



MINISTÉRIO DA CIÊNCIA E TECNOLOGIA

**INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS**

INPE-7250-TDI/694

**UMA APLICAÇÃO DA BUSCA TABU AO PROBLEMA DE  
ROTULAÇÃO CARTOGRÁFICA DE PONTOS**

Missae Yamamoto

Dissertação de Mestrado em Computação Aplicada, orientada pelo Dr. Luiz Antônio Nogueira Lorena e pelo Dr. Gilberto Câmara Neto, aprovada em 18 de dezembro de 1998.

INPE  
São José dos Campos  
1999

**Publicado por:**

**Coordenação de Ensino, Documentação e  
Programas Especiais - CEP**

**Instituto Nacional de Pesquisas Espaciais - INPE**

**Caixa Postal 515**

**12201-970 - São José dos Campos - SP - Brasil**

**Fone: (012) 345.6911**

**Fax: (012) 345.6919**

**E-Mail: [marciana@sid.inpe.br](mailto:marciana@sid.inpe.br)**

- Solicita-se intercâmbio
- We ask for exchange
- Si sollecita intercambio
- On demande l'échange
- Mann bittet un Austausch
- Pidese canje
- Просим обмен
- 歡迎者交換
- 出使物交換の不便

**Publicação Externa - É permitida sua reprodução para interessados.**

MINISTÉRIO DA CIÊNCIA E TECNOLOGIA  
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

**INPE-7250-TDI/694**

**UMA APLICAÇÃO DA BUSCA TABU AO PROBLEMA DE  
ROTULAÇÃO CARTOGRÁFICA DE PONTOS**

Missae Yamamoto

Dissertação de Mestrado em Computação Aplicada, orientada pelo Dr. Luiz Antônio Nogueira Lorena e pelo Dr. Gilberto Câmara Neto, aprovada em 18 de dezembro de 1998.

INPE  
São José dos Campos  
1999

519.863:528.711.7

YAMAMOTO, M.

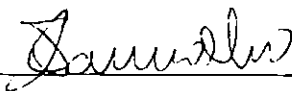
Uma aplicação da busca tabu ao problema de rotulação cartográfica de pontos / M. Yamamoto. – São José dos Campos: INPE, 1998.

132p. – (INPE-7250-TDI/694).

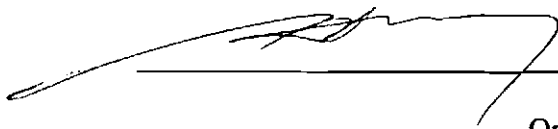
1. Busca tabu. 2. Rotulação cartográfica. 3. Algoritmo de otimização. 4. Produção cartográfica automatizada. 5. Sistemas de informação geográfica (SIG). I. Título.

Aprovado pela Banca Examinadora em  
cumprimento a requisito exigido para a  
obtenção do Título de Mestre  
em **Computação Aplicada**.

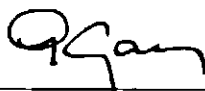
Dr. Solon Venâncio de Carvalho

  
\_\_\_\_\_  
Presidente

Dr. Luiz Antonio Nogueira Lorena

  
\_\_\_\_\_  
Orientador


Dr. Gilberto Câmara Neto

  
\_\_\_\_\_  
Orientador

Dr<sup>a</sup> Cintia Rigão Srich

  
\_\_\_\_\_  
Membro da Banca

Dr. Paulo Cezar Pinto Carvalho

  
\_\_\_\_\_  
Membro da Banca  
Convidado

Candidato(a): Missae Yamamoto



*Aos meus pais Jusaku e Tereza  
e sobrinhos Camila, Marcelo e Hitoshi*





## AGRADECIMENTOS

Ao orientador Dr. Luiz Antônio Nogueira Lorena, meu professor da graduação e do mestrado, uma pessoa muito competente, que esteve presente nas fases deste trabalho orientando, incentivando, apoiando, revisando a parte escrita e contribuindo com sugestões, sempre com profissionalismo e paciência. Espero estar à altura da atenção que sempre me dedicou.

Ao orientador Dr. Gilberto Câmara Neto, uma pessoa muito competente e conhecedor de SIG, que orientou, revisou a parte escrita, contribuiu com sugestões, incentivou e apoiou, e que, em muito influenciou minha carreira desde 1988, quando então, comecei a fazer parte da equipe e dos projetos, dos quais ele foi o gerente.

Ao Ubirajara Moura de Freitas, chefe da Divisão de Processamento de Imagens, pela oportunidade de discussão na fase inicial do trabalho e incentivo.

Ao Guaraci José Erthal, colega e colaborador, que possui uma visão ampla da linguagem C++. Estimulou e ajudou a tornar o processamento da busca tabu mais rápido, além de contribuir com idéias e sugestões.

Ao amigo Fernando Yutaka Yamaguchi, que na última etapa da parte escrita do trabalho, veio em meu socorro. A dissertação não teria sido defendida antes do natal, sem os seus conhecimentos.

Ao Dr. Solon Venâncio de Carvalho, presidente da minha banca examinadora preliminar e final, além de revisar a dissertação e contribuir com sugestões, organizou uma palestra da Dra Cíntia Rigão Scrich sobre busca tabu, que em muito me auxiliou.

A Dra Cíntia Rigão Scrich, membro da banca examinadora final, que além de revisar pacientemente a dissertação, contribuiu com várias idéias e sugestões.

Ao Dr. Paulo Cezar Pinto Carvalho, membro externo da banca examinadora final e Dr. Raimundo Almeida Filho, membro da banca examinadora preliminar, que revisaram a dissertação e contribuíram com sugestões.

Ao Júlio César Lima D'Alge, cartógrafo e colega, que revisou a dissertação na fase preliminar e esclareceu dúvidas sobre cartografia.

Ao Dr. Carlos Ho Shih Ning, colega e professor de computação gráfica, pela oportunidade de discussão e incentivo.

A Roseli Yoshioka Misikami, Mary Minamoto Yamada, Rosa Maria Kato Shimabukuro e Lúbia Vinhas pela amizade, incentivo e apoio.

## RESUMO

A geração de configurações ótimas de rótulos em um mapa é um problema que surge com a produção cartográfica automatizada. O objetivo de uma boa rotulação é mostrar a posição geográfica das entidades com texto associado, de forma legível, e respeitando as convenções cartográficas, com qualidades estética e harmônica na apresentação dessas informações. Abordamos o problema de rotulação cartográfica sob o ponto de vista de otimização combinatória. Nesta perspectiva, a rotulação cartográfica pertence à classe de problemas de difícil solução, conduzindo-nos à necessidade de algoritmos de aproximação, uma vez que não existe nenhum algoritmo exato capaz de solucioná-lo em um intervalo de tempo aceitável. Nossa pesquisa consistiu na avaliação do algoritmo de otimização Busca Tabu aplicado à rotulação cartográfica. A Busca Tabu implementada no SCARTA, um software de produção cartográfico em desenvolvimento pela Divisão de Processamento de Imagens do Instituto Nacional de Pesquisas Espaciais, mostrou-se um algoritmo eficiente, nos casos-teste real e aleatório. Quando comparado a técnicas alternativas como “Simulated Annealing”, algoritmo genético com máscara, e outras descritas na literatura, a Busca Tabu apresentou o melhor desempenho em qualidade. Concluímos que a Busca Tabu é um método recomendável para solução de problema de rotulação cartográfica de pontos, devido a sua simplicidade, praticidade, eficiência e bom desempenho, conjugado à capacidade de gerar soluções de qualidade em tempo computacional aceitável.



## **TABU SEARCH APPLICATION FOR POINT FEATURES CARTOGRAPHIC LABEL PLACEMENT PROBLEM**

### **ABSTRACT**

The generation of better label placement configurations in maps is a problem that comes up in automated cartographic production. The objective of a good label placement is to show the geographic position of the features with their corresponding texts clearly, respecting the cartographic conventions, with an esthetic and a harmonious quality when presenting the information. We approached the label placement problem from a combinatorial optimization point of view. In this perspective, the cartographic label placement belongs to a problem area of difficult solution, leading us to the need of approximation algorithms as there is no exact algorithm that is able to solve this problem within an acceptable amount of time. Our research consisted in the evaluation of the tabu search optimization algorithm applied to cartographic label placement. The tabu search implemented in SCARTA, a cartographic production software, in development by the Image Processing Division of the National Institute of Aerospace Researches, proved to be an efficient algorithm, in real and random test cases. When compared with alternative techniques such as “simulated annealing”, genetic algorithm with mask and others described in literature, the tabu search had the best performance in quality. We concluded that tabu search is a recommended method to solve cartographic label placement problem of point features, due to its simplicity, practicality, efficiency and good performance along with its ability to generate quality solutions in acceptable computational time.



## SUMÁRIO

	<b>Pág.</b>
<b>LISTA DE FIGURAS</b>	
<b>LISTA DE TABELAS</b>	
<b>CAPÍTULO 1 – INTRODUÇÃO</b> .....	17
<b>CAPÍTULO 2 – ROTULAÇÃO CARTOGRÁFICA</b> .....	21
2.1 - Sistemas de informação geográfica .....	21
2.2 - SIG e rotulação cartográfica .....	25
2.3 - O problema de rotulação cartográfica .....	26
2.4 - Técnicas de rotulação de pontos .....	28
<b>CAPÍTULO 3 – ALGORITMOS PARA ROTULAÇÃO DE PONTOS</b> .....	31
3.1 - Busca exaustiva .....	36
3.2 - Algoritmo guloso .....	36
3.3 - “Discrete Gradient Descent” .....	37
3.4 - Algoritmo de Hirsch .....	38
3.5 - “Simulated Annealing” .....	41
3.6 - Algoritmo genético .....	43
<b>CAPÍTULO 4 – BUSCA TABU PARA ROTULAÇÃO CARTOGRÁFICA</b> ...	49
4.1 - Descrição do método .....	49
4.2 - Descrição do método para rotulação cartográfica de pontos .....	52
4.2.1 - Busca Tabu e o problema de rotulação de 6 pontos .....	52
4.2.2 - Pré-computação .....	54
4.2.3 - Configuração inicial .....	60
4.2.4 - Mudança de configuração .....	61
4.2.5 - Memória de longo prazo .....	61

4.2.6 - Função objetivo .....	62
4.2.7 - Lista de soluções Tabu .....	63
4.2.8 - Critério de aspiração .....	64
4.2.9 - Simulação para rotulação de 6 pontos .....	64
<b>CAPÍTULO 5 – RESULTADOS EXPERIMENTAIS .....</b>	<b>73</b>
5.1 - Problema real .....	77
5.2 - Análise comparativa com a literatura .....	80
5.3 - Ambiente de trabalho .....	84
<b>CAPÍTULO 6 – CONCLUSÃO .....</b>	<b>85</b>
<b>REFERÊNCIAS BIBLIOGRÁFICAS .....</b>	<b>89</b>
<b>APÊNDICE A1 – FUNCIONALIDADES DE UM SIG CARTOGRÁFICO ....</b>	<b>93</b>
<b>APÊNDICE A2 – TÉCNICAS DE ROTULAÇÃO DE LINHAS .....</b>	<b>99</b>
<b>APÊNDICE A3 – TÉCNICAS DE ROTULAÇÃO DE ÁREAS .....</b>	<b>105</b>
<b>APÊNDICE A4 .....</b>	<b>111</b>
<b>APÊNDICE A5 .....</b>	<b>123</b>



## LISTA DE FIGURAS

	<b>Pág.</b>
1.1 - Rotulação de cartas .....	18
2.1 - O pacote de software de um SIG .....	23
2.2 - Exemplo de por que a rotulação cartográfica deve ser feito após a escolha da área de estudo e seleção dos planos de informações .....	26
2.3 - Rotulação de entidade ponto, levando em consideração a entidade linha .....	29
3.1 - Conjunto de 8 posições candidatas para o rótulo do ponto .....	31
3.2 - Conjunto de 17 posições candidatas para o rótulo do ponto com suas respectivas penalidades .....	32
3.3 - Rótulos em conflito .....	33
3.4 - Árvore de configurações possíveis para pontos com 2 posições candidatas ....	34
3.5 - (a) mínimo local do algoritmo "Discrete Gradient Descent". (b) uma configuração ótima .....	38
3.6 - (a) posições candidatas para rótulo do ponto no algoritmo de Hirsch. (b) vetor de sobreposição e (c) vetor agregado .....	39
3.7 - Mínimo local do algoritmo de Hirsch .....	40
4.1 - Busca Tabu para evitar o mínimo local .....	51
4.2 - Lista T de soluções Tabu .....	52
4.3 - Posições candidatas e preferência cartográfica .....	53
4.4 - Quantidade de conflitos .....	53
4.5 - Configuração inicial .....	54

4.6 - Posições candidatas .....	55
4.7 - Após a aplicação do algoritmo Busca Tabu, descrito na configuração da Figura 4.5 .....	72
A1.1 - Escala gráfica .....	95
A1.2 - Molde de uma carta .....	96
A2.1 - Rotulação das isolinhas .....	99
A2.2 - Termos utilizados para rotulação das entidades linhas .....	101
A2.3 - Área entre “swath line” e linha silhueta (“skyline”) inferior .....	101
A2.4 - Posição de rótulo com distância média considerado bom .....	102
A2.5 - Calculando grau de curvatura do “swath line” .....	103
A2.6 - Posição do rótulo com o valor de “flatness” considerado bom .....	103
A3.1 - Rotulação da entidade área. (a) Normal (b) Quando os limites da área são importantes .....	105
A3.2 - Esqueleto de uma entidade área .....	107

## LISTA DE TABELAS

	<b>Pág.</b>
3.1 - Tempo usado para gerar configurações possíveis para pontos com 2 posições candidatas .....	35
4.1 - Diagrama de conflitos do ponto P0 .....	56
4.2 - Diagrama de conflitos do ponto P5 .....	56
4.3 - Diagrama de conflitos do ponto P1 .....	57
4.4 - Diagrama de conflitos do ponto P2 .....	58
4.5 - Diagrama de conflitos do ponto P3 .....	59
4.6 - Diagrama de conflitos do ponto P4 .....	60
5.1 - O melhor fator para o tamanho da lista tabu .....	74
5.2 - Número de iterações para recálculos .....	75
5.3 - O melhor fator para o tamanho da vizinhança .....	76
5.4 - Resultados dos testes para diferentes valores de $\alpha_1$ e $\alpha_2$ .....	78
5.5 - Resultados dos testes para diferentes altura de caracter .....	79
5.6 - Resultados dos testes para diferentes escala .....	79
5.7 - Resultados obtidos por Busca Tabu usando o conjunto padrão de dados .....	82
5.8 - Resultados obtidos por vários algoritmos usando o conjunto padrão de dados .....	83
A1 - Formatos de papel .....	93



# CAPÍTULO 1

## INTRODUÇÃO

Este trabalho analisa o problema de rotulação cartográfica e discute alternativas para torná-la mais eficiente e automatizada. O termo “rotulação cartográfica”, neste documento, refere-se ao processo de inserção de texto em um documento cartográfico, aqui denominado “carta”, num ambiente de sistemas de informação geográfica. Segundo Freeman (1991) uma carta pode ser definida como a representação dos aspectos naturais ou artificiais da terra, destinada a fins práticos da atividade humana, permitindo a localização geográfica de pontos, áreas e detalhes. Como um meio de comunicação, uma carta precisa, de alguma forma, apresentar relações espaciais ao seu observador, independente de ele ser um leigo ou um perito, habilidoso na arte de leitura de mapa.

O processo tradicional de produção cartográfica manual leva por volta de 1 ano. Atualmente, a elaboração das cartas é feita utilizando-se o computador e seus acessórios, como digitalizadores, plotters e terminais de vídeo. Os dados da carta são armazenados em banco de dados geográficos do qual podem ser acessados seletivamente. Essa tecnologia resultou numa produção cartográfica incomparavelmente mais rápida e de boa qualidade (Freeman, 1991). Segundo esse autor, um dos maiores desafios da cartografia computadorizada é a rotulação, pois as entidades (ponto, linha, polígono) que devem ser rotuladas podem ocorrer com tal riqueza de diversidade e complexidade que é difícil estabelecer um conjunto fixo de procedimentos para rotulá-las.

O nome fictício da ilha se espalha por toda a área da ilha, acompanhando o formato da mesma; a associação entre o texto e a entidade a que pertence é instantânea. Cidades situadas na costa marítima têm seu texto colocado na água para enfatizar sua localização costeira. A rotulação de texto em áreas não limitadas, como a Egabrag Forest do exemplo, fornece um sentimento intuitivo da extensão e formato da entidade rotulada (Figura 1.1).



Fig. 1.1 - Rotulação de cartas.

FONTE: Freeman (1991, p. 445).

A rotulação é portanto, um problema que precisa ser resolvido na produção cartográfica. Uma carta não deve mostrar apenas a posição geográfica das entidades, mas também algumas de suas propriedades, de tal forma que a informação seja legível, que respeite as convenções cartográficas, que fique clara a que entidade ela pertence, preservando sempre a estética e harmonia na apresentação dessas informações.

Christensen et al. (1995) e Shawn et al. (1996) descrevem um conjunto de técnicas de automatização do processo de rotulação de cartas. No entanto, estas técnicas ainda não estão plenamente incorporadas à grande maioria dos sistemas de informação geográfica disponíveis no mercado. Nestes sistemas, os algoritmos implementados são versões simplificadas e parte da edição e rotulação ainda tem que ser feita manualmente, o que implica num substancial dispêndio de recursos e tempo.

Dentro desse contexto o presente trabalho visa 3 objetivos a saber:

- Contribuir para a melhoria das facilidades de produção cartográfica do SCARTA através da implementação de métodos semi-automáticos de rotulação de pontos. Este software de produção cartográfica, em desenvolvimento pela Divisão de Processamento de Imagens (DPI/INPE), apesar de possuir uma parcela ampla das funcionalidades de um software de produção cartográfica, carece da automatização em rotulação, de modo a evitar que o usuário necessite

fazer um trabalho tedioso de posicionamento manual de textos. Quando as entidades a serem rotuladas são poucas, o SCARTA atende às expectativas razoavelmente, entretanto, se as entidades crescem; a tarefa de rotulação, além de ser tediosa, consome muito tempo, sendo inviável na prática.

- Avaliar o desempenho do algoritmo Busca Tabu aplicado à rotulação cartográfica, comparando com outros algoritmos alternativos para rotulação descritos na literatura, usando o mesmo conjunto padrão de dados de distribuição de pontos descritos no capítulo 5.
- Avaliar o desempenho do algoritmo Busca Tabu aplicado a dados reais.

Para alcançar os objetivos acima descritos as seguintes metas foram cumpridas:

- Revisar a literatura sobre sistemas de informação geográfica (SIG) e SIG cartográfico.
- Revisar a literatura sobre problemas de rotulação de ponto, linha e polígono.
- Analisar, revisar a literatura e selecionar técnicas de rotulação de pontos, linhas e polígonos.
- Revisar a literatura sobre algoritmos de otimização para rotulação de pontos.
- Implementar o algoritmo de otimização Busca Tabu, no ambiente SIG SCARTA.
- Realizar testes do algoritmo Busca Tabu para dados reais e conjunto padrão de dados sugeridos na literatura.

O restante do trabalho está organizado da seguinte forma. O Capítulo 2 faz uma descrição de sistemas de informação geográfica, problemas de rotulação cartográfica e técnicas de rotulação cartográfica. No Capítulo 3 a rotulação de pontos é vista como um problema de otimização combinatória e são estudados 6 algoritmos de otimização aplicados a rotulação de pontos: Busca Exaustiva, Algoritmo Guloso, “Discrete Gradient Descent”, Algoritmo de Hirsch, “Simulated Annealing” e Algoritmo Genético com máscara. No Capítulo 4, apresentamos o algoritmo de otimização Busca Tabu para solucionar o problema de rotulação de pontos. O Capítulo 5 apresenta e discute os resulta-

dos obtidos da aplicação de Busca Tabu à rotulação de pontos, tanto em dados reais quanto em um conjunto padrão de dados sugerido na literatura, a fim de tornar possível a comparação do algoritmo Busca Tabu com os demais algoritmos descritos na literatura. Finalmente o Capítulo 6 apresenta as conclusões e sugere extensões ao trabalho.



## **CAPÍTULO 2**

### **ROTULAÇÃO CARTOGRÁFICA**

Neste Capítulo, fazemos uma revisão bibliográfica sobre Sistemas de Informação Geográfica, rotulação cartográfica e seus problemas, apresentando em seguida técnicas de rotulação de pontos, linhas e polígonos descritos na literatura.

#### **2.1 SISTEMAS DE INFORMAÇÃO GEOGRÁFICA**

O termo Sistema de Informação Geográfica (SIG) aplica-se a sistemas computacionais que manipulam dados geográficos. A palavra sistema deve-se ao fato de SIG ser composto por vários componentes interrelacionados. A palavra informação deve-se ao fato do SIG permitir a conversão de dados em informações a partir de manipulações e consultas interativas sobre os dados armazenados. A palavra geográfica implica que os dados possuem localizações conhecidas ou podem ser calculados em termos de coordenadas geográficas (Barbosa. 1997).

Devido a sua ampla gama de aplicações, que inclui temas como agricultura, floresta, cartografia, cadastro urbano e redes de concessionárias (água, energia e telefonia), há pelo menos três grandes maneiras de utilizar um SIG (Câmara. 1995).

- Como tecnologia de gerenciamento de dados geográficos : os avanços da tecnologia de computadores e de satélites espaciais nas últimas décadas facilitaram a aquisição de dados geográficos por órgãos governamentais e privados, gerando com isto uma massa muito grande de dados. Os SIGs possuem ferramentas que permitem a integração em uma única base cartográfica, informações espaciais provenientes de dados cartográficos, dados de censo, cadastro urbano e rural, imagens de satélite, etc.
- Como suporte para análise espacial de fenômenos: os SIGs oferecem mecanismos para manipular simultaneamente vários dados, os quais vão desde consulta, recuperação e visualização, até combinação de variáveis para análise. O processo de análise dos dados é, geralmente, aquele no qual as relações e signi-

ficados que estão implícitos em um conjunto de dados são extraídos e mostrados de forma explícita (Bonham-Carter. 1996).

- Como ferramenta para produção cartográfica: por possuir facilidades de edição, visualização, acesso rápido e registro geográfico dos dados, os SIGs estão se tornando uma ferramenta de trabalho nos órgãos responsáveis por produção cartográfica.

Um SIG, utilizado para processar dados geográficos, é composto por cinco subsistemas (Figura 2.1):

- interface: define como o sistema é operado e controlado;
- entrada de dados: converte dados capturados em forma digital compatível;
- visualização e plotagem: apresenta resultados em uma variedade de formas como mapas, imagens e tabelas;
- transformação, consulta e análise espacial: provê métodos para processamento de imagens e técnicas para consulta e análise espacial; e
- gerência de dados espaciais: organiza, armazena e recupera dados.

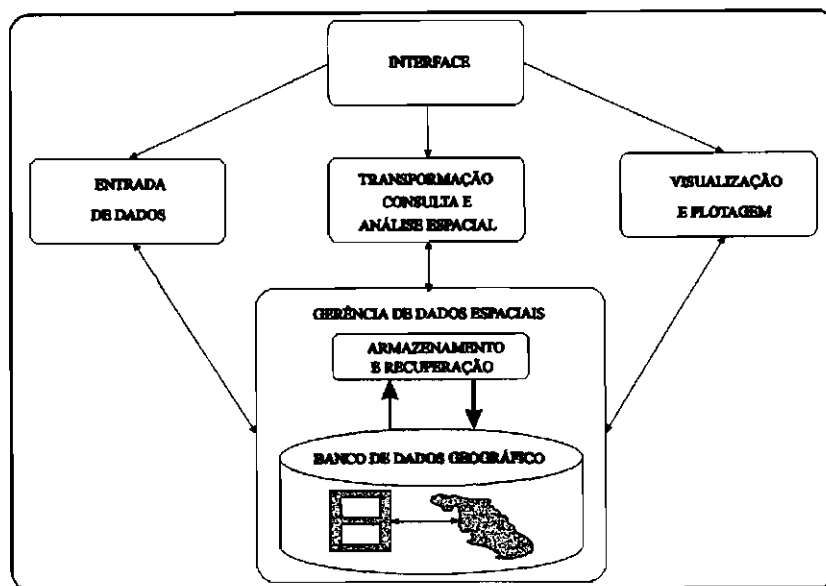


Fig. 2.1 - O pacote de software de um SIG.

FONTE: Câmara (1995, p. 2-6).

Segundo Câmara et al. (1996), os principais tipos de dados em geoprocessamento são:

- mapas temáticos: descrevem a distribuição espacial de uma grandeza geográfica, expressa de forma qualitativa, como mapas de pedologia e aptidão agrícola de uma região.
- mapas cadastrais: distinguem-se de mapas temáticos, pois cada um de seus elementos é um objeto geográfico que possui atributos e pode estar associado a várias representações gráficas. Por exemplo, os lotes de uma cidade são elementos do espaço geográfico que possuem atributos (dono, localização, valor venal, IPTU devido, etc).
- Redes: No caso de redes, cada objeto geográfico (ex. cabo telefônico, transformador de rede elétrica, cano de água) possui uma localização geográfica e está sempre associado a atributos descritivos presentes no banco de dados. Em Geoprocessamento, o conceito de “rede” denota informações associadas a:
  - Serviços de utilidade pública, como água, luz e telefone;

- Redes de drenagem (bacias hidrográficas);
- Rodovias.
- Imagens: obtidas por satélites, fotografias aéreas ou “scanners” aerotransportados, representam formas de captura de informação espacial.
- Modelos Numéricos de Terreno: o termo modelo numérico de terreno – MNT - é utilizado para denotar a representação quantitativa de uma grandeza que varia continuamente no espaço. Comumente associados à altimetria, também podem ser utilizados para modelar grandezas geoquímicas, como teor de minerais ou propriedades do solo, como teor de matéria orgânica, acidez ou condutividade elétrica.

Existem duas grandes classes de representações computacionais de mapas: vetoriais e matriciais (Câmara et al. 1996). A representação matricial consiste no uso de uma malha quadriculada regular sobre a qual se constrói, célula a célula, o elemento que está sendo representado. A cada célula atribui-se um código referente ao atributo estudado, de tal forma que o computador saiba a que elemento ou objeto pertence determinada célula.

No caso de representação vetorial, consideram-se três elementos gráficos: ponto, linha e área (polígono). Os pontos, ou elementos pontuais, abrangem todas as entidades geográficas que podem ser perfeitamente posicionadas por um único par de coordenadas X e Y. Entretanto, além das coordenadas, outros dados não-espaciais (atributos) podem ser arquivados para indicar de que tipo de ponto se está tratando.

As linhas, arcos, ou elementos lineares são um conjunto de pontos conectados. Além das coordenadas dos pontos que compõem a linha, deve-se armazenar informação que indique de que tipo de linha se está tratando, ou seja, a que atributo ela está associada.

As áreas – polígonos - são representados pela lista de linhas que as compõem.

Como pode ser visto a rotulação cartográfica automatizada visa trabalhar apenas com representações vetoriais, ou seja, com os elementos gráficos: ponto, linha, área e seus atributos que se encontram nos bancos de dados geográficos.

Maiores detalhes sobre SIG, consultar Bonham-Carter (1996); Burrough (1987) e Câmara et al. (1996).

## **2.2 SIG E ROTULAÇÃO CARTOGRÁFICA**

Um SIG cartográfico visa utilizar as informações armazenadas em um banco de dado geográfico e gerar documentos cartográficos. Para atender este objetivo, um SIG cartográfico deve ter flexibilidade de rotulação de texto, edição de texto, escolha do tipo de papel, edição de linha, edição de grade de referenciamento geográfico, visualização de informações geográficas, e outras funcionalidades descritas no Apêndice A1. Dentre todas estas funcionalidades, a rotulação cartográfica tem se mostrado um dos maiores desafios para a cartografia computadorizada, pois posicionar os textos requer que a associação não ambígua entre o texto e a entidade correspondente seja alcançada, que não haja sobreposição entre os textos ou entre texto e entidades, que sejam respeitadas as convenções e preferências cartográficas, que o tempo de processamento seja irrisório e que um alto nível de harmonia e qualidade sejam alcançados.

O desenho dos nomes foi sempre a parte de maior responsabilidade profissional para um cartógrafo, devido à natureza deste tipo de representação cartográfica, isto é, o desenho em si, a escolha da letra e a sua colocação ou distribuição em relação aos acidentes e aos outros nomes. E foi, sobretudo, o elemento de mais demorada elaboração em todo o período da construção do mapa. Estima-se que o desenho do letreiro de uma carta constitui cerca de 80% do tempo necessário a toda a operação de feitura do original cartográfico (Oliveira. 1988).

Como pode ser visto, a rotulação cartográfica é um dos problemas que merece atenção especial, pois a execução da tarefa de rotulação é árdua e demorada. Ultimamente, vários algoritmos de rotulação automatizada têm aparecido, mas nenhum é ainda definitivo pois, as entidades a serem rotuladas ocorrem com uma riqueza de diversidade e

complexidade que é difícil encontrar um algoritmo geral para todos os casos. (Freeman, 1991).

### 2.3 O PROBLEMA DE ROTULAÇÃO CARTOGRÁFICA

No contexto deste problema, considera-se que os dados estão armazenados em um banco de dados geográficos na forma de planos de informação, onde cada plano de informação representa um tipo particular de informação geográfica, tais como limite político, rede de estradas, sistema de estrada de ferro, uso da terra, hidrografia, etc. Assim sendo, para construir uma carta, o plano de informação contendo uma informação geográfica específica de uma determinada região deve ser selecionado.

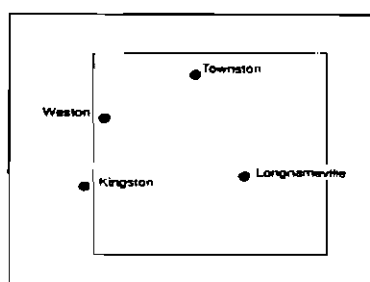


Fig. 2.2 - Exemplo de porque a rotulação cartográfica deve ser feita após a escolha da área de estudo e seleção dos planos de informação.

FONTE: Freeman (1991, p. 447).

Seria conveniente que a rotulação de textos pudesse ser executada apenas uma única vez e a melhor distribuição fosse armazenada juntamente com o plano de informação, mas isso não é possível ( Figura 2.2). Se a carta em construção for de uma área menor do que se encontra no banco de dados, a entidade ponto pode estar dentro da área, mas o texto pode estar fora da área, como em “Weston”, ou a entidade ponto pode estar fora da área, mas o texto estar dentro da área, como em “Kingston”, ou ainda o texto pode ter sido interrompido, aparecendo apenas uma parte dela, como em “Longnameville”.

Mas o problema não se encontra apenas nos limites, uma vez que vários planos de informação de diversas categorias ou tipos de dados podem ser selecionados para fazer

parte da carta, sendo impossível que a sobreposição entre os textos ou entre o texto e a entidade geográfica seja evitada. Portanto a rotulação de textos só deve ser executada depois que:

- Os limites do mapa forem estabelecidos;
- Os planos de informações forem selecionados;
- A escala for escolhida.

A altura e o comprimento do rótulo afetam a qualidade da rotulação, uma vez que quanto maior for o rótulo maior será o espaço ocupado por ele e, por conseguinte, maior a probabilidade de conflito com outros rótulos ou entidades geográficas. Muitas vezes, o simples fato de diminuir um pouco o tamanho dos rótulos acaba com alguns conflitos e melhora a qualidade da rotulação consideravelmente.

Outros fator que deteriora a qualidade da rotulação é o agrupamento natural dos pontos e a relação existente entre a área do mapa considerado, o tipo de papel – A0, A1, A2, A3, A4 – e a escala em uso.

De um modo geral as instruções para uma boa rotulação, segundo Imhof (1962), citado em Freeman (1991), são:

- 1) A associação do rótulo à entidade geográfica a que pertence deve ser imediata aos olhos do observador.
- 2) Não deve haver sobreposição entre os rótulos ou entre o rótulo e as entidades ponto.
- 3) rótulo da entidade área deve ser espaçado de acordo com o seu formato e extensão.
- 4) Pode haver sobreposição entre a entidade linha e seu rótulo. Neste caso a linha deverá sofrer interrupção.

- 5) Rótulos devem ser distribuídos naturalmente sem excessiva aglomeração.
- 6) Para mapas de escala pequena usando projeções, quando as linhas de latitude são mostradas curvadas, todos os rótulos mostrados horizontalmente precisam acompanhar a curvatura dos paralelos.
- 7) Rotulação precisa estar de acordo com padrões e convenções.

Freeman (1991) sugere rotular primeiro as entidades área, pois o texto deverá acompanhar o formato das mesmas, assim como ocupar todo o seu espaço. Em seguida a sugestão é rotular os pontos e por último as linhas, por serem mais flexíveis. Ele sugere ainda o uso de “backtrack”, caso não encontre posições aceitáveis para entidade linha ou ponto.

## 2.4 TÉCNICAS DE ROTULAÇÃO DE PONTOS

As regras básicas de rotulação de pontos, segundo Freeman (1991), são:

- 1) rótulo deverá estar próximo à entidade ponto a que pertence.
- 2) A preferência cartográfica é para os rótulos que se encontram acima das entidades ponto a que pertencem.
- 3) Não pode haver sobreposição com entidade ponto ou outros rótulos; entretanto o rótulo pode ser colocado no espaço entre caracteres que compõe o texto da entidade área. Um rótulo também pode se sobrepor à entidade linha; neste caso, a linha é interrompida.
- 4) Se a entidade ponto está localizada próximo à entidade linha ou próximo ao limite da entidade área, o rótulo deverá ser colocado do mesmo lado que se encontra a entidade ponto em questão.
- 5) Para a entidade ponto localizada ao longo de um curso extenso de água, o rótulo deverá ser colocado na água (esta é uma exceção as regras anteriores). Por exemplo o nome de uma cidade litorânea.



- 6) Rótulos deverão ser colocados horizontalmente, tornando possível a leitura da esquerda para a direita. Quando os paralelos estão encurvados, o rótulo também deve acompanhar a curvatura.
- 7) Não pode haver espaço extra entre os caracteres que compõem o rótulo.
- 8) Em situações extremas, o nome pode ser abreviado. Em último caso, o texto pode ser colocado a alguma distância com um indicador apontando do rótulo para a entidade ponto.

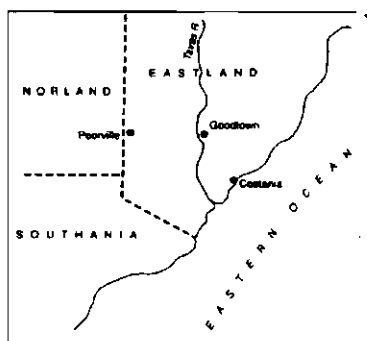


Fig. 2.3 - Rotulação de entidade ponto, levando em consideração a entidade linha.

FONTE: Freeman (1991, p. 451).

Das regras acima, o mais difícil de solucionar são as sobreposições e a colocação de rótulos do mesmo lado a que pertence a entidade ponto quando a vizinhança é entidade linha ou limite de uma entidade área (Figura 2.3). A cidade de “Goodtown” está propriamente rotulada, enquanto que rótulo “Poorville” se encontra em posição imprópria, pela regra 4. O rótulo “Costania” encontra-se rotulado apropriadamente, uma vez que a regra 5 se aplica.

Uma descrição sobre técnicas de rotulação de linhas se encontram no Apêndice A2, no Apêndice A3 se encontram as descrições sobre técnicas de rotulação de áreas e no Capítulo 3 são descritos alguns dos principais algoritmos usados para solucionar o problema de rotulação de pontos.

## CAPÍTULO 3

### ALGORITMOS PARA ROTULAÇÃO DE PONTOS

Como o foco deste trabalho está relacionado à rotulação de pontos, com a proposta de um novo algoritmo para a solução do problema, vamos revisar neste capítulo alguns dos algoritmos existentes para o problema.

A rotulação de pontos é um problema de otimização combinatória e, como tal, dois aspectos merecem a atenção: posições candidatas com preferência cartográfica e a função objetivo.

Entende-se por posições candidatas o conjunto de todas as possíveis posições que o rótulo de um determinado ponto pode ocupar. Estas possíveis posições são escolhidas de acordo com uma padronização cartográfica (Christensen et al. 1995). A Figura 3.1 mostra um conjunto de 8 posições candidatas para o rótulo de um ponto. Os números mostrados indicam a preferência cartográfica; quanto menor for o número, maior é a preferência.

Os algoritmos que consideram a preferência cartográfica, manipulam custos ou penalidades atribuídas a cada posição candidata. Christensen et al. (1995) sugere uma penalidade de  $(r-1)/N$  para cada  $r$ -ésima posição candidata das  $N$  possíveis posições. Assim sendo, no exemplo abaixo, a posição acima à direita recebe uma penalidade 0, acima a esquerda a penalidade  $1/8$  e assim por diante.

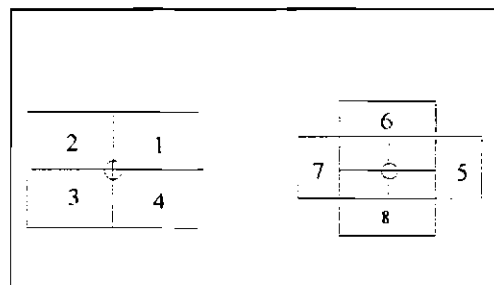


Fig. 3.1 - Conjunto de 8 posições candidatas para o rótulo de um ponto.

FONTE: Christensen et al. (1995, p. 205).

A Figura 3.2 mostra um conjunto de 17 posições candidatas para o rótulo de um ponto com suas respectivas penalidades. Trata-se de uma outra técnica de definição de posições candidatas, proposta por Shawn et al.(1996).

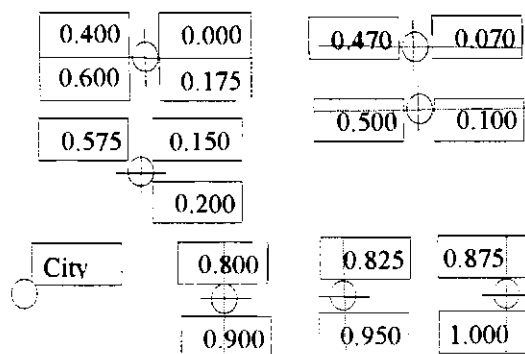


Fig. 3.2 - Conjunto de 17 posições candidatas para o rótulo de um ponto com suas respectivas penalidades.

FONTE: Shawn et al(1996, p. 6).

A função objetivo é uma função a ser otimizada que tem por meta medir a qualidade da rotulação, diferenciando elementos bons dentre as posições candidatas existentes. Segundo Christensen et al.(1995), a qualidade da rotulação depende dos seguintes fatores:

- 1) Número de rótulos em conflito: Conflito entre rótulos diz respeito a interseção entre os retângulos envolventes dos textos. Na Figura 3.3 o número de rótulos em conflito é 3.
- 2) Preferência cartográfica.
- 3) Seleção de ponto.

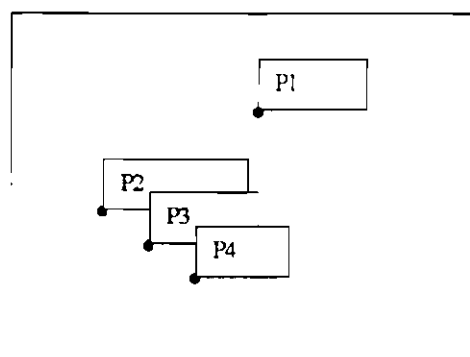


Figura 3.3 - Rótulos em conflito

Entende-se por seleção de ponto o ato de excluir o ponto e o respectivo rótulo se o conflito persistir após um determinado número de tentativas. Quando a seleção de ponto é considerada os pontos também recebem um custo ou penalidade para evitar a deleção de pontos mais importantes, tais como uma capital.

A função objetivo usada é minimizada pois cada posição candidata do rótulo possui uma penalidade mostrando a preferência cartográfica e uma boa rotulação é representada pela melhor combinação.

A otimização combinatória é a área da matemática que estuda problemas que possuem uma ou mais soluções ótimas dentro de um conjunto finito de possíveis soluções.

Um problema de otimização combinatória pode ser definido matematicamente a partir do par  $(\Omega, C)$ , onde  $\Omega$  é o conjunto (finito) de soluções possíveis e  $C$  é uma função  $C : \Omega \rightarrow \mathbb{R}$ , sendo  $\mathbb{R}$  o conjunto dos reais. O objetivo, no problema de otimização, é encontrar uma solução ótima  $S \in \Omega$ , tal que  $C(S) \leq C(S'), \forall S' \in \Omega$ ; ou seja, tal que a função  $C$ , conhecida como função custo do problema, seja minimizada. Como  $\Omega$  é finito, intuitivamente poderíamos, selecionar a solução ótima por enumeração. Essa proposta de enumeração exaustiva funciona teoricamente, mas torna-se inviável para os problemas de interesse, devido ao tamanho extremamente grande – apesar de finito – do conjunto de soluções. Cada pequeno incremento na dimensão do problema introduz uma gama muito grande de possibilidades, configurando uma explosão combinatória de soluções

possíveis, e uma conseqüente explosão no tempo necessário para encontrar a solução ótima (Bjorndal et al. 1995).

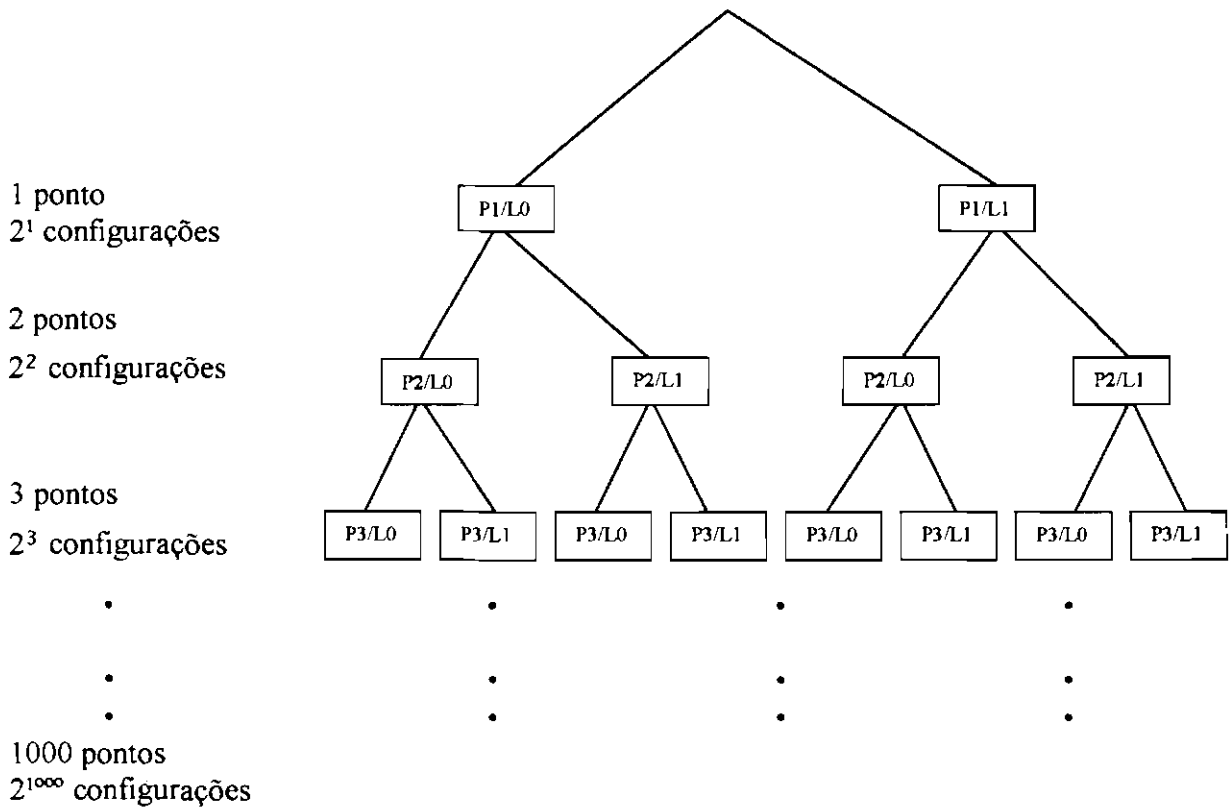


Fig. 3.4 - Árvore de configurações possíveis para pontos com 2 posições candidatas.

Em rotulação cartográfica se considerarmos 2 posições candidatas – L0 e L1 – para cada ponto  $p_i$ , onde  $i$  varia entre 1 e o número de pontos, e desenvolvermos uma árvore de configurações veremos que o número de configurações possíveis aumenta exponencialmente dependendo do número de pontos do leiaute (Figura 3.4).

É possível desenvolver um programa que verifique todas as configurações possíveis de por exemplo 1000 pontos de 2 posições candidatas, mas o que não é possível é esperar, dependendo do número de pontos, o término da execução do programa (Tabela 3.1). O tempo se refere apenas à geração das configurações em uma estação de trabalho SUN – SPARC20, não levando em consideração o tempo de verificação do custo de cada configuração.

**TABELA 3.1 – TEMPO USADO PARA GERAR CONFIGURAÇÕES POSSÍVEIS PARA PONTOS COM 2 POSIÇÕES CANDIDATAS**

Num. de pontos	Num. de configurações	Tempo
10	$2^{10}$	2.52 (seg.)
11	$2^{11}$	9.71 (seg.)
12	$2^{12}$	38.44 (seg.)
13	$2^{13}$	153.47 (seg.)
14	$2^{14}$	10.23 (min)
•	•	•
•	•	•
•	•	•
20	$2^{20}$	698.21 (h) (~ 29 dias)
•	•	•
•	•	•
•	•	•

Esta explosão combinatória mostra que se trata de um problema de otimização combinatória de difícil solução e neste caso muitos pesquisadores não acreditam na existência de um algoritmo exato eficiente e mesmo que esse algoritmo fosse encontrado existe uma grande probabilidade de que seria bastante lento para problemas grandes, assim sendo existe a necessidade de introduzir heurísticas, que não buscam a solução exata, mas uma solução suficientemente boa em termos de custo.

As heurísticas mais elaboradas quando aplicadas à rotulação cartográfica se baseiam em procedimentos de melhoria do leiaute. A partir de uma determinada solução é aplicada uma alteração da posição candidata de um determinado ponto, que acarreta uma variação na configuração, isto é, no custo que demonstra se a alteração proposta trouxe uma solução melhor ou contribuiu para aumentar o número de conflitos ou ainda se os rótulos estão colocados nas posições cartograficamente preferenciais.

Metaheurísticas, também conhecidas como estratégias universais de otimização, procuram abstrair-se das peculiaridades de cada problema, usando analogias que podem ser aplicadas ao mais diversos domínios, através de uma prescrição específica (representação de soluções, avaliação de seus custos e movimentos vizinhos possíveis) do problema em questão. Dentre as metaheurísticas a serem revistas nesta seção, temos o “Simulated Annealing” e o algoritmo genético com máscara. No capítulo 4, descrevemos a proposta de aplicação da Busca Tabu ao problema de rotulação de pontos.

A seguir será feita uma revisão dos principais algoritmos para rotulação de pontos disponíveis na literatura.

### **3.1 BUSCA EXAUSTIVA**

O algoritmo aqui utilizado é o “backtracking”, onde o rótulo do ponto é colocado em uma das posições não obstruídas. Se um determinado ponto não pode ser rotulado porque não existem posições sem conflito, o algoritmo volta para o ponto rotulado anteriormente e muda a posição do rótulo. E assim o algoritmo continua até que todos os pontos tenham sido rotulados.

Para problemas relativamente pequenos os algoritmos de busca exaustiva podem ser usados, mas não são recomendados para problemas de tamanho moderado por causa da sua natureza exponencial, tornando então a abordagem de busca exaustiva impraticável para problemas gerais de rotulação de pontos.

### **3.2 ALGORITMO GULOSO**

O objetivo aqui é limitar o campo de busca. Quando um ponto qualquer não puder ser rotulado porque todas as posições candidatas já se encontram ocupadas, o ponto pode ser retirado se a seleção de pontos for permitida; caso contrário a rotulação do ponto é executada mesmo acarretando conflito de rótulo. Isto evita o retorno ao ponto anterior para trocar a posição do rótulo (Christensen et al. 1995).

Este tipo de algoritmo consegue um resultado onde o custo computacional é baixo, mas a qualidade da rotulação não é tão boa quanto a da busca exaustiva. Mas muitas vezes é usado para se ter o primeiro rearranjo dos rótulos.

### 3.3 “DISCRETE GRADIENT DESCENT”

A idéia básica deste algoritmo é escolher dentre todas as operações disponíveis a que produza a melhor melhoria imediata.

Algoritmo:

- 1) Cada objeto geográfico ou ponto deverá ser rotulado em qualquer uma das posições candidatas.
- 2) Repetir até que a melhora adicional não seja possível
  - Para cada objeto geográfico considerar as posições alternativas do rótulo.
  - Calcular o valor da função objetivo que resultará se o rótulo for movido
  - Implementar o movimento de rótulo que conduz ao melhor resultado.

O maior defeito deste algoritmo é a incapacidade de escapar do mínimo local (Figura 3.5). Para qualquer movimento de rótulo que se faça (Figura 3.5(a)), a função objetivo é no mínimo 5, ou seja, é maior do que a função objetivo corrente que é 4, portanto não consegue produzir uma melhoria imediata e o algoritmo é encerrado, quando na verdade a configuração considerada boa seria a da Figura 3.5(b).



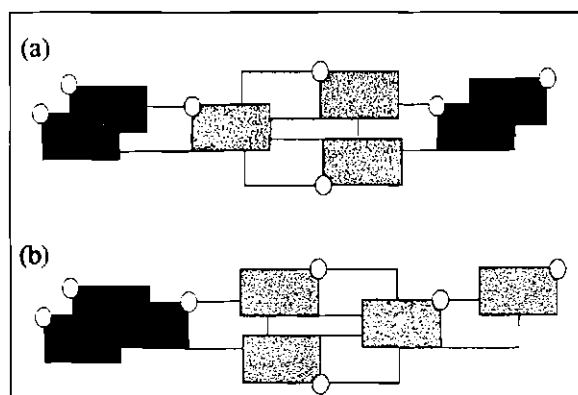


Fig. 3.5 - (a) mínimo local do algoritmo “Discrete Gradient Descent”. (b) uma configuração ótima.

FONTE: Christensen et al. (1995, p. 211)

### 3.4 ALGORITMO DE HIRSCH

Trata-se de um método onde cada ponto possui um conjunto infinito de posições para o rótulo. As posições candidatas do ponto podem tocar no seu círculo envolvente, mas jamais é permitida a interseção entre eles; os rótulos podem deslizar continuamente em torno do círculo que envolve o ponto (Figura 3.6(a)).

Quando o retângulo envolvente do rótulo toca o círculo envolvente do ponto, acima, abaixo, à esquerda ou à direita, é permitido ao retângulo envolvente do rótulo deslizar ao longo do ponto de tangência que mantêm com o círculo que envolve o ponto (Figura 3.6(b)).

No início, cada rótulo é colocado à direita do seu ponto. Cada rótulo é então testado para verificar se existe conflito com outro rótulo ou algum ponto. Para cada conflito existente, calcula-se um vetor de sobreposição baseado na área de interseção. Cada vetor de sobreposição é dividido entre os dois pontos em conflito e estes vetores representam de uma certa maneira o movimento desejado para eliminar o conflito. A soma dos vetores de sobreposição do rótulo em questão é calculado para se obter um vetor agregado que representa a direção ideal de movimento para sanar o conflito entre os rótulos ou entre o rótulo e o ponto. Na (Figura 3.6(c)), o vetor de sobreposição de cada rótulo está representado em pontilhado e o vetor agregado está representado em linha

contínua. Quando o rótulo está envolvido com somente um conflito, a direção do vetor de sobreposição e o vetor agregado são os mesmos.

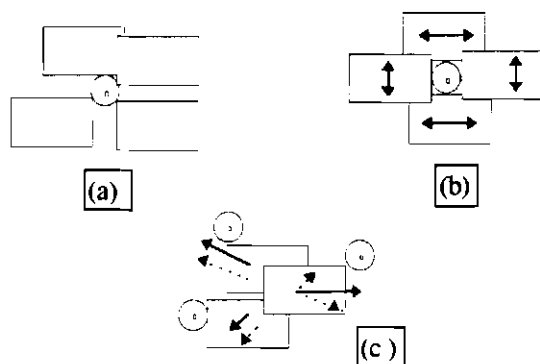


Fig. 3.6 - (a) posições candidatas rótulo do ponto no algoritmo de Hirsch.  
(b) vetor de sobreposição e (c) vetor agregado.

FONTE: Christensen et al. (1995, p. 212).

Quando o vetor agregado de sobreposição tiver sido calculado para todos os rótulos, o algoritmo procura mover os rótulos na direção apontada pelo vetor agregado. Existem duas maneiras de movimento a saber: movimento incremental e o movimento absoluto.

O movimento absoluto reposiciona o rótulo diretamente para a posição indicada pelo vetor agregado e o movimento incremental é guiado por uma série de regras heurísticas para mover o rótulo na direção do vetor agregado. Hirsch (1982) sugere que se altere o uso entre o movimento absoluto e o movimento incremental, porém usando com maior frequência o movimento incremental.

Neste tipo de algoritmo vários rótulos são reposicionados simultaneamente, uma vez que a operação de movimento é aplicada para todos os pontos da área de trabalho. Se compararmos com um sistema físico, o vetor de sobreposição individual representa a força de repulsão entre os objetos em conflito e a soma uma força agregada. Assim, através de movimentos graduais o sistema acaba se acomodando em um mínimo local de energia do sistema. Para evitar o mínimo local o autor sugere que se faça uso do movimento absoluto para pular de um estado de energia para outro.

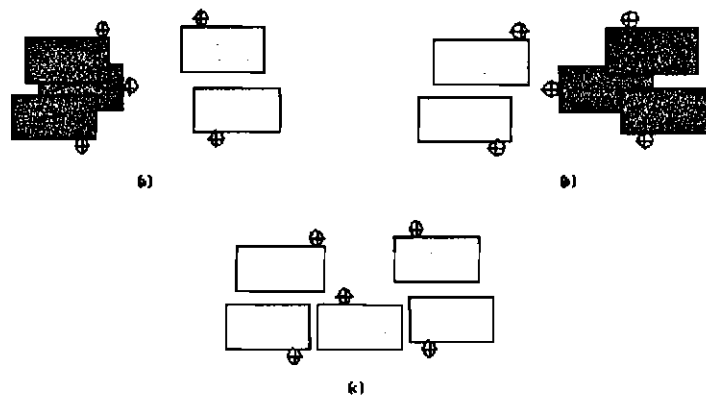


Fig. 3.7 - Mínimo local do algoritmo de Hirsch.

FONTE: Christensen et al. (1995, p. 213).

Existem alguns problemas no algoritmo de Hirsch. São eles:

- Quando muitos rótulos se sobrepõem a um simples rótulo, o vetor agregado calculado se torna muito grande, conduzindo assim a um problema de excesso de movimento incremental.
- Se um rótulo possuir vários vetores de sobreposição apontando radialmente para fora, o reposicionamento do rótulo é um tanto arbitrário.
- Mínimo local é decorrência das heurísticas que o algoritmo utiliza para contornar várias situações de conflito. (Figura 3.7) mostra uma configuração problemática. O algoritmo oscila entre a configuração (a) e configuração (b) sendo que a configuração desejada é a (c). Durante a aplicação do movimento incremental o rótulo é ajustado para cima e para baixo, mas o conflito com outros dois rótulos continua. Durante a aplicação do movimento absoluto, a componente horizontal do vetor de sobreposição domina e a localização do rótulo fica oscilando entre a esquerda e a direita do ponto em questão, esquecendo-se da posição aceitável que se localiza acima e abaixo do ponto em questão.

Com a finalidade de comparar o algoritmo de Hirsch com outros algoritmos, Christensen et al. (1995) reduziu o círculo que envolve o ponto a zero e limitou as posi-

ções candidatas para 20 e 4 posições. O experimento mostrou que o número de posições candidatas é irrelevante.

### 3.5 “SIMULATED ANNEALING”

É um método de busca estocástica que permite, algumas vezes, caminhar para soluções piores desejando no futuro encontrar soluções melhores. Naturalmente isto não é tolerado uniformemente, pois a capacidade do algoritmo para degradar a solução é controlado pelo parâmetro T, chamado de temperatura, que é decrementado com o tempo de acordo com um “schedule annealing”. (Kirkpatrick et al. 1983)

Em analogia com o sistema físico, a função custo representa a energia do sistema físico e a temperatura um parâmetro de controle. Assim sendo se no sistema físico, o cristal for aquecido a uma temperatura bastante elevada e depois submetido a um processo de resfriamento lento, conservando-o em cada temperatura durante um período de tempo e depois decrementando a temperatura de um pequeno fator, obtemos um cristal de altíssima qualidade, enquanto que se o resfriamento for muito rápido, vamos obter um cristal de baixa qualidade. O mesmo ocorre com a rotulação de pontos, portanto, devemos permanecer em uma determinada temperatura por um determinado número de iterações e depois decrementá-lo de um pequeno fator para conseguirmos uma configuração de rótulos de altíssima qualidade. As sugestões do Christensen et al. (1995) para decrementar a temperatura no caso da rotulação de pontos esta descrito em “Schedule Annealing” descrito adiante.

Em altas temperaturas, o algoritmo busca um ajuste de larga escala de modo que regiões circundantes do melhor mínimo local e talvez até mesmo do mínimo global pode ser visitado, e na temperatura próximo a zero, ele comporta-se como um algoritmo de melhoramento iterativo, para fazer os ajustes finos.

Características do algoritmo:

- 1) Para cada ponto colocar o rótulo randomicamente em qualquer das posições candidatas.

- 2) Inicializar a temperatura  $T = T_0$
- 3) Repetir por um número pré-definido de iterações.
  - Decrementar a temperatura  $T$  de acordo com o “schedule annealing”
  - Mover um rótulo para uma nova posição
  - Computar  $\Delta E$ , a mudança na função objetivo causado pelo reposicionamento do rótulo.
  - Se a nova posição do rótulo piorou o estado geral, aceitar o reposicionamento com probabilidade

$$P = 1.0 - \exp(-\Delta E/T) \quad (3.1)$$

onde:

$\Delta E$  = mudança na função objetivo;

$T$  = temperatura.

A implementação de um algoritmo “Simulated Annealing” envolve quatro componentes:

- 1) Configuração Inicial: como uma alternativa pode-se usar um outro método para se fazer a primeira rotulação e depois usar o “Simulated Annealing” como pos-processamento.
- 2) Função objetivo: a escolha da função objetivo afeta a estética do leiaute, a qualidade da solução e a eficiência da busca. Como “Simulated Annealing” é um método estatístico que necessita de um grande número de avaliações para o seu sucesso, as melhores funções objetivos são aquelas em que  $\Delta E$  pode ser computado rapidamente. A função objetivo escolhida por Christensen et al. (1995) foi a de contar o número de rótulos em conflito, se seleção de pontos

não for permitida. Caso a seleção de pontos tenha sido permitida, a função objetivo seria o número de pontos deletados + o número de rótulos obstruídos.

- 3) Mudança de configuração: o rótulo a ser reposicionado pode ser escolhido aleatoriamente ou dentre os que estão em conflito. Quando preferência cartográfica for incluída no problema, não é aconselhável escolher aleatoriamente o rótulo a ser reposicionado pois o movimento pode afetar o valor corrente da função objetivo.
- 4) “Schedule Annealing” sugerido em Chistensen et al. (1993):

$$P = 1.0 - \exp(-\Delta E/T) \text{ (Equação 3.1)}$$

- Quando  $P = 1/3$  e  $\Delta E = 1$ , tem-se  $T_0 \cong 2.5$
- A temperatura é decrementada de 10% a cada 50n iterações, onde n é o número de pontos.
- Se mais que 10n mudanças de configuração tiver sucesso, a temperatura é decrementada imediatamente.
- Repetir o processo para 50 estágios de temperatura.
- Se permanecer 50n iterações de uma determinada temperatura sem aceitar um reposicionamento de texto, o algoritmo entende que é a solução final.

### 3.6 ALGORITMO GENÉTICO

Algoritmo Genético (GA) é um procedimento computacional capaz de emular o processo Darwiniano de seleção natural e mostrou ser bastante interessante em aplicações da otimização combinatória. (Holland. 1975; Goldberg. 1989; Colin. 1995)

Em analogia com o sistema biológico, cada candidato à solução do problema representa o cromossomo ou indivíduo da população e o conjunto de todos os candidatos denomina-se uma geração.

Cada indivíduo da população é representado no Algoritmo Genético por um conjunto de 0 e 1, números inteiros, números de ponto flutuante ou caracteres; e tem um custo associado que determina sua habilidade para sobreviver e produzir descendentes no processo de seleção natural.

O processo de reprodução nos Algoritmos Genéticos pode ser de simples cópia ou de cruzamento dos cromossomos pais, fazendo com que descendentes de cada geração – ou seja, as novas soluções – sejam semelhantes, possuindo muitas de suas características. Vale ressaltar ainda que os descendentes podem sofrer mutação, ou seja, o resultado dos cromossomos pode ser modificado por perturbações aleatórias.

Algoritmo:

- 1) Gerar população inicial aleatoriamente.
- 2) Repetir até alcançar a condição de término
  - Avaliar os indivíduos da população.
  - Selecionar os melhores indivíduos para reprodução
  - Fazer uma das operações:
    - Cópia.
    - Cruzamento.
    - Mutação.

Verner et al. (1997) apresentam uma aplicação de algoritmo genético com máscara ao problema de rotulação. Na aplicação, cada cromossomo representa uma configuração de distribuição de rótulos de um dado conjunto de pontos, onde cada posição no cromossomo corresponde a um ponto e pode assumir uma das posições candidatas que lhe é permitido.

Para o cálculo da função custo de cada cromossomo foram considerados a quantidade de conflitos entre os rótulos e um fator de distância que é a soma das distâncias euclidianas entre o centro do rótulo do ponto em estudo e o centro do rótulo dos próximos 4 pontos que se encontram sobrepostos com o rótulo em estudo.

$$\text{Custo} = a * q - c * d \quad (3.2)$$

onde:

q = quantidade de sobreposição;

d = fator de distância.

O autor conseguiu bons resultados com  $a = 1$  e  $c = 0.0001$

A máscara utilizada é do mesmo tamanho do cromossomo, e cada componente da máscara possui o valor 0 (com sobreposição ou muito próxima ao rótulo com sobreposição) ou 1 (sem sobreposição). Desta maneira, pontos cujo rótulo se encontram em conflito ou pontos próximos a outros pontos com conflito de rótulos será permitida a troca por outra posição candidata através da mutação e cruzamento e talvez corrigir a situação de conflito.

O seguinte exemplo explica em detalhes a operação de cruzamento com máscara para o problema de rotulação de pontos.

Seja:

P1 e P2	cromossomos pais
M1 e M2	máscaras de P1 e P2
U	sequência de 0 e 1 gerado aleatoriamente para ser usado para o cruzamento
c1 e c2	cromossomos filhos intermediários
C1 e C2	cromossomos filhos
20	número de pontos.



Neste exemplo, o cromossomo pai P1 representa uma configuração de distribuição de rótulos de 20 pontos, onde o rótulo ativo do ponto correspondente à primeira posição do cromossomo é a posição candidata 3, o rótulo ativo do ponto correspondente à segunda posição do cromossomo é a posição candidata 7 e assim por diante o cromossomo traz informações sobre as posições candidatas ativas dos 20 pontos. O cromossomo pai P2 também representa uma configuração de distribuição de rótulos de 20 pontos e cada posição do cromossomo corresponde a um ponto e assume uma das 8 posições candidatas que lhe é permitido. U é do mesmo tamanho dos cromossomos pais e é formado por uma sequência de 0 e 1 gerados aleatoriamente, para ser usado no cruzamento.

A operação de cruzamento com máscara é feito em duas etapas. Primeiro os cromossomos filhos intermediários c1 e c2 são gerados através do cruzamento dos cromossomos pais P1 e P2 e depois a opção de máscara é aplicada para gerar os cromossomos filhos.

Para geração dos cromossomos filhos intermediários c1 e c2 são levados em consideração a sequência de 0 e 1 encontrados em U. Assim sendo se em U for encontrado o valor 0 (zero), c1 recebe o gene do pai P2 e c2 recebe o gene do pai P1, mas por outro lado se o valor de U for 1 (um), c1 recebe o gene do pai P1 e c2 recebe o gene do pai P2, como pode ser visto a seguir.

P1: **37281426837516385346**

P2: 27164583552743188214

U: 01100110010010101101

c1: **27264423532713385316**

c2: **37181586857546188244**

Em seguida a opção de máscara é aplicada e os cromossomos filhos gerados.

M1 e M2 são as máscaras do cromossomo pai P1 e P2 e portanto composto de 20 posições, onde cada posição de M1 e M2 possuem o valor 0 (com sobreposição) ou 1 (sem sobreposição).

M1: 10010110000010001011  
M2: 01100001001100001010  
C1: 37184423532713385346  
C2: 37181423852716188216

Na aplicação da máscara as seguintes regras foram respeitadas:

Se  $M1 = 1$  e  $M2 = 0 \rightarrow C1 = P1$  e  $C2 = P2$   
Se  $M1 = 0$  e  $M2 = 1 \rightarrow C1 = P2$  e  $C2 = P2$   
Se  $M1 = 1$  e  $M2 = 1 \rightarrow C1 = P1$  e  $C2 = P2$   
Se  $M1 = 0$  e  $M2 = 0 \rightarrow C1 = c1$  e  $C2 = c2$

No exemplo, a primeira posição dos cromossomos filhos C1 e C2 equivale ao gene de posição 1 do pai P1, pois o valor da primeira posição da máscara M1 é 1 (um) e da máscara M2 é 0 (zero), enquanto que a terceira posição dos cromossomo filhos C1 e C2 é o gene recebido do pai P2, uma vez que o valor da terceira posição da máscara M1 é 0 (zero) e da máscara M2 é 1 (um). Por outro lado a quinta posição do cromossomo filho C1 equivale ao gene de posição 5 do cromossomo filho intermediário c1 e a quinta posição do cromossomo filho C2 equivale ao gene de posição 5 do cromossomo filho intermediário c2, uma vez que a quinta posição da máscara M1 e M2 exibem o valor 0 (zero). O gene de posição 17 do cromossomo filho C1 equivale ao gene de posição 17 do cromossomo pai P1 porque a máscara M1 nesta mesma posição, apresenta o valor 1 (um), assim como o gene de posição 17 do cromossomo filho C2 equivale ao gene de posição 17 do cromossomo pai P2, pois a máscara M2 apresenta valor 1 (um) nesta posição. E assim por diante, todas as demais posições dos cromossomos filhos C1 e C2 são formados respeitando as 4 regras acima descritas.

Os algoritmos de rotulação Busca Exaustiva, Algoritmo Guloso, “Discrete Gradient Descent” e Algoritmo de Hirsch descritos anteriormente são todos métodos que não

conseguem evitar o mínimo local e quanto maior for a densidade dos pontos maior é a probabilidade de cair no mínimo local, degradando assim o desempenho. As operações particulares que o algoritmo incorpora não permitem que se saia de um mínimo local quando este é encontrado, tornando o problema intratável pois não consegue alcançar uma solução aceitável usando qualquer algoritmo prático.

Segundo os resultados dos testes mencionados na literatura Christensen et al. (1995) e Verner et al. (1997), e mostrados no Capítulo 5, o Algoritmo Genético com máscara descrito anteriormente foi o método que alcançou o melhor resultado para rotulação. Ele conseguiu alcançar soluções de boa qualidade, porém o seu tempo de processamento deixou a desejar. Qual seria então o algoritmo adequado para o problema de rotulação? Na tentativa de se encontrar um algoritmo mais apropriado para o problema em questão, com o tempo de processamento mínimo e alta qualidade nas soluções, foi feita a abordagem do problema de rotulação de pontos, usando a Busca Tabu.

## CAPÍTULO 4

### BUSCA TABU PARA ROTULACAO CARTOGRÁFICA DE PONTOS

A Busca Tabu (“Tabu Search”) é um procedimento heurístico proposto por Fred Glover para resolver problemas de otimização combinatória. A idéia básica é evitar que a busca por soluções ótimas termine ao encontrar um mínimo local. (Glover. 1989a, 1989b, 1990; Laguna. 1994; Glover et al. 1995, 1997).

#### 4.1 DESCRIÇÃO DO MÉTODO

Busca Tabu é um algoritmo que faz uma busca agressiva no espaço de soluções do problema de otimização com o intuito de obter sempre os melhores movimentos que não sejam considerados tabu. O termo agressivo neste contexto vem do fato de que o algoritmo sempre escolhe dentre a lista de movimentos candidatos (vizinhança da solução atual) o que produzir a melhor solução, em contrapartida aos algoritmos não agressivos que aceitam a primeira solução que produz alguma melhora uma vez que não possuem o conceito de lista de candidatos.

Uma das principais características da Busca Tabu, baseia-se na sua estrutura de memória, que mantém um histórico seletivo de busca e pode apresentar 3 formas diferentes: curto, intermediário e longo prazo. A memória de curto prazo é a mais usada, porém quando o objetivo é intensificar e diversificar a busca, costuma-se usar memória de longo prazo.

A memória de curto prazo tenta realizar o melhor movimento possível, mas algumas vezes o melhor movimento piora a qualidade da solução, pois a busca está num mínimo local. Quando isto acontece o melhor movimento da próxima iteração será o mínimo local previamente encontrado, caindo em um ciclo e o objetivo das restrições tabu aqui é a de forçar a busca de novas trajetórias evitando assim os ciclos.

O sucesso das aplicações das restrições tabu, deve-se grandemente ao tamanho da lista tabu(t) (Glover et al. 1995, 1997; Furtado. 1995). As funções para determinar t são classificadas como estáticas ou dinâmicas. Nas funções estáticas o parâmetro t permanece

fixo ao longo da busca, enquanto que nas funções dinâmicas este parâmetro é alterado ao longo da busca. Os principais exemplos de tipos de funções empregados são:

- Funções estáticas: Escolhe-se um valor constante, como  $t = 7$  ou

$$t = \text{INT}[(n)^{1/2}] \quad (4.1)$$

onde:

$n$  = dimensão do problema;

INT = valor de  $t$  deve ser truncado para inteiro.

- Funções dinâmicas: Escolhe-se (aleatoriamente ou através de um modelo sistemático)  $t$  variando entre limites  $t_{\min}$  e  $t_{\max}$ , tais como  $t_{\min} = 5$  e  $t_{\max} = 11$  ou

$$t_{\min} = \text{INT}[0.9(n)^{1/2}] \quad (4.2)$$

$$t_{\max} = \text{INT}[1.1(n)^{1/2}] \quad (4.3)$$

onde:

$n$  = dimensão do problema;

INT = valor de  $t_{\min}$  e  $t_{\max}$  devem ser truncados para inteiro.

Opcionalmente, o algoritmo pode algumas vezes ignorar o movimento considerado proibido que se encontra na lista tabu. Este procedimento conhecido como critério de aspiração é usado, por exemplo, quando o movimento produzir uma solução melhor do que a melhor solução até então encontrada.

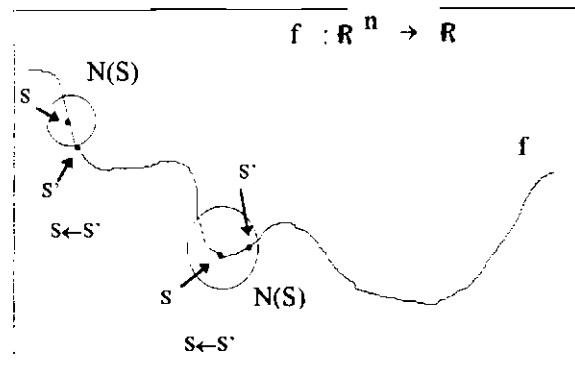


Fig. 4.1 - Busca Tabu para evitar o mínimo local.

Descrição geral do processo de busca:

- Dado  $f: \mathbb{R}^n \rightarrow \mathbb{R}$ ,  $f$  uma função definida em um conjunto de soluções viáveis  $X$  (Figura 4.1).

Seja:

- $S \in X$  uma solução viável.
- $N(S)$  uma vizinhança de uma solução
- $S' \in N(S)$  amostra de soluções na vizinhança
- Escolher a melhor  $S'$  gerada
- $S \leftarrow S'$  (modificação)
- Aceitar a modificação acima mesmo que  $f(S') > f(S)$ , para evitar o mínimo local.
- Para evitar que o procedimento entre em ciclo, usar a lista  $T$  de movimentos Tabu, que contém os últimos  $k$  movimentos realizados (Figura 4.2).
- Se  $S' \in T$  o movimento  $S \leftarrow S'$  não será permitido.

- Quando um movimento é realizado, ele é introduzido em T e (se  $k = |T|$  for fixo) o movimento mais antigo é removido de T.
- Nível de aspiração: quando  $S'$  obtida de S for “suficientemente” boa, pode se permitir o movimento  $S \leftarrow S'$ , ignorando o fato de que o movimento em questão pertence a T.
- Repetir os comandos acima para a nova solução por um número pré-definido de iterações.

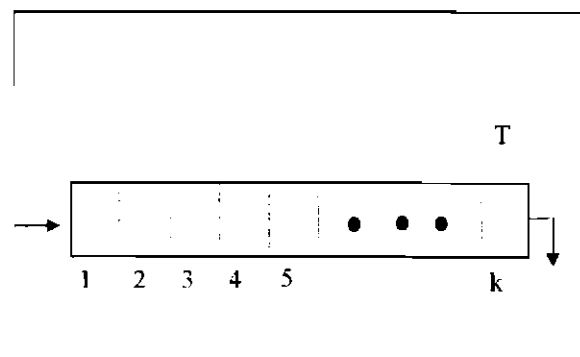


Fig. 4.2 - Lista T de soluções Tabu.

## 4.2 DESCRIÇÃO DO MÉTODO PARA ROTULAÇÃO CARTOGRÁFICA DE PONTOS

### 4.2.1 BUSCA TABU E O PROBLEMA DE ROTULAÇÃO DE 6 PONTOS

Para facilitar a compreensão, o método Busca Tabu será aplicado a um problema de rotulação de 6 pontos. Cada ponto possui 4 posições candidatas ( $L_0$ ,  $L_1$ ,  $L_2$  e  $L_3$ ) e cada posição candidata possui o seu custo (0.0, 0.4, 0.6 e 0.9) (Figura 4.3).

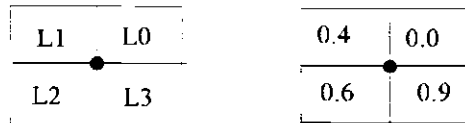


Fig. 4.3 - Posições candidatas e preferência cartográfica.

O algoritmo aqui utilizado para resolver o problema de rotulação contabiliza conflitos de duas maneiras, a saber (Figura 4.4):

- Número de conflitos = 4 (P1 = 0, P2 = 1, P3 = 2, P4 = 1)
- Número de rótulos em conflito = 3

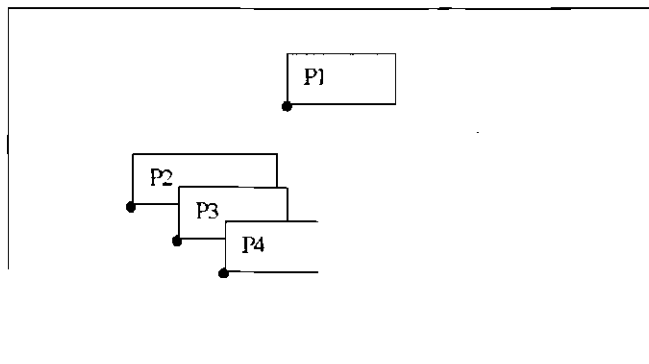


Fig. 4.4 - Quantidade de conflitos.

Número de conflitos: contabiliza os conflitos de cada ponto, com a finalidade de escolher os pontos que farão parte da vizinhança e também, para selecionar o próximo movimento. Detalhes se encontram em 4.2.4 e 4.2.5.

Número de rótulos em conflito: contabiliza a quantidade de conflitos existentes no leiaute, para ser utilizado nos cálculos do tamanho da vizinhança e tamanho da lista tabu



descritos no Capítulo 5. É também a quantidade de rótulos em conflito do leiaute final que esta sendo mostrado ao usuário.

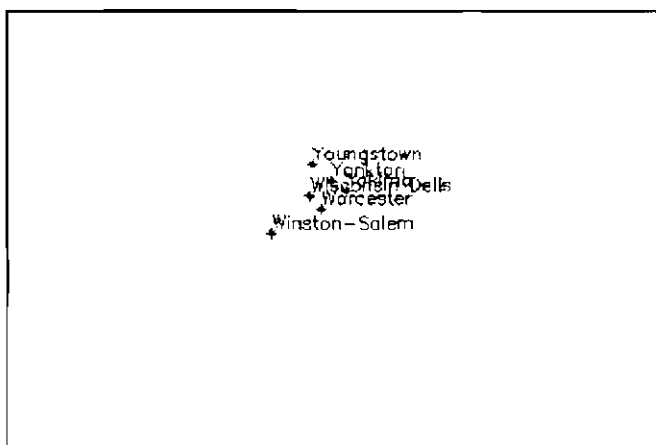


Fig. 4.5 - Configuração Inicial

A Figura 4.5 mostra uma configuração inicial do exemplo de 6 pontos com 12 conflitos, onde o texto correspondente aos pontos são: P0 = Youngstown de 10 caracteres, P1 = Yankton de 7 caracteres, P2 = Yakima de 6 caracteres, P3 = Worcester de 9 caracteres, P4 = Wisconsin Dells de 15 caracteres e P5 = Winston-Salem de 13 caracteres. Os 12 conflitos acima citados são: 1 conflito entre o rótulo L0 do ponto P0 e o rótulo L0 do ponto P1, 3 conflitos entre o rótulo L0 do ponto P1 e o rótulo L0 dos pontos P0 P2 P4, 3 conflitos entre o rótulo L0 do ponto P2 e o rótulo L0 dos pontos P1 P3 P4, 2 conflitos entre o rótulo L0 do ponto P3 e o rótulo L0 dos pontos P2 P4 e 3 conflitos entre o rótulo L0 do ponto P4 e o rótulo L0 dos pontos P1 P2 P3.

Como visto acima cada ponto contabiliza o seu número de conflitos, ou seja se o rótulo ativo L0 do ponto P0 conflita com o rótulo ativo L0 do ponto P1, então o número de conflitos do ponto P0 assim como o número de conflitos do ponto P1 serão incrementados de 1.

#### 4.2.2 PRÉ-COMPUTAÇÃO

O objetivo aqui é pré-computar todos os conflitos existentes, com a finalidade de agilizar o cálculo da função objetivo. Não foi usado aqui nenhum algoritmo especial para se

fazer a pré-computação, porém o tempo gasto nesta etapa não é grande, flutuando em torno do tempo gasto pelo algoritmo Busca Tabu, que se encontra na (Tabela 5.6).

De acordo com o tamanho dos rótulos, a (Figura 4.6) mostra as posições candidatas dos 6 pontos.

P0		P1		P2		P3		P4		P5	
L1	L0	L1	L0	L1	L0	L1	L0	L1	L0	L1	L0
L2	L3	L2	L3	L2	L3	L2	L3	L2	L3	L2	L3

Fig. 4.6 - Posições candidatas.

Todos os conflitos existentes entre os rótulos candidatos são registrados (Tabela 4.1). O ponto P0 possui as posições candidatas L0, L1, L2 e L3; a posição candidata L0 do ponto em questão conflita com a posição candidata L0 e L1 do ponto P1, assim como a posição candidata L1 do ponto P0 conflita com a posição candidata L1 do ponto P1; desta forma, o registro é feito para todas as posições candidatas dos pontos (Tabelas 4.2, 4.3, 4.4, 4.5 e 4.6).

**TABELA 4.1 – DIAGRAMA DE CONFLITOS DO PONTO P0**

<b>P0</b>	L0	P1	L0 L1
	L1	P1	L1
	L2	P1	L1 L2
		P2	L1
		P3	L1
		P4	L0 L1
	L3	P1	L0 L1 L2 L3
		P2	L0 L1
		P3	L0 L1
		P4	L0

**TABELA 4.2 – DIAGRAMA DE CONFLITOS DO PONTO P5**

<b>P5</b>	L0	P2	L2 L3
		P3	L2 L3
		P4	L2 L3
	L1	P3	L2
		P4	L2
	L2		
	L3		

**TABELA 4.3 – DIAGRAMA DE CONFLITOS DO PONTO P1**

P1	L0	P0	L0 L3
		P2	L0 L1
		P4	L0
	L1	P0	L0 L1 L2 L3
		P2	L1
		P4	L0 L1
		P0	L2 L3
	L2	P2	L1 L2
		P3	L0 L1
		P4	L0 L1 L2 L3
		P0	L3
	L3	P2	L0 L1 L2 L3
		P3	L0
		P4	L0 L3

**TABELA 4.4 – DIAGRAMA DE CONFLITOS DO PONTO P2**

P2	L0	P0	L3
		P1	L0 L3
		P3	L0
		P4	L0
	L1	P0	L2 L3
		P1	L0 L1 L2 L3
		P3	L0 L1
		P4	L0 L1
	L2	P1	L2 L3
		P3	L0 L1 L2 L3
		P4	L0 L1 L2 L3
		P5	L0
	L3	P1	L3
		P3	L0 L3
		P4	L0 L3
		P5	L0

**TABELA 4.5 – DIAGRAMA DE CONFLITOS DO PONTO P3**

P3	L0	P0	L3
		P1	L2 L3
		P2	L0 L1 L2 L3
		P4	L0 L3
	L1	P0	L2 L3
		P1	L2
		P2	L1 L2
		P4	L0 L1 L2 L3
	L2	P2	L2
		P4	L2 L3
		P5	L0 L1
	L3	P2	L2 L3
		P4	L3
		P5	L0

**TABELA 4.6 – DIAGRAMA DE CONFLITOS DO PONTO P4**

P4	L0	P0	L2 L3
		P1	L0 L1 L2 L3
		P2	L0 L1 L2 L3
		P3	L0 L1
	L1	P0	L2
		P1	L1 L2
		P2	L1 L2
	L2	P3	L1
		P1	L2
		P2	L2
		P3	L1 L2
	L3	P5	L0 L1
		P1	L2 L3
		P2	L2 L3
		P3	L0 L1 L2 L3
		P5	L0

#### 4.2.3 CONFIGURAÇÃO INICIAL

A configuração inicial considerada foi a de rotular cada ponto com a posição candidata de melhor preferência cartográfica, assim sendo para o exemplo, o rótulo ativo nos pontos P0 = L0, P1 = L0, P2 = L0, P3 = L0, P4 = L0 e P5 = L0.

#### 4.2.4 MUDANÇA DE CONFIGURAÇÃO

Em rotulação de pontos, a vizinhança é composta por uma quantidade de pontos de custo mais alto, dentre todos os pontos do espaço em estudo da configuração corrente. A cada ponto pertencente a vizinhança, lhe é permitido determinar para a solução seguinte o menor custo, escolhendo dentre os seus rótulos candidatos, aquela que produz o menor custo. Assim sendo, cada ponto pertencente a vizinhança tem agora associado a ele, um custo da solução seguinte e o movimento, ou seja, o rótulo a ser ativado.

O algoritmo utilizado para resolver o problema de rotulação é um algoritmo agressivo de busca em vizinhança. A escolha da próxima solução é feita selecionando dentre os movimentos da vizinhança, aquele que não é considerado tabu e que apresenta o menor custo, pois a função objetivo é de minimização ou seja quanto menor o custo total, maior é a qualidade da solução e a estética do leiaute.

A cada 50 iterações, o tamanho da vizinhança foi recalculado usando a expressão:  $1 + \text{INT}(0.05 * \text{num. de rótulos em conflito})$ . O fator 0.05 foi escolhido depois de testes feitos em 9 diferentes configurações de 1000 pontos. A média de rótulos sem conflito das 9 diferentes configurações para os fatores 0.03, 0.04, 0.05, 0.06 e 0.07 mostraram que o fator 0.05 obteve os melhores resultados. Os detalhes encontram-se no Capítulo 5.

#### 4.2.5 MEMÓRIA DE LONGO PRAZO

A composição da vizinhança é feita por um conjunto de pontos de custo mais alto da configuração corrente e o custo destes pontos é então reduzido, para a solução seguinte, escolhendo dentre os seus rótulos candidatos, aquela que produz o menor custo. Assim sendo, os pontos escolhidos para fazer parte da vizinhança são aqueles de custo mais alto, mas muitas vezes o custo de muitos pontos são iguais, principalmente quando a preferência cartográfica não esta sendo contabilizada, e nestes casos quase sempre os mesmos pontos fazem parte da vizinhança. Para diversificar a busca foi introduzido o conceito de frequência normalizada que visa contar a frequência com que o ponto foi visitado. A cada 50 iterações estes contadores são normalizados, dividindo-os pelo seu valor máximo, de tal forma que o ponto mais visitado sofrerá um decréscimo maior no



seu custo do que aqueles que não foram muito visitados. Desta forma, o uso da informação de frequência irá penalizar movimentos que não causam melhora, fazendo com que perca a sua atratividade.

O custo “C(i)” de cada ponto “i” é dado por:

$$C(i) = \alpha_1 \text{ conflitos}(i) + \alpha_2 \text{ preferência}(i) - \text{frequência normalizada}(i) \quad (4.4)$$

onde:

**conflitos (i)** = número de rótulos em conflito no ponto i;

**preferência (i)** = preferência cartográfica ativa dos rótulos em conflito no ponto i;

$\alpha_1$  = nível de consideração a ser dada aos rótulos em conflito;

$\alpha_2$  = nível de consideração a ser dada a preferência cartográfica, quando  $\alpha_2 = 0$ , a preferência cartográfica não será levada em consideração;

Frequência normalizada(i) = diversifica a busca.

Desta forma, no problema de rotulação de pontos foi utilizado a memória de longo prazo, com a finalidade de diversificar a busca, penalizando assim a presença freqüente de pontos que não causam melhora na vizinhança e valorizando outros pontos pouco freqüentes na vizinhança através do uso de memória de frequência.

#### 4.2.6 FUNÇÃO OBJETIVO

A busca tabu aqui utilizada é totalmente determinística e seleciona agressivamente os melhores movimentos admissíveis, logo existe a necessidade de examinar e comparar as opções de movimento, o que acarreta um grande número de cálculos para o seu sucesso, principalmente quando o número de pontos a ser rotulado for grande. Assim sendo as melhores funções objetivo são aquelas em que o custo pode ser computado facilmente,

tornando assim a busca eficiente, ao mesmo tempo que soluções de qualidade possam ser alcançadas beneficiando-se da melhor configuração de rótulos.

E a função objetivo de minimização adotada foi:

$$\sum C(i) \quad (4.5)$$

onde:

$i = 1, 2, 3, \dots$ , número de pontos;

$C(i)$  = custo de cada ponto definido na seção 4.2.5.

#### 4.2.7 LISTA DE SOLUÇÕES TABU

Para solução do problema de rotulação de pontos utilizamos também a memória de curto prazo. Na lista tabu, são guardados os últimos pontos visitados, proibindo assim que se faça movimentos ou melhor trocar o rótulo ativo destes pontos.

Na maioria das aplicações da busca tabu descritas na literatura, o tamanho da lista tabu é estática (Bland e Dawson. 1991). Porém o tamanho da lista tabu implementada neste trabalho é dinâmico, pois o problema de rotulação de pontos necessita de uma lista tabu grande no início para resolver os conflitos entre os rótulos de pontos distintos da carta, evitando assim que se concentre em resolver os conflitos só de uma determinada região da carta. Contudo, quando o número de conflitos diminui, existe a necessidade de uma lista tabu pequena, uma vez que a busca deve ser realizada em algumas regiões da carta, visando apenas um ajuste final. O tamanho da lista tabu adotado foi de  $7 + \text{INT}(0.25 * \text{num. de rótulos em conflito})$  pois assim com o decorrer das iterações o número de rótulos em conflito diminui e em consequência o tamanho da lista tabu. O coeficiente 0.25 assim como o recálculo do tamanho da lista tabu a cada 50 iterações foram estabelecidas depois de testes e experimentos que foram feitos em configurações com 100, 250, 500, 750 e 1000 pontos. Os detalhes encontram-se no Capítulo 5.

Como o tamanho da lista escolhida é dinâmico, ou seja, diminui a cada 50 iterações de acordo com o número de conflitos existentes, adotou-se manter os últimos  $n$  elementos da lista tabu - onde  $n$  é o tamanho da lista corrente - no momento do decréscimo do ta-

manho da lista. Mas por que não reinicializar a lista tabu com zero no momento do decréscimo? Foram feitos alguns testes reinicializando a lista tabu, mas os resultados obtidos não foram tão bons quanto os que foram obtidos mantendo os últimos elementos da lista.

#### **4.2.8 CRITÉRIO DE ASPIRAÇÃO**

Segundo Glover et al. (1995, 1997) e Furtado (1995), existem vários critérios de aspiração, mas a maioria das aplicações de busca tabu, descritas na literatura, utiliza aspiração por objetivo na forma chamada global, onde um critério de aspiração é satisfeito quando o movimento produzir uma solução melhor do que a melhor solução encontrada até então.

O critério de aspiração adotado para o problema de rotulação de pontos foi a aspiração por objetivo na forma chamada global e aspiração “default”. Neste problema a aspiração “default” é utilizada quando todos os movimentos possíveis são classificados como tabu e o critério de aspiração por objetivo não for satisfeito. Neste caso o movimento selecionado é o de mais baixo status tabu.

#### **4.2.9 SIMULAÇÃO PARA ROTULAÇÃO DE 6 PONTOS**

Foram utilizados  $\alpha_1 = \alpha_2 = 1$  ou seja foram consideradas a preferência cartográfica e número de conflitos para o exemplo.

Por se tratar de uma simulação de apenas 6 pontos adotou-se a expressão:  $2 + \text{INT}(0.25 * \text{num. de rótulos em conflito})$  para o cálculo do tamanho da lista tabu, porém se a lista for maior do que 4, o tamanho da lista será obrigatoriamente 4. O tamanho da vizinhança adotada foi de  $2 + \text{INT}(0.05 * \text{num. de rótulos em conflito})$ .

Segundo os testes feitos, notou-se que o cálculo da frequência normalizada, assim como o recálculo do tamanho da lista tabu e o recálculo do tamanho da vizinhança só trazem bons resultados se forem aplicados depois de se ter processado no mínimo 50 iterações, mas na simulação de 6 pontos adotou-se fazer os recálculos a cada 5 iterações com o intuito de mostrar passo a passo como o algoritmo está resolvendo os conflitos existen-

tes entre os rótulos. Por este motivo, o recálculo está de certa forma atrapalhando o bom funcionamento do algoritmo, que podia ter chegado ao estado de não conflito em apenas 8 iterações.

A seguir será mostrado passo a passo como está sendo resolvido o problema de conflito do exemplo dado de 6 pontos.

**Estado inicial:** tamanho da lista tabu = 4

Tamanho da vizinhança = 2

Foram consideradas 5 iterações para o recálculo de:

- tamanho da lista tabu,
- tamanho da vizinhança e
- frequência normalizada.

PONTO	P0	P1	P2	P3	P4	P5
ROTULO ATIVO	L0	L0	L0	L0	L0	L0
CUSTO	1.0	3.0	3.0	2.0	3.0	0.0
TABU	0	0	0	0	0	0
FREQUENCIA	0	0	0	0	0	0
FREQ. NORM.	0.0	0.0	0.0	0.0	0.0	0.0

Custo total = 12.0

Melhor custo = 12.0

Número de conflitos = 12

#### Iteração 0:

Os pontos que possuem o maior custo são o ponto P1 com o rótulo L0 e o ponto P2 com o rótulo L0, ambos de custo 3.0, portanto a vizinhança será formada por estes dois pontos, isto é P1 e P2. As 4 posições candidatas do ponto P1 são avaliadas e escolhida a de menor custo, que no caso é o rótulo L1 de custo 2.4; o mesmo procedimento é aplicado a P2.

É possível notar que dentre os 2 pontos – P1 e P2 – o de menor custo é P1 e o seu status tabu é zero, portanto ele será escolhido e o resultado da escolha pode ser visto abaixo. E assim por diante, o procedimento descrito para a obtenção da próxima configuração é

executado em todas as iterações, até alcançar o número de conflitos igual a zero ou o número de iterações desejado.

Mas quando a preferência cartográfica é considerada, por que finalizar quando o número de rótulos em conflito for igual a zero? A configuração inicial adotada foi a de rotular cada ponto com a posição candidata de melhor preferência cartográfica, mas isto causou os conflitos entre os rótulos, ou seja os pontos rotulados na melhores posições candidatas não apresentou uma boa configuração. O algoritmo Busca Tabu, tenta então, eliminar os conflitos mantendo sempre que possível as melhores posições candidatas, assim sendo, quando o número de rótulos em conflitos for igual a zero, teremos uma configuração com os rótulos colocados, na medida do possível, nas melhores posições candidatas.

vizinhança		
PONTO	P1	P2
MELHOR ROTULO	L1	L1
CUSTO	2.4	3.4

PONTO	P0	P1	P2	P3	P4	P5
ROTULO ATIVO	L0	L1	L0	L0	L0	L0
CUSTO	1.4	2.4	2.0	2.0	3.4	0.0
TABU	0	4	0	0	0	0
FREQUÊNCIA	0	1	0	0	0	0
FREQ. NORM.	0.0	0.0	0.0	0.0	0.0	0.0

Custo total = 11.2

Melhor custo = 11.2

Número de conflitos = 10

**Iteração 1:**

vizinhança		
PONTO	<b>P4</b>	P1
MELHOR ROTULO	<b>L2</b>	L2
CUSTO	<b>1.6</b>	2.6

PONTO	P0	P1	P2	P3	P4	P5
ROTULO ATIVO	L0	L1	L0	L0	L2	L0
CUSTO	1.4	1.4	1.0	1.0	<b>1.6</b>	<b>1.6</b>
TABU	0	3	0	0	4	0
FREQUÊNCIA	0	1	0	0	1	0
FREQ. NORM.	0.0	0.0	0.0	0.0	0.0	0.0

Custo total = 8.0

Melhor custo = 8.0

Número de conflitos = 6

**Iteração 2:**

vizinhança		
PONTO	<b>P5</b>	P4
MELHOR ROTULO	<b>L2</b>	L1
CUSTO	<b>0.6</b>	1.8

PONTO	P0	P1	P2	P3	P4	P5
ROTULO ATIVO	L0	L1	L0	L0	L2	L2
CUSTO	<b>1.4</b>	<b>1.4</b>	1.0	1.0	0.6	0.6
TABU	0	2	0	0	3	4
FREQUÊNCIA	0	1	0	0	1	1
FREQ. NORM.	0.0	0.0	0.0	0.0	0.0	0.0

Custo total = 6.0

Melhor custo = 6.0

Número de conflitos = 4

### Iteração 3:

vizinhança		
PONTO	P0	P1
MELHOR ROTULO	L1	L0
CUSTO	1.8	2.0

PONTO	P0	P1	P2	P3	P4	P5
ROTULO ATIVO	L1	L1	L0	L0	L2	L2
CUSTO	1.8	1.8	1.0	1.0	0.6	0.6
TABU	4	1	0	0	2	3
FREQUÊNCIA	1	1	0	0	1	1
FREQ. NORM.	0.0	0.0	0.0	0.0	0.0	0.0

Custo total = 6.8

Melhor custo = 6.0

Número de conflitos = 4

### Iteração 4:

Nesta iteração é possível verificar que o status tabu do ponto P0 é 4 e do ponto P1 é 1, mas o ponto P1 com o custo 1.0 foi escolhido pois o custo total da nova configuração é 5.6 que é melhor do que o melhor custo total até então alcançado que é 6.0.

vizinhança		
PONTO	P1	P0
MELHOR ROTULO	L0	L0
CUSTO	1.0	1.4

PONTO	P0	P1	P2	P3	P4	P5
ROTULO ATIVO	L1	L0	L0	L0	L2	L2
CUSTO	0.4	1.0	2.0	1.0	0.6	0.6
TABU	3	4	0	0	1	2
FREQUÊNCIA	1	2	0	0	1	1
FREQ. NORM.	0.0	0.0	0.0	0.0	0.0	0.0

Custo total = 5.6

Melhor custo = 5.6

Número de conflitos = 4

### Iteração 5:

Vizinhança		
PONTO	P1	P2
MELHOR ROTULO	L1	L3
CUSTO	1.8	1.9

PONTO	P0	P1	P2	P3	P4	P5
ROTULO ATIVO	L1	L0	L3	L0	L2	L2
CUSTO	0.4	0.0	1.9	1.9	0.6	0.6
TABU	2	3	4	0	0	1
FREQUÊNCIA	1	2	1	0	1	1
FREQ. NORM.	0.0	0.0	0.0	0.0	0.0	0.0

Custo total = 5.4

Melhor custo = **5.4**

Número de conflitos = 2

Nesta iteração foram feitos recálculos do tamanho da lista tabu, tamanho da vizinhança, frequência normalizada e o novo custo de cada ponto com o efeito da frequência normalizada. A frequência normalizada 1(um), passa a ser 0.9 para evitar que fique com custo 0 (zero), significando sem nenhum conflito, para os casos que apresentam 1 (um) conflito.

- Tamanho da lista tabu = 2
- Tamanho da vizinhança = 2

PONTO	P0	P1	P2	P3	P4	P5
ROTULO ATIVO	L1	L0	L3	L0	L2	L2
CUSTO	0.01	0.0	<b>1.44</b>	<b>1.9</b>	0.14	0.14
TABU	0	1	2	0	0	0
FREQUÊNCIA	1	2	1	0	1	1
FREQ. NORM.	0.45	0.9	0.45	0	0.45	0.45



**Iteração 6:**

Vizinhança		
PONTO	P2	P3
MELHOR ROTULO	L0	L1
CUSTO	1.54	2.0

PONTO	P0	P1	P2	P3	P4	P5
ROTULO ATIVO	L1	L0	L0	L0	L2	L2
CUSTO	0.01	0.09	1.54	1.0	0.14	0.14
TABU	0	0	2	0	0	0
FREQUÊNCIA	1	2	2	0	1	1
FREQ. NORM.	0.45	0.9	0.45	0	0.45	0.45

Custo total = 2.93

Melhor custo = **2.93**

Número de conflitos = 4

**Iteração 7:**

vizinhança		
PONTO	P3	P2
MELHOR ROTULO	L3	L3
CUSTO	0.9	1.44

PONTO	P0	P1	P2	P3	P4	P5
ROTULO ATIVO	L1	L0	L0	L3	L2	L2
CUSTO	0.01	0.09	0.54	0.9	0.14	0.14
TABU	0	0	1	2	0	0
FREQUÊNCIA	1	2	2	1	1	1
FREQ. NORM.	0.45	0.9	0.45	0	0.45	0.45

Custo total = 1.83

Melhor custo = 1.83

Número de conflitos = 2

**Iteração 8:**

vizinhança		
PONTO	<b>P4</b>	P3
MELHOR ROTULO	<b>L1</b>	L0
CUSTO	<b>0.01</b>	1.0

PONTO	P0	P1	P2	P3	P4	P5
ROTULO ATIVO	L1	L0	L0	L3	L1	L2
CUSTO	0.01	0.09	<b>0.54</b>	<b>0.9</b>	0.01	0.14
TABU	0	0	0	1	2	0
FREQUÊNCIA	1	2	2	1	2	1
FREQ. NORM.	0.45	0.9	0.45	0	0.45	0.45

Custo total = 1.7

Melhor custo = 1.7

Número de conflitos = 2

**Iteração 9:**

vizinhança		
PONTO	<b>P3</b>	P2
MELHOR ROTULO	<b>L2</b>	L1
CUSTO	<b>0.6</b>	2.34

PONTO	P0	P1	P2	P3	P4	P5
ROTULO ATIVO	L1	L0	L0	L2	L1	L2
CUSTO	0.01	0.09	<b>0.54</b>	<b>0.6</b>	0.01	0.14
TABU	0	0	0	2	1	0
FREQUÊNCIA	1	2	2	2	2	1
FREQ. NORM.	0.45	0.9	0.45	0	0.45	0.45

Custo total = 1.4

Melhor custo = 1.4

Número de conflitos = 2

### Iteração 10:

vizinhança		
PONTO	P2	P3
MELHOR ROTULO	L3	L3
CUSTO	0.44	0.9

PONTO	P0	P1	P2	P3	P4	P5
ROTULO ATIVO	L1	L0	L3	L2	L1	L2
CUSTO	0.01	0.0	0.44	0.6	0.01	0.14
TABU	0	0	2	1	0	0
FREQUÊNCIA	1	2	3	2	2	1
FREQ. NORM.	0.45	0.9	0.45	0	0.45	0.45

Custo total = 1.2

Melhor custo = 1.2

Número de conflitos = 0

O custo total final 1.2, demonstra que apesar de não ter conflitos entre os rótulos, não foi possível colocar todos os rótulos na melhor posição cartográfica.

A Figura 4.7 mostra o resultado da aplicação do método busca tabu no exemplo dado de 6 pontos. Alcançou-se a configuração mostrada em 10 iterações, conseguiu-se sanar todos os conflitos e o custo final foi 1.2 .

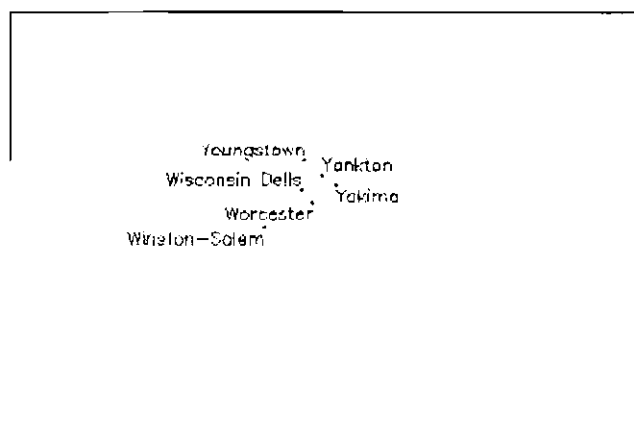


Fig. 4.7 - Após a aplicação do algoritmo Busca Tabu, descrito na configuração da Figura 4.5.

## CAPÍTULO 5

### RESULTADOS EXPERIMENTAIS

Com o intuito de determinar os parâmetros tais como, o tamanho da lista tabu, o tamanho da vizinhança e o número de iterações a ser executado para se fazer os recálculos da frequência normalizada, tamanho da lista tabu e tamanho da vizinhança, foram feitos alguns experimentos que serão detalhados a seguir.

O primeiro parâmetro ajustado foi o tamanho da lista tabu que no caso da rotulação de pontos precisa acompanhar o número de rótulos em conflito, ou seja, quando o número de rótulos em conflito for grande, o tamanho da lista tabu também deve ser grande o bastante para resolver os conflitos entre os rótulos de pontos distintos da carta, evitando assim que se concentre em resolver os conflitos só de uma determinada região da carta. Mas quando o número de rótulos em conflito for pequeno, o tamanho da lista tabu deve ser pequeno para que se realize a busca em algumas regiões da carta, visando assim apenas um ajuste final. Portanto, a expressão adotada para cálculo do tamanho da lista tabu foi  $7 + \text{INT}(x * \text{número de rótulos em conflito})$  onde 7 é o tamanho mínimo da lista e foi escolhido porque segundo Glover et al. (1995) os valores entre 7 e 20 trazem bons resultados para uma variedade de classes de problemas. Para a escolha do fator  $x$  foram feitos alguns experimentos em uma configuração de 1000 pontos fixando os seguintes parâmetros:

- Número de iterações para recálculo: a cada 15 iterações sem melhoria
- Número máximo de iterações = 3000
- Altura do caracter que compõe o texto = 1.0 mm
- Tamanho da vizinhança = 1
- Posições candidatas = 8

O tamanho da vizinhança aqui adotado é 1, porque na época que foram feitos os experimentos para estabelecer o tamanho da lista tabu, ainda não tinha sido incorporado ao código da Busca Tabu o conceito de vizinhança global explicado no capítulo anterior, portanto é como se estivéssemos trabalhando com tamanho de vizinhança igual a 1. Outro parâmetro que ainda não tinha sido estabelecido na época é o número de iterações para recálculo e foi então adotado o recálculo a cada 15 iterações sem melhoria. O número de rótulos em conflito mostrado na Tabela 5.1 são os conflitos mínimos alcançados em 3000 iterações.

Os resultados obtidos nos testes mostraram que o valor razoável para  $x$  é 0.25, pois mostrou o menor número de rótulos em conflito ( Tabela 5.1).

**TABELA 5.1 – O MELHOR FATOR PARA O TAMANHO DA LISTA TABU**

Fator $x$ da (Equação 5.1)	Número de rótulos em conflito
0.10	150
0.20	139
<b>0.25</b>	<b>136</b>
0.30	165

$$7 + \text{INT}( x * nc ) \tag{5.1}$$

onde:

$nc$  = número de rótulos em conflito a cada 15 iterações sem melhora;

$x$  = fator a ser testado.

O segundo parâmetro ajustado foi a escolha do número de iterações para se fazer o recálculo da frequência normalizada, tamanho da vizinhança e tamanho da lista tabu. Novamente foram feitos alguns experimentos em uma configuração de 1000 pontos fixando os seguintes parâmetros:

- Tamanho da lista tabu =  $7 + \text{INT}(0.25 * \text{número de rótulos em conflito})$
- Tamanho da vizinhança = 45
- Posições candidatas = 8
- Altura do caracter que compõe o texto = 1.0 mm
- Número máximo de iterações = 30000

Os resultados da Tabela 5.2 mostram que quando os recálculos são feitos a cada 50 iterações, o número de rótulos em conflito é menor. Assim sendo adotamos fazer os recálculos a cada 50 iterações.

**TABELA 5.2 – NÚMERO DE ITERAÇÕES PARA RECÁLCULOS**

Número de iterações	Número de rótulos em conflito
48	130
49	121
<b>50</b>	<b>119</b>
51	124
52	127
140	131

Talvez a melhor política a ser adotada aqui fosse verificar o melhor número de iterações para cada caso separadamente, ou seja encontrar o melhor número de iterações para recálculo da frequência normalizada, o melhor número de iterações para recálculo do tamanho da lista tabu e o melhor número de iterações para recálculo do tamanho da vizinhança. Este detalhe talvez venha melhorar ainda mais os resultados, portanto fica como sugestão para testes futuros.

O último parâmetro ajustado foi o tamanho da vizinhança explicado no capítulo anterior. A expressão adotada para cálculo do tamanho da vizinhança foi  $1 + \text{INT}(x * \text{número de rótulos em conflito})$ , pois o número de rótulos em conflito influencia no tamanho da vizinhança a ser adotada. O 1 (um) assegura que o tamanho da vizinhança nunca será zero. Neste caso foram feitos experimentos em 9 diferentes configurações de 1000 pontos fixando os seguintes parâmetros:

- Tamanho da lista tabu =  $7 + \text{INT}(0.25 * \text{número de rótulos em conflito})$
- Posições candidatas = 8
- Altura do caracter que compõe o texto = 1.0 mm
- Número máximo de iterações = 30000
- Número de iterações para o recálculo = 50

O fator  $x = 0.05$  foi aqui adotado, pois a média de rótulos em conflito das 9 diferentes configurações mostrou que o fator 0.05 obteve os melhores resultados (Tabela 5.3).

**TABELA 5.3 – O MELHOR FATOR PARA O TAMANHO DA VIZINHANÇA**

Fator x da (Equação 5.2)	Média de rótulos em conflito das 9 diferentes configurações
0.03	110.2
0.04	108.7
0.045	108.4
<b>0.05</b>	<b>107.4</b>
0.06	110.2
0.07	110.6

$$1 + \text{INT}(x * nc) \tag{5.2}$$

onde:

nc = número de rótulos em conflito a cada 50 iterações;  
x = fator a ser testado.

### 5.1 PROBLEMA REAL

Com o objetivo de verificar o desempenho do algoritmo Busca Tabu com relação a dados reais – ocorrência natural de distribuição de pontos – foram utilizados os dados disponíveis em Knuth (1993). Trata-se de um conjunto de 128 pontos referente às cidades de uma região dos Estados Unidos da América e seus respectivos nomes de comprimento variável, o que torna o teste bastante realístico.

Com as informações disponíveis em Knuth (1993) e o uso do SIG SPRING foi possível importar os dados em questão. A área considerada em coordenadas geográficas foi:

Longitude: O 123 0 0 até O 73 0 0

Latitude: N 25 0 0 até N 51 0 0

Projeção: LAMBERT / HAYFORD

Com os dados reais obtidos foram feitos testes com diferentes valores de  $\alpha 1$  (que manipula o nível de consideração a ser dada aos rótulos em conflito) e  $\alpha 2$  (que manipula o nível de consideração a ser dada a preferência cartográfica), uma vez que a nossa função objetivo de minimização (Equação 4.5) descrito em 4.2.5 é:

$$\sum C_i \quad \text{para } i = 1, 2, 3, \dots, \text{número de pontos}$$

onde:

“ $C_i$ ” é o custo de cada ponto:

$$C_i = \alpha 1 \sum \text{conflitos} + \alpha 2 \sum \text{preferência} - \text{frequência normalizada}$$

Para os testes foram considerados:



- Tamanho da lista tabu:  $7 + \text{INT}(0.25 * \text{num. de rótulos em conflito})$
- Tamanho da vizinhança:  $1 + \text{INT}(0.05 * \text{num. de rótulos em conflito})$
- Número de iterações para recálculo: 50
- Altura do caracter que compõe o texto: 1.0 mm
- Posições candidatas: 8

A Tabela 5.4 abaixo mostra os resultados dos testes:

**TABELA 5.4 – RESULTADOS DOS TESTES PARA DIFERENTES VALORES DE  $\alpha_1$  E  $\alpha_2$**

$\alpha_1$	1	1	1	3
$\alpha_2$	10	5	1	1
Iteração	324	140	28	28
Num. de rótulos em conflito	26	4	0	0

Verificou-se que quando o nível de consideração da preferência cartográfica é bastante grande em relação ao nível de consideração dada ao número de rótulos em conflito, o número de rótulos em conflito é grande, mas é possível verificar que os rótulos ocupam a melhor posição cartográfica, na medida do possível. Por outro lado, a medida que o nível de consideração da preferência cartográfica diminui e o nível de consideração do número de rótulos em conflito aumenta, o número de rótulos em conflito diminui, mas verifica-se que os rótulos ocupam qualquer uma das 8 posições candidatas. Os leiautes dos 4 testes da Tabela 5.4 acima se encontram no Apêndice A4.

**TABELA 5.5 – RESULTADOS DOS TESTES PARA DIFERENTES ALTURA DE CARACTER**

Altura do caracter	1.0 mm	1.5 mm
Iteração	28	583
Num. de rótulos em conflito	0	6

Em seguida fixou-se  $\alpha_1 = 3$  e  $\alpha_2 = 1$ ; outras considerações tais como tamanho da lista tabu, tamanho da vizinhança, número de iterações para realizar os recálculos devidos e número de posições candidatas foram mantidas e desta forma foram feitos testes para altura do caracter - que compõe o texto do nome da cidade – igual a 1.0 mm e 1.5mm. O resultado é mostrado na Tabela 5.5 e os respectivos leiautes se encontram no Apêndice A4. Verificou-se aqui que o tamanho do caracter afeta de maneira considerável o leiaute final da carta.

Verificou-se que a escala também afeta consideravelmente o leiaute final da carta. Testes foram feitos para escala 1:23000000 e 1:20000000, fixando  $\alpha_1 = 3$ ,  $\alpha_2 = 1$  e altura do caracter = 1.5 mm; outras considerações tais como tamanho da lista tabu, tamanho da vizinhança, número de iterações para realizar os recálculos devidos e número de posições candidatas foram mantidas. O resultado é mostrado na Tabela 5.6 abaixo e os respectivos leiautes se encontram no Apêndice A4.

**TABELA 5.6 – RESULTADOS DOS TESTES PARA DIFERENTES ESCALA**

Escala	1:23000000	1:20000000
Iteração	1070	4812
Num. de rótulos em conflito	7	2

## 5.2 ANÁLISE COMPARATIVA COM A LITERATURA

Christensen et al. (1995) e Verner et al. (1997) compararam vários algoritmos usando um conjunto padrão de dados gerados randomicamente:

- Região de tamanho 792 x 612 unidades
- Rótulo de tamanho fixo 30 x 7 unidades
- Folha de papel de tamanho 11 x 8.5 polegada

Para comparar a Busca Tabu com outros algoritmos, foi usado o mesmo conjunto padrão de dados e as mesmas condições impostas pelos autores:

- Número de pontos:  $n = 100, 250, 500, 750, 1000$
- Para cada  $n$ , gerar 25 configurações diferentes de distribuição aleatória de pontos através do uso de diferentes sementes.
- Para cada  $n$  calcular a média percentual do número de rótulos sem conflito das 25 configurações.
- Não foi determinada nenhuma penalidade para os rótulos que se situam além do limite da região.
- Foram consideradas 4 posições candidatas.
- Não foi considerada a preferência cartográfica (todas as posições candidatas são igualmente desejáveis).
- Não houve seleção de pontos (não deleta ponto ou rótulo que estiver em conflito na configuração final).
- Os parâmetros utilizados pela Busca Tabu foram:
  - Tamanho da lista tabu:  $7 + \text{INT}(0.25 * \text{num. de rótulos em conflito})$

- Tamanho da vizinhança:  $1 + \text{INT}(0.05 * \text{num. de rótulos em conflito})$
- Número de iterações para recálculo: 50

A média obtida da aplicação do Busca Tabu às 25 configurações estão mostrados na Tabela 5.7, onde as linhas se referem a:

- Iteração: número médio de iterações para alcançar o estado de não conflito entre os rótulos ou o estado de conflito mínimo, respeitando a iteração máxima dada que foi de 50 para 100 pontos, 100 para 250 pontos, 8000 para 500 pontos, 15000 para 750 pontos e 30000 para 1000 pontos.
- Com conflito: número médio de rótulos em estado de conflito
- Sem conflito: número médio de rótulos em estado de não conflito
- Tempo: tempo médio de processamento do algoritmo Busca Tabu para alcançar o estado de não conflito ou conflito mínimo entre os rótulos. O tempo mostrado se refere apenas ao tempo de processamento do algoritmo Busca Tabu, ou seja o tempo usado para o pré-computação dos conflitos existentes entre os rótulos não foi contabilizado.
- Sem conflito(%): trata-se da média percentual do número de rótulos sem conflito das 25 configurações.

**TABELA 5.7 – RESULTADOS OBTIDOS POR BUSCA TABU USANDO O CONJUNTO PADRÃO DE DADOS**

Num. de pontos	100 pontos	250 pontos	500 pontos	750 pontos	1000 pontos
Iteração	7	45	450	6645	21410
Com conflito	0	0	4	24	100
Sem conflito	100	250	496	726	900
Tempo (seg.)	0.01	0.15	3.95	102	732
Sem conflito (%)	100.00	100.00	99.26	96.76	90.00

Um leiaute com configuração de rótulos inicial e configuração de rótulos após aplicação da Busca Tabu para 100, 250, 500, 750 e 1000 pontos encontram-se no Apêndice A5.

Com relação a todos os algoritmos de otimização citados na literatura, o algoritmo Busca Tabu mostrou resultados superiores como pode ser visto na Tabela 5.7, onde as colunas se referem ao percentual médio de rótulos sem conflitos para 100, 250, 500, 750 e 1000 pontos, por diferentes algoritmos da literatura e as linhas mostram o percentual médio de rótulos sem conflito alcançados pelos algoritmos de otimização testados na literatura por Christensen et al. (1995) (Busca Exaustiva, Algoritmo Guloso, “Gradient Descent”, “2-Opt Gradient Descent”, “3-Opt Gradient Descent”, Algoritmo de Hirsch, Zoraster e “Simulated Annealing”), por Verner et al. (1997) (GA sem máscara e GA com máscara) e Busca Tabu por nós testado para configurações de 100, 250, 500, 750 e 1000 pontos. Os resultados mostrados na Tabela 5.8 foram copiados de Verner et al. (1997) e acrescentado de uma linha referente aos resultados obtidos da aplicação do Busca Tabu ao problema de rotulação de pontos.

**TABELA 5.8 – RESULTADOS OBTIDOS POR VÁRIOS ALGORITMOS  
USANDO O CONJUNTO PADRÃO DE DADOS**

Algoritmo	100 pontos	250 pontos	500 pontos	750 pontos	1000 pontos
**Busca Tabu	100.00	100.00	99.26	96.76	90.00
GA com máscara	100.00	99.98	98.79	95.99	88.96
*GA sem máscara	100.00	98.40	92.59	82.38	65.70
“Simulated Annealing”	100.00	99.90	98.30	92.30	82.09
*Zoraster	100.00	99.79	96.21	79.78	53.06
Hirsch	100.00	99.58	95.70	82.04	60.24
* “3-Opt Gradient Descent”	100.00	99.76	97.34	89.44	77.83
* “2-Opt Gradient Descent”	100.00	99.36	95.62	85.60	73.37
“Gradient Descent”	98.64	95.47	86.46	72.40	58.29
Algoritmo guloso	95.12	88.82	75.15	58.57	43.41
Busca exaustiva	84.56	65.63	44.06	29.06	19.53

Adaptada de Verner et al. (1997, p.273).

\* Algoritmos não estudados neste trabalho. Verner et al. (1997); Christensen et al. (1995); Zoraster (1986, 1990, 1991).

\*\* A Tabela 5.8 é uma cópia fiel da tabela encontrada em Verner et al. (1997) com exceção do algoritmo Busca Tabu que foi inserido por nós.

O tempo de processamento médio gasto por algoritmo genético com máscara para resolver o problema de rotulação de 100, 250, 500, 750 e 1000 pontos foi de 6, 49, 414, 1637 e 7256 segundos, usando uma estação SUN-SPARC 10. A Busca Tabu resolveu o problema em 0.01, 0.15, 3.95, 102 e 732 segundos usando uma estação SUN-SPARC 20. Vale ressaltar que o tempo de pré-computação, que flutua em torno do tempo gasto por Busca Tabu, não foi contabilizado. A comparação com relação ao tempo de processamento, deixo a cargo do leitor, uma vez que é difícil de afirmar qualquer coisa, pois trata-se de estações de trabalho diferentes.

### 5.3 AMBIENTE DE TRABALHO

O ambiente de trabalho para o desenvolvimento da técnica constitui do seguinte:

- Estação de trabalho: SUN – SPARC 20;
- Sistema operacional: UNIX – versão Solaris 2.5;
- Compilador: C++ - versão 4.0.1;
- SIG: SCARTA – versão 3.0; e
- SIG: SPRING – versão 3.0 .

## CAPÍTULO 6

### CONCLUSÃO

A rotulação de pontos é um problema de grande importância prática em geoprocessamento e cartografia automatizada. Os algoritmos de rotulação estudados indicam que qualquer método de busca local não consegue alcançar uma solução prática adequada, portanto a busca por soluções heurísticas torna-se importante.

Um dos objetivos desta tese foi a de avaliar o algoritmo de otimização Busca Tabu aplicado a um problema de rotulação de pontos, um problema de difícil solução que tem preocupado vários pesquisadores tais como Christensen, Marks e Shieber (1995) que fizeram um estudo comparativo entre vários algoritmos de otimização aplicado a rotulação cartográfica, chegando a conclusão de que o “Simulated Annealing” era o melhor algoritmo até então encontrado. Verner, Wainwright e Schoenefeld (1997) chegaram a conclusão de que o melhor algoritmo de rotulação de pontos é o Algoritmo Genético com máscara quando comparado a todos os algoritmos descritos por Christensen et al. (1995), utilizando o mesmo conjunto padrão de dados. Com relação a Busca Tabu foram feitos testes utilizando o mesmo conjunto padrão de dados dos trabalhos de Christensen et al. (1995) e Verner et al. (1997) e os resultados mostraram que Busca Tabu apresentou o melhor desempenho em qualidade dos resultados. Isso indica que a Busca Tabu é atualmente a melhor proposta para a rotulação cartográfica de pontos, pois é robusta e capaz de atingir resultados de alta qualidade.

O outro objetivo foi a de avaliar o algoritmo Busca Tabu com relação a dados reais a fim de verificar o seu comportamento em relação a ocorrência natural de distribuição de pontos, uma vez que o agrupamento natural dos pontos e a relação entre a área do mapa considerado, o tipo de papel e a escala em uso dificulta bastante a qualidade da rotulação. Outros fatores que afetam a qualidade da rotulação são o comprimento variável do texto, a altura dos caracteres que compõe os textos dos dados reais e a preferência cartográfica. Foram feitos então testes para verificar a influência da preferência cartográfica manipulando o seu nível de consideração em relação ao nível de consideração da-



dos aos rótulos em conflito. Os resultados mostraram que quanto maior o nível de consideração dado à preferência cartográfica, maior é a dificuldade do algoritmo em alcançar o estado de não conflito entre os rótulos, mas ao mesmo tempo, os rótulos em geral se encontram nas melhores posições, segundo o critério de preferência cartográfica adotada. Com relação à altura do carácter dos textos, o algoritmo mostrou-se bastante sensível e foi possível verificar que o tamanho do carácter que compõe os textos afeta de maneira considerável a qualidade final da carta. Finalmente, é possível afirmar que o desempenho da Busca Tabu foi boa, mesmo quando aplicado a dados reais onde existe um agrupamento natural dos pontos com rótulos de comprimento variável.

Na implementação do algoritmo Busca Tabu no SIG SCARTA, que é também um dos objetivos da tese, o nível de consideração da preferência cartográfica e do número de conflitos são manipulados pelos usuários. A seleção de pontos não é permitida, mas em troca, ferramentas de seletividade e flexibilidade foram incluídas. Entende-se por seletividade, o ato de poder selecionar um determinado rótulo em conflito e excluí-lo através de um editor. Existe ainda a possibilidade de recuperar o último rótulo excluído, se assim se desejar. Flexibilidade é o ato de selecionar o rótulo e modificar sua cor, tamanho ou comprimento e ainda ter a capacidade de movê-lo para outra posição através de um editor. Vale ressaltar que as ferramentas de edição de texto pertencem ao SIG SCARTA, portanto não foram desenvolvidos no presente trabalho. Apenas tornamos possível a sua utilização na rotulação para enriquecer a técnica de rotulação de pontos do SIG SCARTA. Com isso, esperamos que o método Busca Tabu do SIG SCARTA, enriquecido de ferramentas de seletividade e flexibilidade, possa resolver o problema de rotulação cartográfica de pontos ao gerar um resultado de alta qualidade, com tempo de processamento aceitável.

Sugerimos como extensões desta pesquisa:

- Aplicação de outros algoritmos de otimização, para resolver o problema de rotulação cartográfica, implementação dos mesmos e comparação destes algoritmos com todos os demais algoritmos citados neste trabalho.
- Estudar, de forma análoga, a rotulação cartográfica de linhas e áreas.

## REFERÊNCIAS BIBLIOGRÁFICAS

- Barbosa, C. C. F. **Álgebra de mapas e suas aplicações em sensoriamento remoto e geoprocessamento.** 157p. (INPE-7115-TDI-667). Dissertação (Mestrado em Sensoriamento Remoto) – Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 1997.
- Bjorndal, M. H.; Caprara, A.; Cowling, P. I.; Croce, F. D.; Lourenço, H.; Malucelli, F.; Orman, A. J.; Pisinger, D.; Rego, C.; Salazar, J. J. Some Thoughts on Combinatorial Optimisation. **European Journal of Operational Research**, v. 83, p. :253-270, June 1995.
- Bland, J. A.; Dawson, G. P. Tabu search and design optimization. **Computer-Aided Design**, v. 23, n. 3, p. 195-201, Apr. 1991.
- Bonham-Carter, G. F. ed. Geographical information systems for geoscientists: modelling with GIS. In: \_\_\_\_\_ **Computer methods in the geosciences.** New York: Wiley, 1996. 398p. v.13
- Burrough, P. A. **Principles of geographical information systems for land resources assesment.** Oxford: Clarendonn Press, 1987. 193p.
- Câmara Neto, G. **Modelos, Linguagens e Arquiteturas para Bancos de Dados Geográficos.** Tese (Doutorado em Computação Aplicada) – Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 1995.
- Câmara Neto, G. et. al. **Anatomia de Sistemas de Informação Geográfica.** Campinas, Instituto de Computação: UNICAMP, 1996. 193p.

Christensen, J.; Marks, J.; Shieber, S. An empirical study of algorithms for point-feature label placement. **ACM Transactions on Graphics**, v. 14, n. 3, p. 203-232, July, 1995.

Christensen, J.; Marks, J.; Shieber, S. **Placing text labels on maps and diagrams**. London: Academic Press, 1993. 110p.

Edmondson, S.; Christensen, J.; Marks, J.; Shieber, S. **A general cartographic labeling algorithm**. December, 1996. 19p.

Freeman, H. Computer name placement. In: Maguire, D. J.; Goodchild, M. F.; Rhind, D. W. eds. **Geographic information systems: principles and applications**. New York: Longman Scientific, 1991. V.1, Cap. 29, p. 445-456.

Furtado, J. C. **Otimização de leiaute usando busca tabu**. São José dos Campos. 76p. (INPE-5670-TDI/562). Dissertação (Mestrado em Computação Aplicada) – Instituto Nacional de Pesquisas Espaciais, 1995.

Glover, F. Tabu search – part I. **ORSA Journal on Computing**, v. 1, n. 3, p. 190-206, Summer 1989a.

\_\_\_\_\_ Tabu Search – part II. **ORSA Journal on Computing**, v. 2, n. 1, p.4-32, Winter 1989b.

\_\_\_\_\_ Tabu Search – a tutorial. **Interfaces**, v. 20, n. 4, p. 74-94, July-Aug. 1990.

Glover, F.; Laguna, M. Tabu search. In: Colin, R. R. **Modern heuristic techniques for combinatorial problems**. New York: McGraw-Hill, 1995. p. 70-150.

\_\_\_\_\_ **Tabu search**. Boston, Kluwer, 1997. 408p.

- Goldberg, D. E. **Genetic algorithms in search, optimization and machine learning**. New York: Addison-Wesley, 1989. 412p.
- Hirsch, S. A. An algorithm for automatic name placement around point data. **American Cartographer**, v. 9, n. 1, p. 5-17, 1982.
- Holland, J. H. **Adaptation in natural and artificial systems**. Ann Arbor: University of Michigan Press, 1975. 211p.
- Imhof, E. Die anordnung der namen in der karte. In: **Annuaire international de cartographie II**. Zurich: Orell Fuessli Verlag, 1962. p. 93-129.
- Kirkpatrick, S.; Gelatt Jr., C. D.; Vecchi, M. P. Optimization by Simulated Annealing. **Science**, v. 220, p. 671-680, 1983.
- Knuth, D. E. **The stanford graphBase, a platform for combinatorial computing**. New York: Addison-Wesley, 1993. 576p.
- Laguna, M. A guide to implementing tabu search. **Investigación Operativa**, v. 4, p.5-25, 1994.
- Oliveira, C. **Curso de cartografia moderna**. Rio de Janeiro: IBGE, 1988. 152p.
- Reeves, C. R. ed. **Genetic algorithm**. London: McGraw-Hill, 1995. (Modern heuristic techniques for combinatorial problems). p. 151-196.
- Santos, M. C. S. R. **Manual de fundamentos cartográficos e diretrizes gerais para elaboração de mapas geológicos e geotécnicos**. São Paulo: IPT, 1989. 52p.

Verner, O. V.; Wainwright, R. L.; Schoenefeld, D. A. Placing text labels on maps and diagrams using genetic algorithms with masking. **INFORMS Journal on Computing**, v. 9, p. 266-275, 1997.

Zoraster, S. Expert systems and the map label placement problem. **Cartographica**, v. 28, n. 1, p. 1-9, 1991.

\_\_\_\_\_ Integer programming applied to the map label placement problem. **Cartographica**, v. 23, n. 3, p. 16-27, 1986.

\_\_\_\_\_ The solution of large 0-1 integer programming problems encountered in automated cartography. **Operations Research**, v. 38, n. 5, p. 752-759, Sept.-Oct. 1990.

## APÊNDICE A1

### FUNCIONALIDADES DE UM SIG CARTOGRÁFICO

- Manipular o banco de dados: poder recuperar as informações contidas no banco de dados geográficos e apresentá-los da melhor maneira possível. Deve ter, ainda, a facilidade de mudar o visual, como, por exemplo, a cor da linha que representa a estrada.
- Manipular o projeto: poder conectar-se ao projeto de interesse, tendo em vista que o banco de dados contém uma variedade de projetos, cada projeto possuindo o seu limite geográfico da área em estudo e a projeção cartográfica mais adequada.
- Definir características do papel: poder escolher entre os vários formatos padrões (A4, A3, A2, A1 e A0) ou definir uma carta com tamanho personalizado e poder ainda escolher a orientação da folha ou seja a posição do papel (horizontal ou vertical). Os formatos de papel mais usados são mostrados na Tabela A1.1.

**TABELA A1.1 – FORMATOS DE PAPEL**

Formato	Altura	Comprimento
A0	841 mm	1189 mm
A1	594 mm	841 mm
A2	420 mm	594 mm
A3	297 mm	420 mm
A4	210 mm	297 mm

- Definir características dos dados: poder definir escala, limites e posição da área de dados, isto é, moldar a área onde serão apresentados os planos de informação do projeto ativo. A escala, o limite e a posição da área de dados deverão ser calculados automaticamente pelo sistema, levando em consideração o tamanho e a orientação do papel ou possibilitar o fornecimento do mesmo pelo usuário. O usuário deverá ter a liberdade de escolher o posicionamento da área de dados dentro dos limites do

papel escolhido, podendo ainda estabelecer os limites da área de dados no papel e os limites dos dados tanto em coordenadas planas como também em coordenadas geográficas.

- Selecionar os planos de informação: poder escolher os dados que serão mostrados na área de dados, levando em consideração a prioridade de apresentação dos vários planos de informação selecionados.
- Editor de texto: possibilitar ao usuário escrever textos levando em consideração os seguintes atributos: tamanho, espessura, cor, espaçamento entre os caracteres, rotação e tipo de fonte. Poder, ainda, escolher a posição exata de localização do texto tanto em centímetros do papel como também em coordenadas planas ou geográficas. Excluir, recuperar, desativar e ativar o texto também deverá fazer parte do repertório de edição do texto, assim como poder recuperar o texto, fazer algumas mudanças e atualizar o texto em questão. Outros fatores a serem considerados são o alinhamento do texto e o fato de usuário poder reposicionar o texto para a posição desejada.
- Editor de símbolos: poder escolher o símbolo desejado da lista de símbolos apresentado ou poder desenhar o seu próprio símbolo que será inserido na carta. Poder escolher o tamanho do símbolo, a cor, assim como a localização do mesmo. Deletar, recuperar, desativar e ativar o texto também deverá fazer parte do repertório de edição de símbolos, assim como poder mover o símbolo usando o “mouse”.
- Editor de legenda: o objetivo aqui é mostrar um retângulo preenchido seguido por um texto. O preenchimento desse retângulo, identifica os polígonos do mapa temático cujo preenchimento são do mesmo tipo. O texto associado ao retângulo identifica a representação desses polígonos. O editor deve fornecer a capacidade de escolher a localização exata onde as legendas serão colocadas, assim como o tamanho dos retângulos e texto. Vale ressaltar que todas as facilidades descritas para o editor de texto são cabíveis neste contexto.
- Editor de linha: poder desenhar as linhas ortogonais ou não, com espessura e cor desejada, levando em consideração as operações de assistência (extremo, meio, pró-



ximo e normal) de ligação entre as linhas. A edição da linha deve incluir excluir, recuperar, desativar, ativar e alterar os seus atributos.

- Editor de grade: o objetivo aqui é definir grades para referenciamento geográfico, em coordenadas geográficas e/ou planas. O sistema deverá fornecer os limites da grade, deixando a liberdade de mudança por parte do usuário. O usuário deve poder definir tanto em coordenadas planas como também em coordenadas geográficas a linha inicial da grade, assim como o espaçamento entre as linhas. Deve também poder escolher a característica das linhas (cor, espessura, contínua ou cruzamento) e coordenadas (mostrar ou não, cor, estilo, tamanho, rotação, espaçamento entre os caracteres, apresentação geográfica ou plana)
- Escala gráfica: como mostrado na (Figura A1.1) a escala gráfica visa mostrar a redução do mundo real para mapas. O sistema deverá fornecer uma lista de modelos de escala gráfica para ser escolhido pelo usuário ou dar ao usuário a possibilidade de construí-lo através de um editor.



Fig. A1.1 - Escala Gráfica.

- Mapa de localização: poder mostrar em versão miniatura a localização da carta em estudo dentro de um mapa de âmbito maior.
- Articulação de folhas: poder mostrar em versão miniatura a localização do projeto em estudo dentro de um projeto total, que é composto por vários projetos.
- Manipular molde: possibilitar criar, salvar e recuperar os moldes para que textos, símbolos, legendas e formatos pré-definidos (características do papel e características dos dados) possam ser utilizados em várias carta (Figura A1.2).

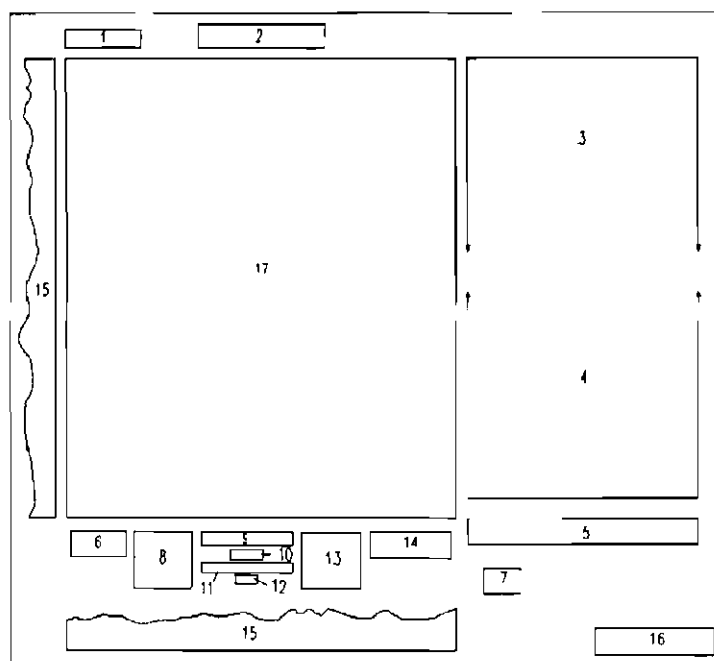


Fig. A1.2 - Molde de uma carta.

FONTE: Santos (1989, p. 35).

onde:

- 1 - Nome e codificação da folha  
Ex: São Paulo - SF-23-Y-C-VI-2
- 2 - Nome dos órgãos contratantes e executor
- 3 - Descrição das unidades temáticas
- 4 - Convenções adotadas
- 5 - Convenções planialtimétricas
- 6 - Informações sobre o mapa-base
- 7 - Declinação magnética
- 8 - Localização da folha
- 9 - Nome da carta
- 10 - Escala numérica
- 11 - Escala gráfica
- 12 - Ano de execução
- 13 - Articulação da folha
- 14 - Projeto e equipe executora
- 15 - Áreas para a localização das seções geológicas

16 - Espaço reservado (optativo) para título da carta e referência bibliográfica do mesmo formato de uma folha geológica 1:50000.

- Manipular carta: possibilita criar, salvar e recuperar as cartas.
- Zoom: ampliar uma região da carta.
- Zoom de tela: ampliar uma região da carta que está mostrada na tela em uma outra janela, tal que a janela do zoom irá mostrar todas as posições que o cursor for percorrendo.
- Navegação: mostrar em tempo real o valor da posição onde se encontra o cursor, em centímetros do papel, ou dos dados, em coordenadas planas ou geográficas.
- Vôo: poder visualizar áreas ocultas do projeto causado pelo zoom.
- Recompôr: poder recompôr a situação anterior ao zoom ou vôo.
- Mostrar itens da carta: poder controlar a visualização dos itens da carta, ativando ou desativando os mesmos.
- Saída: Postscript, HPGL, HPRTL, TIF, GIF são entre outros os tipos de saída desejadas.
- Rotulação: Em cartografia são identificados três tipos de rotulação: rotulação de ponto (cidades, pico de montanhas, escolas, hospitais,...), rotulação de linha (rios, estradas,...) e rotulação de área (oceanos, países, estados,...).

## APÊNDICE A2

### TÉCNICAS DE ROTULAÇÃO DE LINHAS

As regras básicas de rotulação de linhas, segundo Freeman (1991), são:

- 1) Não se deve sobrepor a entidade geográfica com seu rótulo.
- 2) rótulo deverá acompanhar a curvatura da entidade linha, mas nunca em mais de uma direção.
- 3) A linha base do rótulo normalmente é linear.
- 4) Quando a entidade linha é longa, o rótulo deverá ser repetido.
- 5) Os rótulos não deverão ter seus caracteres espalhados, embora espaçamento extra possa ser adicionado entre palavras.
- 6) Os rótulos não deverão ser colocados muito próximo aos extremos das linhas.
- 7) Texto das linhas orientadas verticalmente deverão ter a direção para cima na metade esquerda da carta e a direção para baixo na metade a direita da carta.
- 8) Texto pode ser abreviado, desde que o significado da abreviação seja claro para o observador, como por exemplo “Av.” para “Avenida”.

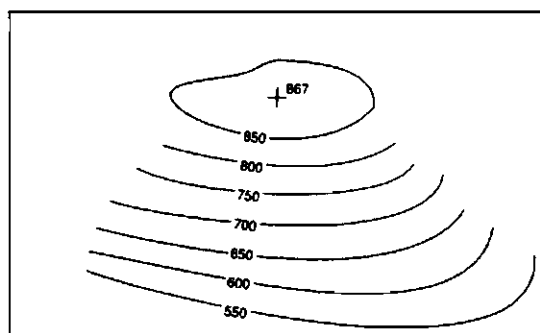


Fig. A2.1 - Rotulação das isolinhas.

FONTE: Freeman (1991, p. 452).

9) As cotas deverão ser colocados em cima das isolinhas ( Figura A2.1).

Segundo Freeman (1991), para rotular uma linha, deve-se analisar toda a sua extensão, a fim de encontrar localizações permissíveis, levando em consideração as regras acima. Quando uma localização satisfatória é encontrada, a seção correspondente da linha é extraída e suavizada para servir de linha base para o rótulo. Para rótulos com palavras múltiplas, o espaçamento entre estas palavras pode ser ajustado para evitar sobreposição com outros rótulos.

Shawn et al. (1996) descreve cinco métricas de posicionamento do texto na linha. São elas: distância média (“AveDist”), distância mínima (“MinDist”), achatamento (“Flatness”), centralização (“Centeredness”) e Posicionamento (“Aboveness”). Antes de descrever as métricas, alguns termos úteis serão definidos.

Uma entidade linha (rio, estrada,...) geralmente é representada por um polígono longo e delgado. Computacionalmente o polígono em questão será referenciado por uma polilinha que representa as suas características e o seu formato.

Linha base (“baseline”) é a linha que serve de base para desenhar o texto (Figura A2.2).

Linha silhueta (“skyline”) é a união dos retângulos envolventes dos caracteres que compõem o texto ( Figura A2.2).

“ $\delta$ ” é a distância considerada ideal entre a “baseline” do rótulo e a linha reta calculada que representa a entidade linha (rio, estrada...). Esta distância deverá variar com a espessura da linha e o tipo de caracter (Figura A2.2). Ou seja:

$$\delta = a + e \tag{A2.1}$$

onde,

a = distância entre a “baseline” e o topo do caracter maiúsculo;

e = espessura / 2.

“swath” é uma faixa longa onde o rótulo se encontra centrado. Os autores recomendam que a sua largura seja 20% maior do que a largura do rótulo (Figura A2.2).

“swath line” é a parte da polilinha próximo ao rótulo (Figura A2.2).

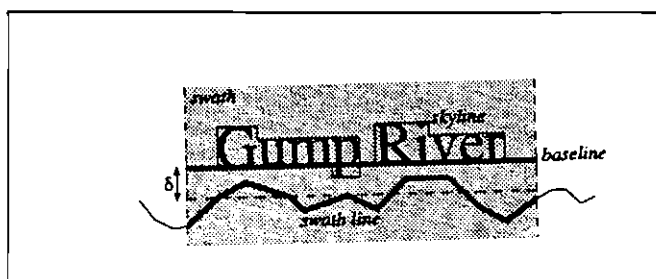


Fig. A2.2 - Termos utilizados para rotulação das entidades linhas.

FONTE: Shawn et al. (1996, p. 7).

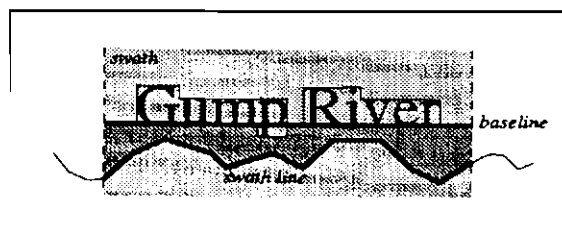


Fig. A2.3 - Área entre “swath line” e linha silhueta (“skyline”) inferior.

FONTE: Shawn et al. (1996, p. 7).

A seguir serão descritas as 5 métricas de posicionamento de rótulos em entidade linhas, segundo Shawn et al. (1996).

“AveDist” mostra, se é adequado a distância média “d” existente entre o rótulo e a linha. (Figura A2.4) mostra posição de rótulo com “AveDist” considerado bom.

$$\text{“AveDist”} = (d - \delta)^2 / \delta^2 \quad (\text{A2.2})$$

onde,

“AveDist” = 0 (ideal);

a = ( área entre “swath line” e “skyline” inferior ) (Figura A2.3);

$d = a / \text{largura do "swath"};$

$\delta = (\text{Equação, A2.1}).$

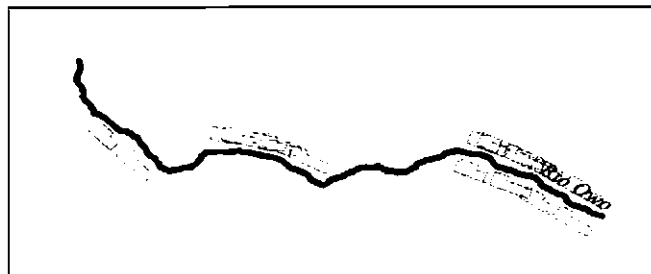


Fig. A2.4 - Posição de rótulo com distância média considerado bom.

FONTE: Shawn et al. (1996, p. 7).

“**MinDist**” mostra, se é adequado a distância mínima “ $d$ ” existente entre o rótulo e a linha.

$$\text{“MinDist”} = (d' - \delta)^2 / \delta^2 \quad (\text{A2.3})$$

onde,

“**MinDist**” = 0 (ideal);

$d'$  = distância mínima permitida entre ponto  $p1$  (ponto pertencente a “swath line”) e ponto  $p2$  (ponto pertencente a “skyline”).

$\delta = (\text{Equação, A2.1}).$

“**Flatness**” é o grau de curvatura do “swath line”.

$$\text{“Flatness”} = d'' / \delta^2 \quad (\text{A2.4})$$

onde,

$a = \Sigma$  (área entre “swath line” e linha reta  $L$ ) (Figura A2.5);

$d'' = a / \text{largura do "swath"};$

$\delta = (\text{Equação, A2.1}).$

O coeficiente “flatness” é tanto menor quanto menos ondulado for a linha em questão ( Figura A2.6).

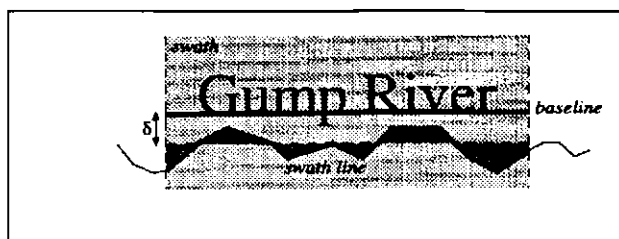


Fig. A2.5 - Calculando grau de curvatura do “swath line”.

FONTE: Shawn et al. (1996, p. 8).

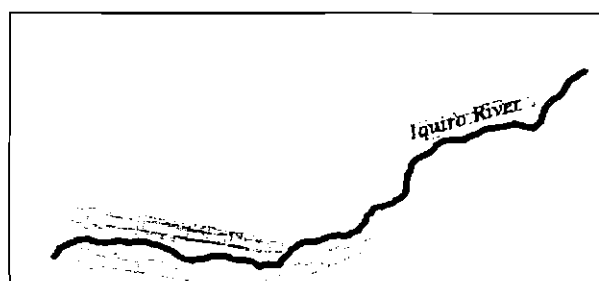


Fig. A2.6 - Posição do rótulo com o valor de “flatness” considerado bom.

FONTE: Shawn et al. (1996, p. 8).

“Aboveness” indica se o rótulo esta acima ou abaixo da entidade linha. Quando “Aboveness” = 0 o rótulo está localizado acima da entidade linha e quando “Aboveness” = 1 o rótulo está localizado abaixo da entidade linha. A preferência é ter o rótulo localizado acima da entidade linha.

“Centeredness” procura uma posição próximo ao centro da entidade linha, pois assume-se que cada entidade linha possui somente um rótulo, portanto é importante que o rótulo seja colocado próximo ao centro da entidade linha em questão. O “Centeredness” ideal é quando o seu valor é igual a zero.

$$\text{“Centeredness”} = |2l - 1| \quad (\text{A2.5})$$



onde,

$$l = l_1 / l_2;$$

$l_1$  = distância entre o ponto  $p$  e extremidade da polilinha;

$l_2$  = comprimento total da polilinha ;

$l = 1/2$  (ideal);

$l = 0$  ou  $1$  (bordas).

Como produzir posições candidatas para rótulos da entidade linha? Shawn et al. (1996) sugere:

- 1) Gerar muitas posições candidatas sem se preocupar com sua validade. Neste ponto ainda não nos preocupamos com a distância entre o rótulo e a entidade linha.
- 2) Ajustar todas as posições candidatas para alcançar o valor ideal da métrica “MinDist” aplicando a translação apropriada.
- 3) Avaliar as posições candidatas de acordo com todas as métricas de linha.
- 4) Deixar as melhores posições candidatas e deletar o resto.

## APÊNDICE A3

### TÉCNICAS DE ROTULAÇÃO DE ÁREAS

A seguir serão apresentados alguns conceitos e descrições que se encontram em Freeman (1991). Segundo ele, o rótulo deverá ocupar todo o espaço da entidade área, respeitando o formato da mesma. O tamanho do carácter deve ser escolhido para indicar a importância relativa pois, desta forma uma percepção implícita do tamanho, extensão e importância da entidade área é transmitida ao observador. Na prática isto nem sempre acontece pois uma área geográfica pode possuir formas irregulares ou pode consistir de partes disjuntas como, por exemplo um arquipélago. O problema torna-se ainda mais complicado quando uma área geográfica não possui um limite explícito, como, por exemplo, “Kamuraga Valley” da (Figura 1.1). Em tais casos a rotulação e a escolha do tamanho da fonte e espaçamento entre os caracteres é crítica pois é através dele que se torna possível comunicar a extensão da área ao observador.

Uma outra maneira de rotular áreas geográficas é escrevendo o nome da área em questão, na parte interna dos segmentos que cercam a área geográfica (Figura A3.1(b)). Nestes casos o observador consegue visualizar os limites precisos da área em questão.

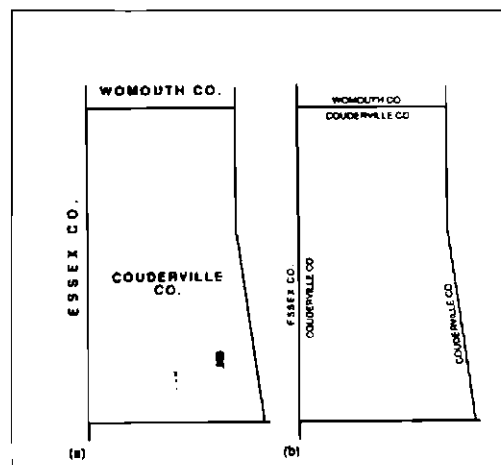


Fig. A3.1 - Rotulação da entidade área. (a) Normal (b) Quando os limites da área são importantes.

FONTE: Freeman (1991, p. 448)

Imhof (1962), citado em Freeman (1991), apresenta o seguinte conjunto adicional de regras:

- 1) A rotulação horizontal dos textos só deverá ser escolhida se além de ser o ideal para a área geográfica em questão, ela é adequada para a área geográfica vizinha.
- 2) Se a escala e projeção da carta é tal que a linha de latitude é mostrada com curvatura, então a linha de base horizontal do texto também precisa estar curvada segundo a linha de latitude.
- 3) A linha de base deverá estar próxima e paralela à linha central dominante da área geográfica.
- 4) Nome deverá ser espaçado tal que o espaçamento inicial e final do texto seja de 1 ½ vezes a largura do espaçamento entre os caracteres.
- 5) Se o texto do rótulo não for totalmente horizontal, a preferência é ter inicialmente o texto na horizontal e depois encurvá-lo, em vez de se ter o texto encurvado no início e horizontal no final.
- 6) Texto da área geográfica vizinha, que pertence a mesma categoria, deverá ser encurvado similarmente, a não ser que o formato da área geográfica em questão apresente uma diferença significativa.
- 7) Se uma área geográfica tem um formato muito curvo, ou consiste de duas ou mais partes separadas por um corpo d'água, o texto pode ser colocado saindo parcialmente para fora dos limites da entidade área em questão. O texto de uma área geográfica não deverá-se sobrepor, mesmo que seja parcialmente, ao rótulo de outra área geográfica.

Para aplicar estas regras, o primeiro passo é identificar os limites da área geográfica a ser rotulada e também o limite das áreas adjacentes. Se uma área não tem limite político ou natural como no caso de floresta, vale, região étnica ou um arquipélago, os limites

precisam ser encontrados. A seguir, o formato da área geográfica precisa ser estudado para decidir onde será a localização da linha de base, que é a linha onde as letras com devido espaçamento serão colocadas, preenchendo a entidade área em questão.

Como determinar a linha de base? A seleção da linha de base é um desafio por causa da infinita variedade de formato das áreas geográficas. Inicialmente, um teste simples pode ser realizado para determinar se a rotulação horizontal é satisfatória. Tal teste é baseado em área/perímetro, largura/altura e forma de rotulação da área vizinha da mesma categoria.

Se a rotulação horizontal não for satisfatória, o esqueleto da entidade área é computado (Figura A3.2).

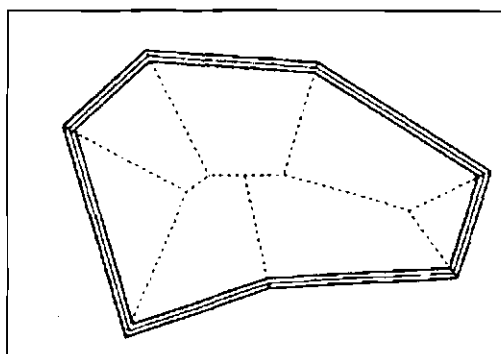


Fig. A3.2 - Esqueleto de uma entidade área.

FONTE: Freeman (1991, p. 449).

O esqueleto normalmente consiste de um número de ramificações e o algoritmo requer que a mais dominante dentre todas seja identificada. Tipicamente esta seria a mais longa ramificação, ou a concatenação de duas ou três ramificações longas que não diferem muito em orientação. Uma curva suave é passada na ramificação selecionada e estendida até os limites dos dois lados terminais. Assim sendo a “baseline” será uma linha paralela à curva suave encontrada, mas deverá ter a compensação de  $\frac{1}{2}$  da altura do carácter maiúsculo.

Como saber se a “baseline” encontrada é a ideal? Por causa da grande variedade de formatos existentes de áreas geográficas, muitas vezes o procedimento acima traz resultados totalmente inaceitáveis. Isto requer que a rotulação seja avaliada posteriormente. Um procedimento especial tem sido desenvolvido para isto com testes de qualidade de rotulação levando em consideração um conjunto de medidas de qualidade, tais como: (1) estar de acordo com o formato dominante; (2) “baseline” com curvatura excessiva; (3) proximidade do caracter das bordas; (4) possível conflito com os rótulos vizinhos de categoria similar; (5) orientação inapropriada da linha de base para uma posição particular da carta.

Se a avaliação indicar qualidade de rotulação inaceitável, o processo é repetido usando um dos algoritmos alternativos de determinação de linha de base ao invés do método de esqueletorização citado acima. Se nenhum dos resultados é satisfatório, escolhe-se a melhor dentre todas as linhas de base encontradas.

Freeman (1991) enfatiza que a qualidade estética da rotulação está ligada à facilidade e naturalidade com que a aparência visual do rótulo realiza seus propósitos de identificar a área geográfica que tenciona rotular. Um observador não tem que estudar o nome e seus vizinhos para estabelecer a qual área geográfica está associada, nem para determinar a extensão da área geográfica rotulada. Neste contexto, ele afirma que a estética não é mera beleza, mas um ingrediente importante que possibilita a um mapa comunicar relações espaciais a seus leitores.

Shawn et al. (1996) assume que a área em estudo seja larga o suficiente para caber o seu rótulo. Logo uma métrica simples é suficiente para satisfazer a rotulação de área. “AreaPos” mede a proximidade do rótulo ao centróide da entidade área em questão.

$$\text{AreaPos} = c / s \quad (\text{A3.1})$$

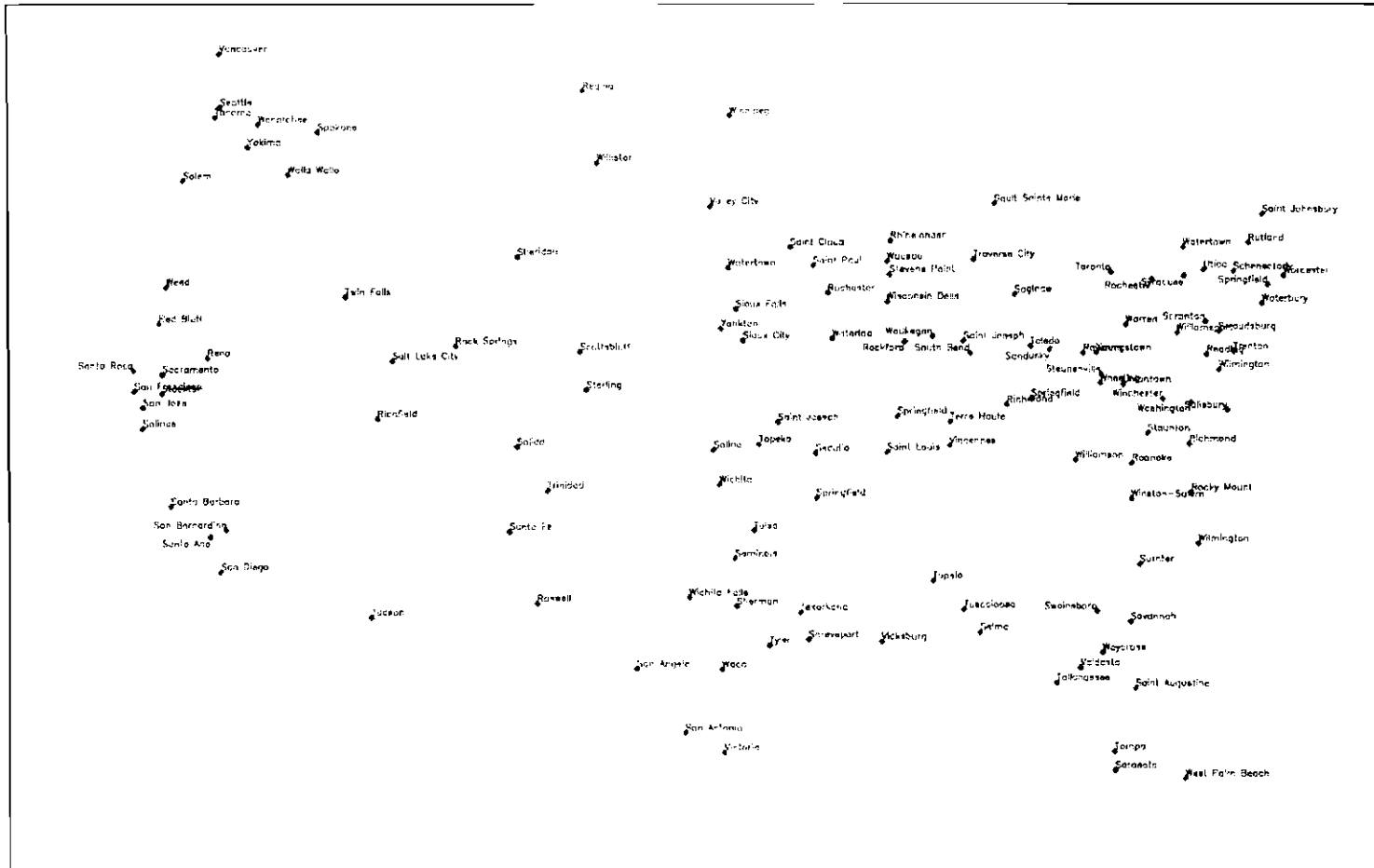
onde,

s é a distância entre o centróide da entidade área e seu vértice mais distante;

c é a distância entre o centro do rótulo e o centróide da entidade área.

c = 0 (ideal).

# USA - CIDADES



APÊNDICE A4

111

POSICÕES CANDIDATAS = 8

ALTURA DO CARACTER = 1.0 mm

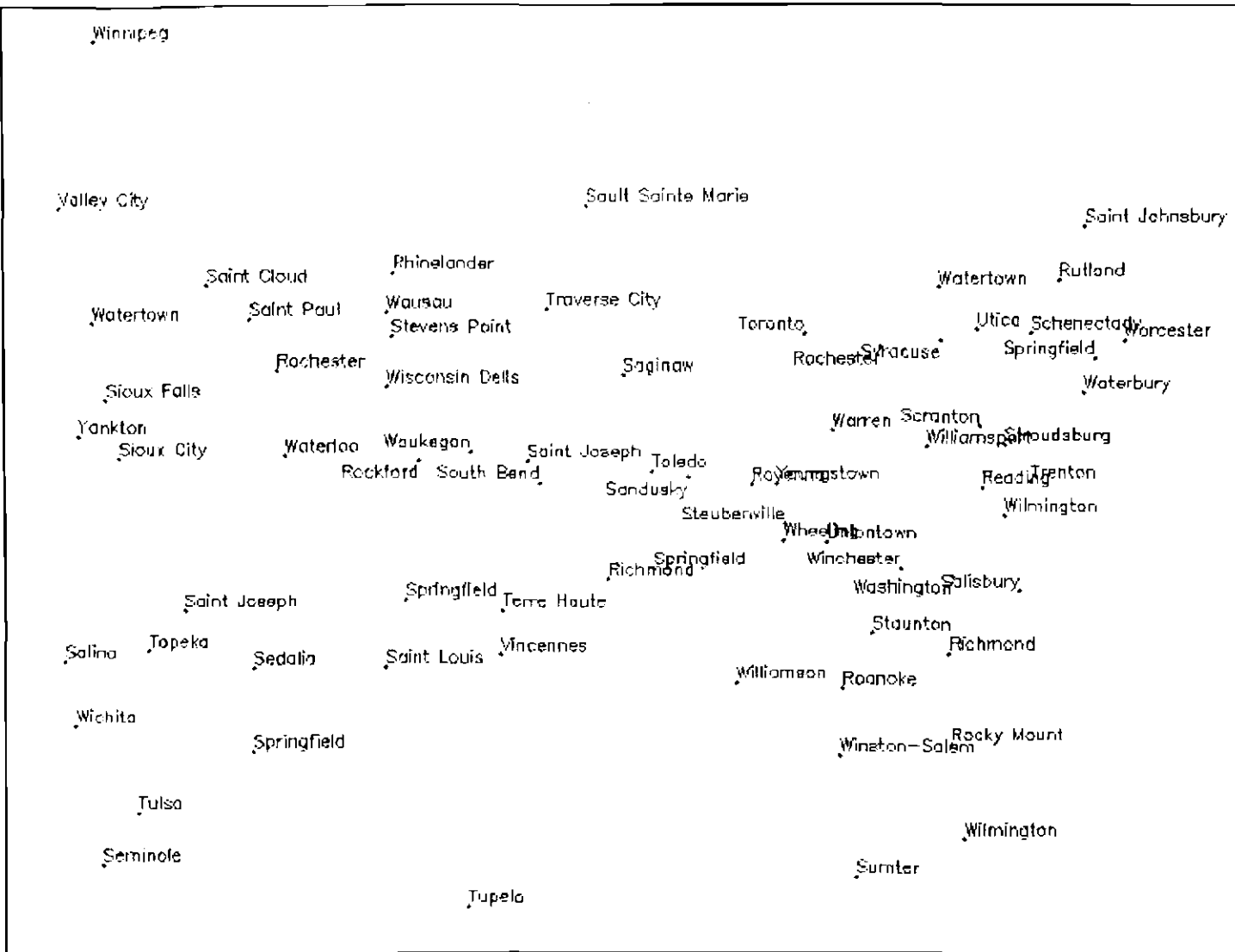
ITERAÇÃO = 306

CONFLITO = 26

ALT A1 = 1

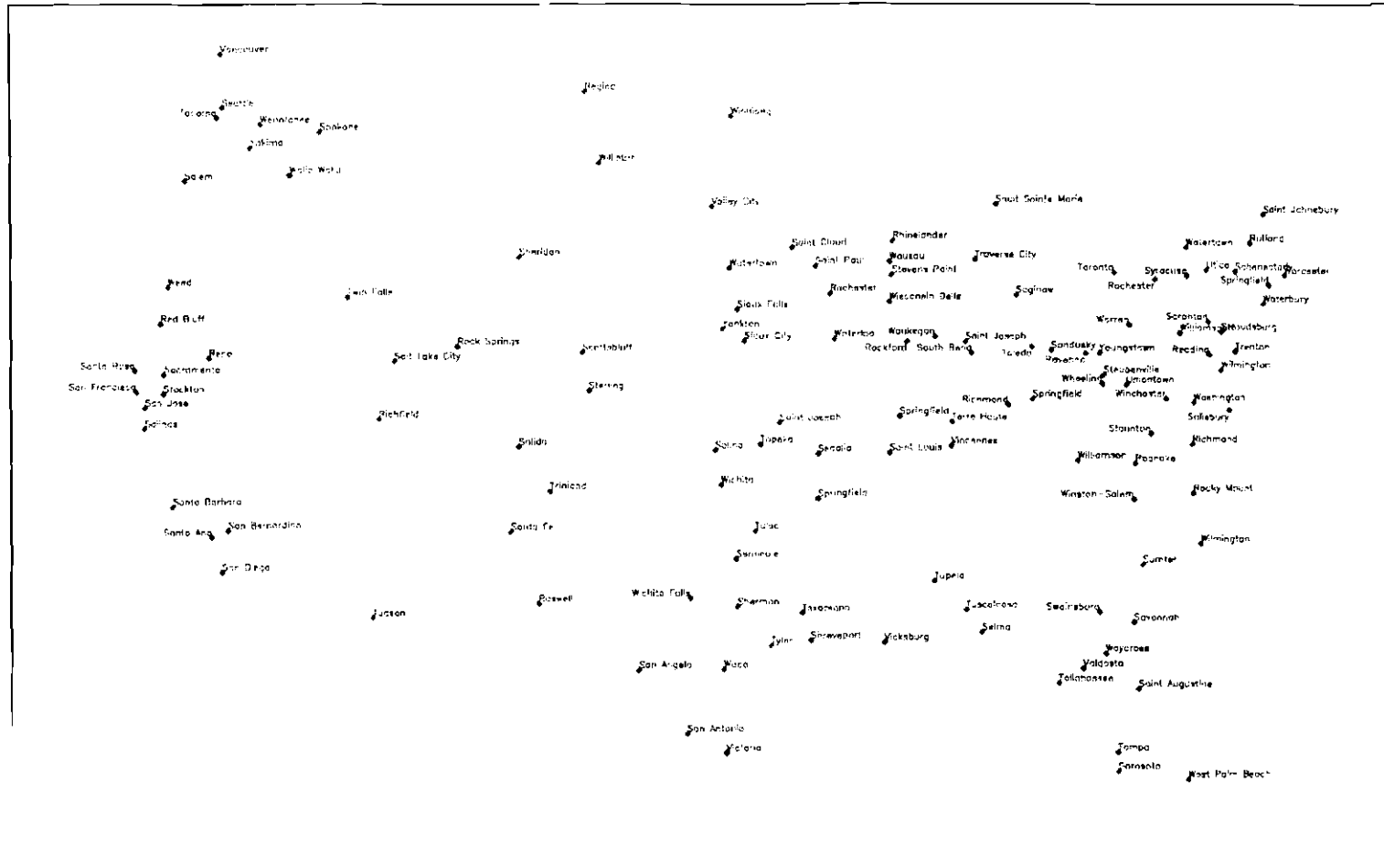
ALT A2 = 10

ESCALA 1:25000000





# USA2 - CIDADES



POSIÇÕES CANDIDATAS = 8

ALFA1 = 1

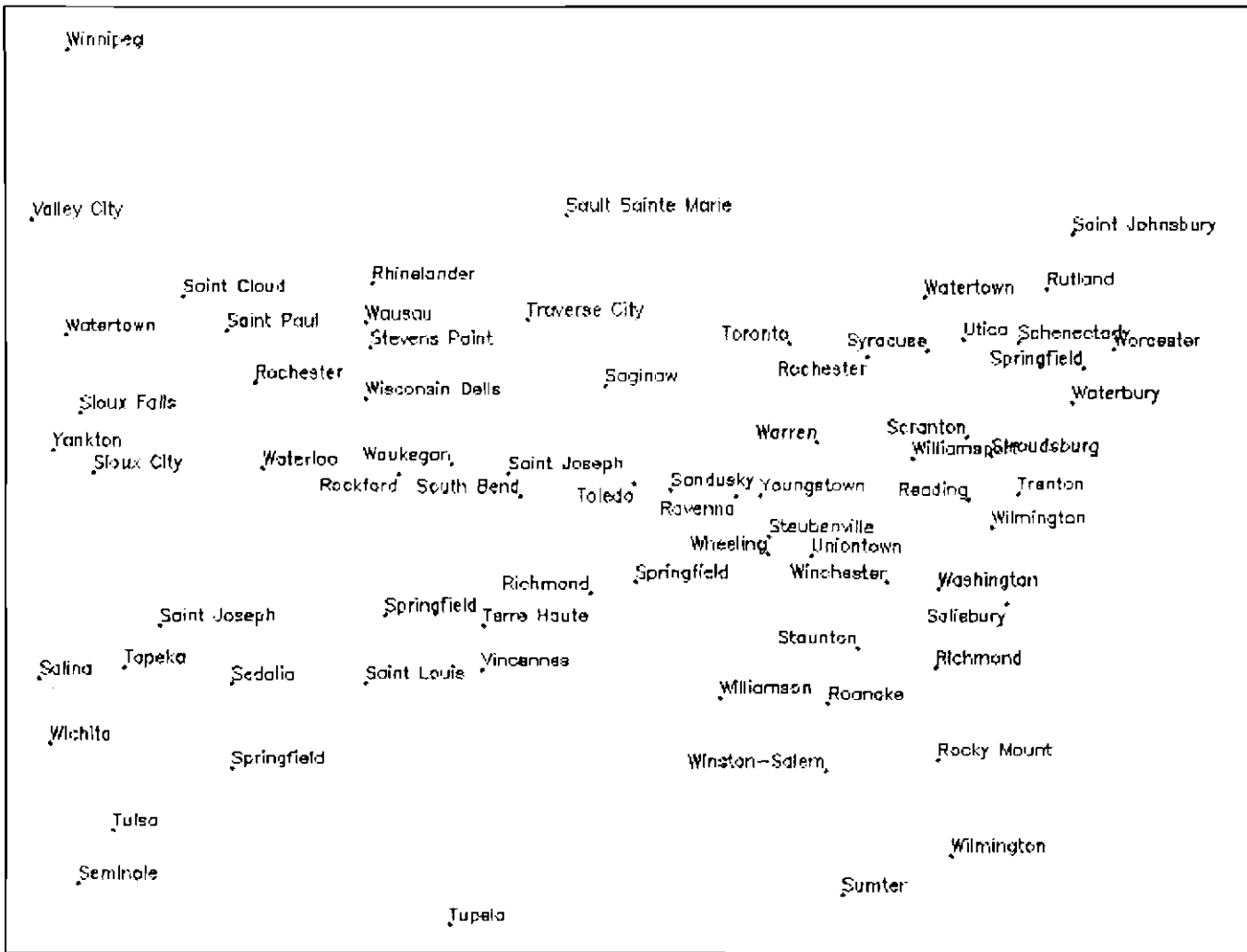
FSCALA 1:23000000

ALTURA DO CARACTER = 1.0 mm

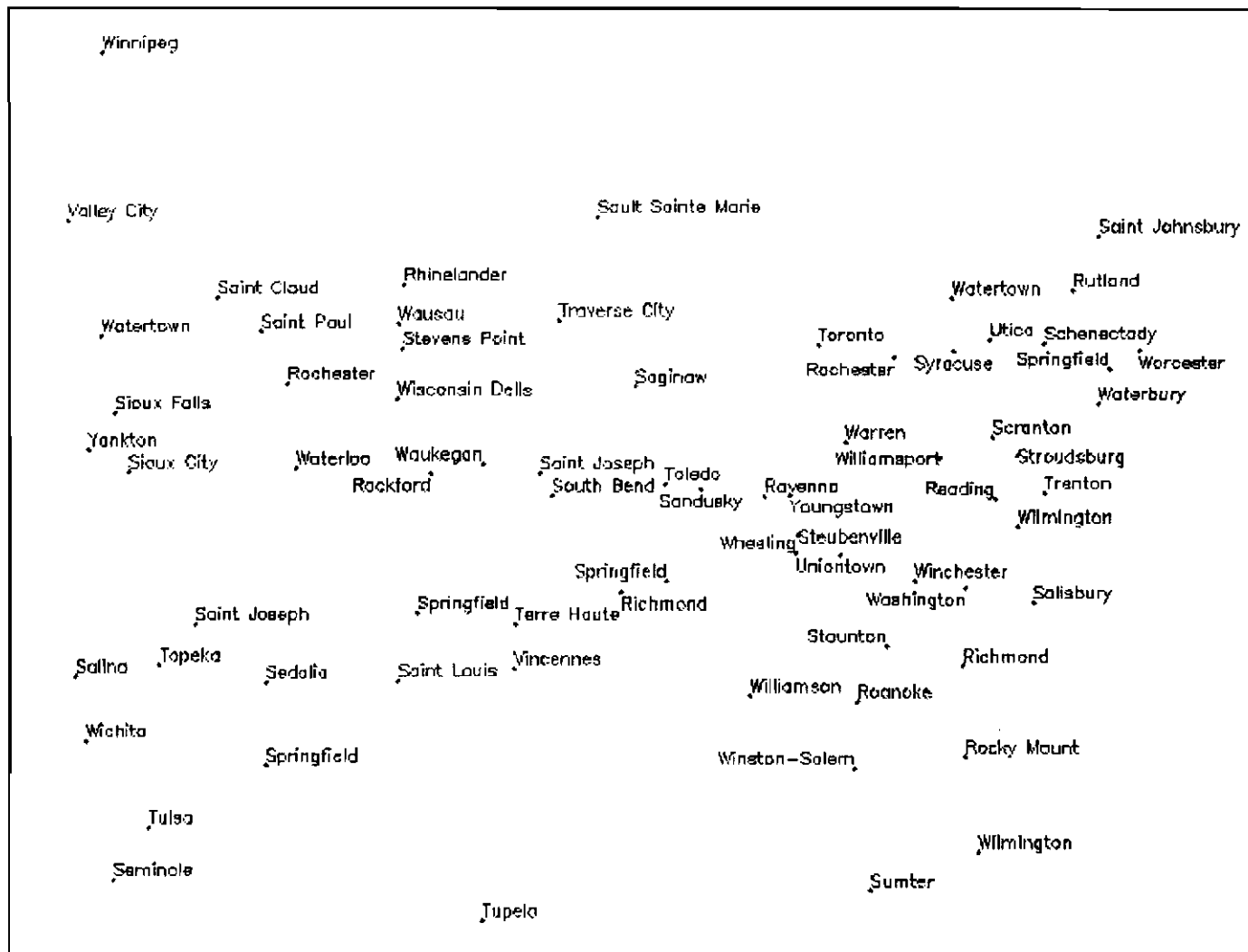
ALFA2 = 5

ITERAÇÃO = 125

CONFLITO = 6

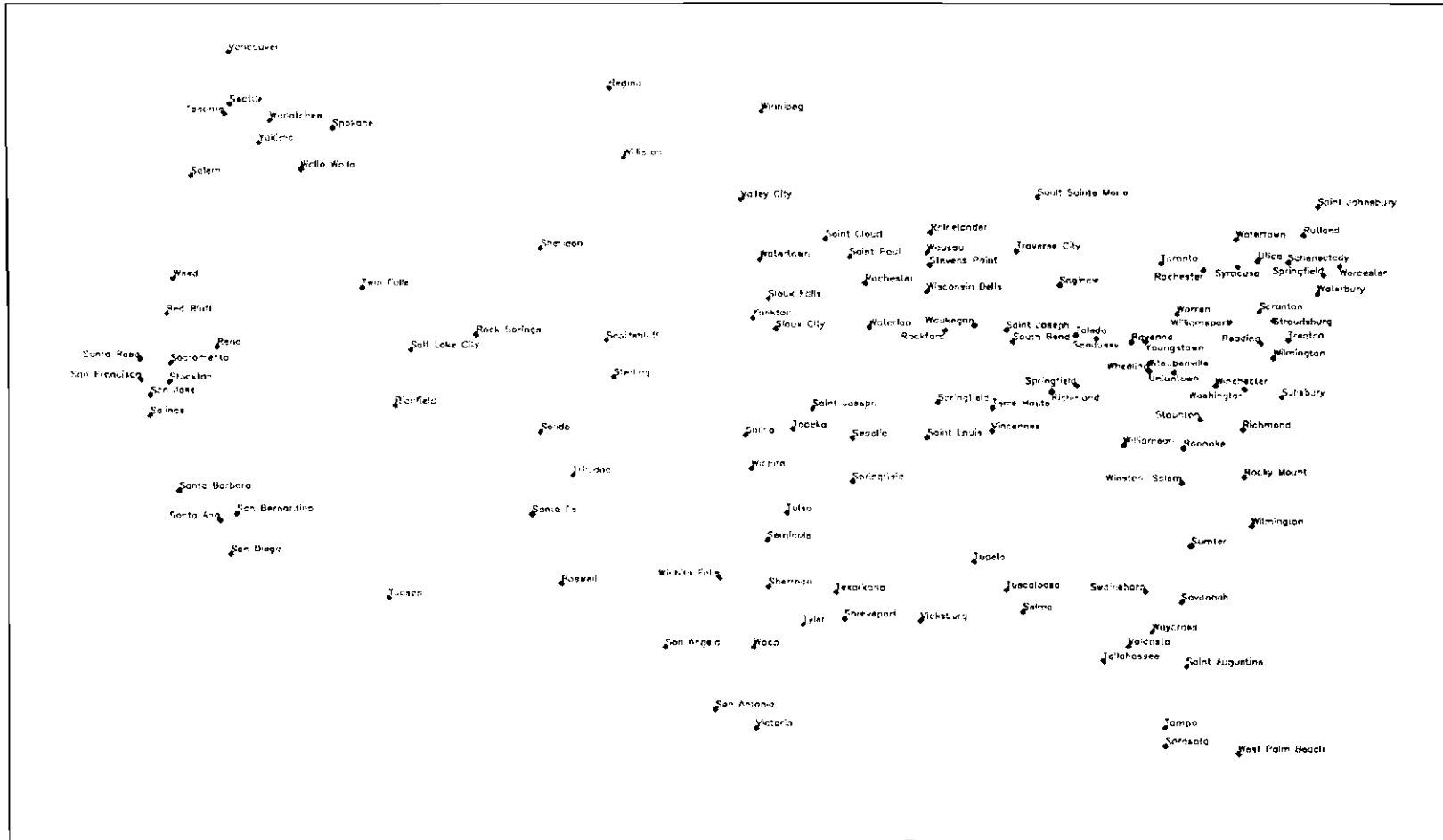






“zoom” da área de agrupamento de pontos da carta USA3 – CIDADES

# USA4 - CIDADES



POSICÖES CANDIDATAS = 117

ALTURA DO CARACTÉR = 1.0 mm

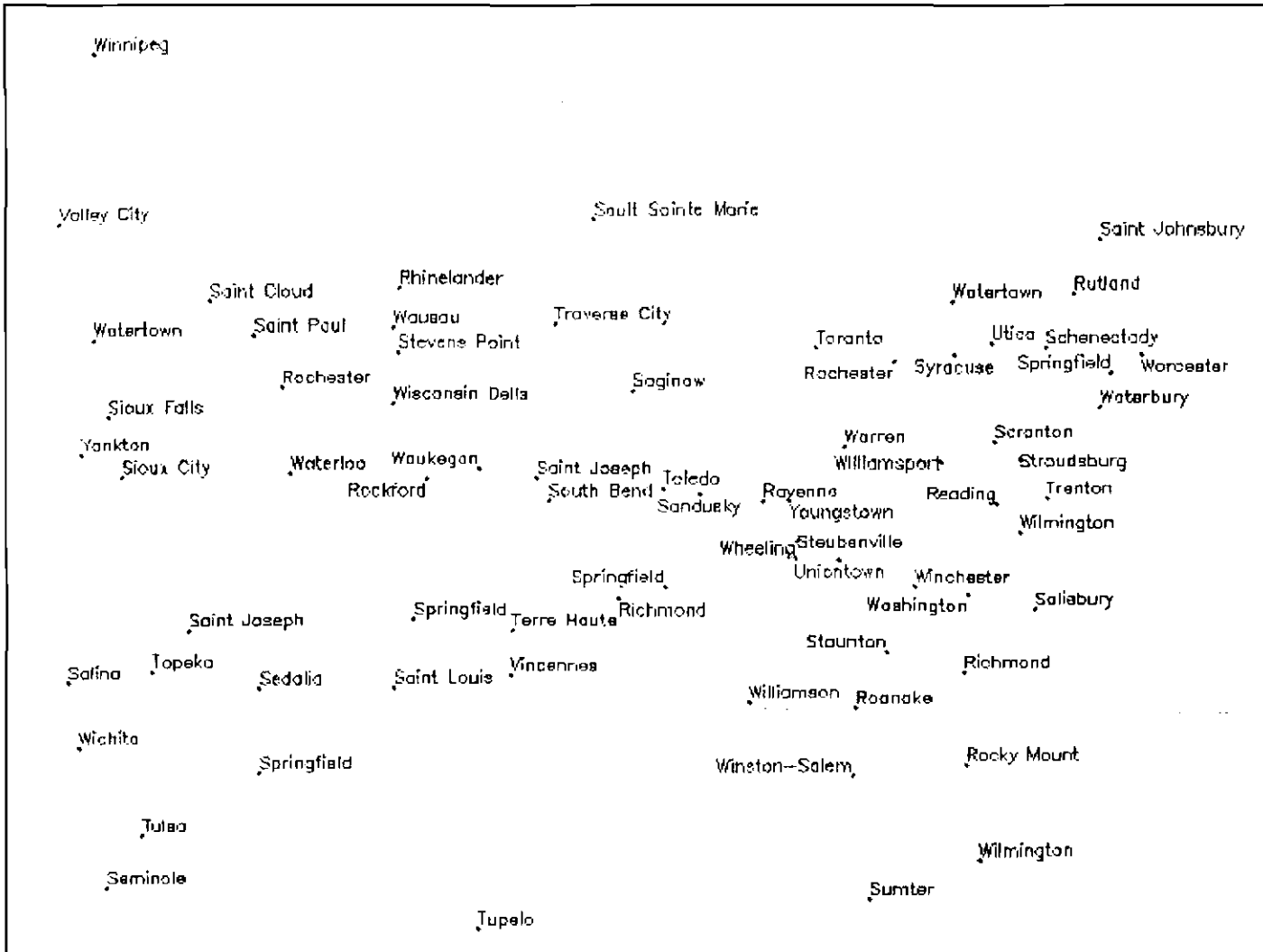
ILUSTRAÇÃO = 32

CONTEÚTO = 0

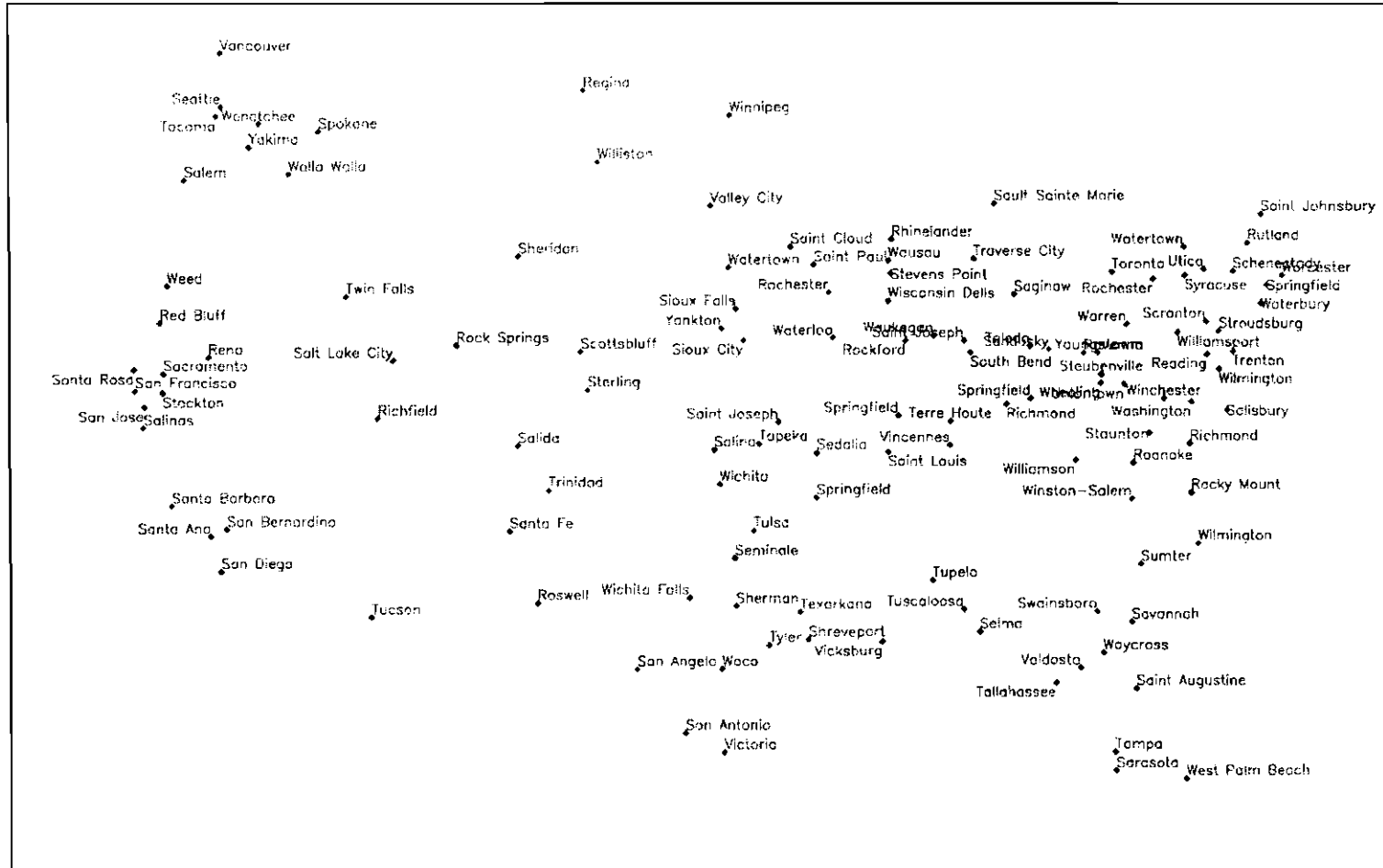
ALFA 1 = 3

ALFA 2 = 1

ESCALA 1:2.300.000



# USA5 - CIDADES



119

POSICÖES CANDIDATAS = 8

ALFA1 = 3

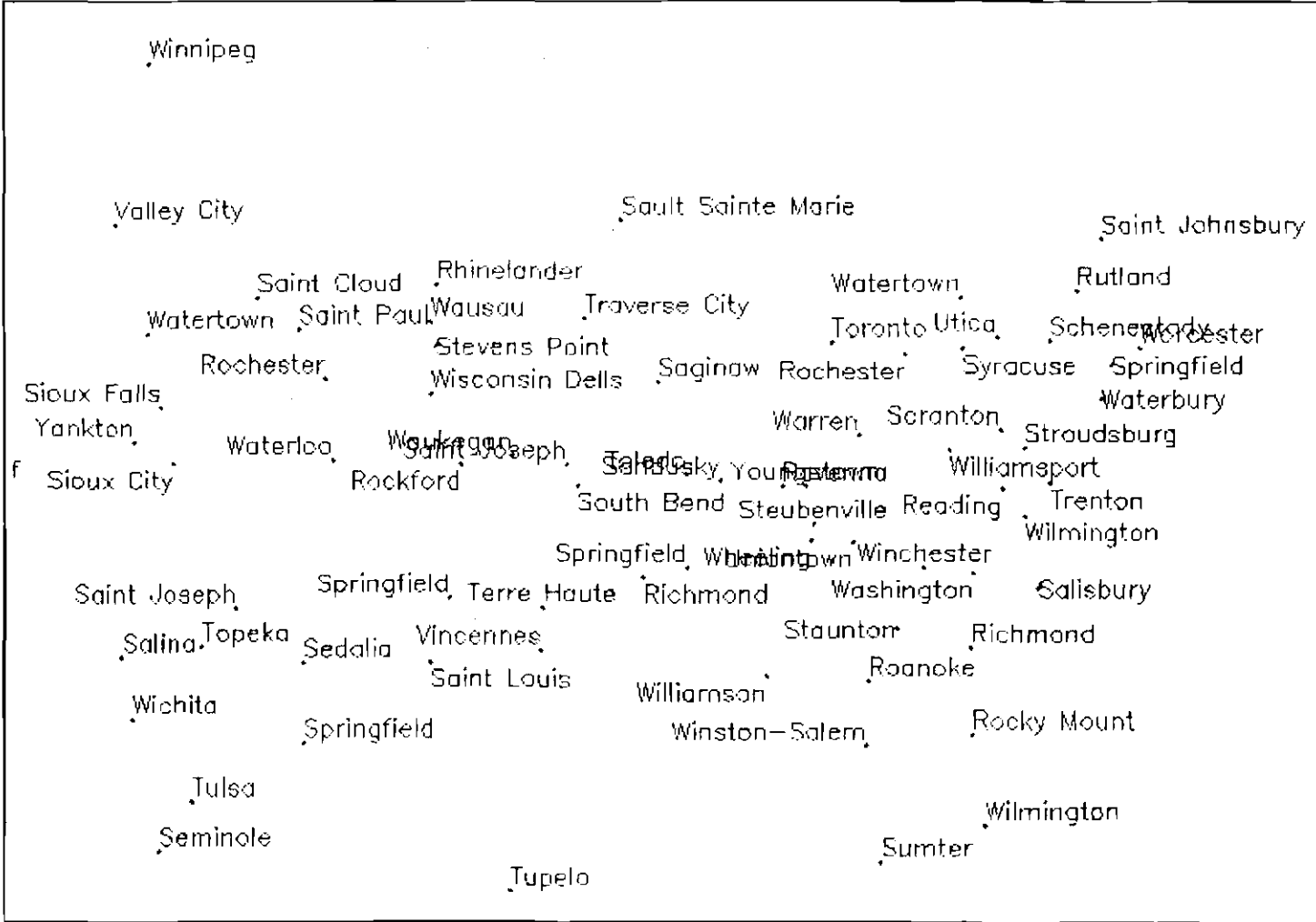
ESCALA 1:23000000

ALTURA DO CARACTER = 1.5 mm

ALFA2 = 1

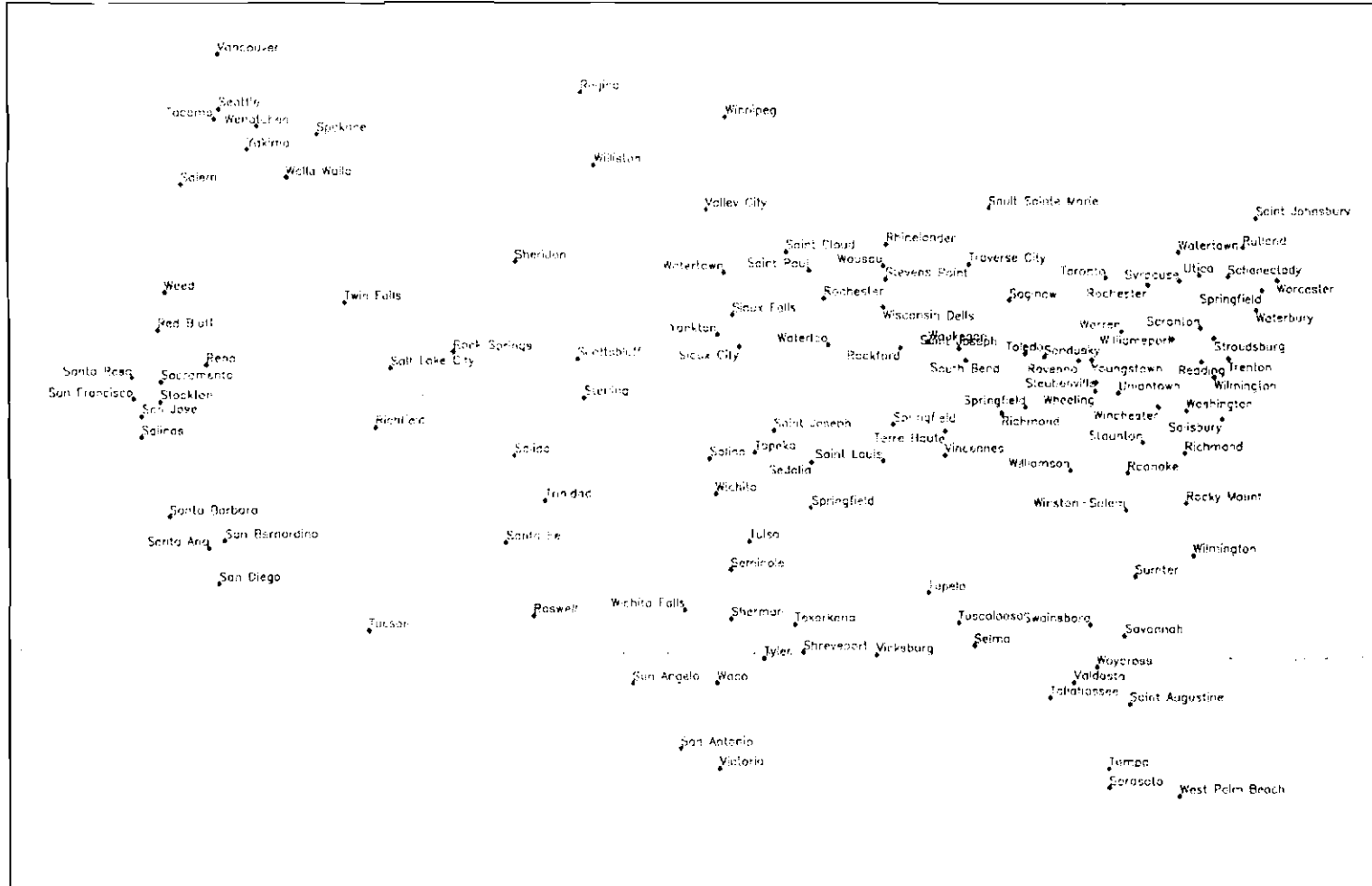
ITERAÇÃO = 560

CONFLITO = 12





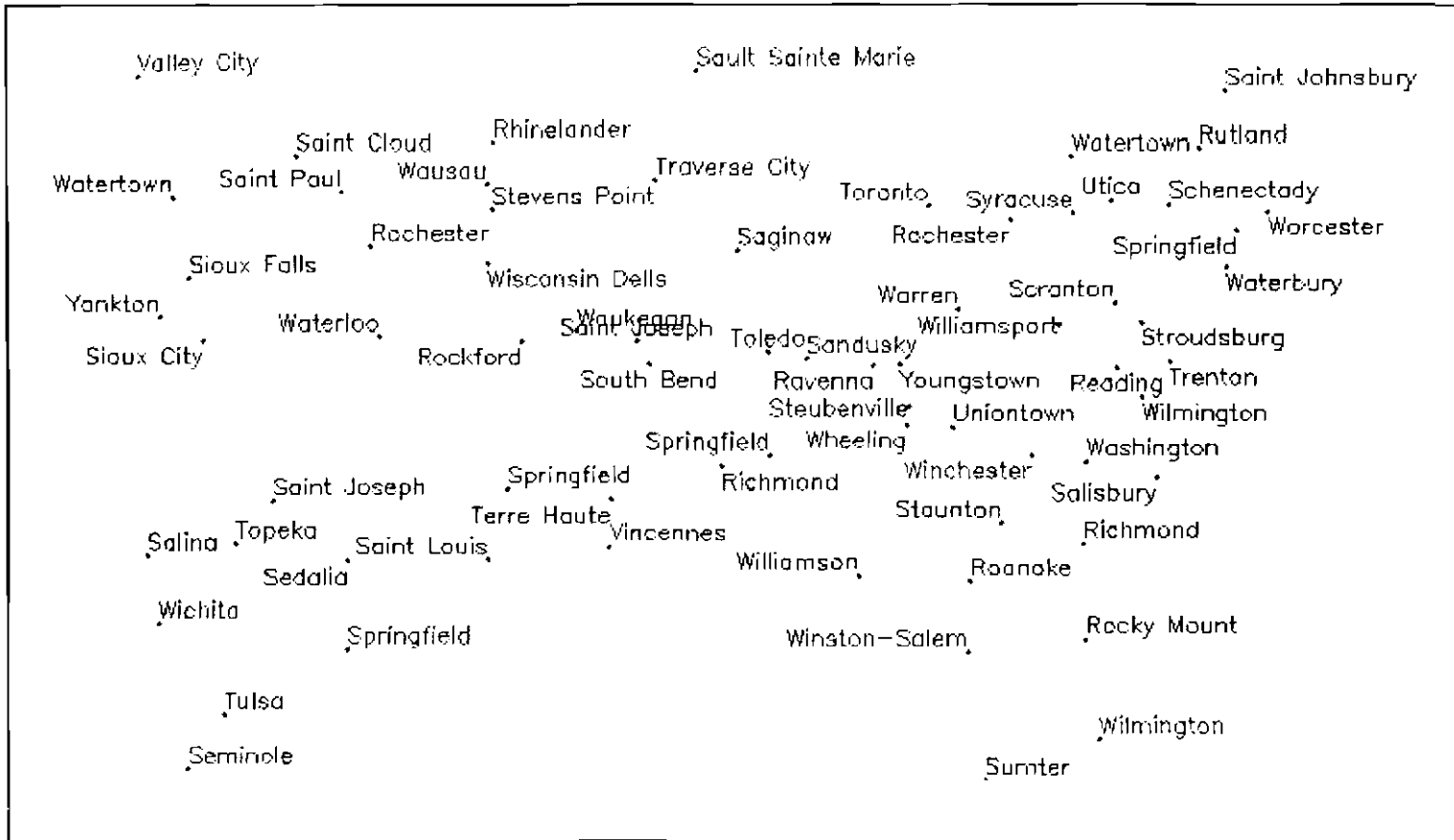
# USAG CIDADES



POSIÇÕES CANDIDATAS = B  
 ALTURA DO CARACTER = 10 mm  
 HORAÇÃO = 4812  
 ROLBUS - M. C. AFILTO = 2

A FA1 = 5  
 A FA2 = 1

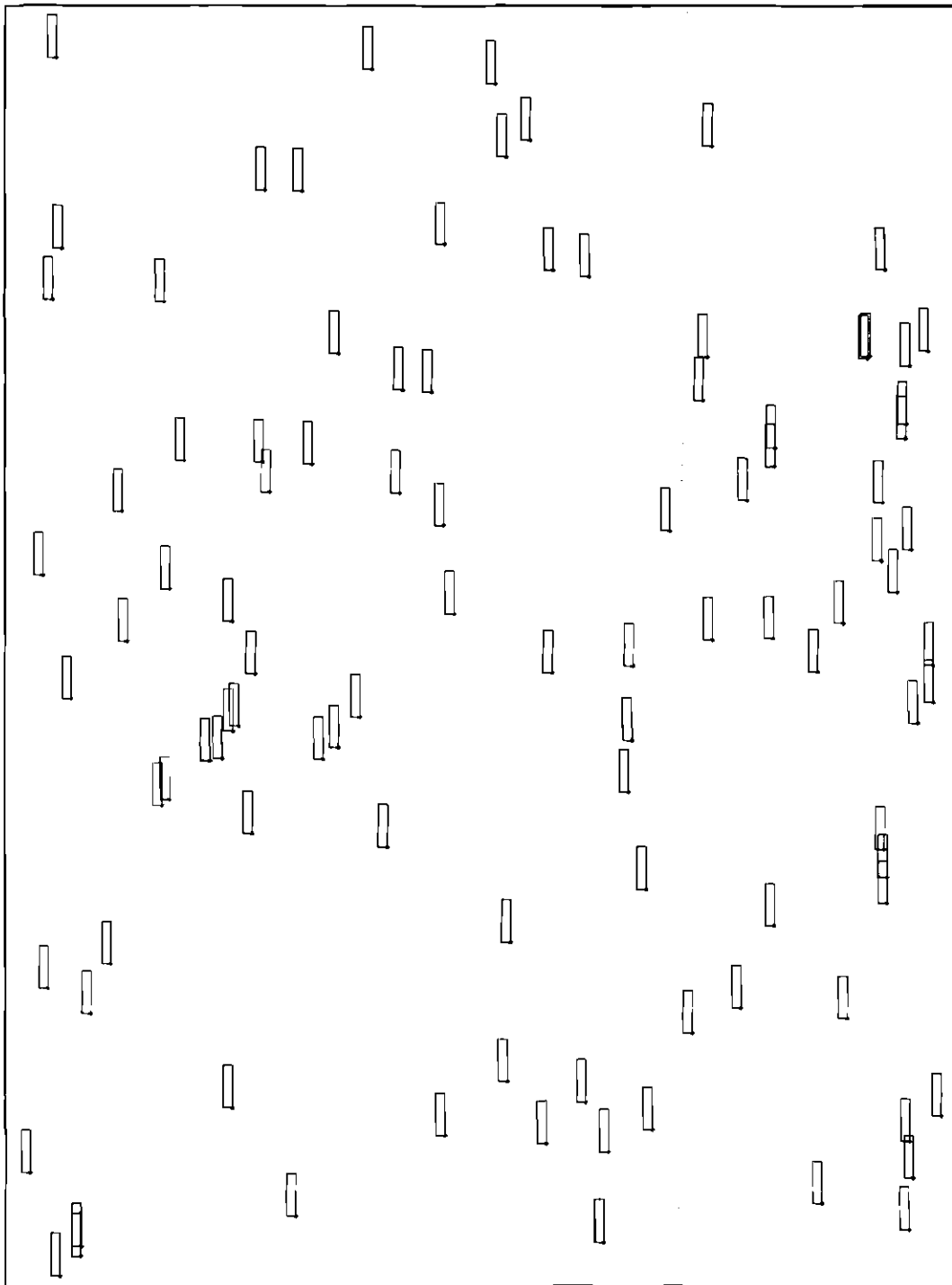
ESCALA = 1:20000000



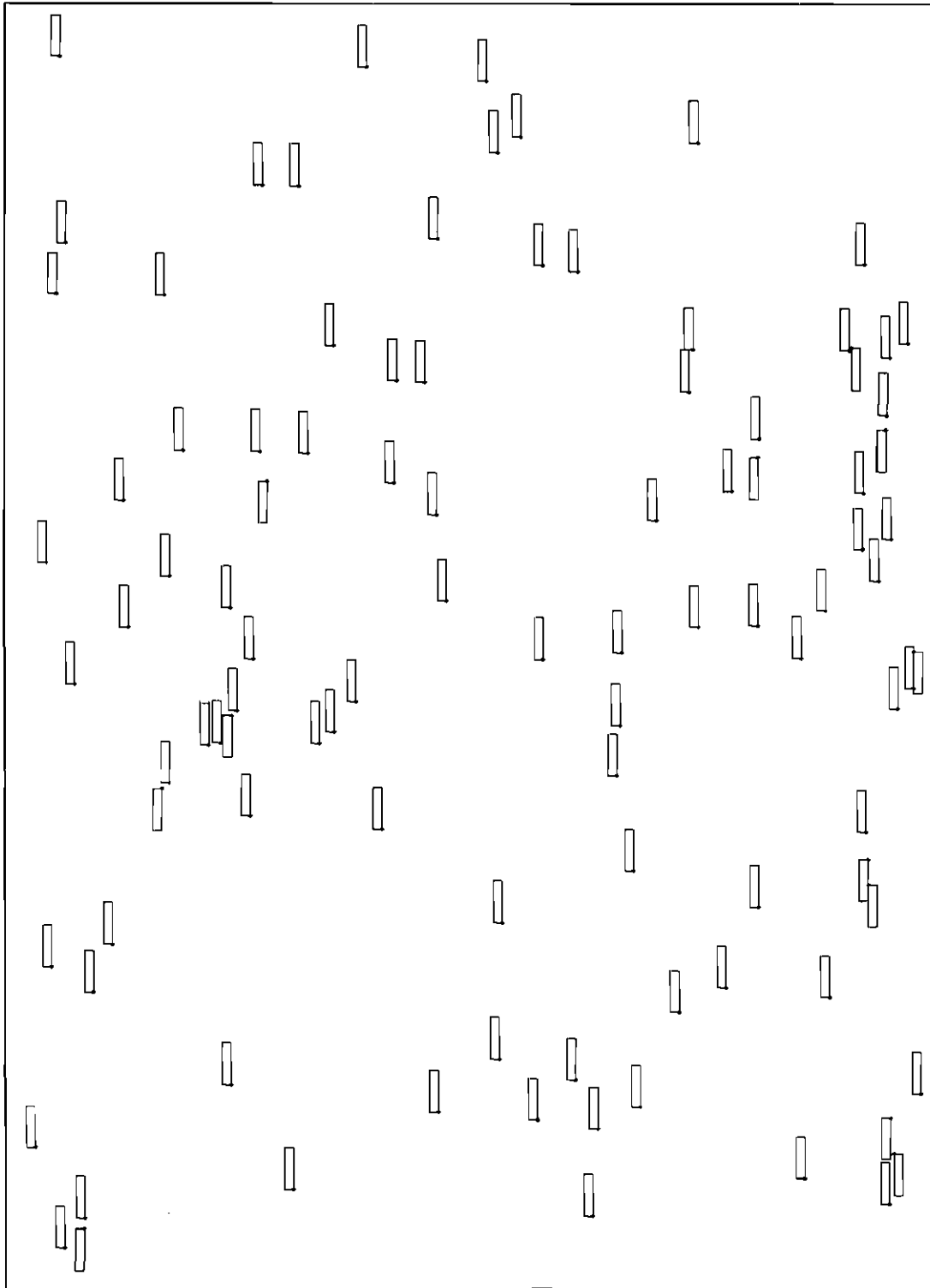
“zoom” da área de agrupamento de pontos da carta USA6 – CIDADES

**APÊNDICE A5**

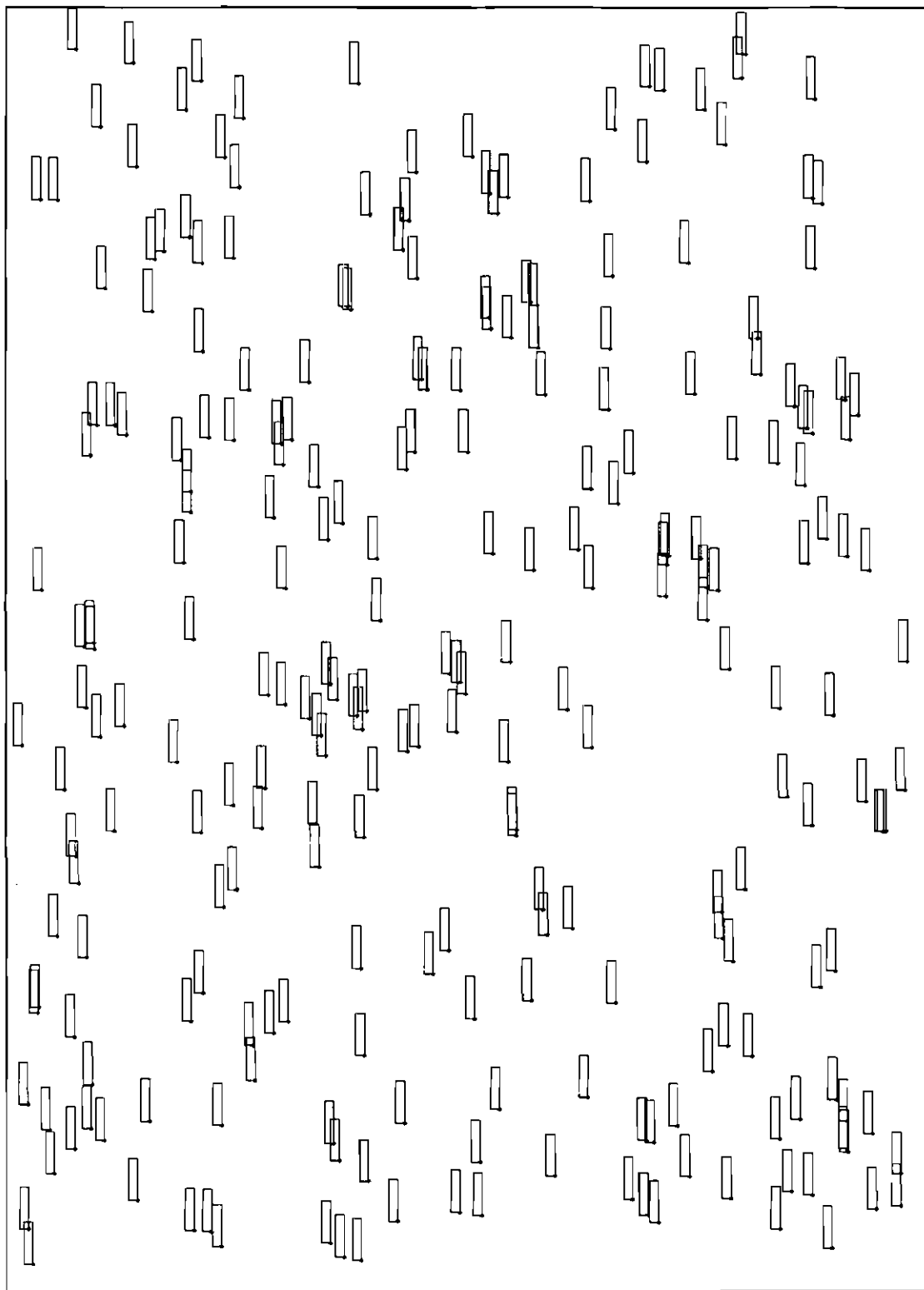
**ROTULAÇÃO DE 100 PONTOS ALEATÓRIOS  
CONFIGURAÇÃO INICIAL  
RÓTULOS EM CONFLITO = 21**



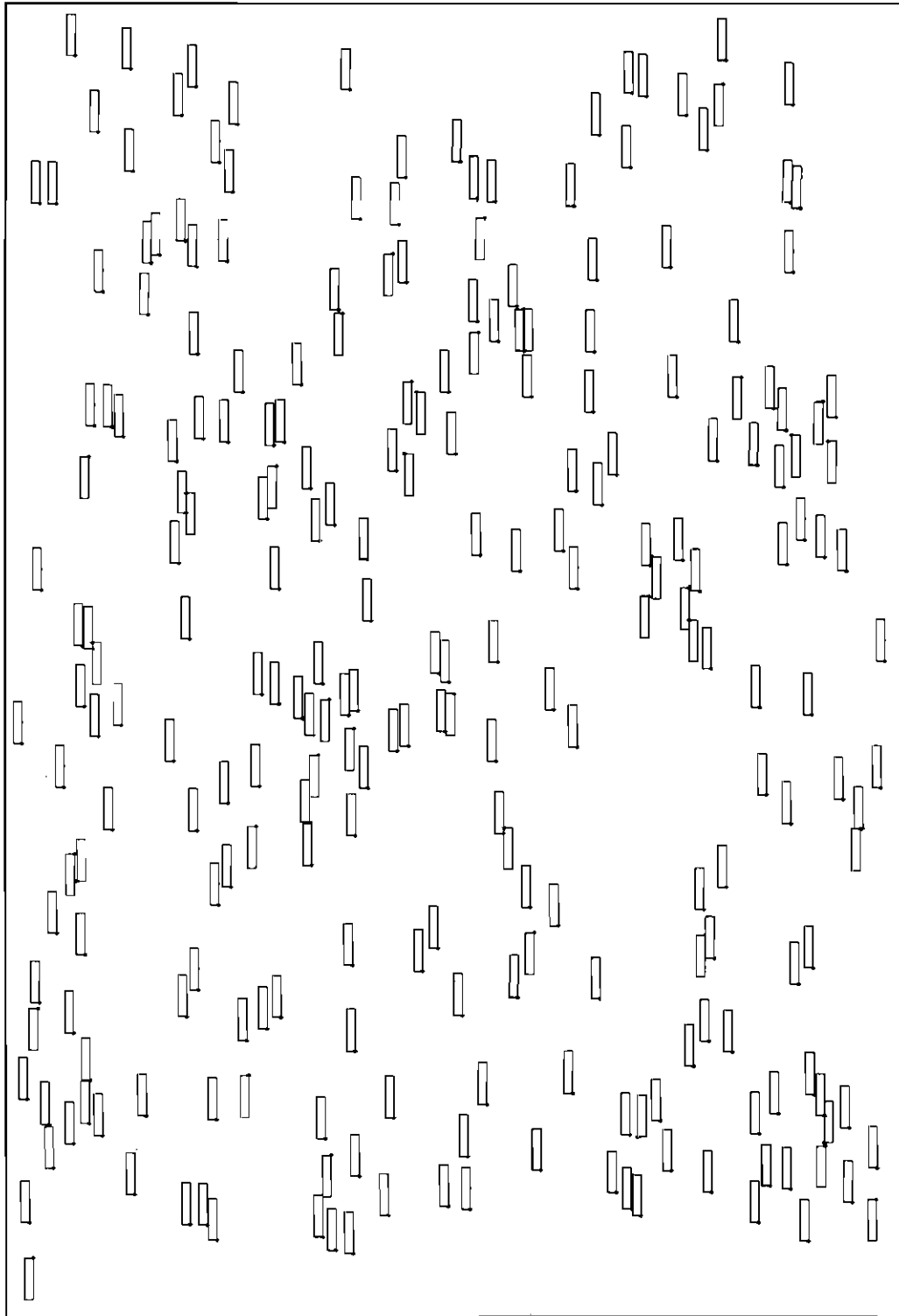
**ROTULAÇÃO DE 100 PONTOS ALEATÓRIOS  
CONFIGURAÇÃO APÓS APLICAÇÃO DA BUSCA TABU  
RÓTULOS EM CONFLITO = 0**



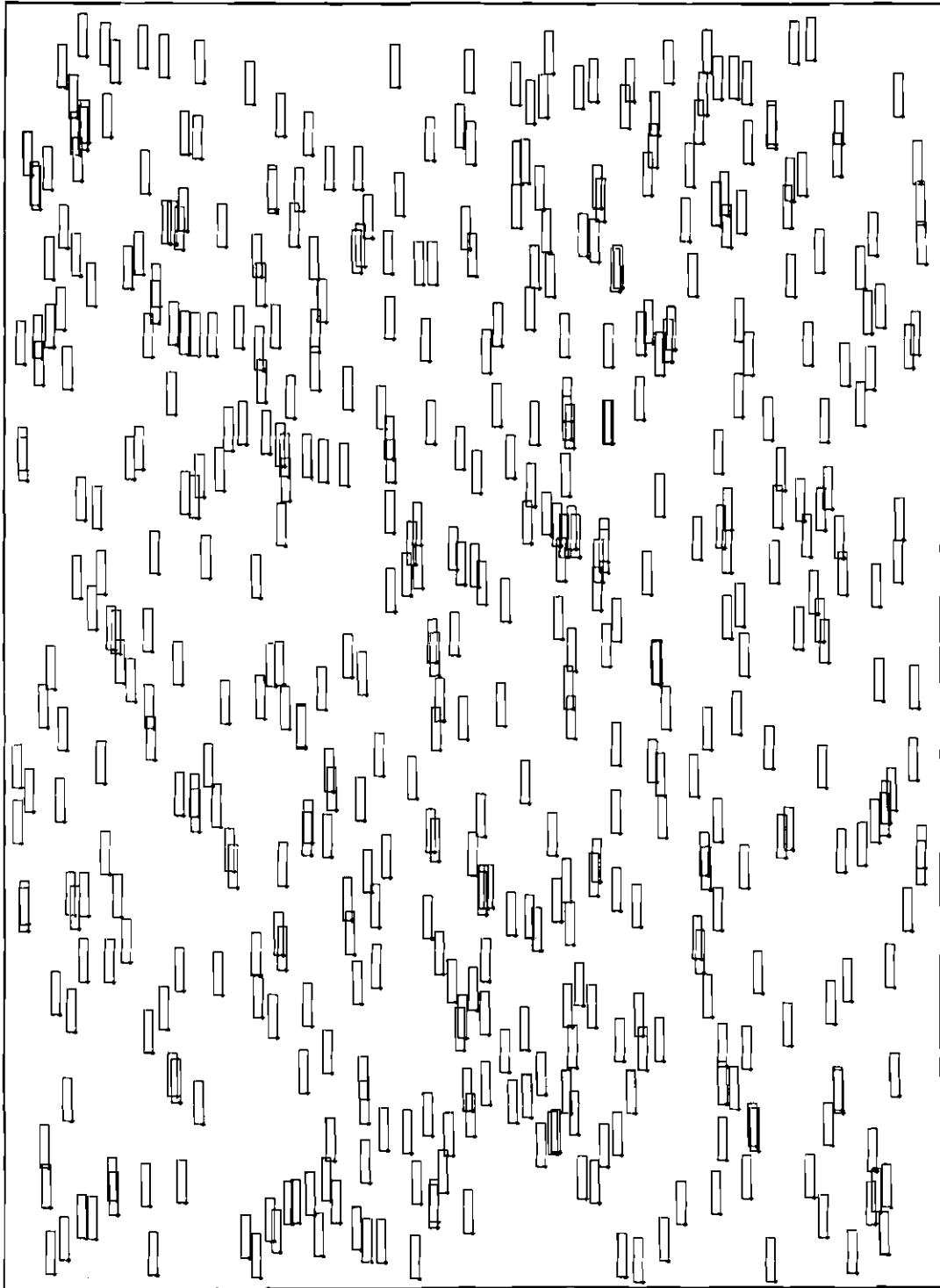
**ROTULAÇÃO DE 250 PONTOS ALEATÓRIOS**  
**CONFIGURAÇÃO INICIAL**  
**RÓTULOS EM CONFLITO = 72**



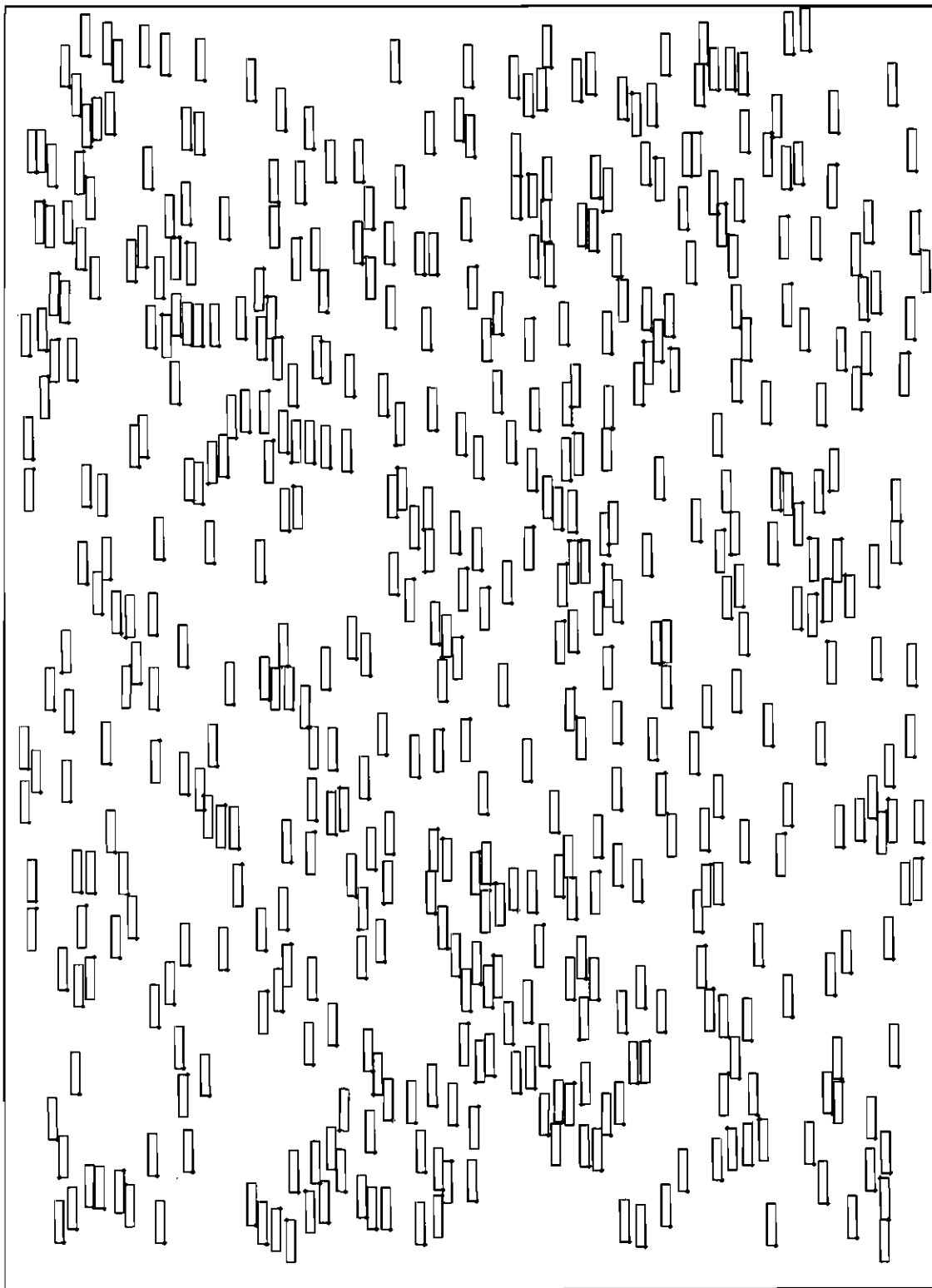
**ROTULAÇÃO DE 250 PONTOS ALEATÓRIOS  
CONFIGURAÇÃO APÓS APLICAÇÃO DA BUSCA TABU  
RÓTULOS EM CONFLITO = 0**



**ROTULAÇÃO DE 500 PONTOS ALEATÓRIOS**  
**CONFIGURAÇÃO INICIAL**  
**RÓTULOS EM CONFLITO = 222**

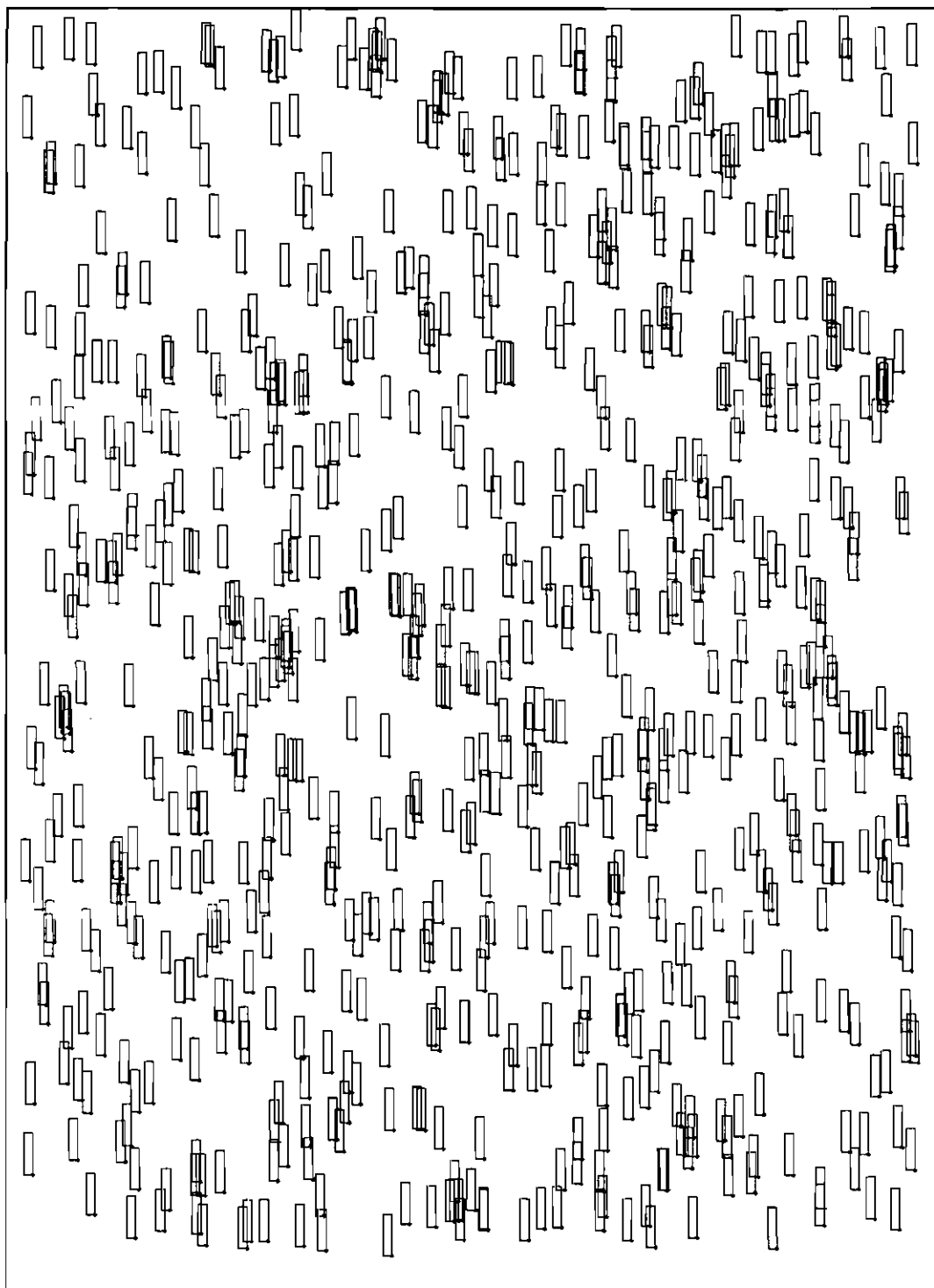


**ROTULAÇÃO DE 500 PONTOS ALEATÓRIOS  
CONFIGURAÇÃO APÓS APLICAÇÃO DA BUSCA TABU  
RÓTULOS EM CONFLITO = 0**

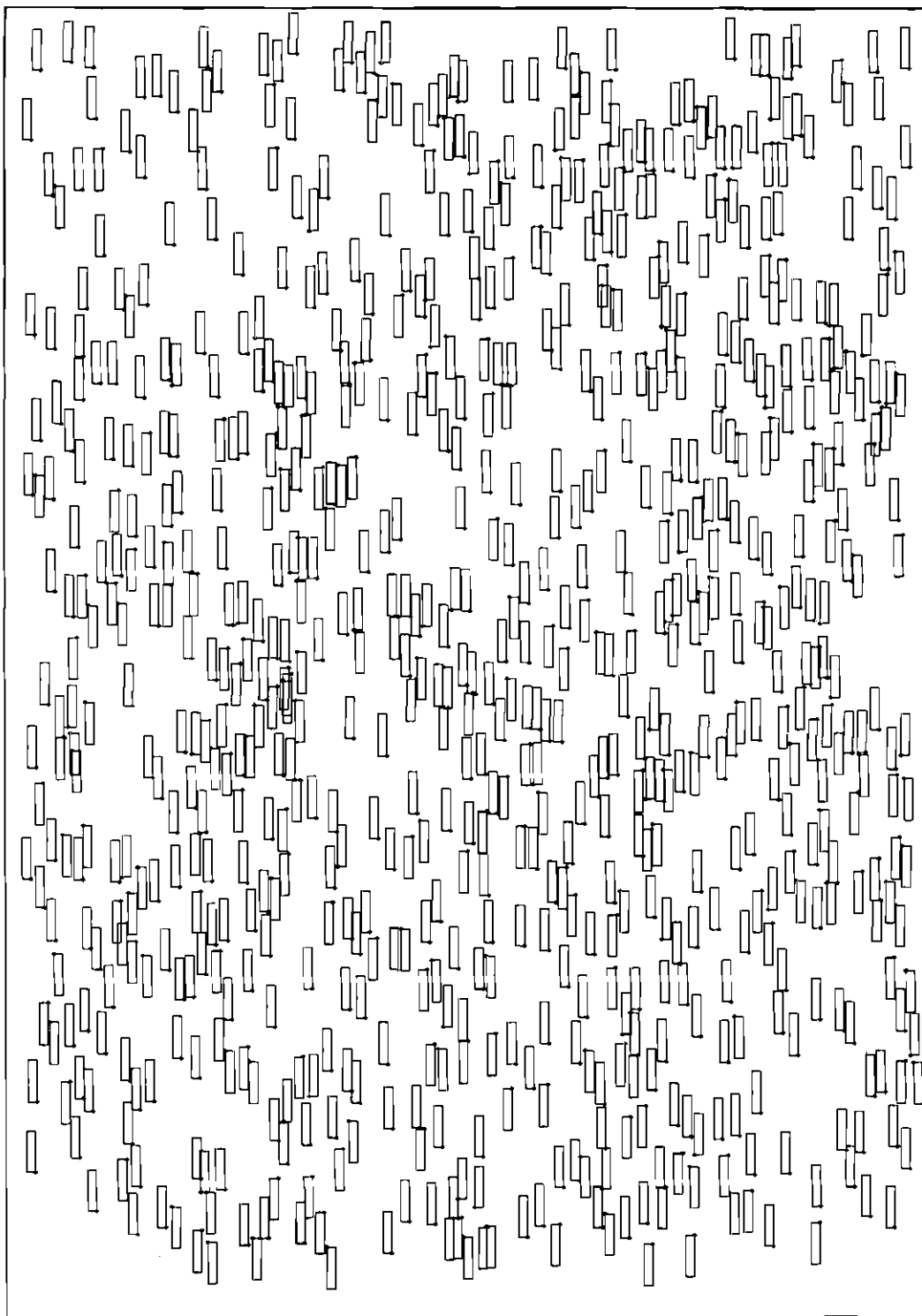




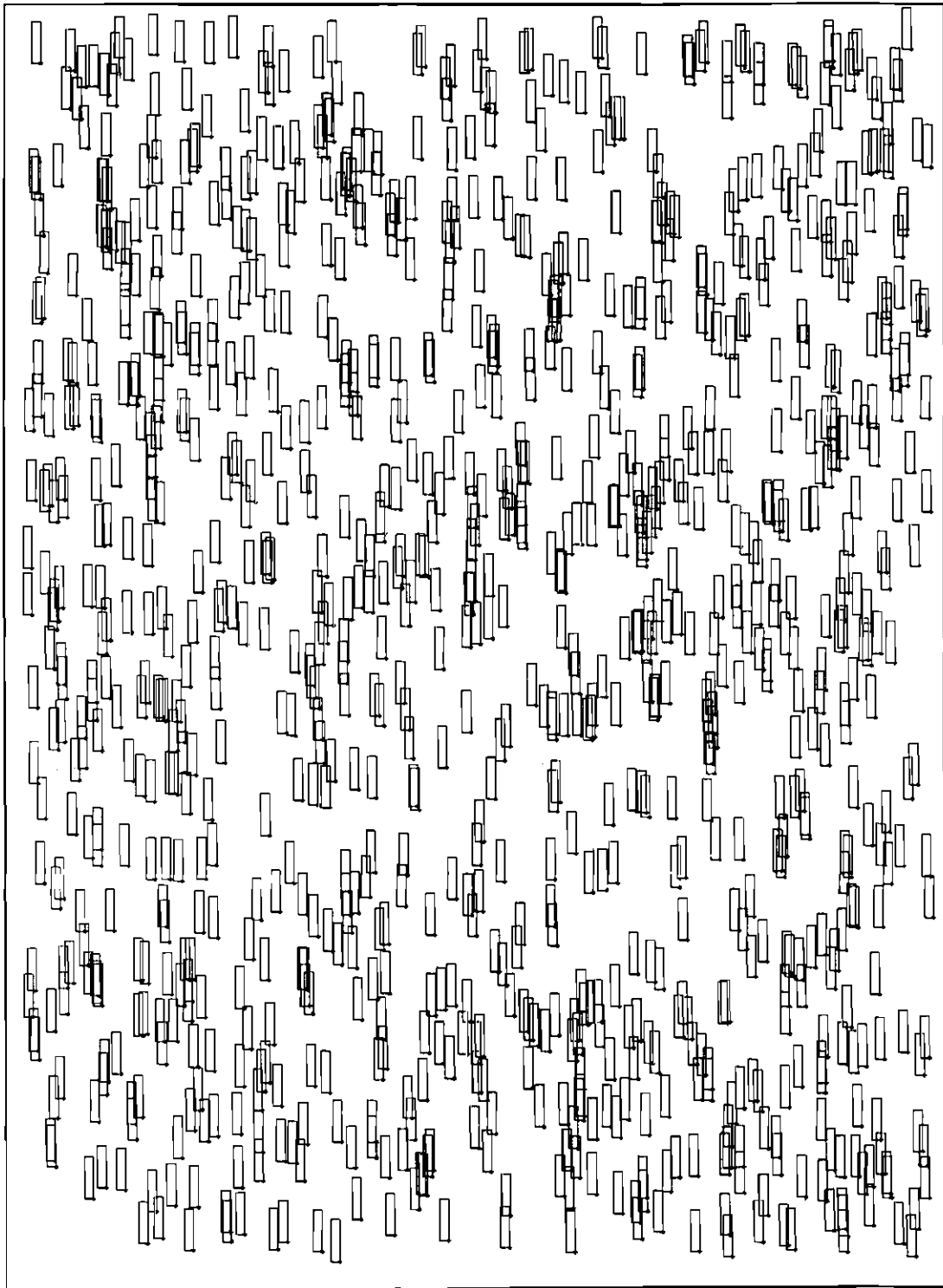
**ROTULAÇÃO DE 750 PONTOS ALEATÓRIOS  
CONFIGURAÇÃO INICIAL  
RÓTULOS EM CONFLITO = 433**



**ROTULAÇÃO DE 750 PONTOS ALEATÓRIOS  
CONFIGURAÇÃO APÓS APLICAÇÃO DA BUSCA TABU  
RÓTULOS EM CONFLITO = 15**



**ROTULAÇÃO DE 1000 PONTOS ALEATÓRIOS**  
**CONFIGURAÇÃO INICIAL**  
**RÓTULOS EM CONFLITO = 703**



**ROTULAÇÃO DE 1000 PONTOS ALEATÓRIOS  
CONFIGURAÇÃO APÓS APLICAÇÃO DA BUSCA TABU  
RÓTULOS EM CONFLITO = 77**

