

Faculdade de Engenharia da Universidade do Porto



Optimização de Rotas

Mariana Soares Maciel Machado Magalhães

VERSÃO PROVISÓRIA

Preparação para a Dissertação realizada no âmbito do
Mestrado Integrado em Engenharia Electrotécnica e de Computadores
Major Automação

Orientador: Prof. Dr. José António Soeiro

15 de Fevereiro de 2011

Índice

Lista de figuras	v
Lista de tabelas	vi
Abreviaturas e Símbolos	vii
Capítulo 1	8
Introdução.....	8
1.1 - Motivação.....	8
1.2 - Descrição do Problema	9
1.3 - Plano de Trabalho.....	9
Capítulo 2	10
Optimização de rotas	10
2.1 Recolha de Resíduos Sólidos Urbanos.....	10
2.2 Abordagem ao Problema	12
2.3 Definições e Conceitos Importantes	14
2.4 Heurísticas.....	16
2.5 Trabalhos Relacionados	28
Capítulo 3	29
Conclusões	29
Capítulo 4	30
Referências Bibliográficas.....	30

Lista de figuras

Figura 1 - Contentores utilizados para resíduos diferenciados	11
Figura 2 - Camião responsável pela recolha de resíduos diferenciados	11
Figura 3 - Contentores para recolha de resíduos indiferenciados	11
Figura 4 - Camião responsável pela recolha de resíduos indiferenciados.....	12
Figura 5 - Diferença entre Recolha ponto-a-ponto e ARP	13
Figura 6 - Exemplo de um Grafo [17].....	14
Figura 7 - Exemplo de um grafo orientado [17]	14
Figura 8 - Grafo com valores associados aos vértices (Rede) [17]	14
Figura 9 - Vizinhança de um ponto [18].....	15
Figura 10 - Máximos e Mínimos Locais e Globais	15
Figura 11 - Algoritmo da heurística do Vizinho mais Próximo.....	18
Figura 12 - Algoritmo da Heurística de Inserção mais Próxima	19
Figura 13 - Diagrama contendo detalhes acerca da heurística de melhoramento k-opt	20
Figura 14 - Diagrama explicativo do algoritmo utilizado para a heurística Pesquisa Local	21
Figura 15 - Algoritmo explicativo da metaheurísticas Arrefecimento Simulado (Simulated Annealing)	23
Figura 16 - Algoritmo utilizado para a Pesquisa Tabu	25
Figura 17 - Diagrama utilizado pelos algoritmos evolucionários.....	26
Figura 18 - Exemplo da fase de crossover	27
Figura 19 - Exemplo da fase de mutação	27

Lista de tabelas

Tabela 1 - Analogia entre Problemas de Optimização e Sistemas Físicos para a utilização da técnica de Arrefecimento Simulado (Simulated Annealing)	23
--	----

Abreviaturas e Símbolos

RSU - Resíduos Sólidos Urbanos

PERSU - Plano Estratégico para os Resíduos Sólidos Urbanos

ARP - Arc Routing Problems

TSP - Traveling Salesman Problem

VRP - Vehicle Routing Problem

SVRP - Stochastic Vehicle Routing Problem

VRPTW - Vehicle Routing Problem with Time Windows

ARP - Arc Routing Problems

GVNT - Guided Variable Neighborhood Thresholding

Capítulo 1

Introdução

1.1 - Motivação

O crescimento da população tem levado a um aumento da produção de resíduos sólidos urbanos, levando também a uma maior degradação das condições ambientais.

Assim, cada vez mais, a recolha de resíduos urbanos é de extrema importância, visto que caso esta não se realize poderá levar a grandes consequências para a saúde pública.

Antigamente, os resíduos urbanos eram recolhidos indiferenciadamente, no entanto em 1997 foi aprovado o Plano Estratégico para os Resíduos Sólidos Urbanos (PERSU) pelo governo. Este pretende encerrar todas as lixeiras do país, a criação de diversos sistemas multimunicipais e intermunicipais para a gestão de resíduos sólidos urbanos, a construção de numerosas infra-estruturas de valorização e eliminação, o lançamento de sistemas de recolha selectiva multimaterial e ainda o licenciamento de entidades gestoras de fluxos especiais de resíduos.

Tendo como objectivo a definição de estratégias, de prioridades e metas entre 2007 e 2016 em termos de resíduos sólidos urbanos, foi aprovado por Portaria de 28 de Dezembro de 2006 o PERSU II.

Como tal, e dada a importância da gestão dos resíduos sólidos urbanos, o processo de recolha destes torna-se essencial para o correcto funcionamento de todo o processo desde a recolha até à reciclagem deste tipo de resíduos. Dependendo dos locais, a recolha destes resíduos poderá ser feita de forma diferenciada ou não.

Na área da recolha dos resíduos de determinada cidade, existirão diversas empresas responsáveis pela realização desta tarefa. Assim, e como qualquer empresa que se encontre no mercado, o objectivo destas passa pela redução de custos e recursos. Um dos campos onde esta minimização poderá ser feita é ao nível das rotas realizadas pelos diversos camiões de recolha de resíduos sólidos.

O trabalho desta dissertação incidirá sobre a optimização destas rotas, tendo em conta diversas restrições, com o objectivo da minimização de custos, recursos e também de distâncias percorridas por cada camião que se encontre de serviço.

O tema da dissertação é bastante importante para as diversas empresas responsáveis pela recolha dos Resíduos Urbanos, pois muitas delas não possuem ainda soluções computacionais capazes de lidar com este tipo de problemas. Trata-se também dum tema bastante abrangente pois muitos dos fundamentos científicos encontram também aplicações em outras áreas. Por outro lado, os problemas envolvidos têm bastante complexidade, quer ao nível das dificuldades e problemas de decisão ao nível da modelação de situações reais mas também pelos problemas computacionais associados à dimensão do espaço discreto de soluções.

Boas soluções permitirão às empresas obter melhores desempenhos, uma redução dos custos, e também a percepção, pelos seus funcionários, da existência de melhores alternativas às que se encontram implementadas no momento.

1.2 - Descrição do Problema

O problema proposto consiste na optimização de rotas na recolha de resíduos sólidos urbanos, através da utilização de métodos seleccionados para esse efeito. Para a obtenção da solução final serão analisados diversos métodos possíveis de serem utilizados, para que seja escolhido o mais eficaz e eficiente de forma a serem obtidos resultados satisfatórios. Após a análise de todos os métodos de resolução associados a este, será definido o algoritmo que permitirá a resolução do problema. No ponto seguinte será apresentado o plano de trabalhos mais detalhadamente.

1.3 - Plano de Trabalho

1. Envolvimento com projectos em curso.
2. Análise e estudo do essencial do estado-da-arte em optimização de rotas em grafos/redes.
3. Consideração e modelação de problema (s) específico (s) de interesse prático no âmbito de serviços ambientais de recolha.
4. Complemento do estudo de métodos de optimização combinatória, nomeadamente de Metaheurísticas.
5. Selecção de métodos potenciais de resolução e desenvolvimento algorítmico.
6. Teste e Avaliação de resultados.
7. Realização do Relatório do projecto.

Capítulo 2

Optimização de rotas

Este capítulo fornecerá informação acerca de como é realizada a recolha de resíduos sólidos urbanos. É de extrema importância perceber como esta tarefa funciona para que o objectivo principal seja atingido, isto é, para que se possa perceber nitidamente a forma como as diferentes rotas percorridas por cada camião responsável pela recolha deste tipo de resíduos possa ser optimizada de forma eficiente.

Assim, neste capítulo serão também encontradas várias referências a trabalhos já realizados por outros autores nesta área, sendo estes de extrema importância para a realização desta dissertação.

Poderão também ser encontradas diversas definições imprescindíveis para a compreensão plena deste tema.

2.1 Recolha de Resíduos Sólidos Urbanos

- **Resíduos Urbanos Sólidos: Definição**

Resíduos sólidos urbanos são definidos como quaisquer resíduos sólidos provenientes das habitações resultantes da actividade doméstica da população de um determinado local.

Estima-se que, todos os dias, em Portugal cada habitante produza 1,3 kg deste tipo de resíduos. Dada a elevada quantidade de resíduos produzidos, é essencial que estes sejam recolhidos de forma frequente, para evitar que se tornem perigosos para a saúde pública.

- **Tipos de Recolha de Resíduos Sólidos Urbanos:**

Esta recolha poderá ser realizada de duas diferentes formas:

Recolha diferenciada - Consiste na recolha, por parte de camiões destinados a esse efeito, de resíduos diferenciados, com vista ao seu reencaminhamento para reciclagem.



Figura 1 - Contentores utilizados para resíduos diferenciados

Tal como pode ser visto na Figura 1, os resíduos são divididos em 3 tipos fundamentais: papel (contentor azul), embalagens (contentor amarelo) e vidro (contentor verde). Assim, visto que estes resíduos terão de ser recolhidos separadamente, os camiões utilizados para esta recolha terão de ser especializados para esse efeito. Na figura seguinte poderá ver-se um camião que se encontra destinado a este tipo de recolha.



Figura 2 - Camião responsável pela recolha de resíduos diferenciados

Alem deste tipo de recolha incidir sobre os ecopontos, em alguns locais também se encontra implementada na recolha porta-a-porta.

Recolha indiferenciada - Consiste na recolha, de todo os resíduos produzidos pela população por um determinado camião. Este tipo de recolha não terá como objectivo o envio deste tipo de resíduos para a reciclagem. Assim, a recolha é feita de forma indiferenciada, ou seja sem a divisão dos resíduos nos 3 grupos mencionados anteriormente.



Figura 3 - Contentores para recolha de resíduos indiferenciados

Na Figura 3 poderão ser vistos dois tipos de contentores utilizados pela população para o depósito de resíduos sólidos, com diferentes tamanhos e capacidades. Como estes não são diferenciados, os camiões utilizados por esta recolha serão camiões de uso normal na recolha de resíduos. Este tipo de camiões pode ser visto na Figura 4.



Figura 4 - Camião responsável pela recolha de resíduos indiferenciados

- **Recolha de RSU's : Funcionamento**

A recolha deste tipo de resíduos é realizada, tal como já foi referido anteriormente, por camiões destinados a esse efeito.

Cada camião terá de percorrer determinada rota, previamente definida. Assim, todos os pontos que necessitam de recolha serão percorridos por esses mesmos camiões. Estas rotas poderão ser determinadas de diferentes formas, sendo este o tema principal desta dissertação.

Outro aspecto importante a reter é a limitação da quantidade de RSU's que determinado camião poderá transportar. Esta quantidade irá variar de camião para camião, assim sendo cada empresa de recolha será responsável pelos camiões que se encontram disponíveis para a realização de determinado serviço.

Quando determinado camião se encontrar cheio, terá de ser esvaziado num aterro ou em determinado local destinado a esse efeito para, posteriormente ser transportado para o aterro.

De seguida, irá ser apresentada a melhor forma de ser abordado o problema da optimização das rotas de recolha.

2.2 Abordagem ao Problema

Dada a complexidade de um problema deste género, devido a todos os pontos que necessitam de ser percorridos numa determinada rota, a melhor forma de serem analisados é através da divisão do problema em problemas mais pequenos e menos complexos. Assim, tendo uma cidade ou localidade, esta vai ser dividida em pequenos sectores, permitindo assim que o número de pontos de recolha seja reduzido, bem como a complexidade do problema em análise.

Estes problemas poderão ser classificados e resolvidos da mesma forma que outros problemas deste tipo, designados por VRP (Vehicle Routing Problem)

Segundo Nuortio[1] existem diversas variantes dos problemas VRP. Sendo estas dependentes das restrições que se deverá ou não ter em conta. O tipo de problema básico, designado VRP, consiste no escalonamento dos veículos para que todos os consumidores sejam

servidos, tendo como objectivo a minimização da distância percorrida por estes. Um determinado veículo começa e acaba a sua viagem num aterro e será responsável pela recolha apenas numa rota pré-definida.

Ainda segundo os mesmos autores poderão ser encontrados mais problemas tendo como base o problema básico de VRP, sendo os seguintes exemplos disso:

- SVRP (Stochastic Vehicle Routing Problem): Algumas das variáveis do problema serão estocásticas, podendo ser os tempos de viagem incertos, encomendas desconhecidas.
- PVRP (Periodic Vehicle Routing Problem): Neste tipo de problemas, para além de ter como base o problema básico de VRP, é também necessário associar os diversos clientes que necessitam de ser servidos a cada dia.
- VRPTW (Vehicle Routing Problem with Time Windows): Neste tipo de problemas, além dos objectivos iniciais de um problema básico de VRP, é necessário ter em conta as janelas temporais nas quais, determinado cliente ou aterro poderá ou não ser visitado. Estas janelas poderão ser violadas, associando a estas violações um determinado custo de penalização.
- VRP com um número limitado de veículos: Nestes problemas terá de ser tido em conta o número de veículos que determinada empresa de recolha poderá disponibilizar, tendo como objectivo o uso destes veículos com o máximo nível de eficácia possível.

Os problemas de VRP, podem ser abordados de duas diferentes formas. Podem ser descritos como problemas de recolha ponto-a-ponto, ou então como Arc Routing Problems (ARP).

Na seguinte figura (Figura 5) encontra-se a forma como estes dois tipos de problemas são abordados.

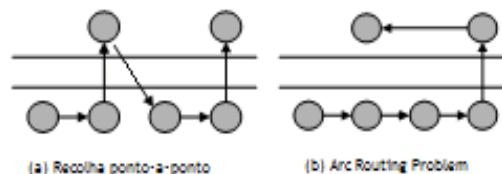


Figura 5 - Diferença entre Recolha ponto-a-ponto e ARP

Nos problemas de recolha ponto-a-ponto o veículo irá ter de visitar os nós, não dando especial importância aos arcos.

Nos ARP's, a importância recai sobre os arcos, não se tendo em atenção os nós. Assim sendo o veículo, em vez de ter de visitar todos os nós de uma determinada rota, terá apenas de percorrer os arcos assinalados.

2.3 Definições e Conceitos Importantes

2.3.1. Grafos e Redes:

Um grafo é constituído por dois elementos: vértices (nós) e arcos (lados ou arestas), sendo que cada aresta irá ligar um par de vértices.

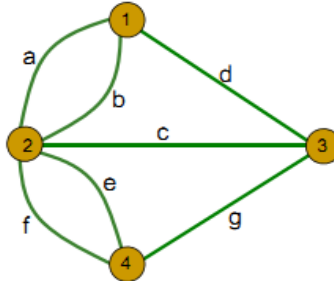


Figura 6 - Exemplo de um Grafo [17]

Na Figura 6 encontra-se o exemplo de um grafo. Como pode ser visto os vértices desse grafo são os seguintes $V=\{1,2,3,4\}$, enquanto os arcos são $A=\{a,b,c,d,e,f,g\}$.

O grafo que se encontra na Figura 6 é exemplo de um grafo não orientado. Um grafo orientado é constituído por pares de vértices ordenados. Isto é, caso se fale do arco associado ao par ordenado (u,v) , significa que o arco começa em u e acaba em v . Na Figura 7 encontra-se um exemplo de um grafo orientado.

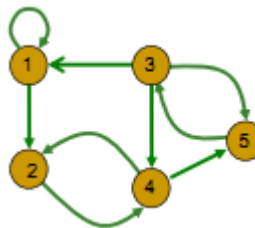


Figura 7 - Exemplo de um grafo orientado [17]

Uma rede é um grafo onde, a cada vértice se encontra associado um determinado numero. Este poderá dizer respeito a diversas características, tais como distancias, custos, capacidades, etc.

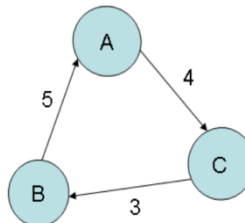


Figura 8 - Grafo com valores associados aos vértices (Rede) [17]

2.3.2. Vizinhaça:

Fazer uma pesquisa numa determinada vizinhaça de um determinado ponto, significa pesquisar numa região próxima desse mesmo ponto. Na próxima figura poderá ver-se graficamente o que significa a vizinhaça de um ponto:

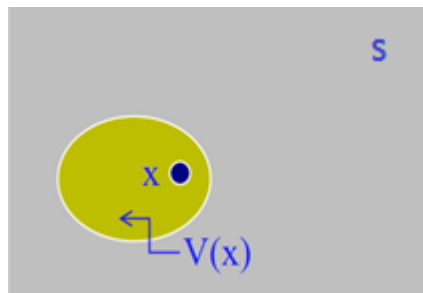


Figura 9 - Vizinhança de um ponto [18]

Na Figura 9, pode ser visto o ponto x e a sua vizinhança, designada por $V(x)$.

2.3.3. Mínimos/Máximos Locais e Mínimos/Máximos Globais:

Nesta secção irão ser explicados os conceitos de mínimos e máximos locais e de mínimos e máximos globais. Estes conceitos são bastante relevantes para o tema em estudo, pois quando se pretende aplicar determinado método a solução obtida poderá ser um mínimo ou um máximo local, ou então um global.

- Mínimo Local: Ponto na vizinhança de determinado ponto que adquire o valor mínimo da função.
- Máximo Local: Ponto na vizinhança de determinado ponto que adquire o valor máximo da função.
- Mínimo global: Ponto mínimo da função em análise.
- Máximo global: Ponto máximo da função em análise.

Na próxima figura poderão ser vistos vários exemplos de máximos e mínimos locais e globais.

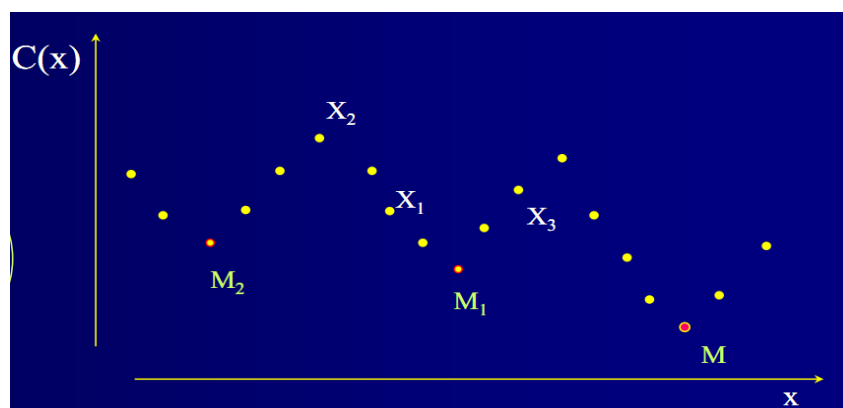


Figura 10 - Máximos e Mínimos Locais e Globais

Pode ver-se na Figura 10 a existência de máximos locais, de mínimos locais (M_2 e M_1), de máximos globais (X_2) e de mínimos globais (M).

2.3.4. TSP (Traveling Salesman Problem)

O problema do caixeiro-viajante (Traveling Salesman Problem - TSP) é um problema bastante conhecido e estudado na área da investigação operacional.

Este problema é constituído por um determinado número de cidades, onde o objectivo passa por minimizar a distância percorrida para que o caixeiro-viajante visite todas as cidades apenas uma vez e que no final volte à cidade inicial.

Apesar de parecer um problema simples, é extremamente complicado encontrar a sua solução óptima, por isso é que este problema é bastante estudado na investigação operacional, sendo que para chegar à sua solução final poderá recorrer-se a diversas técnicas. No ponto seguinte desta dissertação são analisadas diversos métodos possíveis para este e outros problemas.

2.4 Heurísticas

Para a resolução destes problemas existem diversas heurísticas. Neste ponto desta dissertação, serão abordadas diversas heurísticas, que poderão ser utilizadas para a resolução e optimização de problemas relativos a grafos ou redes. No entanto, para a escolha da heurística a utilizar, terá de ser analisado o problema com algum detalhe, para que a escolha recaia na mais indicada para o problema em questão.

Qualquer um dos métodos apresentados de seguida irá possuir vantagens e desvantagens. Determinados métodos, irão ser capazes de retornar a solução óptima do problema, enquanto outros apenas irão retornar mínimos/máximos locais, podendo assim não ser a solução global óptima do problema.

Apesar de algumas destas poderem retornar soluções realmente óptimas para o problema em análise, nem sempre poderão ser utilizadas, visto que esta possibilidade ou não, irá depender do grau de complexidade do problema.

Assim, neste subcapítulo poderão ser encontradas algumas heurísticas desta natureza, para que posteriormente se possa entender mais facilmente a abordagem que irá ser feita ao problema de optimização de rotas na recolha de resíduos urbanos.

2.3.5. Métodos Exactos:

O método exacto que será abordado neste ponto é designado por Branch and Bound. É um método exacto pois a solução que é obtida após a sua aplicação irá ser a solução óptima do problema que se encontra a ser analisado. Estes métodos, apesar de fornecerem a solução óptima, não poderão ser utilizados para a resolução de problemas com um elevado grau de complexidade, pois chegar a uma solução será de extrema dificuldade.

Branch and Bound (Ramificação e Limitação):

Na resolução de problemas de programação inteira irão surgir diversas soluções para o problema. No entanto, a solução que se pretende obter trata-se da solução ótima. Como o número de soluções possíveis pode ser elevado, exigiria bastante esforço computacional para encontrar a solução ótima entre todas as possíveis. Assim, é importante recorrer a um método que seja capaz de examinar apenas um pequeno conjunto de todas as soluções possíveis para o problema. O método Branch-and-Bound é um dos métodos que permite esta abordagem. [19]

Este método surgiu pela primeira vez em 1960, proposto por Land e Doig na resolução de um problema de Programação Inteira, sendo que o nome foi utilizado pela primeira vez na resolução de um problema de Caixeiro-viajante (TSP - Traveling Salesman Problem), sendo posteriormente aplicado em diversos outros tipos de problemas. [19]

O conceito deste método prende-se na separação do conjunto de soluções possíveis ao problema, em subconjuntos sucessivamente mais pequenos, de forma a obter um conjunto tão pequeno que torne fácil o reconhecimento da solução ótima. Logo, e dadas estas características, este método não será capaz de ser aplicado a um problema demasiado complexo, que possua um elevado número de soluções.

2.3.6. Heurísticas Construtivas:

O nome, Heurísticas Construtivas, deve-se à forma como a solução final é encontrada. Esta solução é construída através da adição de pontos à solução. Isto é, cada ponto é adicionado individualmente em cada iteração tendo em conta determinado critério, até que a solução seja encontrada, isto é, até que todos os pontos se encontrem presentes na solução final. Os critérios utilizados para a adição dos pontos irão depender do tipo de heurística que se encontra a ser utilizada. De seguida, irão ser apresentadas três destas heurísticas, que possuem diferentes critérios.

Vizinho Mais Próximo:

A solução é construída através da adição de todos os pontos individualmente à solução final. Esta solução é encontrada quando todos os pontos se encontram nela. O critério utilizado para a sua adição é a distância entre dois pontos, sendo que o ponto escolhido para ser adicionado será o mais próximo do selecionado anteriormente. Na seguinte figura poderá ser visto o diagrama que explica o algoritmo utilizado para esta heurística.

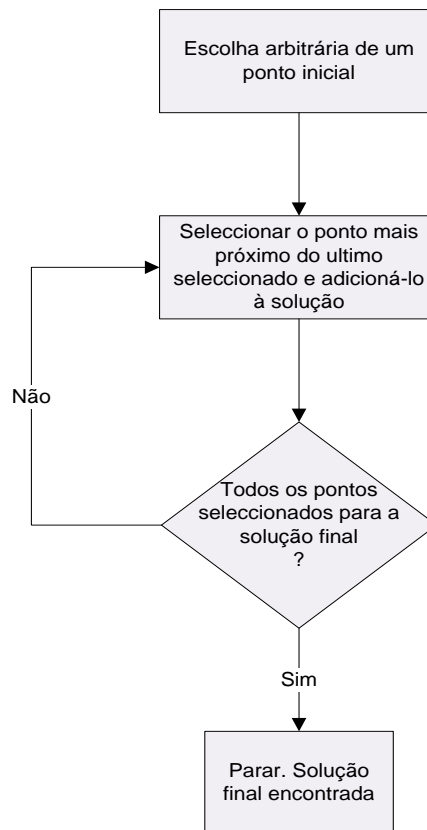


Figura 11 - Algoritmo da heurística do Vizinho mais Próximo

Heurística de inserção Mais Próxima:

Nesta heurística poderão ser encontradas 3 principais fases [19]:

- Inicialização: Escolha do primeiro ponto que irá fazer parte da solução, e também do seguinte com menor custo/distancia, construindo assim uma solução parcial
- Seleccção: Seleccção de qual o próximo ponto a ser inserido na solução parcial, tendo em conta a minimização do custo/distancia.
- Inserção: Inserção desse mesmo ponto na solução parcial, entre os dois pontos escolhidos inicialmente.

Na figura seguinte poderá ser visto o diagrama que compreende o algoritmo utilizado para construção de uma solução através da heurística de inserção mais próxima.

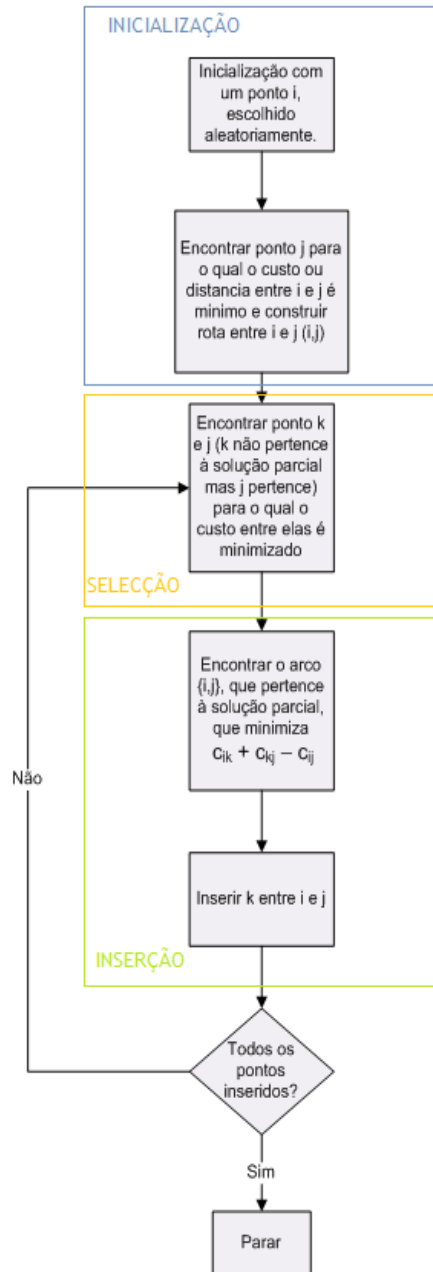


Figura 12 - Algoritmo da Heurística de Inserção mais Próxima

Heurística de inserção Mais Distante:

Segundo [19] para a construção de uma solução através da heurística de inserção mais distante irá utilizar-se os mesmos passos mencionados para a de inserção mais próxima, diferindo apenas na fase da selecção, visto que, no lugar de se pretender que esse custo seja minimizado, pretende-se a sua maximização, daí o nome dado a esta heurística.

2.3.7. Heurísticas de Melhoramento:

Este tipo de heurísticas começa com uma solução existente e possível, e vão-se fazendo melhorias sucessivas a essa solução. A solução inicial poderá ser obtida através de heurísticas construtivas ou então de forma aleatória.

Heurística k-opt:

Esta heurística consiste na melhoria de uma solução encontrada inicialmente através da mudança de k arestas de sítio, ate que seja obtida uma solução melhor que a inicialmente obtida.

Na próxima figura encontra-se o algoritmo descrito detalhadamente:

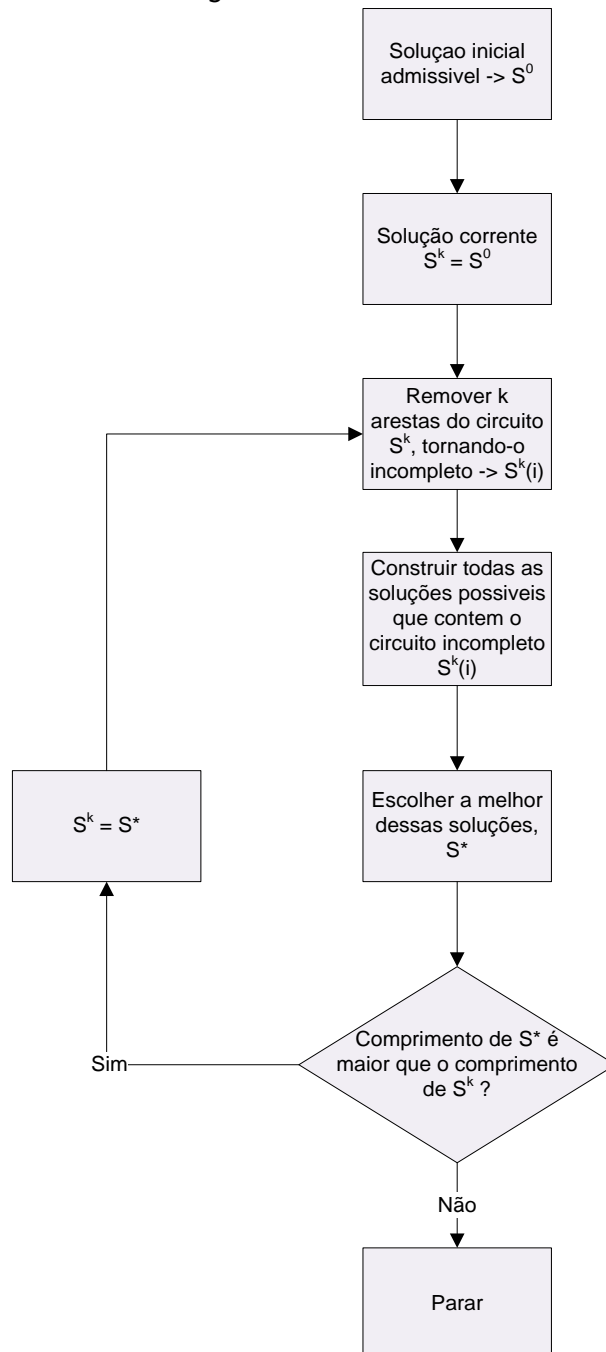


Figura 13 - Diagrama contendo detalhes acerca da heurística de melhoramento k-opt

Pesquisa Local:

Para este método é importante ter em atenção o conceito de vizinhança já mencionado anteriormente. A pesquisa local baseia-se no seguinte princípio: tem início numa primeira solução, e em cada iteração é obtida uma nova solução na vizinhança da solução encontrada anteriormente.

Na seguinte imagem encontra-se um diagrama onde poderá ser analisado o algoritmo utilizado pelo método de Pesquisa Local.

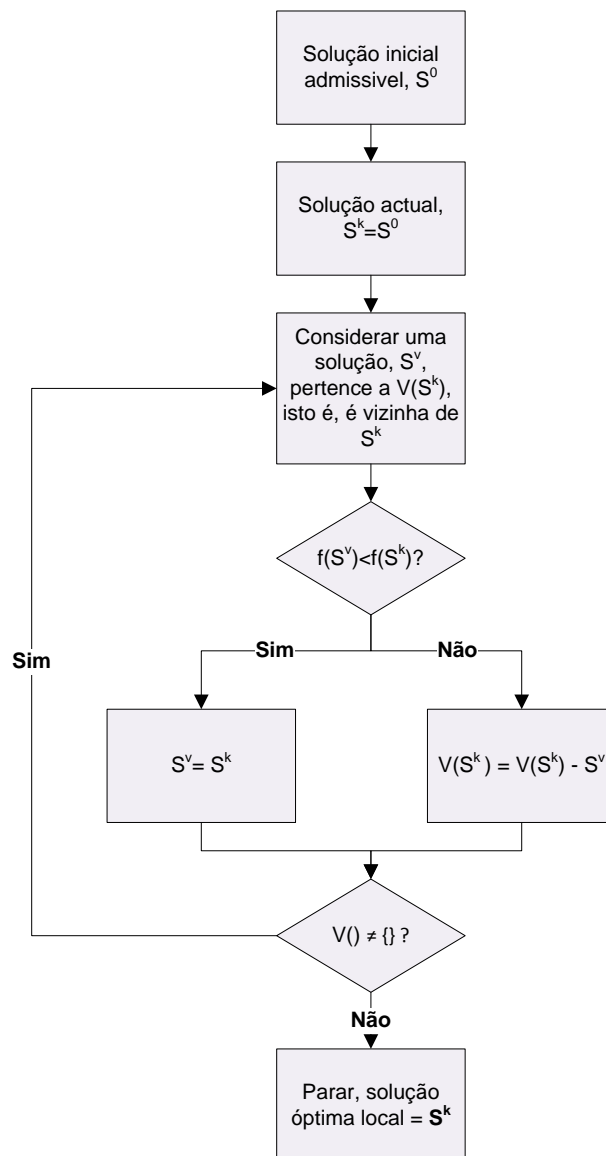


Figura 14 - Diagrama explicativo do algoritmo utilizado para a heurística Pesquisa Local

Esta heurística possui uma desvantagem muito grande que se prende com o facto da solução óptima local obtida através desta, ser bastante diferente da solução óptima global do

problema, pois o desempenho do algoritmo vai depender fortemente da solução inicial e da vizinhança escolhida. Uma forma de se ultrapassar isto é proceder-se a várias repetições desta heurística para um conjunto diversificado de vizinhanças escolhidas.

2.3.8. Heurísticas Compostas:

Estas heurísticas dizem respeito a métodos que, para atingirem determinada solução, precisam de duas fases. Para as heurísticas de melhoramento, inicialmente é necessário que exista uma solução possível. Para atingir esta mesma solução poderá utilizar-se uma qualquer heurística construtiva, sendo que depois, para se proceder à melhoria desta solução é necessário utilizar-se uma heurística de melhoramento, como tal estamos perante uma heurística composta.

2.3.9. Metaheurísticas:

Para as metaheurísticas, os métodos que irão ser explicados de seguida são os seguintes: *Simulated Annealing*, Pesquisa Tabu, Algoritmos Evolucionários e *Ant Colony Algorithms*.

Taillard [2], abordou estas quatro metaheurísticas, explicando detalhadamente cada uma delas e apresentando no final alguns casos de estudo, nos quais, a solução final encontrada é obtida através da utilização dos diferentes métodos que se encontram descritos de seguida.

Simulated Annealing:

Como pode ser visto pelo trabalho apresentado por Taillard [2], este método, designado por *Simulated Annealing* (Arrefecimento Simulado) foi apresentado inicialmente por 3 investigadores da IBM, S.Kirkpatrick, C.D. Gelatt e M.P.Vechi [Kirkpatrick et al.,1983], em 1982, apenas publicado em 1983. Ao mesmo tempo foi também elaborado um trabalho parecido desenvolvido por V.Cerny [Cerny,1985], publicado em 1985.

Ainda segundo o mesmo autor [2], esta técnica irá fornecer uma grande eficácia na resolução de problemas complexos, como por exemplo no design de circuitos eléctricos, no processamento de imagem, na recolha de resíduos urbanos porta a porta, e não só.

Segundo Taillard [2] este método tem por base um método utilizado pelos físicos para a modificação do estado de determinado material, através de um parâmetro ajustável.

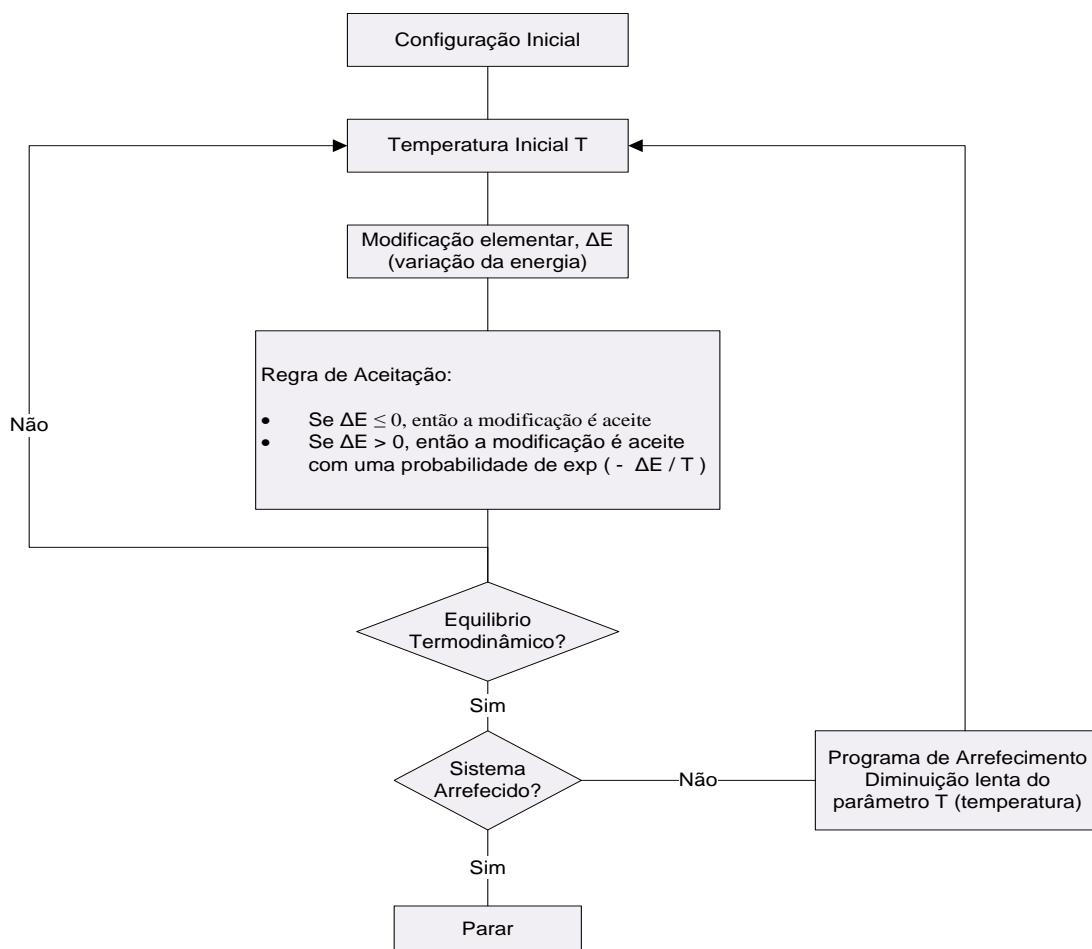


Figura 15 - Algoritmo explicativo da metaheurísticas Arrefecimento Simulado (Simulated Annealing)

Visto que este método tem como base um processo utilizado por físicos poderá fazer-se uma analogia entre estes dois métodos, isto é, entre determinado problema de optimização e um sistema físico.

Problema de Optimização	Sistema Físico
Função Objectivo	Energia livre
Parâmetros do Problema	“Coordenadas” das partículas
Encontrar uma boa configuração, que poderá ser ou não a configuração óptima	Encontrar estados de menor energia

Tabela 1 - Analogia entre Problemas de Optimização e Sistemas Físicos para a utilização da técnica de Arrefecimento Simulado (Simulated Annealing)

Tendo em atenção o diagrama que se encontra na Figura 15 e a analogia presente na Tabela 1, ambos baseados no trabalho realizado pelo autor Taillard [2], um problema de optimização consiste no uso de um determinado parâmetro de controlo, que realiza o papel

da temperatura num sistema físico. Esse parâmetro deverá ter o mesmo efeito que a temperatura num sistema físico:

- Caso a temperatura seja baixada de forma bem controlada deverá condicionar o número de estados acessíveis e levar o sistema até ao estado óptimo.
- Caso a temperatura seja baixada de forma abrupta então levará o sistema até um mínimo local.

Após a breve explicação feita anteriormente sobre este método, é importante perceber quais as vantagens e desvantagens associadas a ele. Uma das desvantagens deste método encontra-se nos ajustes que se tem de fazer, um deles é o ajuste da diminuição da temperatura, pois para a sua realização é necessário que o utilizador tenha um bom conhecimento de como este deverá ser feito evitando assim que este valor seja demasiado elevado. Outra das desvantagens, prende-se ao facto deste método exigir um elevado tempo computacional durante a sua utilização. Como vantagem pode ser apontado o facto de este método ser bastante flexível, podendo acompanhar a evolução do problema. Este método fornece resultados bastante satisfatórios para problemas de grande complexidade.

Pesquisa Tabu

Segundo o autor Taillard [2] o método de pesquisa tabu surgiu em 1986 por F.Glover. A principal característica deste método é a utilização certos mecanismos inspirados pela memória humana, permitindo assim a aprendizagem de lições através do passado.

Segundo o mesmo autor,[2], este método consiste numa solução contendo apenas uma configuração actual (no início, nenhuma solução) que é actualizada a cada iteração realizada. Em cada iteração realizada, a passagem da configuração designada por s para a configuração designada por t , envolve duas fases:

- A primeira constrói o conjunto de vizinhos possíveis de s , $V(s)$.
- A segunda é responsável pela avaliação da função objectivo f do problema para cada configuração pertencente a $V(s)$. A configuração t , que sucede a s , na serie de soluções construídas pelo método tabu, será a configuração de $V(s)$ onde f possui o valor mínimo. No entanto, esta solução é aceite mesmo que seja pior que a solução s , $f(t) > f(s)$.

Devido aos passos a serem seguidos neste método, ele poderá levar a uma configuração que já foi anteriormente aceite numa iteração passada, levando assim à existência de um ciclo infinito. De forma, a permitir que isso aconteça, é criada uma lista tabu, que contem todos os m movimentos ($t \rightarrow s$), que são os opostos dos últimos movimentos realizados de $s \rightarrow t$. O algoritmo utilizado pela pesquisa tabu poderá ser analisado na figura seguinte.

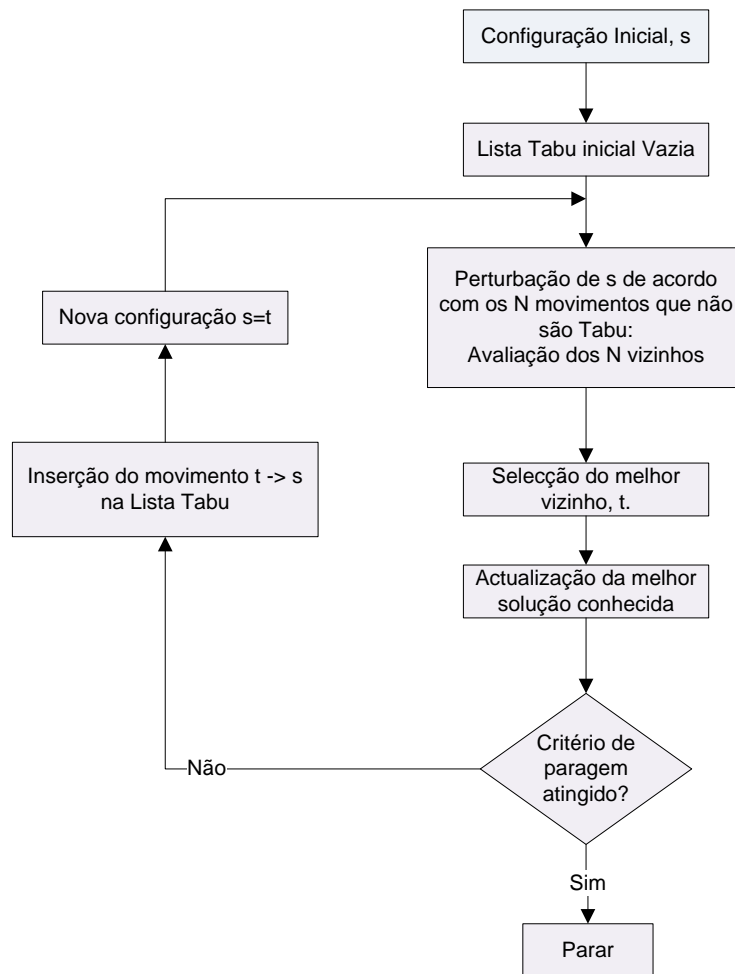


Figura 16 - Algoritmo utilizado para a Pesquisa Tabu

Segundo o mesmo autor, Taillard [2], este método irá fornecer excelentes resultados para certos problemas de optimização, pois são necessários menos parâmetros que requerem ajuste que o método Simulated Annealing, tornando assim a Pesquisa Tabu, num método mais simples a ser utilizado. No entanto, os diversos mecanismos adicionais irão trazer um aumento do nível de complexidade.

Algoritmos Evolucionário: Algoritmo Genético

Segundo Taillard[2], os algoritmos evolucionarios surgiram nos anos 50, mais propriamente em 1957. Estes algoritmos tem como base a evolução biologica das especies. Os algoritmos geneticos são os mais importantes nesta area. Apesar de inicialmente estes algoritmos não serem muito utilizados, nos ultimos tempos verificou-se uma enorme evolução no uso destes devido ao aumento da capacidade dos computadores utilizados.

De acordo com o mesmo autor, a população inicial é formada por N pontos escolhidos aleatoriamente, onde a cada um destes N individuos, cada um deles designados por x, irá estar associado um certo valor que representa a aptidão associado. O valor associado a cada

um dos indivíduos x irá ser mais alto consoante o valor de $z(x)$ seja mais elevado, isto quando se pretende a minimização da função objectivo $z(x)$. O objectivo deste algoritmo passa pelo melhoramento a aptidão dos indivíduos, simulando os dois principais mecanismos que governam a evolução dos seres vivos, propostos pela teoria de C. Darwin:

- Seleccção: suporta a sobrevivencia dos seres com melhor aptidão
- Reprodução: permite a mistura, recombinação e variação das características hereditárias dos pais, de forma a formar descendentes com novas potencialidades.

Na pratica, os indivíduos poderão ser um conjunto de valores inteiros, um vector de numeros reais, uma string de digitos binarios ou capazes da combinação destas representações em problemas mais complexos.

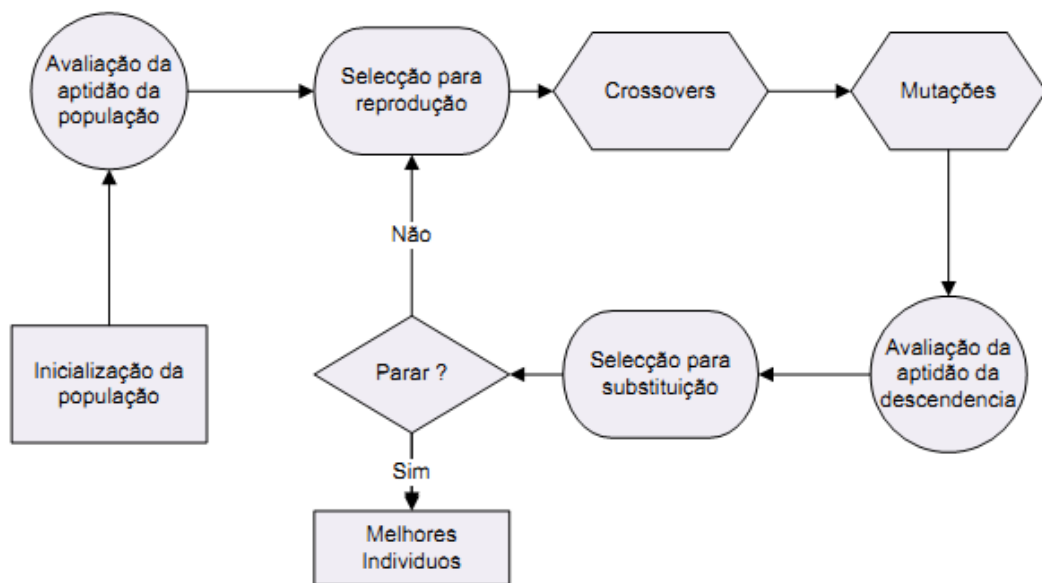


Figura 17 - Diagrama utilizado pelos algoritmos evolucionários

Tal como Taillard[2] descreve e como pode ser visto pela Figura 17, este algoritmo é constituído por quatro fases:

- Fase de seleccção: selecciona quais os indivíduos que irão fazer parte da reprodução
- Fase de reprodução: aplica operadores de variação de forma a serem criados novos indivíduos. Estes operadores utilizados para a criação de novos indivíduos designam-se por Crossover (Figura 18) e Mutação (Figura 19)
- Fase de avaliação da aptidão dos indivíduos -
- Fase de substituição - Seleccionar os novos indivíduos com maior aptidão, substituindo os que possuem pior aptidão.

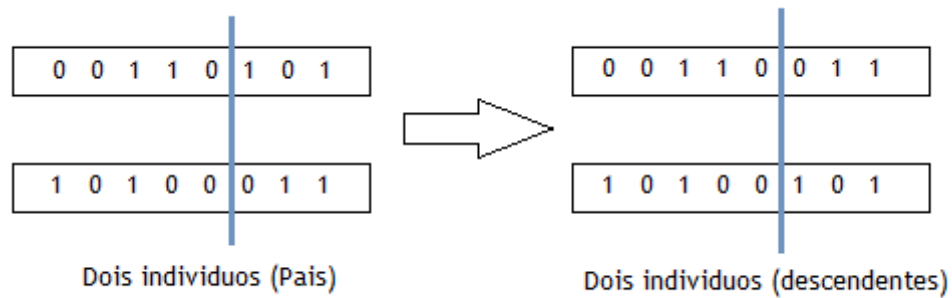


Figura 18 - Exemplo da fase de crossover

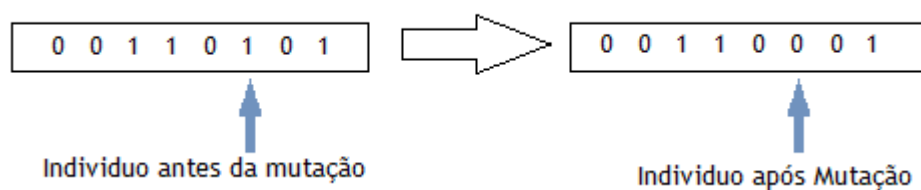


Figura 19 - Exemplo da fase de mutação

Ant Colony Algorithms

Segundo Taillard[2] estes algoritmos foram propostos por Colorni, Dorigo e Manniezzo em 1992. Estes são capazes de simular a capacidade colectiva de resolução de alguns problemas, através do funcionamento das colónias de formigas, nos quais os membros possuem algumas dificuldades ao nível da sua capacidade.

Os entomologistas, através do estudo do comportamento destes insectos, chegaram à conclusão que estes seguem sempre o mesmo caminho sendo também mais curto possível, quando se encontram à procura de alimento. Isto deve-se ao facto de quando estas irão depositar uma substancia designada feromona. A existência desta substancia ao longo de determinado caminho irá ser tanto maior quanto o número de formigas que o percorrem, fazendo assim com que os melhores caminhos sejam reforçados, deteriorando os piores.

Estes algoritmos irão possuir uma grande desvantagem, que é a sua flexibilidade. Na realidade, caso seja adicionado ao problema um novo caminho mais curto, este poderá não ser o caminho escolhido, pois o que foi escolhido anteriormente irá possuir uma maior quantidade de feromona, levando as formigas a escolher esse caminho em deterioração do novo caminho mais curto.

Segundo o mesmo autor, o primeiro algoritmo deste tipo foi aplicado ao Problema do Caixeiro-viajante (Travelling Salesman Problem), no entanto os resultados obtidos por este não foram de todo animadores, devido a isso foram feitas novas pesquisas neste campo, surgindo novos algoritmos que produziram resultados bem mais satisfatórios.

2.5 Trabalhos Relacionados

Visto que esta dissertação irá incidir sobre a optimização de rotas de recolha de resíduos urbanos torna-se importante abordar uma secção acerca de trabalhos já realizados nesta área. Assim, nesta secção irão ser expostos alguns trabalhos já realizados e estudados por diversos autores.

O trabalho realizado por Sahoo et al [15] diz respeito à aplicação de métodos a recolha de resíduos nos Estados Unidos da América, através de uma aplicação informática, designada por WasteRoute. Poderão ser analisados os resultados obtidos após a construção de rotas por esta aplicação, deparando-se com uma redução dos custos relativamente às rotas que se encontrariam em execução inicialmente.

Em [Mourão, Nunes e Prins, 2008] [14] podem encontrar-se algumas das heurísticas descritas anteriormente, aplicadas a um determinado problema. O problema em questão é abordado como um problema SARP (Sectoring Arc Routing Problem), e a sua solução é obtida através de heurísticas de 2 fases. Na primeira fase, é aplicada a heurística CTH (Circuit Task Heuristic) e numa segunda fase é a STH (Single Task Heuristic), obtendo no final resultados bastante satisfatórios relativamente aos que se encontravam em execução anteriormente.

[Kulkar,1996][10] aposta na optimização da recolha de resíduos na cidade de Bruxelas, obtendo no final, resultados positivos relativamente ao que se encontrava implementado anteriormente. Estas soluções são encontradas através de problemas que são divididos em duas fases distintas, sendo cada uma delas responsável pela optimização de duas funções objectivo diferentes, tornando-o num problema mais simples de ser resolvido.

[Nuortio,2006][1] dedica-se à optimização de rotas de recolha de resíduos em duas zonas diferentes da Finlândia. Para a sua resolução, aposta em métodos de duas fases, na primeira fase utiliza uma heurística híbrida de inserção, na segunda opta pela aplicação de uma metaheurística designada por GVNT (Guided Variable Neighborhood Thresholding). Para o cálculo do mínimo caminho a ser percorrido, em termos de distância ou de tempo de viagem, é utilizado o algoritmo de Dijkstra.

Na realidade, muitos trabalhos poderão ser encontrados nesta área, sendo que os que aqui são apresentados são apenas uma pequena parcela dos existentes.

Capítulo 3

Conclusões

Após a realização do estado-de-arte do tema em análise, chega-se à conclusão que este se trata de um tema bastante abrangente e complexo.

A sua complexidade deve-se à grande quantidade de pontos ou arcos a ter conta, enquanto se faz a construção das rotas a serem seguidas, bem como ao número de restrições que poderão existir.

Quanto à sua abrangência prende-se ao facto de serem encontrados bastantes trabalhos nesta área, cada um deles recorrendo a diferentes métodos para a sua resolução.

Apesar disso, é um tema com bastantes aplicações na vida real, trazendo inúmeras vantagens para as empresas que os pretendam investigar e posteriormente aplicar, não só no campo da recolha de resíduos urbanos, mas em outras áreas também.

Capítulo 4

Referências Bibliográficas

1. Nuortio, T., et al., *Improved route planning and scheduling of waste collection and transport*. Expert Systems with Applications, 2006. 30(2): p. 223-232.
2. Dréo Pétrowski and S. Taillard, *Metaheuristics for Hard Optimization: Methods and Case Studies*, ed. Springer. 2006.
3. Ayininuola, G. and M. Muibi, An engineering approach to solid waste collection system: Ibadan North as case study. Waste Management, 2008. 28(9): p. 1681-1687.
4. Bautista, J., *Solving an urban waste collection problem using ants heuristics*. Computers & Operations Research, 2008. 35(9): p. 3020-3033.
5. Benjamin, A.M. and J.E. Beasley, *Metaheuristics for the waste collection vehicle routing problem with time windows, driver rest period and multiple disposal facilities*. Computers & Operations Research, 2010. 37(12): p. 2270-2280.
6. Caballero, R., et al., *Solving a multiobjective location routing problem with a metaheuristic based on tabu search. Application to a real case in Andalusia*. European Journal of Operational Research, 2007. 177(3): p. 1751-1763.
7. Carvalho, M.M.V.T.d., *Optimização de circuitos e indicadores de recolha de resíduos urbanos. Caso de estudo: Município de Almada*. 2008, Universidade Nova de Lisboa: Lisboa.
8. Dang Vu Tung, A.P., *Vehicle routing-scheduling for waste collection in Hanoi*. European Journal of Operational Research, 2000.
9. Deoliveirasimonetto, E. and D. Borenstein, *A decision support system for the operational planning of solid waste collection*. Waste Management, 2007. 27(10): p. 1286-1297.

10. Kulcar, T., *Optimizing solid waste collection in Brussels*. European Journal of Operational Research, 1996. 90: p. 71-77.
11. Mourao, M., A. Nunes, and C. Prins, *Heuristic methods for the sectoring arc routing problem*. European Journal of Operational Research, 2009. 196(3): p. 856-868.
12. Mourão., M.C., A.C. Nunes., and C. Prins., *Heuristic methods for the sectoring arc routing problem*. European Journal of Operational Research, 2008.
13. N.Meegoda, K.A.F.L.A.-M.H.-N.H.a.J., *Optimization of Municipal Solid Waste Collection System: Case Study*. ASCE, 2009.
14. Nunes, A.C.d.C., *Sectorização de Redes em Problemas com Procura nos Arcos e Limitações de Capacidade*. 2009, Universidade Técnica de Lisboa.
15. Sahoo, S., et al., *Routing Optimization for Waste Management*. Interfaces, 2005. 35(1): p. 24-36.
16. Teixeira, J., A. Antunes, and J. Desousa, *Recyclable waste collection planning--a case study*. European Journal of Operational Research, 2004. 158(3): p. 543-554.
17. Ferreira, José Soeiro, *Grafos*
18. Ferreira, José Soeiro, *Optimização Combinatória*
19. Carravilha, M.A, Oliveira, J.F., *Programação Inteira: Resolução por Branch and Bound, 2010*
20. Carravilha, M.A, Oliveira, J.F., *Heuristics and Local Search, 2009*