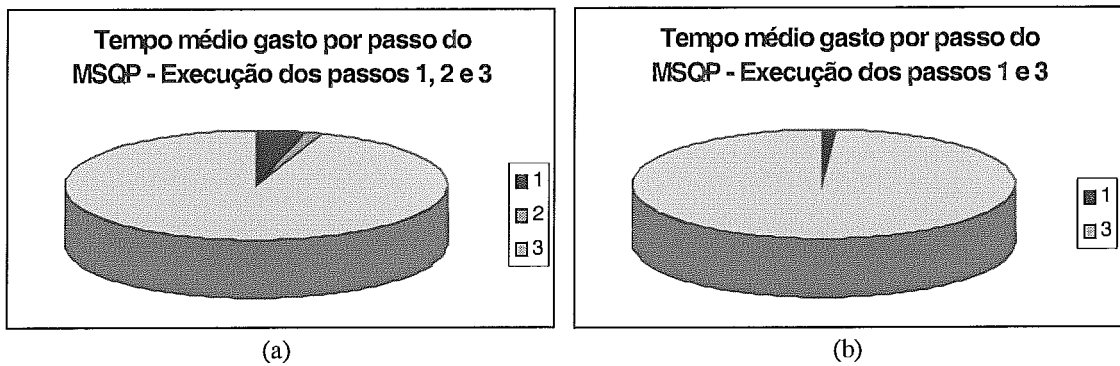


Figura 5.1 - Distribuição média do tempo de processamento na execução das junções executando os passos 1, 2 e 3 (a) e 1 e 3 (b) do MSQP.

É importante lembrar que a execução da junção sem o segundo passo usa uma árvore sem aproximações. Assim sendo, na execução da junção com os passos 1 e 3, o terceiro passo é responsável por uma fatia ainda maior de tempo devido a dois motivos: o conjunto de entrada para este passo é maior, correspondendo a todos os pares de objetos que possuem interseção entre seus MBRs, pois não há um filtro intermediário; e o primeiro passo consome um tempo menor, uma vez que a árvore tende a ser mais compacta necessitando de um número menor de acessos a disco para percorrê-la.

Os tempos referentes a cada um dos passos foi medido através da diferença de duração entre a execução com o passo e a execução com os passos antecessores ao mesmo. Por exemplo, o tempo de execução referente ao terceiro passo corresponde ao tempo de execução dos três passos menos o tempo de execução dos passos 1 e 2. Já o tempo de execução do segundo passo corresponde à duração da execução dos passos 1 e 2 menos o tempo de execução do passo 1. Essa forma de medição é necessária porque os passos são executados de forma intercalada e não de forma sequencial.

Os tempos de execução das junções de teste com os três passos e somente com o primeiro e terceiro são comparados na tabela 5.11. A figura 5.2 ilustra as diferenças.

Tabela 5.11 - Tempo total de execução das junções

Junção	Tempo total de execução (Seg)	
	Passos 1, 2 e 3	Passos 1 e 3
A	23,7	27,1
B	12,5	28,5
C	807,3	1354,5
D	47,0	137,4
E	752,3	1415,8
F	92,0	241,7
G	721,3	1048,1
H	143,2	241,7

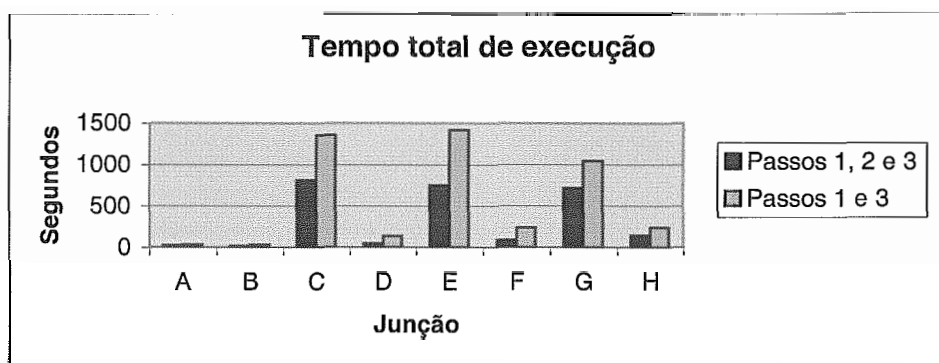


Figura 5.2 - Comparação entre os tempo totais de execução

Porém, medir apenas os tempos de execução não fornece uma medida precisa para avaliar o ganho real devido ao uso do filtro raster no segundo passo do MSQP. Isto ocorre porque o terceiro passo pode ser desempenhado por outros algoritmos com os quais não teríamos condições de afirmar que os resultados de tempo de execução permanecem válidos. Como exemplos temos a *Árvore R\** de trapézios e a detecção simbólica de interseção, apresentada em HUANG (1997), as quais podem substituir com vantagens um algoritmo padrão de *plane-sweep*.

Medimos então o número de testes de interseção exata efetuados e o número de acessos a disco. A redução desses números pelo uso do filtro raster permite avaliar o ganho real obtido no processamento da junção. Qualquer que seja o algoritmo empregado no terceiro passo, se reduzirmos seu conjunto de entrada estaremos diminuindo de forma

equivalente o seu tempo de execução. Por outro lado, ao reduzir o conjunto de entrada para o terceiro passo também estamos diminuindo a necessidade de recuperar as representações reais dos objetos, o que é refletido diretamente no número de acessos a disco. As tabelas 5.12 e 5.13 e as figuras 5.3 e 5.4 mostram a redução tanto do número de testes de interseção exata quanto do número de acessos a disco.

Tabela 5.12 - Número de testes de interseção exata das junções

Junção	Número de testes de interseção exata	
	1, 2 e 3	1 e 3
A	1480	3896
B	1093	4768
C	23304	34782
D	1615	5470
E	26685	55163
F	3751	6119
G	6916	15485
H	2350	6119

Tabela 5.13 - Número total de acessos a disco das junções

Junção	Número total de acessos a disco	
	1, 2 e 3	1 e 3
A	4731	10410
B	4653	14623
C	226617	341471
D	12881	36581
E	202532	343451
F	26387	37046
G	131790	146456
H	25118	37046

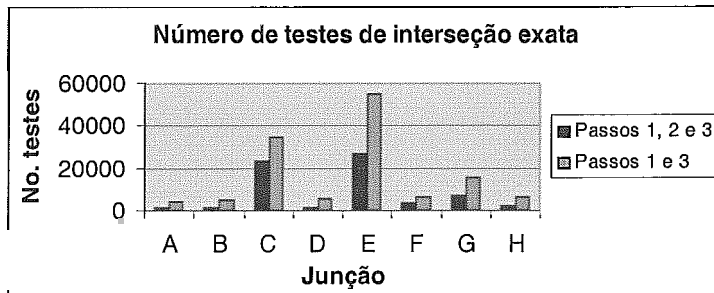


Figura 5.3 - Comparação entre os números de teste de interseção exata

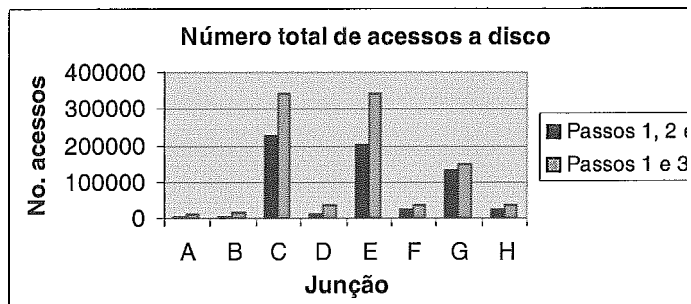


Figura 5.4 - Comparação entre os totais de acesso a disco

## 6 Conclusões

Este capítulo destina-se ao relato das conclusões obtidas durante a realização deste trabalho. Nosso principal objetivo foi propor uma aproximação raster para polilinhas, a 5CDRS (desenvolvida em conjunto com AZEVEDO (2001)), e avaliar a sua eficiência na execução de junções com polígonos representados pela aproximação raster 4CRS.

A seção 6.1 apresenta as considerações finais sobre as contribuições desta tese. A seção 6.2 é dedicada as direções para continuidade do trabalho.

### 6.1 Contribuições da tese

A avaliação e constatação da eficiência do uso de filtros raster no processamento de junções espaciais foram os objetivos que guiaram este trabalho. Os níveis médios de redução do número de testes de interseção exata alcançaram 58%, o que afetou significativamente o tempo total de execução. Consideramos estes níveis satisfatórios, uma vez que a média de redução dos filtros para polígonos apresentados em BRINKHOFF (1993b) é de 46%.

Outra utilidade da aproximação 5CDRS é a possibilidade de apresentar uma visualização rápida e simplificada de um conjunto de polilinhas, fator extremamente importante para aplicações interativas.

Finalmente, o presente trabalho pode ser usado como um guia para o estudo do processamento de consultas espaciais, uma vez que apresenta e referencia várias estruturas espaciais responsáveis por diversas etapas do processo.

### 6.2 Trabalho futuros

Pretendemos avaliar os limites estabelecidos neste trabalho para o número máximo de células na grade e tamanho máximo da aproximação 5CDRS, com o intuito de estabelecer um ponto ideal entre custo e benefício ou diretrizes para a determinação dos mesmos.



O uso das aproximações raster para avaliação de outros tipos de predicados que não o de interseção é outro ponto a ser considerado.

A adaptação do módulo de execução de junções para permitir o uso de outras aproximações, assim como o seu acoplamento com o módulo de visualização promete gerar uma ferramenta muito útil para uso didático e a avaliação de novas propostas.

## 7 Bibliografia

- ABEL, D. J., 1985, “Some elemental operations on linear quadtrees for geographic information systems”, *Computer Journal*, v. 28, n. 1 (Feb), pp. 73-77.
- ABEL, D. J., SMITH, J. L., 1983, “A data structure and algorithm based on a linear key for a rectangle retrieval problem”, *Computer Vision, Graphics, and Image Processing*, v. 24, n. 1 (Oct.), pp. 1 – 13.
- ARGE, L., et al., 1998. “Scalable Sweeping Based Spatial Join”. In: *24th International Conference on Very Large Databases*, New York City, NY, USA, August, 1998.
- AZEVEDO, L. G., 2001, *Filtros Raster para Junções de Polilinhas*, Tese de M.Sc., COPPE/UFRJ, Rio de Janeiro, RJ, Brasil.
- BERCHTOLD, S., KEIM, D. A., KRIEGEL, H. P., 1996. “The X-Tree: An Index Structure for High-Dimensional Data”, In: *Proceedings of 22nd International Conference on Very Large Data Bases*, Bombay, India 1996.
- BRINKHOFF, T., KRIEGEL, H. P, SCHNEIDER, R., et al., 1994. “Multi-step Processing of Spatial Joins”. In: *Proceedings of the 1994 ACM-SIGMOD Intl. Conference*, Minneapolis, USA, May 1994.
- BRINKHOFF, T., KRIEGEL, H. P, SCHNEIDER, R., 1993b. “Comparison of Approximations of Complex Objects Used for Approximation-based Query Processing in Spatial Database Systems”. In: *Proceedings of 9th International Conference on Data Engineering*, Vienna, Austria, Mar 1993.

- BRINKHOFF, T., KRIEGEL, H. P., SEEGER, B., 1993a. "Efficient processing of Spatial Joins Using R-Trees". In: *Proceedings of the 1993 ACM-SIGMOD Intl. Conference*, Washington, DC, USA, May 1993.
- BRINKHOFF, T., KRIEGEL, H. P., SEEGER, B., 1996. "Parallel Processing of Spatial Joins Using R-Trees", In: *Proceedings of 12th International Conference on Data Engineering*, New Orleans, LA, USA 1996.
- DEWITT, D., PATEL, J. M., 1996. "Partition Based Spatial-Merge Join". In: *Proceedings of the 1996 ACM-SIGMOD Conference*, Montreal, Canada, June 1996.
- GORDON, S. R., et al., 1994. *Final Report on Status of Spatial/Map Databases*. Technical Report of Oak Ridge National Laboratory, June 1994.
- GUTTMAN, A., 1984. "R-Trees: A Dynamic Index Structure for Spatial Searching". In: *Proceedings of the ACM SIGMOD Intl. Conf on Management of Data*, Boston, MA, USA, May 1984.
- HUANG, Y. W., JONES, M. C., RUNDESTEINER, E. A., 1997. "Improving Spatial Intersect Joins Using Symbolic Intersect Detection". In: *Proceedings of the 5th International Symposium on Advances in Spatial Databases, SSD'97*, Berlin, Germany, July 1997.
- KEDEM, G., 1982, "The quad-CIF tree: a data structure for hierarchical on-line algorithms", In: *Proceedings of the Nineteenth Design Automation Conference*, pp. 352-357, Las Vegas, June.

- KNUTH, D. E., 1973, *The Art of Computer Programming*. 2 ed. New York, Boston, MA, USA.
- KOUDAS, N., SEVCIK, K., 1997. "Size Separation Spatial Join". In: *Proceedings of ACM SIGMOD International Conference on Management of Data*, Tucson, Arizona, USA, May 1997.
- MAMOULIS, N., PAPADIAS, D., 1999. "Integration of Spatial Join Algorithms for Joining Multiple Inputs". In: *Proceedings of the ACM SIGMOD Conference on the Management of Data*, Philadelphia, PA, USA, May 1999.
- MING, L. L., RAVISHANKAR, C. V. "Generating Seeded Trees From Data Sets". In: *Proceedings of the 4th International Symposium on Large Spatial Databases*, Portland, ME, USA, August, 1995.
- MING, L. L., RAVISHANKAR, C. V. "Spatial Hash-Joins". In: *Proceedings of the 1996 ACM SIGMOD Conference*, Montreal, Canada, June 1996.
- MONTEIRO, R. S., AZEVEDO, L. G., ZIMBRAO, G., SOUZA, J. M., 2001, *Polyline Join Evaluation Using Raster Approximation, ES-550/01*, Programa de Engenharia e Sistemas e Computação.
- MORTON, G. M., "A computer oriented geodetic data base and a new technique in file sequencing", IBM Ltd., Ottawa, Canada, 1966.
- ROGER, D. F., 1986. *Procedural Elements For Computer Graphic*. 1 ed, McGraw-Hill Book Company, NYC, 1986.

- SAMET, H., 1990. *The Design and Analysis of Spatial Data Structure*, 2 ed, NYC, Addison-Wesley Publishing Company, 1990.
- SWARCFITER, J. L., MARKENZON, L., 1994. *Estruturas de Dados e seus Algoritmos*, 2 ed, Rio de Janeiro, Livros Técnicos e Científicos Editora, 1994.
- SHAFFER, C. A., 1986, *Application of alternative quadtree representations*. Ph. D. dissertation, University of Maryland, College Park, MD, USA.
- SUTHERLAND, I. E., SPROULL, R. F., SCHUMACKER, 1974, “A characterization of ten hidden-surface algorithms”, *ACM Computing Surveys*, v. 6, n. 1 (Mar), pp. 1-55.
- WILF, J. M., 1981, “Chain code”, *Robotics Age*, v. 3, n. 2, pp. 12-18.
- ZIMBRÃO, G., 1999, *Avaliação de Junções em Bancos de Dados Espaciais*, Tese de D.Sc., COPPE/UFRJ, Rio de Janeiro, RJ, Brasil.
- ZIMBRÃO, G., SOUZA, J. M., 1998. “A Raster Approximation For Processing of Spatial Joins”. In: *Proceedings of the 24th International Conference on Very Large Databases, VLDB'98*, NYC, USA, August 1998.
- ZIMBRÃO, G., SOUZA, J. M., 1997. “Realização Eficiente de Junções Espaciais Utilizando Hash-Join”. In: *Anais do XII Simpósio Brasileiro de Banco de Dados*, Fortaleza, CE, Brasil, Outubro, 1997.

ZIMBRÃO, G., SOUZA, J. M., MONTEIRO, R. S., AZEVEDO, L. G., 2000. “Filtro Raster para Junção de Polilinhas”. In: *Anais do XV Simpósio Brasileiro de Banco de Dados*, João Pessoa, Brasil, Outubro 2000.

ZIMBRÃO, G., SOUZA, J. M., MONTEIRO, R. S., AZEVEDO, L. G., 2001, *Filtro Raster para Junções de Polilinhas usando Posições de Entrada e Saída, ES-544/01*, Programa de Engenharia e Sistemas e Computação.

# Anexo A

Este anexo apresenta os dados da Amazônia (cedidos pelo IBAMA) usados neste trabalho. São exibidas telas do módulo de visualização desenvolvido ao longo desta tese em conjunto com AZEVEDO (2001).

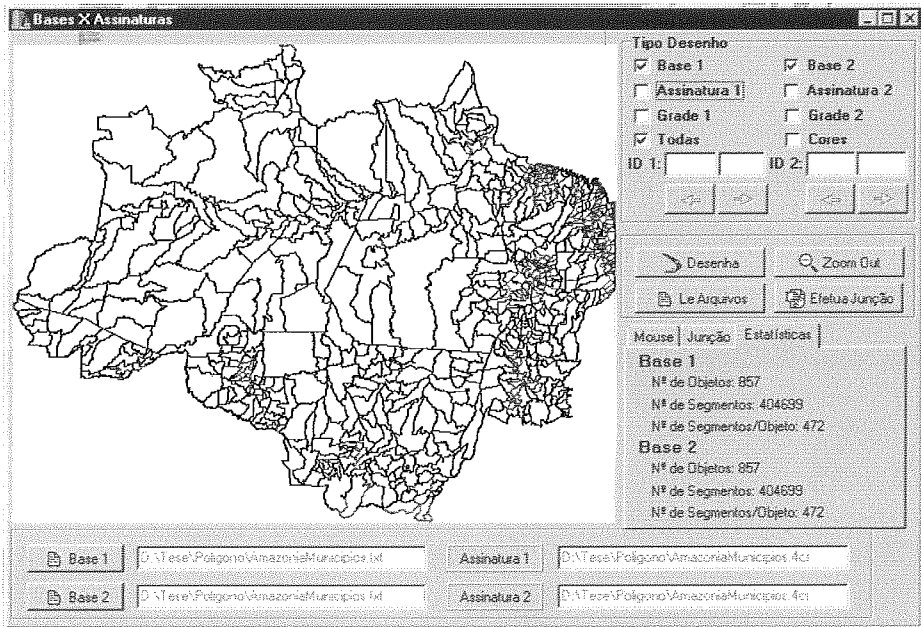


Figura A.1 - Municípios da Amazônia.

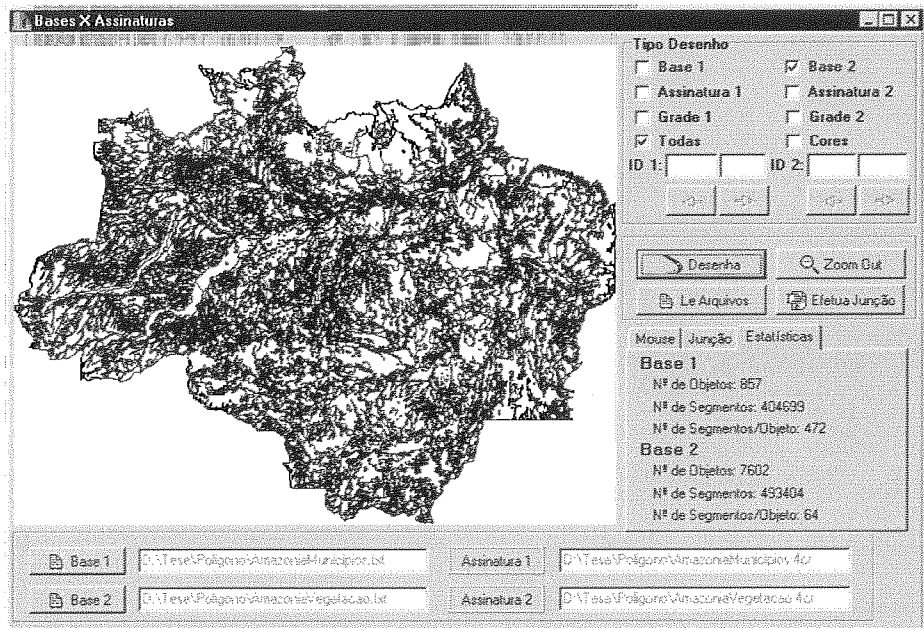


Figura A.2 - Vegetação da Amazônia.

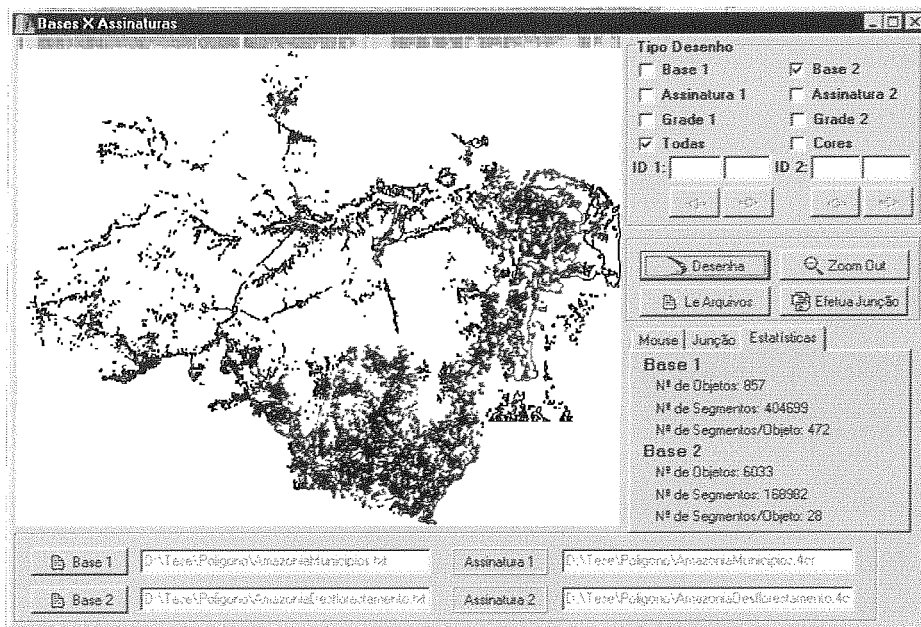


Figura A.3 – Áreas de Desflorestamento da Amazônia.

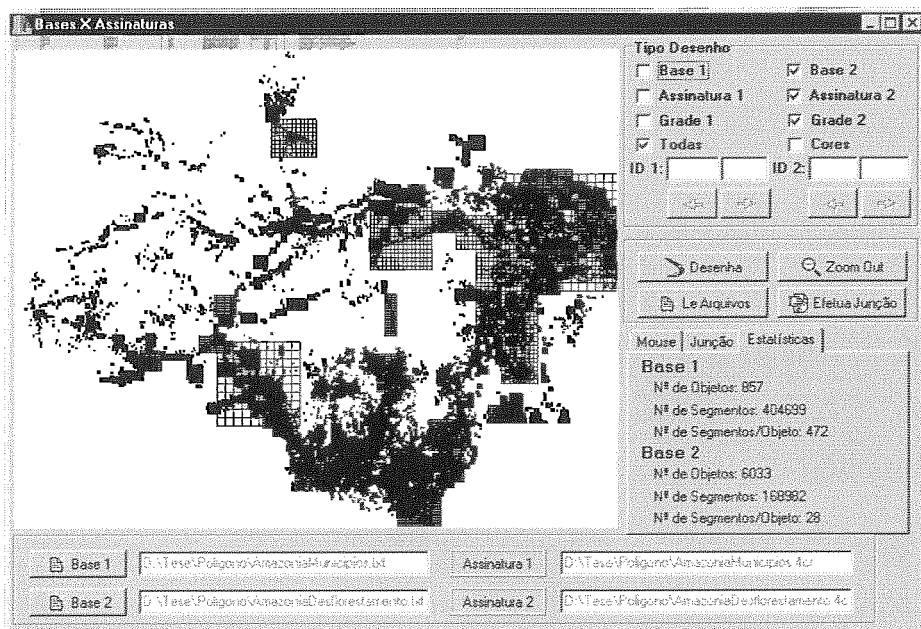


Figura A.4 – Áreas de Desflorestamento da Amazônia com apresentação da aproximação 4CRS e grade.



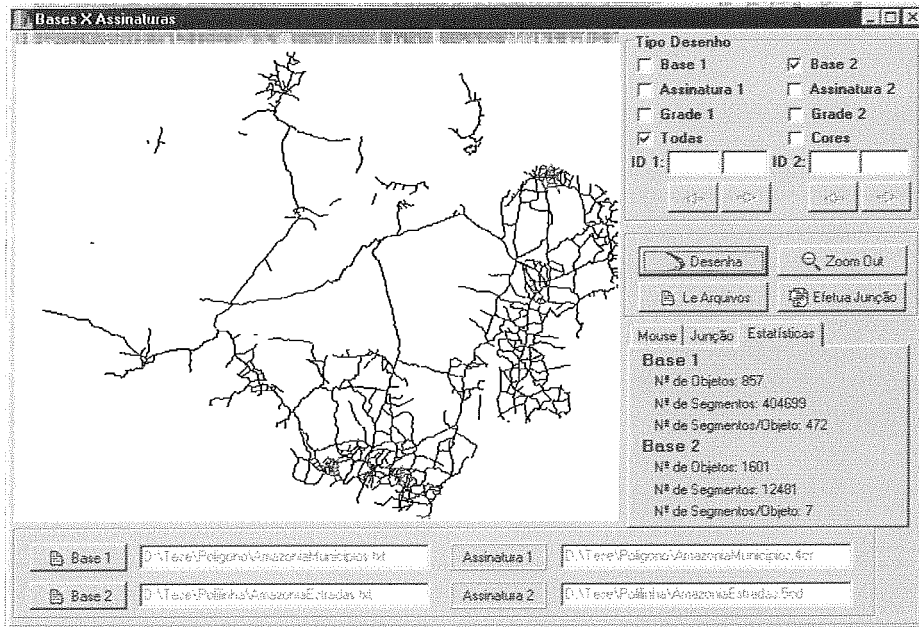


Figura A.5 – Estradas da Amazônia.

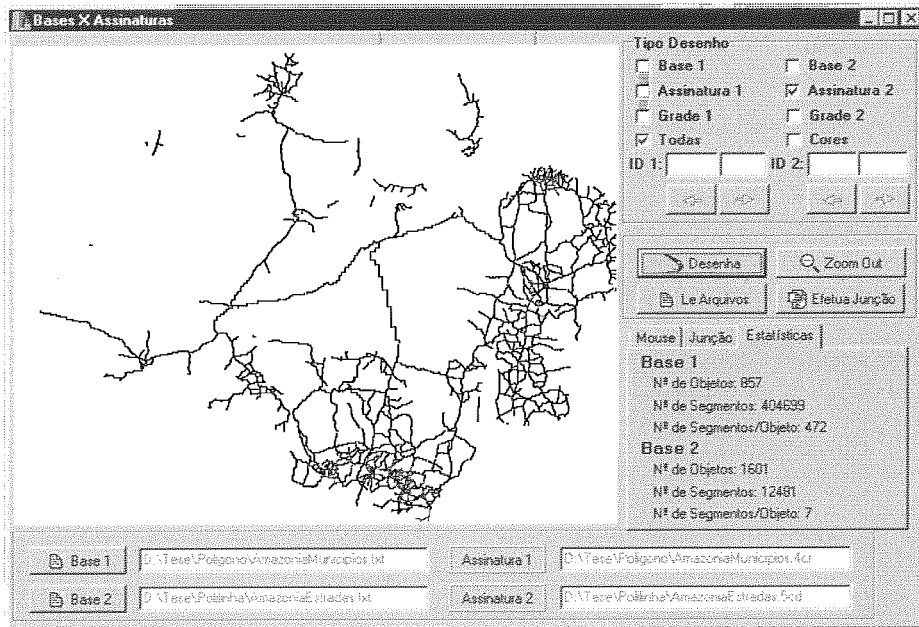


Figura A.6 – Aproximação 5CDRS das Estradas da Amazônia.

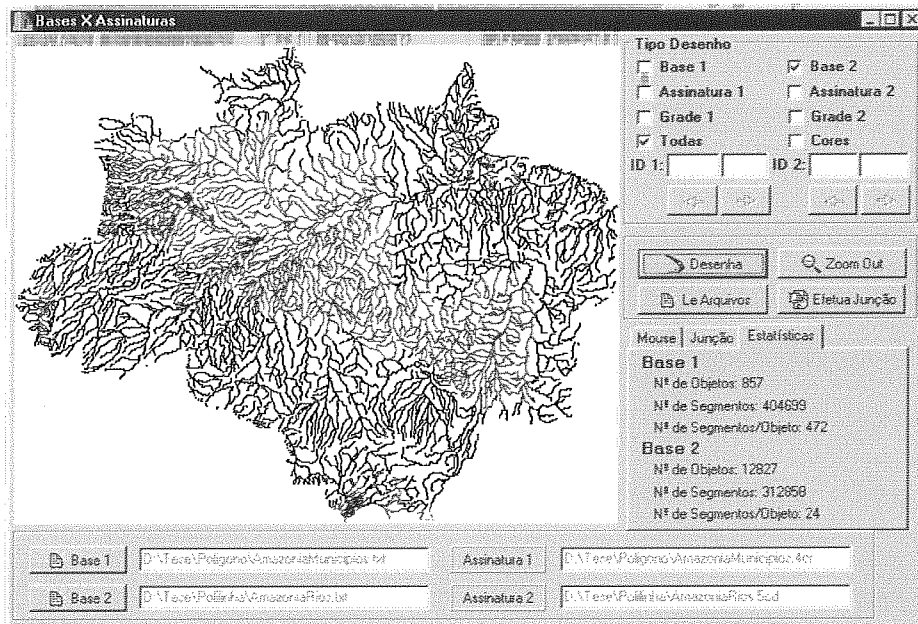


Figura A.7 – Rios da Amazônia.

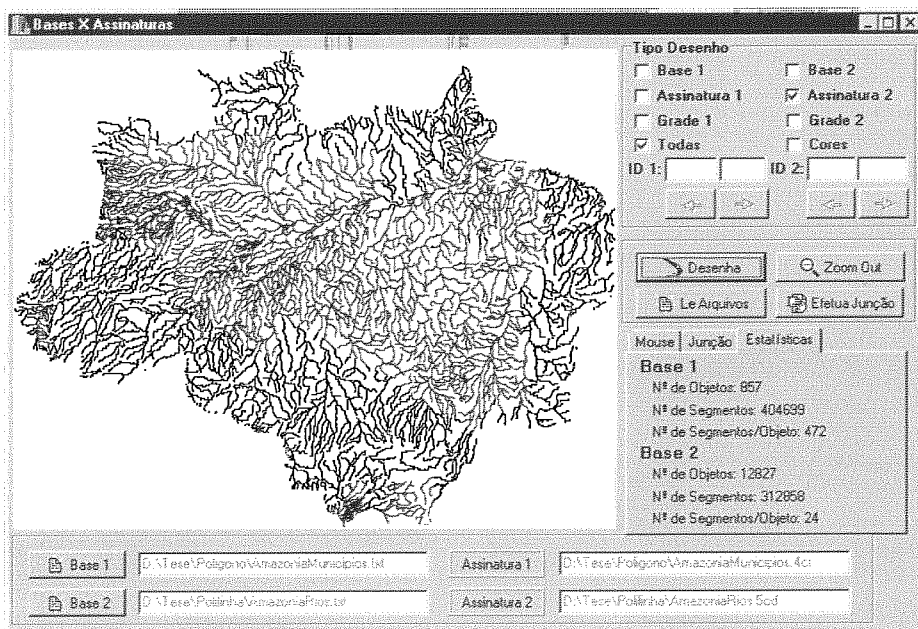


Figura A.8 – Aproximação 5CDRS dos Rios da Amazônia.

## Anexo B

Este anexo apresenta os dados dos Estados Unidos (fornecidos pela ESRI) usados neste trabalho. São exibidas telas do módulo de visualização desenvolvido ao longo desta tese em conjunto com AZEVEDO (2001).

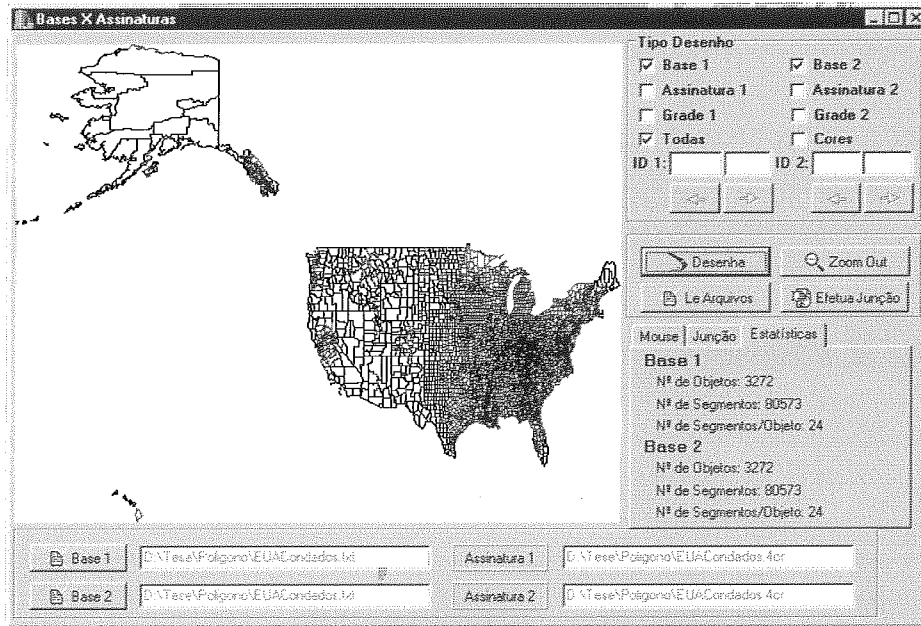


Figura B.1 – Condados dos Estados Unidos.

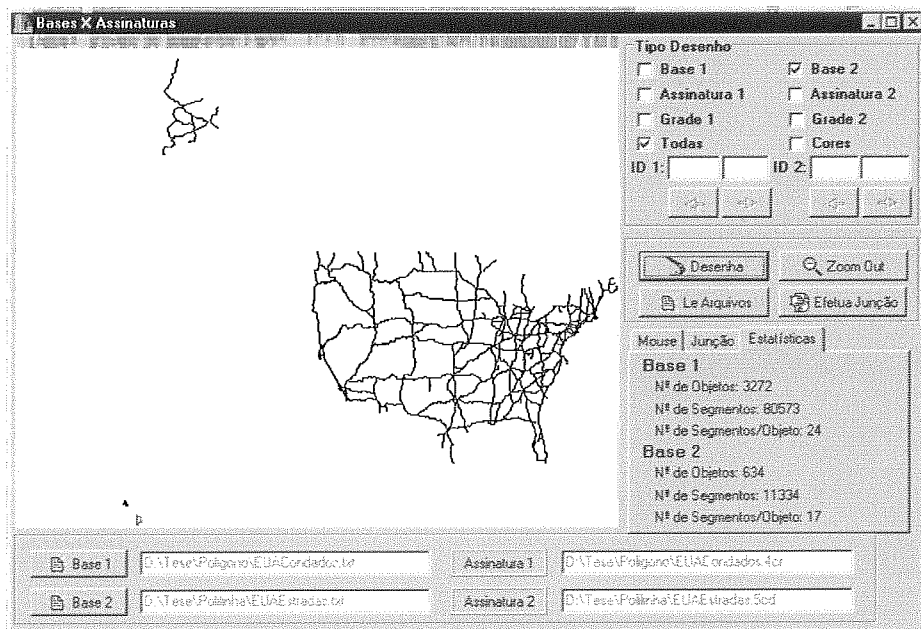


Figura B.2 – Estradas dos Estados Unidos.

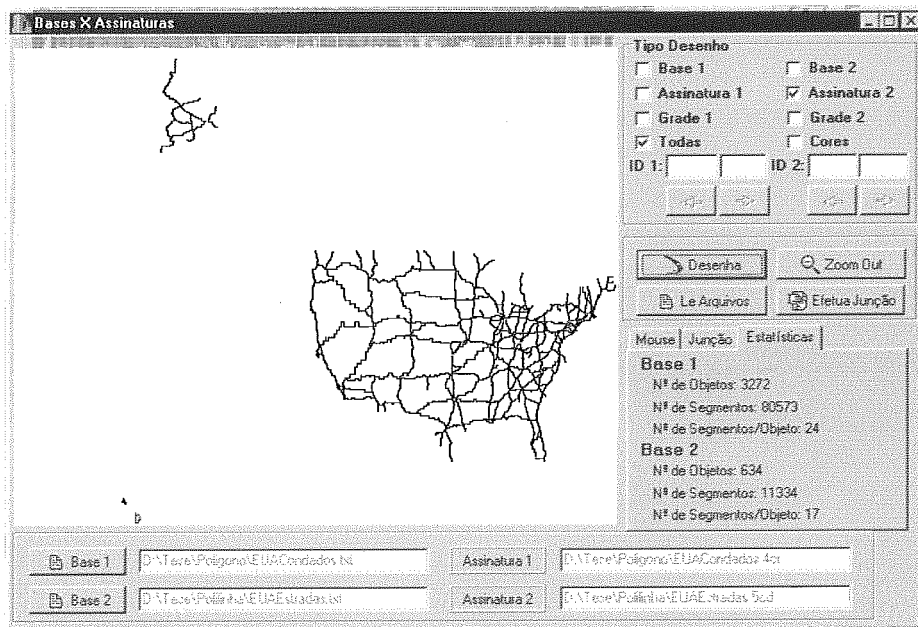


Figura B.3 – Aproximação 5CDRS das Estradas dos Estados Unidos.

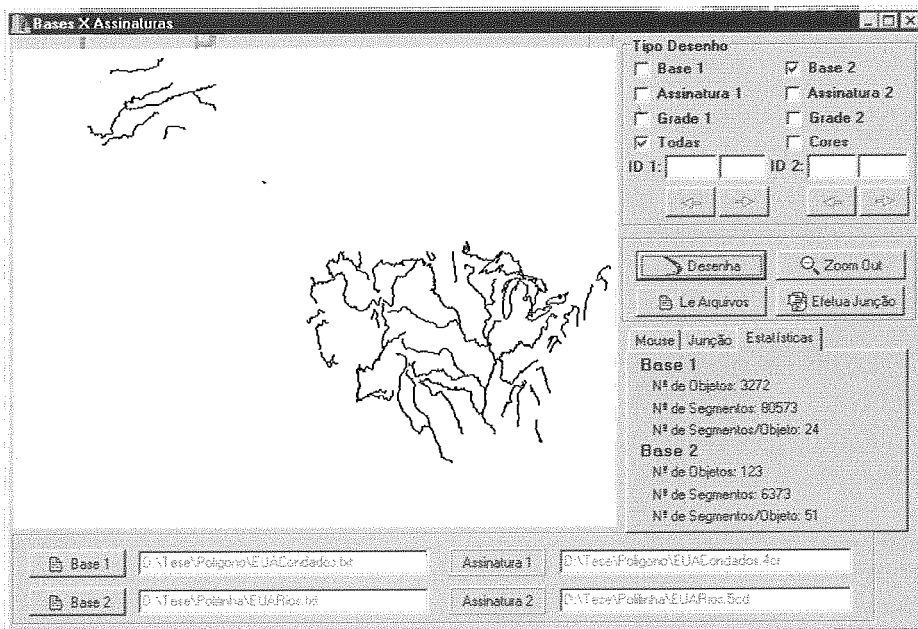


Figura B.4 – Rios dos Estados Unidos.

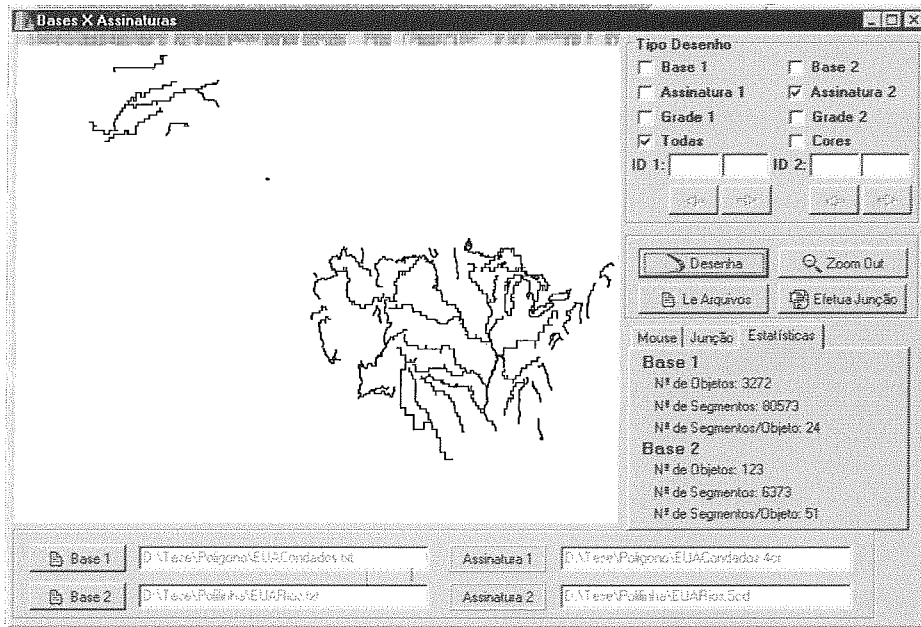


Figura B.5 – Aproximação 5CDRS dos Rios dos Estados Unidos.

## Anexo C

Este anexo apresenta a visualização das aproximações de forma mais detalhada. As funcionalidades de zoom e seleção de objeto do módulo de visualização foram usadas para obter as telas abaixo.

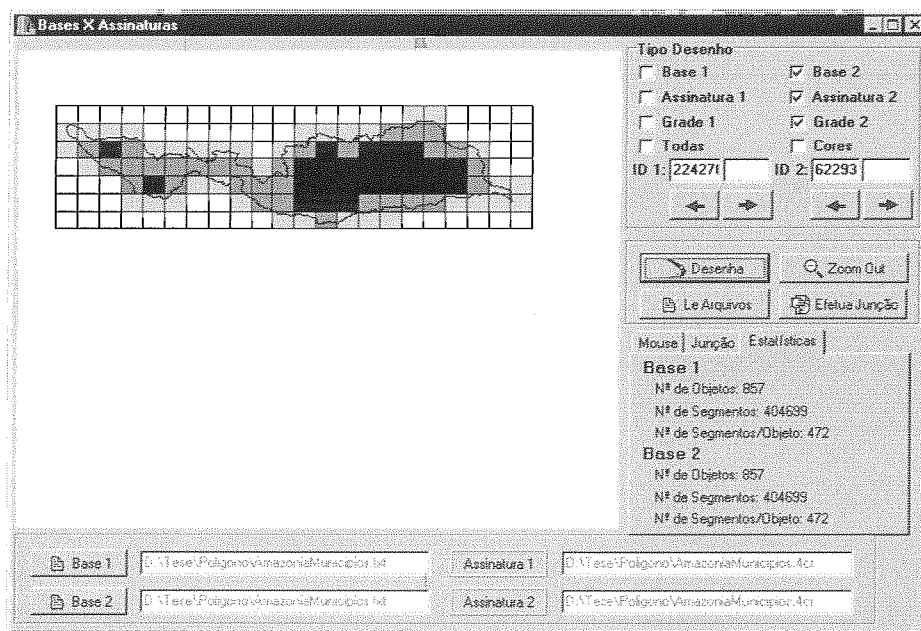


Figura C.1 – Polígono com aproximação 4CRS e grade.

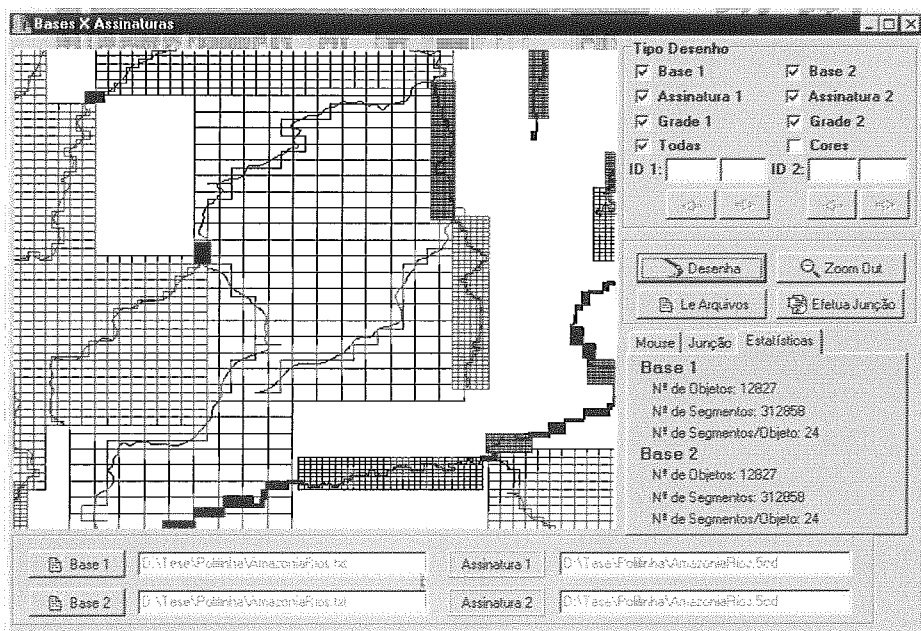


Figura C.2 – Polilinhas com aproximação 5CDRS e grade.