



ROTEAMENTO DE VEÍCULOS COM BASE EM SISTEMAS DE INFORMAÇÃO GEOGRÁFICA

Roberto Diéguez Galvão

COPPE/Universidade Federal do Rio de Janeiro
Caixa Postal 68507 - 21945-970 - Rio de Janeiro - RJ

Júlio Francisco Barros Neto

Depto. de Estatística e Matemática Aplicada
Universidade Federal do Ceará - Fortaleza - Ceará

Virgílio J. M. Ferreira Filho

Escola de Engenharia/Universidade Federal do Rio de Janeiro
Caixa Postal 68548 - 21941-590 - Rio de Janeiro - RJ

Horácio Brescia de Sousa Henriques

COPPE/Universidade Federal do Rio de Janeiro
Caixa Postal 68507 - 21945-970 - Rio de Janeiro - RJ

Resumo

Um algoritmo para roteamento de veículos é integrado a um Sistema de Informação Geográfica, de cuja base de dados obtém as informações necessárias para o roteamento e no qual mostra as rotas resultantes. O algoritmo de roteamento utilizado tem como base a metaheurística de simulated annealing, que apresenta neste caso resultados computacionais de boa qualidade. Descrevemos a interface do algoritmo com um SIG específico e a aplicação do sistema resultante a um programa de entregas simulado, no bairro de Copacabana, no Rio de Janeiro.

Palavras-chave: roteamento de veículos, sistemas de informação geográfica, métodos heurísticos.

1. Introdução

Sistemas de distribuição física têm como um de seus componentes o roteamento de veículos de entrega, que consiste em obter rotas que minimizem custos de distribuição de uma frota de veículos operando a partir de um ou mais depósitos centrais, observadas as restrições relevantes à operação de entrega específica.

Este problema aparece, por exemplo, na distribuição de mercadorias e serviços a partir de depósitos centrais, na coleta de correspondência pelos Correios, na coleta e entrega de crianças por ônibus escolares, no roteamento de helicópteros na indústria do petróleo (GALVÃO & GUIMARÃES, 1990) e no roteamento de manutenção

preventiva em fábricas. Existe uma vasta literatura sobre roteamento, relacionada à formulação de diversos tipos de problema, métodos de solução correspondentes e aplicações. Não é objetivo deste artigo fazer uma revisão bibliográfica detalhada da área. O leitor interessado deve recorrer, entre outras, às excelentes revisões de BODIN *et al.* (1983), FISCHER (1994), GOLDEN & ASSAD (1988) e LAPORTE (1992).

Sistemas de Informação Geográfica (SIG's) foram criados para tratar dados referenciados espacialmente. Estes sistemas manipulam dados obtidos em fontes tais como mapas e imagens de satélites, entre outras, recuperando e combinando informações e permitindo vários tipos de análise. SIG's têm sido usados com frequência crescente em diversas áreas tais como análise e monitoramento do meio ambiente, planejamento urbano e regional, estudo de recursos naturais, controle de redes de transporte, distribuição de energia. Sobre este assunto pode-se consultar, por exemplo, TOMLINSON, CALKINS & MARBLE (1987), EHLER (1994) e YANG & SATHISAN (1994). A utilização de tais sistemas em larga escala se tornou possível graças à disponibilidade, aos custos razoáveis, a várias tecnologias como cartografia digital, ao desenvolvimento de bases de dados e ao processamento digital de imagens. Um SIG combina tipicamente estas tecnologias com técnicas de análise e manipulação de informação espacialmente referenciada.

A introdução de SIG's foi uma consequência prática do desenvolvimento da computação gráfica nos anos 80. Os modelos computacionais da Pesquisa Operacional (PO) estavam, concomitantemente, sendo analisados de maneira crítica. Por exemplo, estava se tornando cada vez menos aceitável apresentar resultados de simulação de sistemas operacionais na forma de simples

tabelas e gráficos. Vários *software* gráficos de simulação foram desenvolvidos durante os anos 80, nos quais as interações entre elementos dos sistemas simulados e a formação e o gerenciamento de filas são representados em forma gráfico-visual. O papel de modelos de otimização da PO tendo em vista estas novas tecnologias começou a ser amplamente discutido. Ficou evidente que os modelos da PO teriam de ser cada vez mais embutidos em sistemas computacionais gráficos e de forma transparente ao usuário não-especialista.

O que nos propomos a investigar no presente artigo é a integração de algoritmos de roteamento a Sistemas de Informação Geográfica. É nesse contexto que embutimos um algoritmo de roteamento desenvolvido com base na técnica de *simulated annealing* (SA) em um sistema de roteamento com base em SIG's previamente desenvolvido (ver GALVÃO *et al.*, 1994). O uso de um algoritmo de roteamento SA justifica-se por sua simplicidade e facilidade de programação para computadores, pelo bom desempenho do algoritmo em termos de qualidade das soluções obtidas (ver Apêndice) e pelos baixos tempos computacionais envolvidos.

A utilização de informações geográficas precisas se tornou um ponto central no desenvolvimento de sistemas de distribuição. A integração de bases de dados geográficos e sistemas de roteamento de veículos e a localização e o monitoramento de veículos operando na distribuição de mercadorias e serviços (ver GOLDEN & ASSAD, 1988), são aplicações importantes em logística. Em particular, a integração de algoritmos de roteamento a SIG's é um tema atual e pouco desenvolvido do ponto de vista comercial (ver DAVIS, 1996).

Este artigo está organizado da seguinte maneira: na Seção 2 discutimos, brevemente, métodos de solução para o problema de roteamento de veículos,

definimos o problema específico tratado no artigo e descrevemos sua solução por meio de *simulated annealing*. Isto é seguido, na Seção 3, pela descrição da integração do algoritmo de roteamento a um SIG. Uma

aplicação do esquema proposto a um programa de entregas simulado, no bairro de Copacabana, no Rio de Janeiro, é mostrada na Seção 4. Finalmente, apresentamos na Seção 5 as conclusões.

2. Definição de um Problema de Roteamento e sua Solução por *Simulated Annealing*

Problemas de roteamento de veículos são problemas de otimização combinatoria pertencentes à categoria NP-árduos (*NP-hard*), o que significa que o tempo computacional necessário à obtenção da solução ótima é uma função exponencial do tamanho do problema. Em consequência, mesmo para as formulações mais simples, algoritmos exatos são utilizados na prática apenas para problemas de tamanho médio (com não mais de 50 clientes), para evitar tempos computacionais excessivos. Na grande maioria das aplicações práticas são utilizados métodos heurísticos, que produzem soluções aproximadas de boa qualidade em tempo reduzido.

LAPORTE & NOBERT (1987) e LAPORTE (1992) apresentam revisões bibliográficas detalhadas de algoritmos exatos e heurísticos para problemas de roteamento. Neste estudo será apresentado apenas um resumo dos principais métodos heurísticos disponíveis, com a finalidade de situar o leitor no tópico.

O primeiro método heurístico desenvolvido para o problema foi o algoritmo de *economias* de CLARKE & WRIGHT (1964). Melhoramentos do método de Clarke & Wright foram desenvolvidos por GASKELL (1967), YELLOW (1970), GOLDEN, MAGNANTI & NGUYEN (1977) e PAESSENS (1988). Uma heurística que generaliza o conceito de *economias* de Clarke & Wright foi proposto por MOLE & JAMESON (1976).

Todas as heurísticas mencionadas acima são do tipo construtivo, nas quais rotas são

construídas ao longo do procedimento. Outra classe de métodos heurísticos são as heurísticas de melhoria iterativa: obtida uma solução inicial por meio de uma heurística construtiva, aplica-se por exemplo um algoritmo de troca de arestas (LIN, 1965; LIN & KERNINGHAM, 1973) a cada uma das rotas obtidas por meio da heurística inicial, como no método 3-otimal de CHRISTOFIDES & EILON (1969) e no algoritmo de RUSSEL (1976).

O algoritmo de GILLET & MILLER (1974), também conhecido por algoritmo de *varredura*, representa uma idéia diferente, sendo um método de duas fases que tem por base a estratégia de “agrupar primeiro, rotear em seguida”. Outros algoritmos que usam a mesma estratégia são as heurísticas de CHRISTOFIDES, MINGOZZI & TOTH (1979) e FISHER & JAIKUMAR (1981). BEASLEY (1983) propõe métodos de solução com base na idéia de “rotear primeiro, agrupar em seguida”.

Mais recentemente, diversos autores têm recorrido ao uso de metaheurísticas do tipo *simulated annealing* e busca tabu para a solução de problemas de roteamento. Sobre o assunto ver por exemplo GOLDEN & SKISCIM (1986), OSMAN (1993), GENDREAU, HERTZ & LAPORTE (1994). A técnica de *simulated annealing*, pela simplicidade dos conceitos utilizados e facilidade de programação para computadores, tem sido a mais utilizada entre as metaheurísticas.

2.1 A Técnica de *Simulated Annealing*

Simulated Annealing é uma metaheurística sobreposta a métodos iterativos de busca local e representa uma variação de métodos tradicionais descendentes (ascendentes) de otimização local ou busca por vizinhança. Seja um problema de otimização combinatória representado por um par (Ω, Z) , onde Ω é o conjunto de todas as soluções viáveis e Z é uma função objetivo a ser minimizada ou maximizada. Em um problema de minimização busca-se uma solução ótima $S^* \in \Omega$ tal que $Z(S^*) \leq Z(S) \forall S \in \Omega$ (veja PAPADIMITRIOU & STEIGLITZ, 1982). Um mecanismo de geração de vizinhanças define para cada solução $S \in \Omega$ o conjunto de todas as soluções $S' \in \Omega$ que podem ser geradas de acordo com certas regras. Uma transição de S para S' em $N(S)$ é chamada de **movimento** e $N(S)$ é chamada de **vizinhança** de S . Uma solução S é chamada de **mínimo local** em relação ao mecanismo de geração

se e somente se $Z(S) \leq Z(S') \forall S' \in N(S)$ (veja OSMAN & CHRISTOFIDES, 1994).

Em algoritmos descendentes de busca local apenas movimentos que reduzam o valor da função objetivo são aceitos. Além destes, *simulated annealing* ocasionalmente aceita movimentos que aumentem o valor da função objetivo, por intermédio de uma estratégia de aceitação probabilística.

SA aceita um movimento de S para $S' \in N(S)$ que piore a solução com probabilidade $e^{-\Delta/T}$, na qual $\Delta = Z(S') - Z(S)$ e T é um parâmetro positivo chamado **temperatura**. Os valores de T são atualizados segundo uma **função de redução de temperatura**. Um algoritmo com base em SA termina quando um dos critérios de parada é atingido. Para revisões da teoria e aplicações de SA o leitor interessado pode recorrer a VAN LAARHOVEN & AARTS (1987), CONOLLY (1992), EGGLESE (1990) e DOWSLAND (1993), entre outros.

2.2 Um Problema de Roteamento Capacitado com Restrições de Distância (PRC-RD)

Com o objetivo de investigar a integração de modelos de roteamento a SIG's, decidimos utilizar o seguinte problema capacitado de roteamento com restrições de distância, que chamamos de PRC-RD. Seja $G=(V,A)$ um grafo direcionado no qual $V = \{v_0, \dots, v_n\}$ é um conjunto de vértices e $A = \{(v_i, v_j), v_i, v_j \in V, i \neq j\}$ é um conjunto de arcos. v_0 representa um depósito central e todos os outros v_i representam clientes com demandas d_i por um dado produto ou serviço. Os clientes devem ser supridos a partir do depósito central por uma frota de m veículos homogêneos, cada um com capacidade Q . O custo de viajar entre os vértices v_i e v_j é dado por uma matriz de custos não-negativa $C=[c_{ij}]$. O problema consiste em, dada uma frota compatível com

os requisitos do problema, obter um conjunto de rotas de custo total mínimo, de modo que cada cliente seja visitado uma única vez e que as seguintes restrições sejam satisfeitas:

R1 - a demanda de cada cliente seja totalmente atendida;

R2 - a capacidade de cada veículo não seja excedida;

R3 - o tempo total gasto por um veículo em sua rota (ou, alternativamente, a distância total viajada pelo veículo) não exceda um limite superior T (L) pré-definido. Se c_{ij} representa o tempo de viagem e cada cliente tem um tempo de serviço não-negativo γ_i , então
$$\sum_{i \in R} \sum_{j \in R} c_{ij} + \sum_{i \in R} \gamma_i \leq T$$
 deve ser satisfeita para

toda rota $r \in \mathcal{R}$, na qual \mathcal{R} é o conjunto das rotas na solução problema.

Embora funções objetivo diferentes possam ser definidas para descrever o custo total do sistema, na formulação acima busca-se minimizar a distância total viajada pela frota de veículos (minimização dos cus-

tos variáveis). Como alternativas podemos ter a minimização do número de veículos utilizados (minimização dos custos fixos), ou a minimização de uma combinação de custos fixos e variáveis.

2.3 Solução do PRC-RD através de *Simulated Annealing*

A aplicação de SA a um problema de otimização combinatória exige a definição de: (i) uma solução inicial viável para o problema; (ii) um mecanismo de geração de vizinhanças; (iii) uma estratégia de seleção de soluções vizinhas; (iv) valor inicial do parâmetro de controle (*temperatura* T_0); (v) *função de redução de temperatura*; (vi) critérios de parada. Como a ênfase do presente artigo está na integração de um algoritmo de roteamento a um SIG, omitiremos o detalhamento destes tópicos para o PRC-RD, mas o leitor interessado poderá consultar BARROS NETO (1995). Vale notar que o algoritmo inclui uma rotina de otimização de rotas individuais por meio

da solução de Problemas do Caixeiro Viajante (PCV's).

O algoritmo SA desenvolvido para o PRC-RD segue os passos padronizados para esta técnica de solução e também não será descrito aqui; uma descrição completa do mesmo pode ser também encontrada em BARROS NETO (1995). Este algoritmo foi testado em problemas disponíveis na literatura e os resultados obtidos comparados com outros métodos de solução desenvolvidos para o problema. Uma comparação do desempenho (qualidade das soluções obtidas) de diversos métodos heurísticos disponíveis para o PRC-RD é mostrada no Apêndice.

3. Integração do Algoritmo a um *Software* SIG

Desenvolvemos um *software*, que denominamos de **FazRota**, para servir de interface entre o algoritmo SA e o *software* SIG **MapInfo**, versão 3.0. Em sua versão atual, **FazRota** obriga os dados de roteamento (endereços de depósito e clientes, informações sobre a frota de entrega, demandas e tempos de serviço dos clientes, comprimentos máximos de rotas) a serem armazenados em sua base de dados. Uma nova versão em desenvolvimento extrairá parte destes dados do SIG correspondente.

Após dados do depósito, clientes, frota de entrega e outras informações relevantes para o roteamento serem selecionadas de sua base de dados, **FazRota** associa vértices a

interseções de ruas e a clientes a serem visitados. Levando em conta a direção do tráfego, o *software* gera um grafo direcionado correspondente. O algoritmo de Floyd é então aplicado a este grafo; a matriz de distâncias gerada pelo mesmo é uma das informações de entrada para o algoritmo SA. Em seguida, este algoritmo é então executado de dentro de **FazRota**. A solução produzida pelo mesmo é processada pelo *software* e os arquivos de interface com **MapInfo** são gerados. **MapInfo** é finalmente acionado, e o usuário pode utilizar esse *software* para mostrar as rotas obtidas em mapas e formato tabular, ou para proceder a outros tipos de análise com os dados. Um diagrama de blocos do sistema

de roteamento que combina **FazRota** com alguns módulos de **MapInfo** é mostrado na Figura 1.

Graças a sua interface amigável, com janelas do tipo *pop-up*, o usuário não encontra dificuldades em utilizar **FazRota**. O software também produz relatórios detalhados relativos a cada rota de entrega, facilitando as tarefas dos motoristas dos caminhões de entrega.

Uma ilustração de **FazRota** é mostrada nas Figuras 2 a 5. Respectivamente, as

Figuras 2 e 3 mostram a base de dados dos depósitos e um subconjunto dos clientes a serem visitados no programa de entregas da Seção 4. A Figura 4 ilustra os algoritmos que podem ser ativados por **FazRota**: o algoritmo de Floyd, o algoritmo de roteamento de Gillet & Miller e o algoritmo SA deste artigo. Parte do relatório correspondente a uma rota de entrega (relatório do motorista) é mostrada na Figura 5.

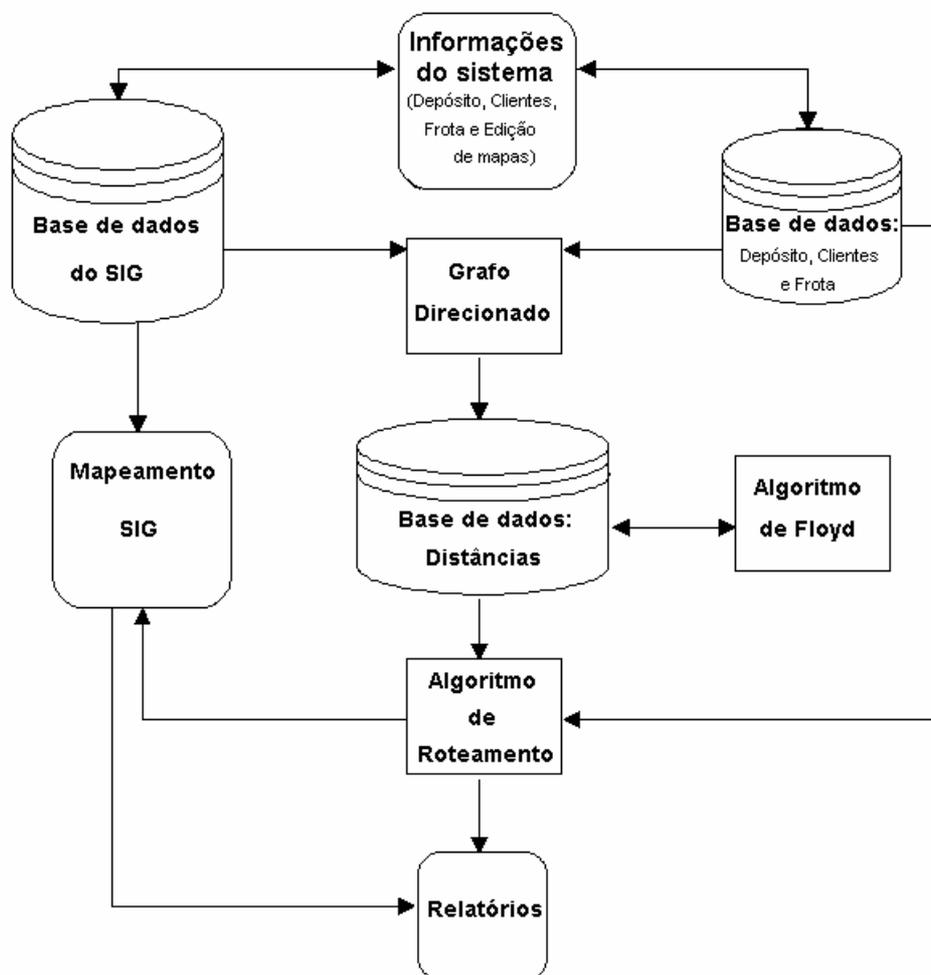


Figura 1 - Diagrama de Blocos do Sistema de Roteamento



Figura 2 - Base de Dados dos Depósitos



Figura 3 - Endereços dos Clientes a serem Visitados



Figura 4 - Algoritmos Ativados por FazRota

```

-----
Capacidade:1600|Dist. Total s/ TD:7981 m|Dist. Total c/ TD:47981 m
-----
>>>  ROTEIRO:  Deposito:"R. Ronald de Carvalho" 250,
              "R. Ronald de Carvalho",
              "Av. Nossa Senhora de Copacabana",

/=====\
| Parada:001 | LE BEC FIN
\=====/
| Endere#o: "Av. Nossa Senhora de Copacabana" 178
| Demanda :      174.00
\=====/

              "Av. Nossa Senhora de Copacabana",
              "R. Belford Roxo",
              "R. Min. Viveiros de Castro",
              "R. Rodolfo Dantas",

/=====\
| Parada:002 | CAFE BAR MACAU
\=====/
| Endere#o: "R. Rodolfo Dantas" 87
| Demanda :      67.00
\=====/

```

Figura 5 - Relatório do Motorista (Parcial) para uma Rota de Entrega

4. Aplicação a um Programa de Entregas em Copacabana

O sistema de roteamento desenvolvido no presente artigo foi aplicado a um programa de entregas simulado, no bairro de Copacabana, no Rio de Janeiro. Este programa consiste na entrega de refrigerantes a 40 bares e restaurantes em Copacabana. Os bares, os restaurantes e seus endereços são reais e foram escolhidos da lista telefônica.

Entretanto, esta aplicação corresponde apenas a um teste do sistema de roteamento, sendo fictício por não corresponder a uma entrega real de mercadorias. A frota de entrega é composta por 5 caminhões, cada um com capacidade $Q = 1.600$ quilos e a distância é restrita a não mais de 100 quilômetros por rota, incluídos aí os tempos de entrega equivalentes. As distâncias

viajadas são mostradas em metros e o tempo de entrega a cada cliente foi fixado no equivalente de 5 quilômetros de distância viajada. O depósito foi, também, localizado em um endereço em Copacabana.

O mapa do referido bairro Copacabana foi manualmente alimentado ao **MapInfo** e é mostrado na Figura 6. No mesmo está incor-

porada a direção do tráfego nas ruas de Copacabana. Quando o algoritmo SA foi aplicado ao programa de entregas foram produzidas 4 rotas, cujos detalhes estão na Tabela 1. O conjunto das 4 rotas agregado por **MapInfo** ao mapa de Copacabana é mostrado na Figura 7 e um mapa da rota 3, na figura 8.

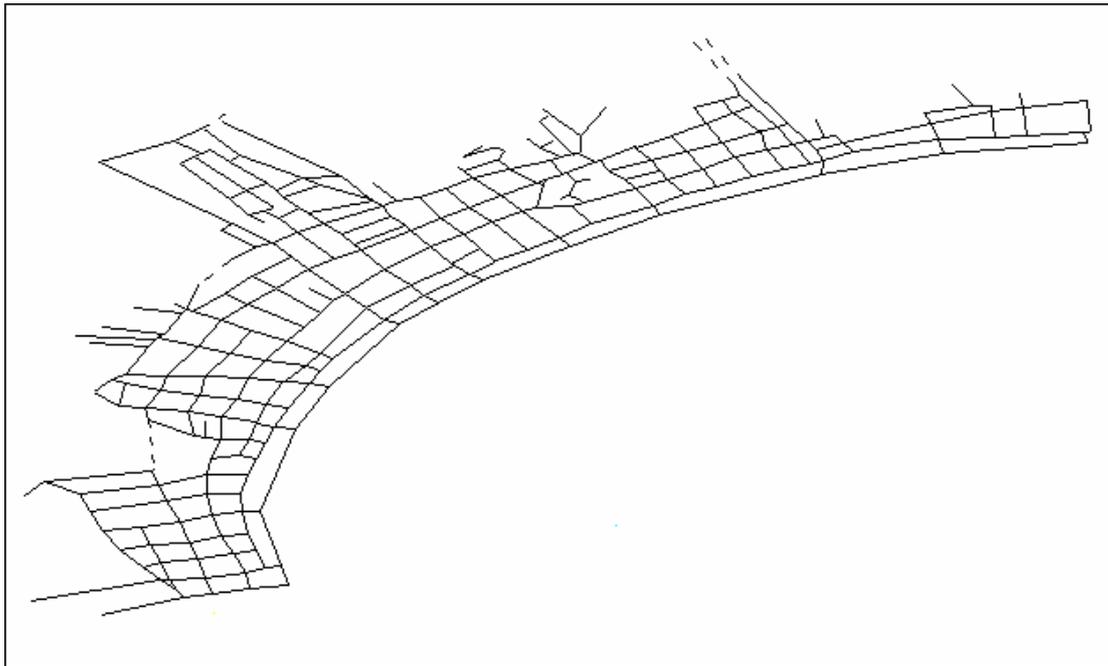


Figura 6 - Mapa de Copacabana Alimentado ao MapInfo

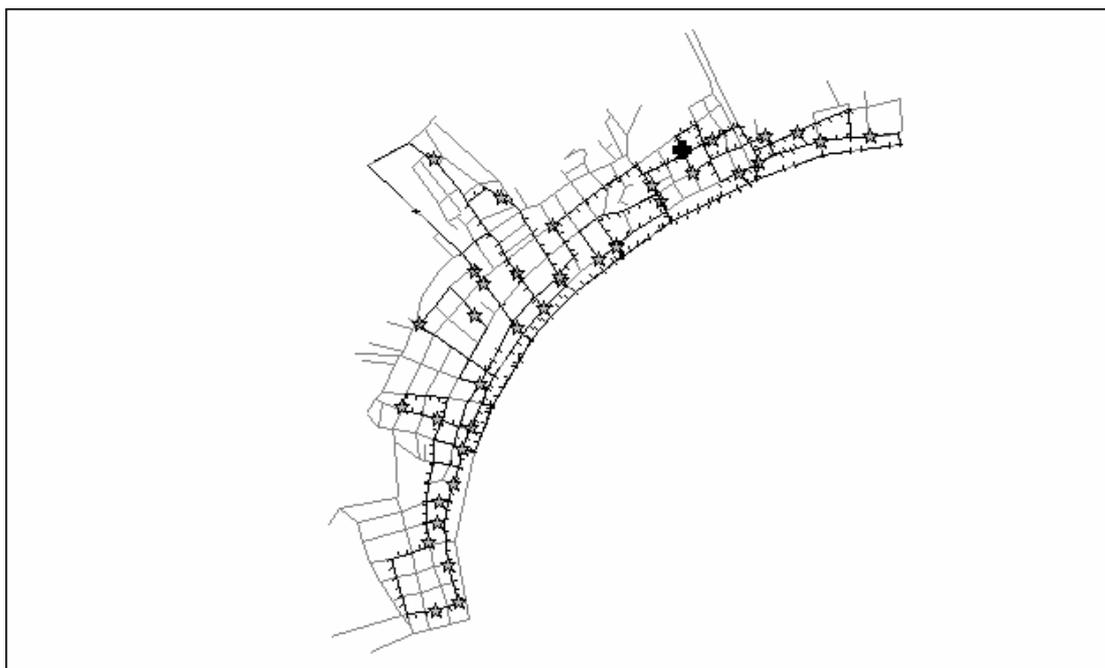


Figura 7 - Conjunto de Rotas Mostrado no Mapa de Copacabana

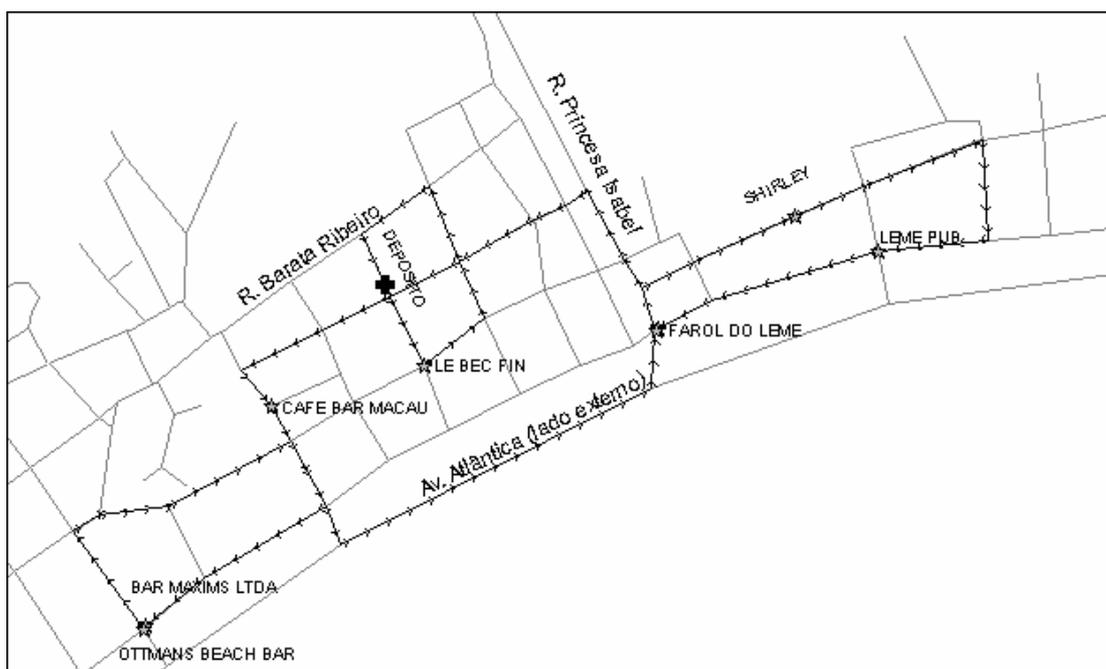


Figura 8 - Mapa da Rota 3

Tabela 1 - Detalhes de 4 Rotas Produzidas pelo Algoritmo SA

Rota 1			Rota 2			Rota 3			Rota 4		
Cliente	Demanda	Distância									
Depós.			Depós.			Depós.			Depós.		
26	107	1492,8	7	120	1698,1	34	174	502,0	17	58	1715,8
40	344	997,4	21	137	1884,3	5	67	273,5	9	90	485,1
23	275	439,5	38	40	1064,2	32	209	932,0	2	93	724,4

36	194	2292,7	25	228	2501,2	14	148	933,2	27	117	1768,1
33	243	711,7	41	186	591,9	31	202	1907,3	3	142	877,8
15	183	431,5	22	69	1128,1	11	166	479,5	30	55	993,4
35	201	607,0	39	302	415,6	28	199	916,7	24	81	578,3
19	38	2952,8	13	77	827,3	10	211	1255,6	6	100	531,0
Depós.		1383,9	18	178	490,2	Depós.		781,0	37	132	810,8
			16	83	949,8				4	111	934,1
			Depós.		742,5				8	45	848,4
									20	247	442,7
									12	142	2762,2
									29	86	749,8
									Depós.		364,4
TOTAL	1585	11309,3	TOTAL	1420	12293,2	TOTAL	1376	7980,8	TOTAL	1499	14586,3

5. Conclusões

A integração de modelos de PO a sistemas computacionais gráficos está se tornando uma prática bem estabelecida a partir da introdução destes sistemas na década de 80. O uso de SIG's em várias áreas de aplicação importantes foi possível devido à disponibilidade e aos custos moderados de várias tecnologias correlatas. A integração de bases de dados geográficos e sistemas de roteamento de veículos é uma aplicação importante de SIG's na distribuição física.

O presente artigo demonstra a viabilidade de tal integração. O algoritmo SA foi desenvolvido e testado no contexto do PRC-RD, produzindo rotas de boa qualidade em tempo computacional reduzido. Este algoritmo foi integrado com sucesso a um *software* SIG disponível no mercado por um programa de interface simples, o **FazRota**. Um dos maiores esforços dispendidos no desenvolvimento do sistema de roteamento foi a aquisição de dados geográficos para o *software* SIG, especificamente do mapa de Copacabana em nosso caso. A

disponibilidade no âmbito comercial deste tipo de dados implica em custos de desenvolvimento mais baixos e utilização mais ampla de sistemas de roteamento com base em SIG's.

O potencial da utilização comercial da integração de algoritmos de roteamento a SIG's é bastante grande. Além do algoritmo SA aqui descrito, outros algoritmos de roteamento podem ser facilmente incorporados ao *software*, tanto para o PRC-RD como para outros problemas de roteamento. A única limitação atual, conforme já mencionado, está na aquisição dos dados geográficos necessários.

O uso de SIG's viabiliza não só uma melhor apresentação gráfica das soluções, como também permite, através do uso de uma base de dados única (geográfica), a integração de outros modelos de apoio à gerência logística tais como: sistemas de monitoramento de veículos, modelos de localização de depósitos de distribuição, análise da demanda, entre outros.

Referências Bibliográficas:

BARROS NETO, J.F.: *O uso de "simulated annealing" na solução de problemas de roteamento de veículos*. Tese de Mestrado, COPPE/UFRJ, 1995.

BEASLEY, J.E.: "Route first - cluster second methods for vehicle routing". *Omega*, Vol. 11, p. 403-408, 1983.

- BODIN, L.D., GOLDEN, B.L., ASSAD, A. & BALL, M.:** "Routing and scheduling of vehicles and crews: the state of the art". *Computers and Operations Research*, Vol. 10, p.69-211, 1983.
- CHRISTOFIDES, N. & EILON, S.:** "An algorithm for the vehicle dispatching problem". *Operational Research Quarterly*, Vol. 20, p. 309-318, 1969.
- CHRISTOFIDES, N., MINGOZZI, A. & TOTH, P.:** "The vehicle routing problem". In: Christofides, N., Mingozi, A., Toth, P. & Sandi, C. (Eds.), *Combinatorial Optimization*, John Wiley & Sons Ltd., p. 315-338, 1979.
- CLARKE, G. & WRIGHT, J.W.:** "Scheduling of vehicles from a central depot to a number of delivery points". *Operations Research*, Vol. 12, p. 568-581, 1964.
- CONOLLY, D.:** "General purpose simulated annealing". *Journal of the Operational Research Society*, Vol. 43, p. 494-505, 1992.
- DAVIS, C.:** "GIS e Roteamento". *Fator GIS*, Vol. 4, n. 16, p. 9, 1996.
- DOWSLAND, K.A.:** "Simulated annealing". In: C.R. Reeves (Ed.), *Modern Heuristic Techniques for Combinatorial Problems*, Blackwell Scientific Publications, Oxford, 1993.
- EGLESE, R.W.:** "Simulated annealing: a tool for operational research". *European Journal of Operational Research*, Vol. 46, p. 271-281, 1990.
- EHLER, G.B.:** "Expertise: A knowledge-based spatial decision support system for industrial site evaluation". *Proceedings of the Fourteenth Annual ESRI User Conference*, p. 191-200, 1994.
- FISHER, M.L. & JAIKUMAR, R.:** "A generalized assignment heuristic for vehicle routing". *Networks*, Vol. 11, p. 109-124, 1981.
- FISHER, M.L.:** "Vehicle Routing". In: *Networks and Distribution*, Elsevier Science Publishers B.V., Amsterdam, 1994.
- GALVÃO, R.D. & GUIMARÃES, J.:** "The control of helicopter operations in the Brazilian oil industry: issues in the design and implementation of a computerized system". *European Journal of Operational Research*, Vol. 49, p. 266-270, 1990.
- GALVÃO, R.D., FERREIRA FILHO, V.J.M. & HENRIQUES, H.B.S.:** "Roteamento baseado em Sistemas de Informação Geográfica". *Anais do XXVI Simpósio Brasileiro de Pesquisa Operacional*, p. 708-713, 1994.
- GASKELL, T.J.:** "Bases for vehicle fleet scheduling". *Operational Research Quarterly*, Vol. 18, p. 281-295, 1967.
- GENDREAU, M., HERTZ, A. & LAPORTE, G.:** "A tabu search heuristic for the vehicle routing problem". *Management Science*, Vol. 40, p. 1276-1290, 1994.
- GILLETT, B.E. & MILLER, L.R.:** "A heuristic algorithm for the vehicle-dispatch problem". *Operations Research*, Vol. 22, p. 341-349, 1974.
- GOLDEN, B., MAGNANTI, T.L. & NGUYEN, H.Q.:** "Implementing vehicle routing algorithms". *Networks*, Vol. 7, p. 113-148, 1977.
- GOLDEN, B.L. & ASSAD, A.A. (Eds.):** *Vehicle Routing: Methods and Studies*, Elsevier Science Publishers B.V., Amsterdam, 1988.
- GOLDEN, B.L. & SKISCIM, C.C.:** "Using simulated annealing to solve routing and location problems". *Naval Research Logistics Quarterly*, Vol. 33, p. 261-279, 1986.
- LAPORTE, G. & NOBERT, Y.:** "Exact algorithms for the vehicle routing problem". In: S. Martello, G. Laporte, M. Minoux & C. Ribeiro (Eds.), *Surveys in Combinatorial Optimization*, Elsevier Science Publishers B.V., Amsterdam, p. 147-184, 1987.
- LAPORTE, G.:** "The vehicle routing problem: an overview of exact and approximate methods". *European Journal of Operational Research*, Vol. 59, p. 345-358, 1992.
- LIN, S.:** "Computer solutions of the travelling salesman problem". *Bell System Technical Journal*, Vol. 44, p. 2245-2269, 1965.
- LIN, S. & KERNIGHAM, B.W.:** "An effective heuristic algorithm for the travelling salesman problem". *Operations Research*, Vol. 21, p. 498-516, 1973.
- MOLE, R.H. & JAMESON, S.:** "A sequential route-building algorithm employing a generalized savings criterion". *Operational Research Quarterly*, Vol. 27, p. 503-511, 1976.
- OSMAN, I.H.:** "Metastrategy simulated annealing and tabu search algorithm for the vehicle routing problem". *Annals of Operations Research*, Vol. 41, p. 421-451, 1993.
- OSMAN, I.H. & CHRISTOFIDES, N.:** "Capacitated clustering problems by hybrid simulated annealing and tabu search". *International Transactions in Operational Research*, Vol. 1, p. 317-336, 1994.
- PAESSENS, H.:** "The savings algorithm for the vehicle routing problem". *European Journal of Operational Research*, Vol. 34, p. 336-344, 1988.
- PAPADIMITRIOU, C.H. & STEIGLITZ, K.:** *Combinatorial Optimization: Algorithms and Complexity*, Prentice Hall Inc., Englewood Cliffs, N.J., 1982.

RUSSEL, R.A.: "An effective heuristic for the m-tour traveling salesman problem with some side constraints". *Operations Research*, Vol. 25, p.517-524, 1976.

TOMLINSON, R.F., CALKINS, H.W. & MARBLE, D.F.: "Current and potential uses of geographic information systems: the North American experience". *International Journal of Geographical Information Systems*, Vol. 1, p. 203-218, 1987.

VAN LAARHOVEN, P.J.M. & AARTS, E.H.L.: *Simulated Annealing: Theory and Applications*, Dordrecht: D. Reidel, 1987.

YANG, X. & SATHISAN, S.K.: "Development of a Gis-based routing model". *Proceedings of the Fourteenth Annual ESRI User Conference*, p 1076-1091, 1994.

YELLOW, P.: "A computational modification to the savings method of vehicle scheduling". *Operational Research Quarterly*, Vol. 21, p 281-283, 1970.

Apêndice: Comparação do Desempenho de Diversos Algoritmos para o PRC-RD

O algoritmo SA, desenvolvido no presente artigo, foi testado em problemas disponíveis na literatura, mostrados na Tabela 2 abaixo. O algoritmo foi testado tanto para valores inteiros como para valores reais dos custos c_{ij} . Os valores inteiros foram obtidos pelo arredondamento das distâncias euclidianas calculadas a partir das coordenadas (x,y) de depósito e clientes.

A Tabela 3 mostra as soluções obtidas pelo algoritmo SA para PRC-RD (tanto para

valores inteiros como para valores reais de c_{ij}), em comparação com resultados disponíveis na literatura para várias outras heurísticas existentes sobre o mesmo problema, para os problemas-teste da Tabela 1. O algoritmo SA foi executado em um microcomputador 486DX/2 de 66Mhz com 8Mb de memória RAM. Na Tabela 2 cada solução é mostrada na forma **a/b**, na qual **a** é o comprimento total das rotas e **b** é o número de veículos utilizados.

Tabela 2 - Problemas-teste Obtidos da Literatura

Problema	Origem	Nº de Clientes	Local. Depósito *	Capac. Veículos	Tamanho Máx. da Rota	Tempo Serviço Cliente **
G1	Gaskell (1967)	21	(145,215)	6000	200	10
G2	Gaskell (1967)	22	(266,235)	4500	240	10
G3	Gaskell (1967)	29	(162,354)	4500	240	10
G4	Gaskell (1967)	32	(292,425)	8000	240	10
C1	Christofides & Eilon (1969)	50	(30,40)	160	-	-
C2	Christofides & Eilon (1969)	50	(30,40)	160	200	10
C3	Christofides & Eilon (1969)	75	(40,40)	140	-	-
C4	Christofides & Eilon (1969)	75	(40,40)	140	160	10
C5	Christofides & Eilon (1969)	100	(35,35)	200	-	-
C6	Christofides & Eilon (1969)	100	(35,35)	200	230	10

C7	Christofides <i>et al.</i> (1979)	150	(35,35)	200	-	-
C8	Christofides <i>et al.</i> (1979)	150	(35,35)	200	200	10
C9	Christofides <i>et al.</i> (1979)	199	(35,35)	200	-	-
C10	Christofides <i>et al.</i> (1979)	199	(35,35)	200	200	10

* A localização é dada pelas coordenadas (x,y) do depósito.

** O tempo de serviço do cliente é dado em unidades equivalentes de distância.

Tabela 3 - Resultados Computacionais do Problema PRC-RD

Algoritmo	Problema													
	G1	G2	G3	G4	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10
Heurístico														
Simul. Annealing-Inteiro (este artigo)	585/4	949/5	873/4	809/4	521/5	553/6	837/10	903/11	825/8	862/9	1049/12	1159/14	1318/17	1413/18
Simul. Annealing-Real (este artigo)	585,28/ 4	948,68/ 5	875,35/ 4	810,13/ 4	529,22/ 5	558,93/ 6	849,83/ 10	913,94/ 11	837,94/ 8	884,81/ 9	1082,07/ 12	1196,59/ 14	1376,70/ 17	1458,75/ 18
Clarke & Wright (1964) (savings, sequential)	648/4	949/5	1017/5	850/5	585/6	619/-	900/11	976/-	887/8	973/-	1204/12	1426/16	1540/17	1800/21
Clarke & Wright (1964) (savings, multiple)	598/4	955/5	963/5	839/5	579/6	-	901/11	-	864/8	-	1083/12	1289/15	1386/17	1505/19
Gaskell (1967) (savings, λ)	602/4	977/5	971/5	821/5	-	-	-	-	-	-	-	-	-	-
Gaskell (1967) (savings, π)	598/4	990/6	939/5	850/5	-	-	-	-	-	-	-	-	-	-
Christofides & Eilon (1969) (3-optimal)	585/4	949/5	875/4	810/4	556/5	-	876/10	-	863/8	-	-	-	-	-
Gillet & Miller (1974)	-	-	-	-	532/5	560/-	874/10	933/-	851/8	888/-	1079/12	1230/15	1389/17	1518/19
Mole & Jameson (1976)	-	-	-	-	575/5	-	910/10	-	882/8	-	1259/12	1289/15	1545/17	1770/20
Russel (1976)	585/4	956/5	875/4	810/4	525/5	-	854/10	-	833/8	-	-	-	-	-
Fisher & Jaikumar (1981)	-	-	-	-	524/5	560/-	857/10	916/-	833/8	885/-	1014/12	1230/15	1420/17	1518/19
Paessens (1988)	585/-	956/-	937/-	812/-	557/-	-	861/-	-	858/-	-	1083/-	1221/-	1359/-	1476/-
Gendreau, Hertz & Laporte (1994)	-	-	-	-	524,61/-	555,43/-	835,32/-	909,68/-	826,14/-	865,94/-	1029,64/-	1162,89/-	1300,89/-	1403,21/-

VEHICLE ROUTING BASED ON GEOGRAPHICAL INFORMATION SYSTEMS

Abstract

An algorithm for vehicle routing is embedded into a Geographical Information System (GIS), from the database of which it extracts the information needed for the routing and where it displays the resulting routes. The routing algorithm is a simulated annealing metaheuristic that produces good quality routes in reduced computational times. We describe the embedding of the algorithm into a specific GIS software and the application of the routing system to a simulated delivery schedule in the neighbourhood of Copacabana, in Rio de Janeiro.

Key words: vehicle routing, geographical information systems, heuristic methods.