



Alexandre Duarte Aquino Marques

**RESOLUÇÃO DE PROBLEMAS DE OTIMIZAÇÃO
DE SISTEMAS DE DISTRIBUIÇÃO DE ÁGUA -
NOVOS PROCEDIMENTOS PARA O
MELHORAMENTO DO DESEMPENHO DE
ALGORITMOS HEURÍSTICOS**

**OPTIMISATION OF WATER DISTRIBUTION
NETWORKS — NEW PROCEDURES TO IMPROVE
MODERN HEURISTICS' PERFORMANCE**

Dissertação de Mestrado Integrado em Engenharia Civil, na área de Especialização em Hidráulica, Recursos Hídricos e Ambiente,
orientada pelo Professor Doutor Maria da Conceição Morais de Oliveira Cunha.

Coimbra, 28 de setembro de 2017



UNIVERSIDADE DE COIMBRA



FCTUC DEPARTAMENTO DE ENGENHARIA CIVIL
FACULDADE DE CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE DE COIMBRA

Alexandre Duarte Aquino Marques

**RESOLUÇÃO DE PROBLEMAS DE OTIMIZAÇÃO
DE SISTEMAS DE DISTRIBUIÇÃO DE ÁGUA -
NOVOS PROCEDIMENTOS PARA O
MELHORAMENTO DO DESEMPENHO DE
ALGORITMOS HEURÍSTICOS**

**OPTIMISATION OF WATER DISTRIBUTION
NETWORKS – NEW PROCEDURES TO IMPROVE
MODERN HEURISTICS’ PERFORMANCE**

Dissertação de Mestrado Integrado em Engenharia Civil, na área de Especialização em Hidráulica, Recursos Hídricos e Ambiente, orientada pelo Professor Doutor Maria da Conceição Morais de Oliveira Cunha.

Esta Dissertação é da exclusiva responsabilidade do seu autor.
O Departamento de Engenharia Civil da FCTUC, declina qualquer
responsabilidade legal ou outra em relação a erros ou omissões
que possam surgir

Coimbra, 28 de setembro de 2017

AGRADECIMENTOS

À Professora Maria da Conceição Cunha, pela sua orientação, transmissão de conhecimentos e por ter fomentado o meu interesse na área da Hidráulica, Recursos Hídricos e Ambiente. O seu espírito crítico e exigência de rigor tiveram um papel importante, não só para alcançar a qualidade final deste trabalho, como também no meu crescimento enquanto pessoa.

Ao João Rosa Marques, pela sua disponibilidade e simpatia. A ele devo-lhe grande parte dos meus escassos conhecimentos em programação, essenciais para desenvolver a componente informática deste trabalho. Agradeço-lhe, ainda, por ter disponibilizado o programa que serviu de base para esta dissertação.

À Tatiana Silva, pelo seu apoio constante ao longo de todo o tempo que estamos juntos. Todas as dificuldades que me foram surgindo tornaram-se mais facilmente superáveis com ela a meu lado.

A toda a minha família pelas mensagens de encorajamento e motivação. Aos meus pais, pelo amor incondicional, educação e por sempre se terem esforçado em garantir uma boa qualidade de vida para os seus filhos. Às minhas irmãs, pela boa disposição contagiante. Aos meus avós, por todo o carinho e apoio concedido ao longo da minha existência.

A todos os que, direta ou indiretamente, fizeram parte do meu percurso académico e contribuíram para que a conclusão do meu curso se tornasse uma realidade. Deixo um especial agradecimento aos meus colegas da especialidade em Hidráulica, André Moura e Guilherme Ribeiro, pela amizade e pelos momentos de boa disposição passados, e também aos meus amigos de longa data, Roberto Marques e Cristiano Laranjeiro, que contribuíram de forma significativa para moldar a minha personalidade.

RESUMO

Os sistemas de distribuição de água constituem infraestruturas de importância crucial nas sociedades atuais. Trata-se de sistemas cuja concretização é bastante onerosa, devendo ser tomadas iniciativas no sentido de propor soluções para o seu dimensionamento que simultaneamente permitam minimizar os seus custos e dar resposta à procura com adequado desempenho hidráulico. Os modelos de otimização podem dar um importante contributo para este objetivo. Nos últimos anos, têm aparecido variados modelos que, pelas suas características matemáticas, têm sido resolvidos através de diferentes heurísticas e metaheurísticas. No entanto verifica-se que em problemas de maiores dimensões é ainda necessário investir na melhoria desses métodos para que boas soluções possam ser encontradas eficientemente.

Nesta dissertação é testado o desempenho de uma metaheurística híbrida construída a partir dos conceitos do Recozimento Simulado e da Entropia Cruzada e é aplicada a problemas típicos de dimensionamento de sistemas de distribuição de água. O principal objetivo é melhorar a eficiência em encontrar soluções quase-ótimas relativamente à eficiência do Recozimento Simulado (sem hibridização).

Palavras-chave: Otimização; Sistemas de Distribuição de Água; Metaheurísticas; Recozimento Simulado; Entropia Cruzada

ABSTRACT

Water distribution networks are infrastructures of crucial importance in today's societies. These are systems whose implementation is quite expensive, and initiatives must be taken to propose solutions for their design that simultaneously allow to minimize their costs and meet the demand with adequate hydraulic performance. Optimization models can make an important contribution to this goal. In recent years, several models have appeared that, due to their mathematical characteristics, have been solved through different heuristics and metaheuristics. However, it is found that in larger problems it is still necessary to invest in the improvement of these methods.

In this dissertation, the performance of a hybrid metaheuristic created from the concepts of Simulated Annealing and Cross-Entropy and applied to typical problems of water distribution networks, is tested. The main objective is to improve the efficiency of finding quasi-optimal solutions when compared with Simulated Annealing's efficiency.

Keywords: Optimization; Water Distribution Networks; Metaheuristics; Simulated Annealing; Cross-Entropy

ÍNDICE

ÍNDICE DE FIGURAS	vi
ÍNDICE DE QUADROS	vii
ÍNDICE DE GRÁFICOS	viii
SIMBOLOGIA	ix
SIGLAS / ACRÓNIMOS	xi
1 INTRODUÇÃO.....	1
1.1 Enquadramento Geral	1
1.2 Objetivos e Âmbito da Dissertação.....	3
1.3 Organização da Dissertação	3
2 REVISÃO BIBLIOGRÁFICA	4
2.1 Introdução	4
2.2 Propriedades das Metaheurísticas	6
2.3 Aplicação a Problemas Reais.....	10
2.3.1 Mudanças na Formulação do Problema.....	10
2.3.2 Redução do Espaço de Procura	11
2.3.3 Melhoria da Eficiência Computacional	12
2.3.4 Aptidão para Encontrar Soluções Ótimas.....	13
2.3.5 Identificação do Melhor Critério de Convergência	13
2.3.6 Simplificação do Processo de Decisão Final	14
2.3.7 Incorporação da Incerteza.....	15
2.3.8 Implementação do Algoritmo.....	16
2.4 Revisão de Heurísticas de Otimização.....	17
2.4.1 Heurísticas de Pesquisa Local	17
2.4.2 Recozimento Simulado.....	19
2.4.3 Pesquisa Tabu	20
2.4.4 Pesquisa em Vizinhanças Variáveis	21
2.4.5 Método da Entropia Cruzada.....	22
2.4.6 Algoritmos Genéticos	24
2.4.7 Hibridização de Metaheurísticas	25
3 FORMULAÇÃO DO MODELO DE OTIMIZAÇÃO.....	29
3.1 Função Objetivo.....	30
3.2 Restrições de Conservação da Massa	30
3.3 Restrições de Conservação da Energia	30
3.4 Restrições de Pressão Mínima / Máxima.....	31
3.5 Restrições de Velocidade Máxima / Mínima.....	32

3.6	Restrições de Diâmetro Mínimo	32
3.7	Restrições de Domínio das Variáveis de Decisão	33
3.8	Modelo Matemático Completo	33
4	DESCRIÇÃO DO ALGORITMO	34
4.1	Parâmetros de Controlo / Solução Inicial	34
4.2	Geração de Soluções Candidatas	36
4.3	Critério de Metropolis	37
4.4	Saída do Ciclo Interno	37
4.5	Saída do Ciclo Externo	37
4.6	Solução Final e Índices de Desempenho / Segunda Fase	38
4.7	Alterações Propostas	38
4.7.1	Novos Parâmetros de Controlo	39
4.7.2	Geração da Matriz de Probabilidades Inicial / Atualização da Lista de Soluções / Critério de Atualização da Matriz de Probabilidades	40
4.7.3	Escolha de um Subconjunto de Melhores Soluções	41
4.7.4	Atualização da Matriz de Probabilidades	41
4.7.5	Nova Geração de Soluções Candidatas	42
5	ESTUDOS DE CASO	43
5.1	Two-Loop Network	43
5.2	Hanoi Network	43
5.3	Fossolo Network	44
5.4	Balerna Irrigation Network	45
6	RESULTADOS	47
6.1	Simulated Annealing	48
6.2	Simulated Annealing com Segunda Fase	50
6.3	Simulated Annealing Híbrido	51
7	CONCLUSÃO	57
7.1	SA vs. SAR	57
7.2	SAR vs. SACER	57
7.3	Trabalhos Futuros	58
	REFERÊNCIAS BIBLIOGRÁFICAS	60

ÍNDICE DE FIGURAS

Figura 5.1 - Esquema da Two-Loop Network.....	43
Figura 5.2 - Esquema da rede de Hanoi.....	44
Figura 5.3 - Esquema da rede de Fossolo.....	44
Figura 5.4 - Esquema da rede de Balerna.....	45

ÍNDICE DE QUADROS

Quadro 2.1 - Características do espaço das soluções admissíveis, influência das mesmas na dificuldade de um problema e exemplos de métricas (Adaptado de Maier et al., 2014).....	7
Quadro 2.2 - Algumas métricas genericamente aceites para quantificar a intensificação e a diversificação de um algoritmo	9
Quadro 4.1 - Exemplo hipotético de matriz de probabilidades	39
Quadro 5.1 - Diâmetros comerciais disponíveis e respetivos custos unitários.....	46
Quadro 6.1 - Resultados para o algoritmo SA.....	48
Quadro 6.2 - Resultados para o algoritmo SAR	50
Quadro 6.3 - Parâmetros de controlo usados para o algoritmo SACER.....	51
Quadro 6.4 - Resultados para o algoritmo SACER.....	52

ÍNDICE DE GRÁFICOS

Gráfico 2.1 - Problema de otimização hipotético	18
Gráfico 2.2 - Exemplo hipotético do espaço de procura de um algoritmo híbrido obtido por adição de um método de pesquisa local.....	26
Gráfico 4.1 - Fluxograma do algoritmo utilizado (adaptado de Cunha, 1999 e Cunha & Sousa, 1999, 2001).....	35
Gráfico 4.2 - Fluxograma da geração de soluções candidatas (adaptado Cunha & Sousa, 2001).....	36
Gráfico 4.3 - Fluxograma das alterações propostas.....	40
Gráfico 6.1 - Influência do parâmetro α no andamento do algoritmo SA	49
Gráfico 6.2 - Influência do parâmetro μ_1 no andamento do algoritmo SA.....	49
Gráfico 6.3 - Efeito da Segunda Fase	50
Gráfico 6.4 - Influência do parâmetro β	53
Gráfico 6.5 - Influência do parâmetro γ	53
Gráfico 6.6 - Influência do parâmetro N	53
Gráfico 6.7 - Influência do parâmetro ρ	53
Gráfico 6.8 - Influência do parâmetro ρ no andamento do algoritmo SACER.....	55
Gráfico 6.9 - Influência do parâmetro N no andamento do algoritmo SACER	55
Gráfico 6.10 - Andamento dos algoritmos SACER e SAR para o caso HAN	56
Gráfico 6.11 - Andamento dos algoritmos SACER e SAR para o caso FOS.....	56

SIMBOLOGIA

A	Coeficiente da fórmula de Hazen-Williams [-]
B	Expoente da fórmula de Hazen-Williams [-]
C	Expoente da fórmula de Hazen-Williams [-]
C_k	Coeficiente de resistência da fórmula de Hazen-Williams [-]
D	Conjunto de diâmetros comerciais disponíveis
$D_{min,k}$	Diâmetro mínimo admissível para as condutas [m]
E	Energia do sistema físico (Recozimento Simulado)
F	Função objetivo
H_i	Energia total nos nós da rede [m]
I_{jk}	Função que toma o valor de 0 ou 1 dependente do diâmetro utilizado na conduta j da solução k (Entropia Cruzada) [-]
M_p	Conjunto de condutas pertencentes a uma malha da rede
N	Tamanho da amostra (Entropia Cruzada) [-]
N_i	Conjunto de pontos diretamente ligados a um nó da rede
P_t	Probabilidade de aceitação à temperatura t (Recozimento Simulado) [-]
T	Temperatura do sistema físico (Recozimento Simulado)
V	Função vizinhança
V_k	Função vizinhança k (Pesquisa em Vizinhanças Variáveis)
X	Espaço das soluções admissíveis
c	Custo unitário das condutas [\$/m] [€/m]
d_i	Consumo nos nós da rede [m ³ /s]
ea	Elasticidade de aceitação (Recozimento Simulado) [-]
f_k	Coeficiente de resistência da fórmula de Darcy-Weisbach [-]
g	Aceleração gravítica [9.8 m/s ²]
h_i	Pressão nos nós da rede [m]
$h_{max,i}$	Pressão máxima admissível nos nós da rede [m]
$h_{min,i}$	Pressão mínima admissível nos nós da rede [m]
i	Solução atual/corrente
j	Solução candidata
j_k	Solução candidata referente à vizinhança k (Pesquisa em Vizinhanças Variáveis)
k_B	Constante de Boltzmann (Recozimento Simulado)
k_{max}	Número de vizinhanças utilizadas (Pesquisa em Vizinhanças Variáveis) [-]
l	Comprimento das condutas [m]

ll	Número de diâmetros comerciais (Entropia Cruzada) [-]
m	Número de condutas da rede [-]
n	Número de nós da rede [-]
nl	Número de malhas naturais da rede [-]
p	Probabilidade de aceitação do critério de Metropolis (Recozimento Simulado) [-]
pa_{lim}	Probabilidade de aceitação limite (Recozimento Simulado) [-]
pl	Número de malhas fictícias da rede [-]
p_{ij}^t	Probabilidade de se instalar o diâmetro comercial i na conduta j à t -ésima iteração (Entropia Cruzada) [-]
\hat{p}_{ij}^t	Estimador de p_{ij}^t (Entropia Cruzada) [-]
q_{ij}, q_k	Caudal nas condutas da rede [m ³ /s]
r	Número aleatório compreendido entre 0 e 1 [-]
rr	Número de reservatórios na rede [-]
t	Temperatura do algoritmo (Recozimento Simulado)
t_0	Temperatura inicial do algoritmo (Recozimento Simulado)
v_k	Velocidade do fluxo de água nas condutas da rede [m/s]
$v_{max,k}$	Velocidade máxima admissível nas condutas da rede [m/s]
$v_{min,k}$	Velocidade mínima admissível nas condutas da rede [m/s]
x	Variáveis de decisão / diâmetro das condutas / solução corrente
x_t	Variáveis de decisão à iteração t
x'	Solução ótima local
x^*	Solução ótima global
y	Solução candidata (Recozimento Simulado)
z_i	Cota geométrica dos nós da rede [m]
ΔH_k	Perda de carga nas condutas [m]
Δz_p	Diferença de cotas entre reservatórios [m]
Φ	Diâmetros comerciais das condutas [m]
α	Coefficiente de arrefecimento (Recozimento Simulado) [-]
β	Coefficiente de suavização (Entropia Cruzada) [-]
γ	Frequência de atualização da matriz de probabilidades (Entropia Cruzada) [-]
μ_1	Parâmetro relativo ao critério de saída do ciclo interno (Recozimento Simulado) [-]
μ_2	Parâmetro relativo ao critério de saída do ciclo externo (Recozimento Simulado) [-]
ρ	Coefficiente de redução da amostra (Entropia Cruzada) [-]

SIGLAS / ACRÓNIMOS

ANN – Rede Neural Artificial (*Artificial Neural Network*)

API – Interface de Programação de Aplicações (*Application Programming Interface*)

BIN – Rede de irrigação da cidade de Balerma

CE – Método da Entropia Cruzada (*Cross-Entropy Method*)

CPU – Unidade de Processamento Central (*Central Processing Unit*)

DLL – Biblioteca de Vínculo Dinâmico (*Dynamic-Link Library*)

FOS – Rede de abastecimento da cidade de Fossolo

GA – Algoritmos Genéticos (*Genetic Algorithms*)

GLS – Pesquisa Local Guiada (*Guided Local Search*)

GRASP – Procedimento de Pesquisa Guloso, Aleatório e Adaptativo (*Greedy Randomized Adaptive Search Procedure*)

GRG – Gradiente Reduzido Generalizado (*Generalized Reduced Gradient*)

HAN – Rede de abastecimento da cidade de Hanoi

ILS – Pesquisa Local Iterativa (*Iterative Local Search*)

PC – Computador Pessoal (*Personal Computer*)

PSO – Otimização por Enxame de Partículas (*Particle Swarm Optimisation*)

SA – Recozimento Simulado (*Simulated Annealing*)

SACER – *Simulated Annealing* com *Cross-Entropy* e Segunda Fase

SAR – *Simulated Annealing* com Segunda Fase

SDA – Sistemas de Distribuição (Pública) de Água

TLN – Rede de abastecimento hipotética com 2 malhas (*Two-Loop Network*)

TS – Pesquisa Tabu (*Tabu Search*)

VNS – Pesquisa em Vizinhanças Variáveis (*Variable Neighbourhood Search*)

1 INTRODUÇÃO

1.1 Enquadramento Geral

A água é essencial para assegurar a sobrevivência de todas as espécies de seres vivos existentes no planeta. Apesar da sua abundância no globo terrestre, apenas uma pequena porção dessa água está disponível para satisfazer as necessidades do Homem e não se encontra uniformemente distribuída. Como consequência, existem atualmente, comunidades com problemas sérios de carência de água e no futuro a tendência será para piorar (principalmente em países desenvolvidos). Portugal poderá vir a sofrer de stress hídrico agravado (razão entre a procura e a disponibilidade de água) nos próximos 20 anos, tal como relatado pelo *World Resources Institute* (WRI@, 2015). É, por esse motivo, deveras importante consciencializar a população para esta ameaça iminente e para a importância que uma adequada gestão dos recursos hídricos tem para retardar este efeito.

Uma das áreas da gestão de recursos hídricos onde se devem tomar medidas é nos sistemas de distribuição pública de água. Em algumas regiões de Portugal, as perdas nestes sistemas chegam a atingir valores intoleráveis (>60%), sendo a média no país de aproximadamente 30% (ERSAR, 2016).

Por sistemas de distribuição pública de água (SDA) compreendem-se todos os elementos necessários para efetuar o adequado transporte e abastecimento de água, a partir de um reservatório de distribuição, até aos utilizadores em quantidade e pressão adequadas. Da lista de elementos mais comuns encontrados em SDA, fazem parte: reservatórios (*e.g.* distribuição, extremidade); um conjunto de condutas interligadas entre si, tal que a sua disposição em planta forma uma rede composta por malhas e/ou ramais; estações elevatórias; e um conjunto de acessórios, tais como válvulas, descargas de fundo, *etc.*

Aquando do dimensionamento ou reabilitação de SDA, o projetista é confrontado com inúmeras soluções possíveis. No passado (e em muitos casos presentemente), a forma de atuar dependia, em grande parte, do conhecimento adquirido pelo projetista através de projetos semelhantes realizados anteriormente, apoiado por modelos de simulação hidráulica, quando disponíveis. Ou seja, o projetista idealizava uma solução inicial que posteriormente seria ajustada por forma a cumprir com restrições de natureza regulamentar, técnica ou de outro tipo eventualmente. Restrições essas que seriam verificadas com a ajuda de modelos de simulação hidráulica. Sem

desvalorizar o trabalho até agora efetuado, a verdade é que na maior parte das soluções que foram implementadas teria sido possível otimizá-las por forma a minimizar diversos fatores, tais como o custo associado à conceção ou exploração do sistema, a vulnerabilidade do sistema ou ainda a quantidade de perdas do sistema (tema que está, cada vez mais, na ordem do dia de países desenvolvidos). É neste ponto que devem entrar os mais recentes desenvolvimentos da Investigação Operacional.

A Investigação Operacional surgiu no decorrer da 2ª Guerra Mundial como forma de resolução de problemas de elevada complexidade logística militar. Atualmente, é uma área disciplinar indispensável para lidar com alguns dos principais problemas do quotidiano, entre os quais encontram-se os problemas que envolvem a gestão de recursos hídricos, e mais concretamente, os problemas de dimensionamento de SDA. Algumas das ferramentas desenvolvidas no âmbito da Investigação Operacional, tais como as metaheurísticas, podem ajudar na tomada de decisões que terão influência não só no presente, mas também num futuro mais ou menos distante. Posto isto, uma pergunta inevitável poderá surgir: “Mas de que forma podem as metaheurísticas ajudar na tomada de decisões?”. A melhor forma de responder a esta pergunta é expondo-a na forma de um exemplo.

Imagine-se que é necessário tomar uma decisão relativamente a um SDA, gravítico, composto por um reservatório de distribuição (que já se encontra implementado) e por uma rede de 20 condutas, a dimensionar, com a possibilidade de utilização de 10 diâmetros diferentes. As diversas partes interessadas chegaram a um acordo sobre os diversos objetivos a alcançar, sendo que o objetivo essencial é o de que a água deve chegar aos consumidores em quantidade e pressão suficientes e os custos decorrentes da aquisição das condutas devem ser minimizados. Este problema contém apenas variáveis discretas, pelo que é um típico problema de otimização combinatoria em que o número de soluções possíveis é finito e é igual a 10^{20} soluções.

Suponha-se agora, num ambiente ideal, que dispõe da mais recente tecnologia informática e tem na sua posse um PC com uma capacidade de processamento total de 20GHz. Se forem necessários 10^{-5} segundos para obter uma solução do problema, então levaria milhões de anos a enumerar todas as possíveis soluções deste problema. A impraticabilidade deste procedimento (apesar de se garantir uma solução ótima) leva a que sejam usadas outras formas de resolver este tipo de problema. As metaheurísticas são uma opção viável, permitindo resolver problemas desta dimensão em poucos segundos, não garantindo que a solução encontrada seja ótima, contudo.

1.2 Objetivos e Âmbito da Dissertação

A presente dissertação, insere-se diretamente no campo de aplicação do dimensionamento otimizado de SDA podendo ser extrapolável para outras áreas da gestão de recursos hídricos. Para além disso, as metaheurísticas em geral são um tema corrente desta dissertação, com especial destaque para o Recozimento Simulado.

Fundamentalmente, pretende-se com esta dissertação, contribuir para a investigação e desenvolvimento de métodos heurísticos híbridos que visam melhorar o desempenho de algoritmos de otimização existentes quando aplicados a problemas típicos de otimização do dimensionamento de SDA. Outros objetivos passam por dar a conhecer a aplicabilidade das diferentes metaheurísticas e servir de elemento de apoio para trabalhos futuros neste setor.

1.3 Organização da Dissertação

A dissertação encontra-se organizada em 7 capítulos, o primeiro dos quais é a presente introdução.

No segundo capítulo será feita uma revisão bibliográfica acerca de metaheurísticas. Começar-se-á por introduzir a definição de metaheurística e explicar o porquê da sua popularidade. Seguidamente, indicar-se-ão algumas das suas características e os principais problemas encontrados aquando da sua aplicação a problemas reais, fazendo referência, sempre que possível, ao setor do dimensionamento otimizado de SDA. Por fim, são expostos exemplos de diversas metaheurísticas amplamente utilizadas pela comunidade científica.

No terceiro capítulo será apresentado o modelo matemático geral correspondente aos problemas de dimensionamento otimizado de SDA.

No quarto capítulo será descrito o algoritmo utilizado. Primeiro, descrever-se-á o algoritmo base e de seguida descrever-se-á as alterações efetuadas, justificando as razões para tal.

No quinto capítulo expor-se-ão os problemas que foram alvo de análise nesta dissertação e identificar-se-ão as principais diferenças entre eles.

No sexto capítulo serão apresentados alguns resultados de diversas simulações efetuadas. O objetivo será comparar o comportamento do algoritmo original relativamente ao algoritmo depois de alterado de diversas formas.

Por fim, no sétimo e último capítulo serão elaboradas algumas conclusões em relação ao trabalho desenvolvido e são dadas algumas sugestões de trabalhos a realizar futuramente.

2 REVISÃO BIBLIOGRÁFICA

2.1 Introdução

O aparecimento de metaheurísticas¹ como forma de resolução de problemas ligados ao campo dos recursos hídricos já data da década de 90 (*e.g.* Dougherty & Marryott, 1991). Desde então, o interesse por esta área tem levado a uma procura incessante pelo conhecimento mais aprofundado da forma como operam as diversas metaheurísticas e de que forma se relacionam com os tipos de problemas mais comuns na gestão de recursos hídricos (*i.e.*, calibração de modelos, planeamento, conceção e manutenção de sistemas de recursos hídricos, *etc.*).

As metaheurísticas vêm colmatar uma série de lacunas, que ainda hoje não foram possíveis de ultrapassar, referentes aos métodos de otimização clássicos². As vantagens mais relevantes, que justificam a grande aceitação por parte da comunidade científica, são seguidamente expostas (Maier *et al.*, 2014):

- Facilidade na compreensão das estratégias de otimização, devida à analogia (em alguns casos) com fenómenos simples de interpretação biológica, química, física, entre outras;
- É relativamente fácil adicionar os modelos de simulação hidráulica existentes no mercado à componente da otimização. Desta forma, o decisor tem a possibilidade de escolher o modelo de simulação com que se sente mais à vontade para trabalhar e continuar a confiar nos resultados obtidos. Isto é possível, uma vez que, qualquer metaheurística pode ser aplicada (bem ou mal, é um assunto discutido posteriormente) independentemente do tipo de problema a tratar;
- Relativamente ao ponto anterior, uma das vantagens consequentes é a de se conseguir realizar processamento em paralelo de forma bastante mais simples;

¹ Uma metaheurística pode ser definida como um processo iterativo que controla a forma de atuação das diversas heurísticas subordinadas com vista a utilizar de modo ponderado os conceitos de diversificação e intensificação do espaço de pesquisa para criar um algoritmo que se guia de modo eficiente para soluções quase-ótimas, escapando assim de eventuais ótimos locais (para uma definição mais abrangente ver *e.g.* Blum & Roli, 2003). As metaheurísticas são, por esse motivo, considerados métodos de pesquisa globais. Como exemplos mais comuns podem referir-se os Algoritmos Genéticos, o Recozimento Simulado, a Pesquisa Tabu, entre outros.

² Métodos de otimização clássicos são aqueles que se baseiam em conceitos puramente matemáticos e que sob algumas condições garantem a obtenção da solução ótima global. Técnicas como a Programação Linear, Dinâmica e Não-Linear são alguns exemplos.

- Talvez a principal vantagem seja a de que, as metaheurísticas possibilitam a resolução de problemas de elevado grau de dificuldade matemática, permitindo encontrar soluções quase ótimas de forma mais eficiente.

As referidas metaheurísticas são comumente aglomeradas em dois grupos distintos, sendo eles os seguintes:

- Metaheurísticas baseadas numa população (*e.g.* Algoritmos Genéticos (GA); Entropia Cruzada (CE));
- Metaheurísticas baseadas num único ponto (ou, equivalentemente designadas, metaheurísticas baseadas numa trajetória) (*e.g.* Recozimento Simulado (SA); Pesquisa Tabu (TS)).

A principal diferença entre elas, como o próprio nome indica, reside no número de soluções analisadas simultaneamente em cada iteração.

Geralmente, não é conhecido o melhor algoritmo de otimização a aplicar a um determinado problema e, quando é, deve-se à variedade e quantidade de experiências já desenvolvidas nessa área. Pode ser demonstrado (Wolpert & Macready, 1997) que um determinado algoritmo de otimização não consegue levar à melhor solução para todos os problemas possíveis. De facto, é facilmente observável que caso se tenha uma função objetivo contínua, diferenciável e estritamente convexa (ou côncava)³ um método baseado em gradientes locais terá um melhor desempenho quando comparado com um dos métodos de pesquisa globais exemplificados, porém, quando o problema é de otimização combinatoria é expectável que o inverso ocorra (Maier *et al.*, 2014). Ainda assim, os dois grupos de metaheurísticas seguiram trajetórias de popularidade diferentes, havendo, até ao momento, uma subtil preferência pela escolha de algoritmos baseados numa população, para o caso de problemas relacionados com a gestão de recursos hídricos.

O dimensionamento otimizado de SDA não é exceção, este tema já começou a ser estudado mesmo antes das metaheurísticas mais conhecidas terem surgido (*e.g.* Schaake & Lai, 1969; Alperovits & Shamir, 1977), nestes casos, eram feitas aproximações do problema real para que métodos de otimização clássicos pudessem ser usados, tais como admitir que os diâmetros das condutas podiam ser representados por variáveis contínuas. Com a popularidade das metaheurísticas a crescer de dia para dia, era uma questão de tempo até estes métodos começarem a ser aplicados ao dimensionamento otimizado de SDA, Savic & Walters (1997) com os GA, Cunha & Sousa (1999) com o SA, Fanni *et al.* (2000) e Cunha & Ribeiro (2004) com a TS e outros exemplos poderiam ser dados. Atualmente, os objetivos não se prendem

³ No âmbito da otimização, se o espaço das soluções admissíveis for um conjunto convexo e a função objetivo convexa então o problema tem apenas um mínimo, que é também o mínimo global. Se o espaço das soluções admissíveis for convexo e a função objetivo côncava então o problema tem apenas um máximo, que é também o máximo global. Para outras situações o problema poderá admitir diversos mínimos/máximos locais.

apenas com a minimização dos custos de um SDA, havendo a preocupação de garantir a sustentabilidade dos mesmos, seja numa perspectiva ambiental (*e.g.* Marques *et al.*, 2015b), numa perspectiva de gestão de riscos (*e.g.* Cunha & Sousa, 2010), ou qualquer outra que se considere adequada.

No contexto desta dissertação, é importante rever algumas questões acerca destes métodos de otimização “modernos” que têm vindo a acompanhar a sua própria evolução, tais como:

- Em que ponto está o nosso conhecimento acerca das propriedades dos algoritmos existentes e de que forma podem ser influenciadas?
- Podemos melhorar o desempenho desses algoritmos quando aplicados a problemas reais?
- Como funcionam esses algoritmos e quais as principais diferenças entre eles?

Devido à quantidade de informação disponível com o propósito de responder a estas perguntas, optou-se por dividir cada assunto por secções. Na última secção deste capítulo são introduzidos alguns conceitos de pesquisa local, essenciais, para uma melhor compreensão do funcionamento dos diversos algoritmos aí apresentados (restringe-se a revisão a alguns métodos que se julga serem importantes para a compreensão deste trabalho).

2.2 Propriedades das Metaheurísticas

Como já foi referido, não existe nem deverá existir num futuro próximo um algoritmo ideal, capaz de resolver todos os problemas de modo mais eficiente que outro algoritmo qualquer. Ainda assim, é possível restringir a utilização de um determinado algoritmo a um subconjunto de problemas por forma a que esse algoritmo tenha uma eficiência bem acima da média. Para esse efeito, é necessário saber de que forma é influenciado o desempenho de um algoritmo de acordo com as características de um problema.

Maier *et al.* (2014) resumem algumas das principais características dos problemas de otimização, relativas ao espaço das soluções admissíveis, e relacionam-nas com a dificuldade em obter soluções quase-ótimas. Para além disso, fornecem exemplos de metodologias para quantificar essas mesmas características (*i.e.*, métricas). Algumas dessas relações são apresentadas no Quadro 2.1.

Todas estas características são comuns a problemas de dimensionamento de SDA, em especial, a multimodalidade e a falsa aparência devem-se, fundamentalmente, à formação de malhas na rede e a proeminência pode ser o resultado de um troço da rede com um comprimento significativamente maior que os restantes.

Esta quantificação da dificuldade associada a um problema é muitas vezes posta em causa, sendo o principal motivo a não existência de uma definição rigorosa e consensual do conceito de “dificuldade” num problema de otimização (Kallel, 1998). No entanto, a avaliação da probabilidade (eficácia) e rapidez (eficiência) com que um algoritmo de otimização consegue localizar uma solução quase-ótima conhecida pode ajudar a melhor caracterizar esse termo. Torna-se óbvio que o algoritmo de otimização envolvido, e respetivos parâmetros de controlo, também terão influência nessa dificuldade, não devendo ser postos de parte aquando da respetiva avaliação.

Quadro 2.1 - Características do espaço das soluções admissíveis, influência das mesmas na dificuldade de um problema e exemplos de métricas (Adaptado de Maier *et al.*, 2014)

Característica	Dificuldade encontrada no problema	Exemplo de métrica
Modalidade	Quanto mais ótimos locais existirem mais difícil é encontrar boas soluções.	<i>Iterated Local Search</i> (Merz, 2004)
Neutralidade	Regiões planas no espaço das soluções admissíveis, onde o valor da função objetivo se mantém constante, podem bloquear os algoritmos e limitar a aptidão para encontrar melhores soluções.	<i>Genetic Distance Operator</i> (Katada & Ohkura, 2006)
Proeminência	Se uma variável de decisão tiver uma grande influência no valor da função objetivo, então, essa variável deve convergir para valores ótimos antes das outras variáveis influenciarem a solução obtida.	<i>Saliency Measure</i> (Gibbs <i>et al.</i> , 2011)
Falsa Aparência	Uma função objetivo pode aparentar ter o seu ótimo global localizado junto a bons ótimos locais enquanto que na realidade está bastante afastado dos mesmos. Torna-se, assim, mais difícil determinar a solução ótima.	<i>GA-Deception</i> (Deb & Goldberg, 1994)

Apesar de no passado a pesquisa desenvolvida ter-se focado, em grande parte, apenas na eficácia e eficiência da solução final, é também necessário ter em conta o comportamento do algoritmo ao longo do processo de otimização, para perceber qual é a dinâmica envolvida no processo iterativo de geração de novas soluções, querendo com isto explicar a forma como um algoritmo se movimenta no espaço das soluções admissíveis para que seja possível:

- Conseguir localizar e explorar zonas promissoras;
- Aproveitar a informação recolhida por forma a convergir para a solução ótima ou para a fronteira de Pareto (caso de problemas multiobjetivo);
- Evitar a convergência prematura para um ótimo local.

Na literatura da especialidade, o comportamento dos operadores de um algoritmo (ou até dos próprios algoritmos) era geralmente classificado, qualitativamente, com base na sua tendência

para explorar o espaço das soluções admissíveis ou para convergir para um ótimo local (Colorni *et al.*, 1996). A constante necessidade de melhorar a informação transmitida acerca de um algoritmo, por forma a aplicá-lo corretamente e facilitar o desenvolvimento do mesmo, levou a que análises comportamentais mais rigorosas fossem implementadas mais recentemente. Para o efeito, têm-se utilizado critérios que podem ser quantificados durante a execução de um algoritmo, com o objetivo de caracterizar o comportamento do mesmo. Maier *et al.*, (2014) propõem a utilização de alguns dos seguintes critérios:

- Avaliações no espaço das soluções admissíveis⁴, como os seguintes exemplos de critérios:
 - Medição da distância (topográfica), da qual resulta uma medição direta do grau de convergência de um algoritmo;
 - Medição da direção, de onde é possível caracterizar uma direção predominante segundo a qual o algoritmo focou a sua pesquisa;
 - Medição do histórico de frequências, que, como o nome evidencia, utiliza informação relativa à frequência de soluções no histórico de soluções geradas por um algoritmo para poder classificar esse algoritmo;
 - Medição da fiabilidade da solução, que mede o tempo gasto, pelo algoritmo, a explorar zonas admissíveis e não admissíveis;
- Avaliações no espaço objetivo⁵ (aplicável a problemas multiobjetivo), como a caracterização da evolução da fronteira de Pareto. São geralmente utilizadas para comparar a qualidade da pesquisa realizada por diversos algoritmos;
- Avaliações dependentes do algoritmo usado, que tentam estabelecer a relação entre os parâmetros de um algoritmo e o tempo despendido pelo mesmo para intensificar ou diversificar a procura.

Como já foi referido, a avaliação do comportamento de um algoritmo não deve ser menosprezada em relação à avaliação do seu desempenho (esta última que mereceu especial destaque na investigação realizada no passado), contudo, uma boa avaliação do desempenho de um algoritmo é de extrema importância, não só para poder comparar diretamente diversos algoritmos, mas também, porque em alguns casos proporciona uma melhor compreensão do comportamento do algoritmo.

O desempenho de um algoritmo está normalmente associado a quatro características fundamentais (algumas delas já mencionadas), são elas as seguintes:

- Eficácia - Para algoritmos com apenas uma função objetivo, a eficácia pode ser medida diretamente através da razão, ou distância, entre a solução obtida e a melhor solução alguma vez encontrada. Para algoritmos com várias funções objetivo a eficácia também

⁴ O espaço das soluções admissíveis é definido pelo domínio de aplicação das variáveis de decisão e é também conhecido por região admissível.

⁵ O espaço objetivo é definido pelo domínio de aplicação das funções objetivo e é limitado pela fronteira de Pareto.

pode ser medida, mas com alguma relatividade envolvida. Para tal é necessário avaliar (Zitzler *et al.*, 2000):

- A distância entre as soluções finais obtidas pelo algoritmo e a fronteira de Pareto;
- A cobertura do espaço não dominado;
- A extensão das soluções não dominadas.
- Eficiência - Teoricamente, a eficiência é medida pelo esforço computacional necessário para que um algoritmo encontre a solução ótima de um problema. Como normalmente não são conhecidas as soluções ótimas, a eficiência pode ser avaliada por comparação das soluções obtidas entre diversos algoritmos para um esforço computacional utilizado (*e.g.* número total de iterações realizadas, tempo despendido).
- Fiabilidade - Para problemas com apenas uma função objetivo, a fiabilidade pode ser representada pela média e desvio-padrão das melhores soluções encontradas em diversas execuções do algoritmo. Para algoritmos com várias funções objetivo, a fiabilidade pode ser representada pela *Empirical Attainment Function*, que mede a probabilidade de um algoritmo encontrar uma solução melhor ou igual do que uma solução previamente definida (López-Ibáñez *et al.*, 2010);
- Robustez - É definida como a maior ou menor sensibilidade aos parâmetros de controlo do algoritmo. Pode ser avaliada a partir da variação dos resultados obtidos por alteração dos parâmetros do algoritmo. Um algoritmo robusto irá apresentar boas soluções mesmo com parâmetros significativamente diferentes em cada execução.

No Quadro 2.2 são apresentadas algumas métricas que são de prática corrente no âmbito da caracterização da intensificação e da diversificação de algoritmos multiobjetivos.

Quadro 2.2 - Algumas métricas genericamente aceites para quantificar a intensificação e a diversificação de um algoritmo

Métricas de Medição da Intensificação	Autor(es)	Métricas de Medição da Diversificação	Autor(es)
<i>Hypervolume</i>	Zitzler, 1999	<i>Schott's Spacing</i>	Knowles & Corne, 2002
<i>Error Ratio</i>	Knowles & Corne, 2002	<i>Chi Square Deviation</i>	Srinivas & Deb, 1994
<i>ϵ-performance</i>	Kollat <i>et al.</i> , 2008	<i>Diversity Measure</i>	Deb & Jain, 2002
<i>Generational Distance</i>	Van Veldhuizen & Lamont, 1998	<i>Hole Relative Size</i>	Collette & Siarry, 2005
<i>Maximum Pareto Front Error</i>	Van Veldhuizen, 1999		
<i>Additive ϵ-indicator</i>	Zitzler <i>et al.</i> , 2003		

2.3 Aplicação a Problemas Reais

A aplicação de metaheurísticas a problemas de otimização está sujeita a uma série de desafios decorrentes das condições reais em que os referidos problemas se inserem, entre os quais se podem citar (Maier *et al.*, 2014):

- Mudanças na formulação do problema;
- Redução do espaço de procura;
- Melhoria da eficiência computacional;
- Aptidão para encontrar soluções ótimas;
- Identificação do melhor critério de convergência;
- Simplificação do processo de decisão final;
- Incorporação da incerteza;
- Implementação do algoritmo.

Dada a relevância de cada um destes temas, optou-se por expô-los convenientemente em subsecções. Ainda assim, estes desafios não devem ser vistos como independentes, estando quase sempre inerentemente ligados entre si.

2.3.1 Mudanças na Formulação do Problema

Em problemas reais, a formulação dos problemas torna-se mais complexa, estando envolvidas mais variáveis de decisão, mais restrições e mais objetivos. Para além disso, em problemas reais as restrições e os objetivos são muitas vezes confundidos, na medida em que, uma restrição pode tornar-se num objetivo e vice-versa. É devida a essa incerteza que reside grande parte da dificuldade de modelar um problema real. Enquanto que num problema académico é conhecida à partida a formulação do problema, ou então, é dada de forma implícita, num problema real a formulação do problema depende da existência de um consenso entre o analista (quem executa as metaheurísticas), o decisor final (*e.g.* entidades governamentais) e outras partes interessadas (*e.g.* consumidores finais) para chegar a uma formulação aceite por todos. Alguns estudos recentes (Babbar-Sebens & Minsker, 2008, 2010, 2012) consideram a possibilidade de usar metodologias participativas, onde as diversas partes interessadas colaboram na formulação do problema.

A respeito de objetivos refira-se que, os problemas de otimização da gestão de recursos hídricos são muitas vezes caracterizados por terem diversos objetivos conflituantes (*e.g.* sociais, ambientais, económicos, técnicos). Caso os diversos objetivos pudessem ser aglomerados numa única função objetivo, existiria apenas uma solução ótima do problema, porém, o trabalho de Franssen (2005) revelou que era muito difícil criar uma única função de agregação consistente. Por outro lado, algumas dificuldades também se opõem à utilização de demasiados objetivos, tais como a perda da perceção do espaço objetivo e conseqüentemente a perda de perceção dos *trade-off* disponíveis na fronteira de Pareto.

Os problemas estudados nesta dissertação são bem conhecidos da literatura e a sua formulação é unívoca (ver cap. 3), pelo que os problemas da ambiguidade (função objetivo/restrições) e multiplicidade de funções objetivos não são aplicáveis na presente dissertação.

2.3.2 Redução do Espaço de Procura

Uma das consequências do aumento das variáveis de decisão, e conseqüente aumento do espaço das soluções admissíveis, é o facto de tornar muito mais difícil para um algoritmo encontrar soluções razoavelmente próximas da solução ótima. Este facto é um efeito da chamada “praga da dimensionalidade”, que reflete, no pior dos cenários, o aumento exponencial do espaço das soluções admissíveis com o aumento das variáveis de decisão (Bellman, 1957). Assim sendo, limitar a procura no espaço das soluções admissíveis leva a uma melhor eficiência computacional e como resultado, há uma maior probabilidade de encontrar soluções de melhor qualidade. Contudo, em problemas reais, uma redução do espaço de procura⁶ leva, geralmente, à aproximação do espaço das soluções admissíveis, seja porque as variáveis de decisão têm que ser definidas previamente à otimização ou porque a interação entre elas não permite que se procedam a simplificações.

Os métodos que visam reduzir o espaço de procura existentes podem ser divididos em dois grupos, sendo eles os seguintes (Maier *et al.*, 2014):

- Métodos baseados em mudanças no espaço de procura;
- Métodos baseados em mudanças na forma de pesquisa de um algoritmo.

Os primeiros métodos são vulgarmente aplicáveis a algoritmos baseados numa população, isto porque o princípio de alguns destes métodos baseia-se na formação de uma malha de pontos adaptativa, pertencente ao espaço de procura. No início da execução do algoritmo é definida uma malha de pontos “grosseira” na qual se instala a população de soluções, essa malha vai sendo refinada com o decorrer do algoritmo, por forma a tornar-se mais “fina” e poder assim albergar um maior número de soluções possíveis. Alguns exemplos que utilizam esta metodologia são a *Dynamic Parameter Encoding* (Schraudolph & Belew, 1992) e a *Messy GA* (Goldberg *et al.*, 1989).

Os segundos métodos utilizam algum tipo de conhecimento adquirido do problema a resolver por forma a encontrar um “atalho” até à solução ótima. Uma das técnicas usadas é o chamado condicionamento, que não é mais do que a utilização de algumas boas soluções do problema, conhecidas previamente, como soluções iniciais do algoritmo.

⁶ Ambos, o espaço de procura e o espaço das soluções admissíveis, referem-se ao espaço multidimensional onde se encontram as soluções admissíveis de um determinado problema, porém, o espaço de procura está ligado à forma como o algoritmo se movimenta no espaço das soluções admissíveis, não implicando que contenha todas as soluções admissíveis possíveis para esse mesmo problema.

Um dos principais objetivos desta dissertação passa efetivamente pela redução do espaço de procura de um algoritmo através dos segundos métodos. Para tal, é recolhida informação acerca de um problema enquanto o algoritmo está a ser executado por forma a que a mesma seja usada como um indicador da direção a ser seguida pelo algoritmo.

2.3.3 Melhoria da Eficiência Computacional

Nos últimos anos, ambos os campos da Investigação Operacional e da Informática contribuíram de forma significativa para melhorar a eficiência com que os problemas de otimização são resolvidos. Apesar da utilização dos mais recentes avanços informáticos serem uma opção tentadora, existem outras opções para melhorar a eficiência computacional sem ser a aquisição de um computador com uma CPU mais rápida, tais como (Maier *et al.*, 2014):

- O uso de metamodelos (*Surrogate Models*);
- O uso da paralelização;
- Usar informação heurística para reduzir o espaço de procura (já abordado em 2.3.2).

Geralmente, para problemas em que o espaço das soluções admissíveis é extremamente grande, ou que requerem uma resposta rápida (*e.g.* gestão, em tempo real, de albufeiras), ou ainda, em que os modelos utilizados são demasiado complexos, é essencial utilizar metodologias que melhorem a eficiência computacional. A complexidade associada aos fenómenos físicos que regem o movimento da água em condutas combinada com um, possível, espaço das soluções admissíveis muito grande fazem dos problemas de dimensionamento de SDA um dos exemplos em que é necessário melhorar a eficiência computacional.

Metamodelo (ou traduzindo diretamente da expressão original, modelo de substituição) é um modelo aproximado do modelo real, usado geralmente para substituir funções objetivo e/ou restrições que são muito complexas e que consomem uma grande quantidade de recursos computacionais. Existem duas abordagens comumente associadas a metamodelos. Uma das abordagens é usualmente referida como “redução do modelo”, em que o modelo complexo é substituído por um modelo de menor dimensão, onde um subconjunto de variáveis de decisão são projetadas no espaço (como se fossem eliminadas em termos da formulação do problema). Esta abordagem já foi aplicada a diversos problemas de gestão de recursos hídricos como é o caso do trabalho desenvolvido por McPhee & Yeh (2008) na área das águas subterrâneas, ou o trabalho de Castelletti *et al.* (2012) na área do controlo da qualidade da água. A outra abordagem é obtida por aproximação (regressão) a um conjunto de pontos determinados através de simulações feitas com o modelo real. Salienta-se o trabalho desenvolvido por Broad *et al.* (2005), no qual é comparada a resolução de um problema de otimização do dimensionamento de um SDA através do modelo real com a resolução por um metamodelo utilizando esta última abordagem. Aí é constatado que utilizando o metamodelo é possível reduzir o tempo de execução do algoritmo de otimização em valores da ordem das centenas de vezes, sem perder

a qualidade da solução final⁷. O único defeito está relacionado com o tempo despendido para determinar o conjunto de pontos usados para treinar o metamodelo. Por fim, evidencia-se, também, o trabalho de Razavi *et al.* (2012) onde é elaborada uma revisão do desenvolvimento e aplicação de metamodelos a problemas de gestão de recursos hídricos.

O processamento em paralelo, ou em nuvem, permite reduzir o esforço computacional por transmitir parte desse esforço para outros elementos. Existem várias opções disponíveis para o efeito, como sejam, os múltiplos núcleos de processamento, coprocessadores ou até clusters de PC. Estas técnicas são de relativa fácil implementação, existindo *software* disponível para aplicação das mesmas.

2.3.4 Aptidão para Encontrar Soluções Ótimas

As metaheurísticas são geralmente descritas salientando a vantagem de serem métodos prontos a serem aplicados em qualquer tipo de problema. Apesar da veracidade da afirmação anterior, também já aqui foi referido que a escolha do algoritmo e seus parâmetros de controlo influencia significativamente a qualidade das soluções finais encontradas.

É frequente encontrar artigos científicos cujo algoritmo usado para resolver determinado problema é parametrizado por meio de uma sucessão de execuções do tipo tentativa e erro. Sem infligir qualquer demérito a este método, atualmente, parece mais adequado automatizar o processo de seleção e afinação dos parâmetros aquando da execução do algoritmo (e é este o método que ganha cada vez mais seguidores), essa adaptação dos parâmetros pode ser feita com o decorrer do tempo de execução do algoritmo, com o decorrer do progresso até à solução final, ou existe ainda a possibilidade de utilizar, ou não, um determinado operador com base no sucesso anterior para gerar boas soluções (*e.g.* Hadka & Reed, 2013). Pode, ainda, ser usado um conjunto de parâmetros escolhido com base em teorias existentes (*e.g.* Gibbs *et al.*, 2010; Hu & Hu, 2011).

2.3.5 Identificação do Melhor Critério de Convergência

Definir um critério de convergência tem como finalidade decidir quando é que o algoritmo deve parar por forma a evitar iterações desnecessárias e consequente perda de eficiência, garantindo que o algoritmo não converge prematuramente e não se perde qualidade nas soluções finais.

⁷ Para não perder a qualidade da solução final é comparada a solução ótima do metamodelo com a solução correspondente ao modelo real, posteriormente aplica-se um algoritmo de pesquisa local até a solução convergir para o máximo do modelo real, pelo que a qualidade da solução final, é assim mantida.

Nos métodos clássicos, os critérios de convergência são unívocos devido ao facto de que as propriedades analíticas do problema permitem chegar sempre à solução ótima. Tal não é o caso das metaheurísticas, pelo que é necessário definir um critério de convergência aproximado. Esses critérios podem ser classificados da seguinte maneira (Zielinski *et al.*, 2005):

- Critérios baseados na exaustão. O algoritmo termina quando for atingido um limite máximo de avaliações da função objetivo ou de iterações, normalmente dependente do tempo disponível;
- Critérios baseados no progresso. O algoritmo termina quando não se verificam melhorias significativas na função objetivo após um determinado número de iterações;
- Critérios baseados na distribuição (aplicável a algoritmos baseados numa população). O algoritmo termina quando a distância máxima em relação à melhor solução encontrada ou o desvio-padrão da população estiver abaixo de um determinado limite;
- Critério da referência (aplicável a algoritmos baseados numa população). O algoritmo termina quando um certo número de soluções convergirem;
- Critério combinado. Utiliza combinações dos critérios acima mencionados.

A escolha do critério de convergência mais acertado pode parecer um pouco subjetiva, porém, deve ter-se em conta as características do problema e até do próprio algoritmo na definição do mesmo, para além do mais e na falta de mais informação, pode ser útil definir vários critérios de convergência e verificar em que tipos de problemas são melhor sucedidos.

2.3.6 Simplificação do Processo de Decisão Final

É importante, neste ponto, deixar claro que a otimização deve ser usada como uma técnica de apoio à decisão e nunca como um fornecedor de soluções a implementar diretamente num sistema, no mínimo, as soluções obtidas devem ser alvo de uma análise mais ou menos aprofundada, para não permitir que pequenos pormenores ponham em causa todo um sistema. Para além do suporte à decisão, a otimização é, também, uma excelente ferramenta para sistematizar informação e poder, assim, adquirir uma perceção estruturada do que se passa na realidade.

É comum nos problemas de gestão de recursos hídricos existirem objetivos conflitantes, como já aqui foi referenciado (subsecção 2.3.1), que levam a diferentes visões da definição de uma boa decisão. Por esse motivo, é importante facultar ao decisor as ferramentas necessárias para que ele possa interpretar da melhor forma os resultados obtidos através da otimização e, assim, tomar decisões que representem o melhor compromisso entre os objetivos do problema e as diversas partes interessadas.

O decisor pode definir as suas preferências de otimização *a priori* ou *a posteriori* do próprio processo de otimização. Quando estas são definidas *a priori*, a estrutura do modelo de

otimização é previamente definida com base nas preferências do decisor, pelo que a solução final reflete essas mesmas preferências. Porém, o que muitas vezes é observado na prática é que “os decisores não sabem o que querem enquanto não souberem aquilo que podem ter” (adaptado de Loucks, 2012). Com vista à transposição desse obstáculo, também é possível ao decisor definir as suas preferências numa abordagem *a posteriori* (apesar da inerente relatividade envolvida), para tal é necessário determinar a fronteira de Pareto, o que pode levar a dois problemas principais:

- Aumento do esforço computacional (que pode ser aliviado com as medidas propostas nas subsecções prévias);
- Dificuldade na visualização e interpretação do conjunto de soluções definidas pela fronteira de Pareto⁸.

2.3.7 Incorporação da Incerteza

A incerteza está presente em todo o processo de otimização da gestão de recursos hídricos, desde a formulação do problema até às soluções obtidas. Segundo Mays (1996) a incerteza pode ser definida como a ocorrência de eventos que estão para além do nosso controlo. No contexto em que se insere esta dissertação, a incerteza pode advir das mais diversas origens, tais como:

- Fatores naturais a curto prazo, como a quantidade de água disponível numa fonte, influenciada pela variabilidade da precipitação (temporal e espacial);
- Fatores naturais a longo prazo, como as, importantes, alterações climáticas;
- Eventos extremos, não apenas desastres naturais (incêndios, sismos, furacões, ...) mas também eventos como atentados (11 de setembro, 2001) ou acidentes (Central Química de Jilin, 2005)⁹;
- Fatores socioeconómicos, como o consumo de água *per capita* de uma comunidade e a sua futura evolução;
- Fatores políticos, como uma possível mudança da legislação existente.

Para além destas, outro tipo de incertezas decorrentes da modelação dos processos físicos (*e.g.* leis de resistência) ou da análise de dados (*e.g.* perdas em condutas) devem ser tidas em conta. Note-se que, a previsão de acontecimentos futuros está sempre associada a incerteza que será tanto maior quanto maior o horizonte de previsão estipulado.

Existem inúmeras outras fontes de incerteza de maior ou menor relevância dependentes de cada caso. Parece correto dizer, neste contexto, para “esperar o inesperado”. O preço a pagar pela

⁸ Estudos mostram que esta barreira pode ser mitigada com o uso de estruturas de suporte visual tais como métodos gráficos com incorporação de um código de cores ou formas para problemas com um número de dimensões elevado (Kollat & Reed, 2007), ou até com a projeção de alguns objetivos envolvidos no problema.

⁹ A respeito de eventos extremos refira-se que apesar da sua reduzida probabilidade de ocorrência, os prejuízos decorrentes destes são avultados, pelo que não podem simplesmente serem desprezados.

não consideração da incerteza pode variar desde resultados com soluções subótimas até mesmo soluções impraticáveis.

As fontes de incerteza frequentemente interagem entre si e são de elevado grau de complexidade, pelo que se torna muito difícil, senão mesmo impossível, caracterizar todas elas no processo de otimização. Quando considerada é geralmente de forma implícita, através de abordagens reativas, como por exemplo, através de um número limitado de cenários, contudo, ultimamente tem-se utilizado modelos estocásticos (*e.g.* Método de Monte Carlo), juntamente com diversos cenários, para um número restrito de fontes de incerteza chaves na tentativa de caracterizar explicitamente a incerteza. Algumas críticas têm sido apontadas a esta alternativa, seja por em muitos casos não se disporem de dados que permitam caracterizar estocasticamente as mais importantes fontes de incerteza, ou seja pelas simplificações efetuadas a nível temporal e/ou espacial, contudo, estas suposições também são válidas para outras abordagens existentes. A forma mais aceite pela comunidade científica de incorporar a utilização de modelos estocásticos em problemas de otimização, é aquela em que é introduzido no modelo uma medida de robustez (ou de fiabilidade, risco, *etc.*) como indicador do desempenho do sistema, respeitando assim os princípios de abordagem proativa. A respeito dos SDA é de notar que estes têm sido concebidos através de abordagens determinísticas ao usar valores de referência para caracterizar certos tipos de incerteza e considerar que algumas falhas poderiam surgir no decorrer da vida útil do sistema. Segundo Cunha & Sousa (2010) este tipo de abordagens reativas deve ser progressivamente trocado por abordagens proativas que incluam a incerteza desde o início do processo de decisão.

Salienta-se, novamente, a importância do papel que o decisor desempenha. Com a incorporação da incerteza nos problemas reais de recursos hídricos é espetável que exista um espectro de soluções muito mais flexível, pelo que cabe ao decisor determinar quanto está disposto a agravar os custos de um sistema para diminuir a sua vulnerabilidade, ou vice-versa.

2.3.8 Implementação do Algoritmo

A implementação do algoritmo através de um código e ligação com o modelo de simulação pode ser feita de diversas formas, sendo importante realçar que a eficiência computacional depende da facilidade com que é implementado o código. A implementação dos algoritmos é comumente feita através dos seguintes métodos (Maier *et al.*, 2014):

- Linguagem de programação (*e.g.* C++; Java);
- Bibliotecas compartilhadas com API (*e.g.* DLL);
- Pacotes e *Toolboxes* escritos para Python, MatLab, *etc.*;
- Macros e extensões para folhas de cálculo.

Pelo facto da comunidade de utilizadores dos diversos algoritmos de otimização estar dispersa, continua a utilizar-se algoritmos orientados para fins específicos de forma pouco objetiva, no entanto, recentemente tem existido alguns esforços para tentar catalogar os diversos métodos existentes numa tentativa de padronizar a implementação dos diferentes algoritmos, muitas vezes dificultada pelo facto de que a complexidade dos problemas reais não deixar outra solução senão implementar modificações *ad-hoc* no método escolhido.

2.4 Revisão de Heurísticas de Otimização

Por forma a uma boa compreensão do trabalho desenvolvido nos capítulos posteriores, são expostos, nesta secção, os fundamentos da pesquisa local e são revistas algumas das suas aplicações em metaheurísticas largamente utilizadas pela comunidade científica.

2.4.1 Heurísticas de Pesquisa Local

A pesquisa local é um conceito muito utilizado em otimização, infelizmente, a sua definição não é de todo consensual entre a comunidade científica, o que leva a que cada autor tenha a sua própria visão do que é a pesquisa local. No âmbito desta dissertação, considere-se que a pesquisa local diz respeito, apenas, a métodos que utilizem uma vizinhança da solução¹⁰ e não permitam a escolha de soluções piores no decorrer do mesmo (frequentemente chamados de algoritmos gulosos), ou seja, apesar de métodos como o SA e a TS utilizarem os conceitos de pesquisa local, estes são vistos como (meta-)heurísticas de pesquisa global por permitirem a escolha de soluções com um pior custo (ou benefício).

Seja dado um problema de otimização ($\min F(x)$ ou $\max F(x)$), com um determinado espaço de soluções admissíveis (X) e onde a cada solução ($x : x \in X$) corresponde um determinado custo ($F(x)$) (ou benefício se o problema for de maximização), seja, também, definida uma vizinhança da solução em questão ($V(x) : V(x) \subseteq X$)¹⁰. Partindo de uma solução ($x_t = i$), a pesquisa local pode ser sumarizada como um procedimento iterativo, onde em cada iteração é escolhida uma nova solução pertencente à vizinhança da primeira solução de acordo com regras bem estabelecidas, mas é escolhida de tal forma que haja uma melhoria no custo (ou benefício), ou pelo menos, que a nova solução não seja pior que a sua antecessora.

¹⁰ Geralmente, a estrutura de uma vizinhança é obtida por aplicação de pequenas alterações na solução à qual se refere a vizinhança, chamadas de alterações elementares (Pirlot, 1996). Tomando como exemplo um problema de dimensionamento de uma rede de distribuição de água: caso a variável de decisão em causa seja um diâmetro de uma tubagem (variável discreta), então, uma possível alteração elementar passa por considerar o diâmetro comercial imediatamente acima ou abaixo do atual; caso a variável de decisão em causa seja a altura piezométrica num nó (variável contínua), então, uma possível alteração elementar passa por aplicar um pequeno incremento de pressão à pressão já existente ($h \pm \Delta h$).

Existem duas formas típicas de abordar a pesquisa local. A primeira, denominada de *iterative improvement*, escolhe aleatoriamente uma solução candidata da vizinhança e se houver uma melhoria no custo em relação à solução atual ($F(j) < F(i)$) então a solução atual é atualizada ($x_{t+1} \leftarrow j$), caso contrário mantém-se ($x_{t+1} \leftarrow i$), e o processo repete-se até que não haja melhorias significativas (esta abordagem não tem em conta toda a vizinhança). A segunda, denominada de *steepest descent*, escolhe a solução da vizinhança que retorna um menor custo para o problema, ou mantém a solução atual caso o custo associado a todos os seus vizinhos seja maior.

Como é facilmente perceptível, a pesquisa local por si só, geralmente, não leva a soluções de grande qualidade, ficando rapidamente retida em ótimos locais que apenas excepcionalmente coincidiriam com o ótimo global. Uma observação do Gráfico 2.1 demonstra isso mesmo, nele está representado um problema de otimização hipotético onde se pretende minimizar o custo associado a uma só variável de decisão. Ao aplicar uma das abordagens de pesquisa local ao problema rapidamente se encontraria a solução x' , que é um mínimo local, contudo, a solução pretendida, x^* , não pode ser encontrada prosseguindo com a mesma abordagem. Com vista a ultrapassar esta dificuldade são introduzidas novas componentes (*e.g.* aleatoriedade, memória, ...) nos algoritmos originais, por forma a que estes possam escapar dos ótimos locais e acima de tudo, para diversificar a pesquisa, não só intensificando-a. Ao longo do tempo foram-se explorando e melhorando estas ideias para dar forma às metaheurísticas que conhecemos hoje em dia (*e.g.* SA, TS, VNS (Pesquisa em Vizinhanças Variáveis), GRASP (Procedimento de Pesquisa Guloso, Aleatório e Adaptativo), GLS (Pesquisa Local Guiada), ILS (Pesquisa Local Iterativa), ...).

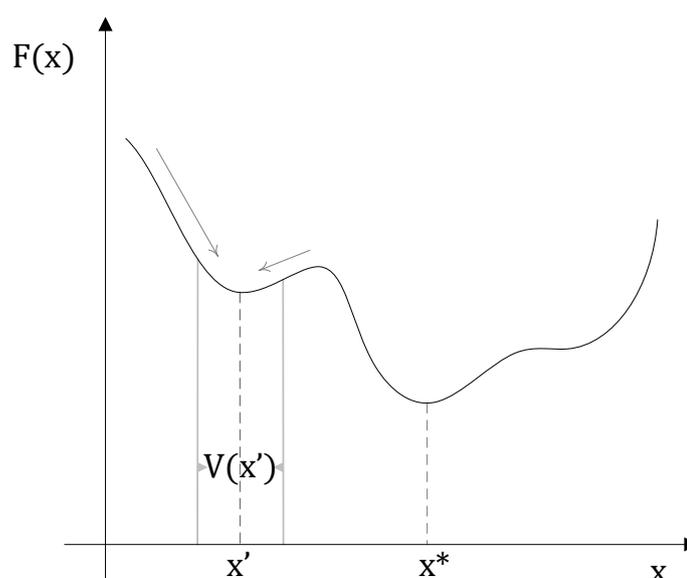


Gráfico 2.1 - Problema de otimização hipotético

2.4.2 Recozimento Simulado

O termo *Simulated Annealing* foi pela primeira vez introduzido por Kirkpatrick *et al.* (1983) e formulado independentemente pelos mesmos autores e por Černý (1985). A teoria por detrás deste algoritmo baseia-se na analogia para com o processo de recozimento de metais, onde a temperatura de um metal é aumentada até um pouco abaixo da sua temperatura de fusão e de seguida o metal é arrefecido lentamente, por forma a que a sua estrutura atómica se vá rearranjando, até que no final do processo a estrutura está de tal forma arranjada que a energia do metal é mínima.

Segundo o critério de Metropolis *et al.* (1953), o processo de recozimento pode ser simulado através de uma sequência de estados, em que um estado j é gerado a partir de um estado i por uma pequena alteração no sistema e é sempre aceite caso conduza a um decréscimo na energia do sistema, ou então é aceite com uma probabilidade não nula definida por $\exp\left(\frac{E_i - E_j}{k_B \cdot T}\right)$, onde E_i e E_j são as energias do sistema para os estado i e j respetivamente, k_B é a constante de Boltzmann e T é a temperatura a que se encontra o sistema.

Analogamente, no problema real os estados correspondem a soluções e a energia do sistema num dado estado corresponde ao custo da respetiva solução. A temperatura é vista no problema real como um parâmetro de controlo do qual vai depender a qualidade da solução final. O critério de admissibilidade de Metropolis pode ser traduzido da seguinte forma para o problema real (adaptação de Aarts *et al.*, 1997)

$$P_t\{\text{aceitar } j\} = \begin{cases} 1 & , \text{ se } F(j) \leq F(i); \\ \exp\left(\frac{F(i) - F(j)}{t}\right) & , \text{ se } F(j) > F(i). \end{cases}$$

Existem algumas características básicas do algoritmo, definidas pelo utilizador, que têm influência direta no desempenho do mesmo. Estas, geralmente, diferem de problema para problema e devem ser alvo de um estudo cuidadoso. São elas as seguintes:

- A temperatura a partir da qual o algoritmo é iniciado, t_0 ;
- O número de iterações necessárias para o algoritmo convergir para uma distribuição estacionária associada à cadeia de Markov numa dada temperatura (Aarts *et al.*, 1997);
- O critério de arrefecimento. Habitualmente é usado o critério geométrico que consiste em definir um taxa de arrefecimento, α , de valores típicos entre 0.8 e 0.99 (Aarts *et al.*, 1997), de tal forma que

$$t_{k+1} = \alpha \cdot t_k \Leftrightarrow t_k = \alpha^k \cdot t_0;$$

- O(s) critério(s) de convergência (ver subsecção 2.3.5).

2.4.3 Pesquisa Tabu

Foi Fred Glover quem propôs o uso de uma nova metaheurística, inovadora na altura, que combinava a pesquisa local com estruturas de memória adaptativa (Glover, 1986, 1989, 1990). A função dessas estruturas de memória é a de não permitir que o algoritmo fique estagnado num ótimo local ao proibir certos movimentos entre soluções de serem realizados. Para além disso, devem assegurar, como já foi referido na subsecção 2.4.1, que a diversificação do espaço de procura é garantida.

Tome-se, como exemplo, o problema hipotético descrito anteriormente. Imagine-se que a estrutura da vizinhança é simétrica¹¹ (*i.e.*, $i \in V(j) \wedge j \in V(i), \forall i, j \setminus \{i = j\} \in X$). A Pesquisa Tabu aceita sempre uma nova solução da vizinhança, mesmo que isso implique um aumento do custo, isto poderia levar a um ciclo interminável ($x_t \leftarrow i; x_{t+1} \leftarrow j; x_{t+2} \leftarrow i; x_{t+3} \leftarrow j; \dots$), contudo, se se impedir que o algoritmo retorne à solução antecessora, por introdução desse movimento numa lista tabu (lista de movimentos proibidos), é possível escapar de ótimos locais e prosseguir a busca no restante espaço de procura.

Esta pode ser vista como a função básica das listas tabu. Porém, se toda a informação relativa a um movimento for armazenada numa entrada da lista tabu, para além da garantia da não existência de movimentos cíclicos não ser verificada, pode-se dar o caso do algoritmo tornar-se ineficiente por perder muito tempo a analisar a admissibilidade de cada movimento e necessitar de uma grande capacidade de memória (Pirlot, 1996). Para contornar esta desvantagem, é normal não armazenar toda a informação, mas apenas alguns atributos de um determinado movimento¹².

Infelizmente, este último procedimento pode, por sua vez, ser demasiado limitativo na medida em que o algoritmo pode restringir movimentos para soluções que realmente trariam uma redução do custo final, ou, em casos extremos, o algoritmo pode mesmo chegar a estagnar (quando todos os possíveis movimentos a partir de uma solução estão incluídos na lista tabu). Como forma de ultrapassar este problema, são criados critérios de aspiração que não são mais do que contratabus por permitirem que certos movimentos, que gerem soluções suficientemente boas, sejam realizados, independentemente de estarem na lista tabu. Existe, como se depreende, uma certa liberdade na escolha destes critérios devido à indefinição de “soluções

¹¹ Não é uma condição estritamente necessária.

¹² Tome-se novamente o exemplo do dimensionamento de uma rede de distribuição de água. Imagine-se que o movimento da solução i para a solução j é definida por alteração do diâmetro de uma conduta de Φ_1 para Φ_2 e por alteração de uma altura piezométrica de p_1 para p_2 . Esse movimento pode ser descrito pelos seguintes atributos (1,2,1,2). O correspondente movimento a ser restringido é (2,1,2,1). Contudo, se assim o desejar, é possível seleccionar apenas alguns desses atributos a colocar na lista tabu, como por exemplo, (2,1,0,0), que corresponde a impedir todos os movimentos que alterem o diâmetro da conduta de Φ_2 para Φ_1 , independentemente da altura piezométrica.

suficientemente boas”, todavia, como foi referido, estes critérios são muito importantes para garantir a qualidade da solução final devendo, por esse motivo, ser criteriosamente escolhidos.

Tal como acontece no SA, existem diversas características básicas do algoritmo, definidas pelo utilizador, que têm influência direta no desempenho do mesmo. Descrevem-se as fundamentais seguidamente:

- A definição dos atributos associados a cada movimento a serem armazenados nas entradas da lista tabu;
- A dimensão da lista tabu. Refira-se a possibilidade de alterar a dimensão com o decorrer do algoritmo;
- A definição dos critérios de aspiração;
- O(s) critério(s) de convergência.

2.4.4 Pesquisa em Vizinhanças Variáveis

A VNS surgiu quando Mladenović & Hansen (1997) descobriram que “a mudança sistemática da vizinhança acompanhada de uma das abordagens de pesquisa local permitia obter uma nova metaheurística simples e eficaz”¹³.

De facto, na versão básica da VNS, definem-se um número finito de vizinhanças para uma solução ($V_k(x): V_k(x) \in X, k = 1, \dots, k_{máx}$), em contraste com uma só vizinhança usado por outras metaheurísticas. A melhor forma de compreender o algoritmo é através da exposição do procedimento básico adotado pelo mesmo, como se faz de seguida (adaptação de Hansen & Mladenović, 2001):

0. Definem-se as estruturas das $k_{máx}$ vizinhanças. Inicia-se a contagem de k em 1. Até k atingir $k_{máx}$ procede-se do seguinte modo;
1. Define-se uma solução aleatória (j_k) pertencente à k -ésima vizinhança de uma solução inicial ($x_t = i$);
2. A partir da solução aleatória (j_k) aplica-se uma abordagem de pesquisa local à escolha;
3. Agora existem duas opções:
 - 3.1. Se a solução gerada (j'_k) for melhor que a inicial ($F(j'_k) < F(i)$, se o problema for de minimização), então atualiza-se a nova solução inicial ($x_{t+1} \leftarrow j'_k$) e a pesquisa recomeça no ponto 1, definindo o contador de k novamente para 1 ($k \leftarrow 1$);
 - 3.2. Caso a solução gerada não seja melhor que a inicial, então a solução inicial é mantida ($x_{t+1} \leftarrow i$) e a pesquisa retoma no ponto 1, atualizando o contador de k ($k \leftarrow k + 1$).

¹³ Adaptação do autor de Mladenović & Hansen (1997).

O facto de se usar uma solução aleatória (j_k) previne o aparecimento de ciclos no decorrer do algoritmo que poderiam existir caso fosse utilizada alguma regra determinística (Hansen & Mladenovi, 2001). Refira-se, ainda, que a VNS é uma metaheurística bastante flexível, existindo até ao momento diversas variantes da mesma, com a sua eficiência comprovada.

Para a VNS, as principais características a definir são:

- O parâmetro $k_{m\acute{a}x}$;
- A estrutura das $k_{m\acute{a}x}$ vizinhanças e a sua ordenação;
- O(s) critério(s) de convergência.

2.4.5 Método da Entropia Cruzada

Em contraste com as metaheurísticas referidas anteriormente, a CE (Rubinstein, 1997) é um método que se enquadra no grupo de metaheurísticas baseadas numa população. O seu objetivo inicial era o de estimar com precisão a probabilidade de ocorrência de eventos extremos, mais tarde verificou-se que esta metodologia também poderia ser aplicada com sucesso a problemas de otimização por transformação do problema determinístico num problema estocástico.

A CE pode ser estruturada num ciclo de dois passos essenciais. O primeiro diz respeito à geração de uma amostra de soluções de acordo com um determinado mecanismo (distribuição probabilística). O segundo diz respeito à atualização do mecanismo tendo por base a informação recolhida da amostra para que na próxima iteração a amostra gerada seja melhor.

Para estimar uma probabilidade que se sabe ser muito pequena, não é de todo eficiente simular o sistema em condições, ditas, normais. Para tal, a CE utiliza uma técnica de redução da variância denominada de “amostragem por importância” em que é utilizada uma distribuição probabilística diferente da original para gerar amostras com uma maior probabilidade de ocorrência de eventos extremos (de Boer *et al.*, 2005).

Para problemas de otimização combinatoria, aplicado ao dimensionamento de SDA (ver cap.3), a CE envolve o seguinte procedimento (adaptado de Perelman & Ostfeld, 2007):

0. Definição da matriz de probabilidades inicial,

$$p_{ij}^0, \forall i \in \{1, \dots, ll\} \wedge j \in \{1, \dots, mm\},$$

onde ll é o número de diâmetros comerciais disponíveis e mm é o número de condutas a dimensionar. p_{ij}^t representa a probabilidade de se instalar o diâmetro comercial i na conduta j à t -ésima iteração. Sem outras informações que digam o contrário, é comum utilizar a distribuição uniforme de probabilidades para a iteração inicial, que corresponde a atribuir a mesma probabilidade a todos os diâmetros comerciais, ou seja

$$p_{ij}^0 = 1/ll, \quad \forall i \in \{1, \dots, ll\} \wedge j \in \{1, \dots, mm\};$$

1. Gerar uma amostra de soluções admissíveis de tamanho N ($x_1, \dots, x_k, \dots, x_N$) a partir da matriz de probabilidades p_{ij}^t ;
2. Calcular a função objetivo das N soluções da amostra ($F(x_1), \dots, F(x_k), \dots, F(x_N)$) e ordená-las por ordem crescente

$$F_1(x) \leq \dots \leq F_k(x) \leq \dots \leq F_N(x);$$

3. Escolher um subconjunto da amostra que contenha as melhores soluções. Tal pode ser feito a partir de uma percentagem ρ da amostra ($F_1(x), \dots, F_k(x), \dots, F_{\rho \cdot N}(x)$);
4. Atualizar a matriz de probabilidades com base no subconjunto de melhores soluções. Tal, pode ser feito através do seguinte estimador:

$$\hat{p}_{ij}^{t+1} = \frac{\sum_k^{\rho \cdot N} I_{jk}(\Phi_i)}{\rho \cdot N}, \quad \forall i \in \{1, \dots, ll\} \wedge j \in \{1, \dots, mm\},$$

onde $I_{jk}(\Phi_i)$ é uma função que depende do diâmetro comercial i e toma o valor de 1 se na solução k o próprio estiver instalado na conduta j ou toma o valor nulo caso contrário;

5. Em problemas de otimização combinatórios, é prática comum aplicar um coeficiente de suavização (β) no estimador \hat{p}_{ij} . Desta forma é, como que, introduzida uma certa inércia à atualização da matriz por forma a impedir que o método convirja precocemente para ótimos locais. Tal, pode ser representado da seguinte forma:

$$p_{ij}^{t+1} = \beta \cdot \hat{p}_{ij}^{t+1} + (1 - \beta) \cdot p_{ij}^t;$$

6. Se forem verificados os critérios de convergência, então a melhor solução é a que corresponde um valor da função objetivo igual a $F_1(x)$ e o método termina. Caso contrário, incrementa-se o contador de iterações em uma unidade ($t \leftarrow t + 1$) e recomeça-se o procedimento no passo 1.

Para a CE, as principais características a definir são:

- O tamanho da amostra, N ;
- Os parâmetros ρ e β , ambos definidos entre 0 e 1;
- O(s) critério(s) de convergência.

2.4.6 Algoritmos Genéticos

Certamente das metaheurísticas mais conhecidas hoje em dia, os GA pertencem ao grupo de metaheurísticas baseadas numa população e a sua conceção deve-se, em grande parte, a John Holland, que publicou as suas ideias no livro *Adaptation in Natural and Artificial Systems* (Holland, 1975). O principal objetivo de Holland era reproduzir, de forma análoga, o processo de seleção natural darwiniano que é comum a todas as espécies de seres vivos.

Uma das razões que torna os GA extremamente populares entre a comunidade científica é o facto de possibilitarem a representação de um problema de otimização em termos e denominações biológicas de forma bastante precisa. Os GA distinguem-se de outras metaheurísticas pelo facto de usarem uma representação codificada do conjunto de variáveis de decisão que compõem um problema de otimização. Essa representação codificada das variáveis de decisão é vista como o genótipo/cromossoma de um indivíduo (indivíduo esse que é visto como uma solução distinta do problema de otimização a resolver) enquanto que a representação natural é vista como o fenótipo. A representação codificada é composta por uma sequência de caracteres (geralmente números binários) onde cada elemento da sequência é visto como um gene e os caracteres possíveis para esse elemento os alelos.

A estratégia de otimização das GA é composta por um ciclo de 4 passos fundamentais (Sivanandam & Deepa, 2008). A seleção, onde geralmente são escolhidos para reprodução os indivíduos (soluções) que apresentarem a melhor aptidão para sobreviver (valor da função objetivo). A reprodução, onde são gerados novos cromossomas a partir da recombinação dos cromossomas dos indivíduos selecionados anteriormente. A avaliação, onde é avaliada a aptidão de sobrevivência dos novos indivíduos gerados. Por fim, a reposição, onde os elementos mais velhos e menos aptos são substituídos pelos novos elementos gerados.

Existem diversas formas de proceder à seleção de indivíduos, mas a forma mais comum é baseada numa probabilidade de escolha que é diretamente proporcional à aptidão de sobrevivência do indivíduo. Para efetuar a reprodução são geralmente usados dois operadores básicos bastante conhecidos do campo da genética, são eles (Sivanandam & Deepa, 2008):

- O cruzamento, onde o cromossoma descendente contém partes inalteradas dos cromossomas parentes. A forma mais comum de realizar este procedimento é através do cruzamento em um ponto, nesta situação o cromossoma de um parente é quebrado a determinada altura e é repostado pelo cromossoma do outro parente;
- A mutação, onde um ou mais genes de um cromossoma são alterados de forma aleatória. A forma mais comum de realizar este procedimento é através da utilização de uma distribuição de Bernoulli para cada gene (geralmente com uma probabilidade baixa).

Tal como a seleção, também existem diversas formas de proceder à reposição de uma população, uma das formas recorrentes consiste em repor parte da população “menos apta” pela nova geração de indivíduos criada (conceito que pode ser comparado ao de lacuna geracional).

Resumindo a principal informação acerca dos GA, expõe-se de seguida a metodologia a utilizar para resolver problemas de otimização:

0. Criar uma população de cromossomas inicial e avaliar a sua aptidão;
1. Criar novos cromossomas, repetindo este passo, até que parte da população possa ser reposta:
 - 1.1. Seleção de 2 ou mais cromossomas parentes para reprodução;
 - 1.2. Cruzamento dos cromossomas parentes;
 - 1.3. Mutação dos cromossomas descendentes;
 - 1.4. Avaliação da aptidão dos novos cromossomas;
 - 1.5. Acolhimento dos novos cromossomas na população;
2. Reposição de parte da população para dar entrada aos novos cromossomas descendentes;
3. Se forem verificados os critérios de convergência, então é indicado qual o cromossoma mais apto (solução quase-ótima) e o seu valor de aptidão (valor da função objetivo). Caso contrário, o processo recomeça no passo 1.

Para os GA, as principais características a definir são:

- A forma de proceder à seleção e cruzamento parental;
- A probabilidade de mutação da descendência e a sua introdução na população;
- O(s) critério(s) de convergência.

2.4.7 Hibridização de Metaheurísticas

Inevitavelmente, com o aparecimento de inúmeras novas metaheurísticas aparece, também, uma tendência para ligar os conceitos das mesmas entre si. O principal objetivo é o de tirar partido das vantagens referentes a cada metaheurística e, se possível, reduzir as desvantagens (as correntemente chamadas sinergias).

As metaheurísticas daí resultantes são referidas como metaheurísticas híbridas¹⁴, e o seu aparecimento data de finais da década de 80, mais concretamente com o aparecimento dos algoritmos meméticos (Moscato, 1989).

No início, a comunidade científica, na generalidade, mostrou-se cética a esta abordagem, contudo, as metaheurísticas híbridas começaram a ganhar mais seguidores quando estes se aperceberam que não era possível que um algoritmo resolvesse todos os problemas de forma mais eficiente do que qualquer outro (Blum *et al.*, 2011). Hoje em dia, é largamente aceite pela comunidade científica que o futuro dos métodos de otimização modernos passa pela mistura/hibridização de diferentes componentes dos diversos algoritmos no seu estado “puro”.

¹⁴ As metaheurísticas híbridas não incluem necessariamente conceitos exclusivos das diversas (meta-)heurísticas como se verá na classificação proposta para este tipo de metaheurísticas.

Uma possível hibridização, simples e bastante eficiente, é conseguida ao introduzir uma das abordagens de pesquisa local diretamente nos algoritmos de otimização. Com isto, pretende-se acrescentar uma componente adicional de intensificação (devendo-se reduzir a respetiva componente no algoritmo inicial, regulada através dos parâmetros de controlo, por forma a obter uma melhor eficiência). Desta forma, obtém-se um algoritmo híbrido com as componentes de diversificação e intensificação bem estruturadas. No grupo de metaheurísticas baseadas numa população, os algoritmos meméticos anteriormente referidos, são um exemplo de aplicação bem-sucedido deste conceito. No grupo de metaheurísticas baseadas numa trajetória também é possível aplicar estes conceitos (*e.g.* Zeferino *et al.*, 2009). Em ambos os casos, a principal função do algoritmo original é encarregar-se de diversificar a pesquisa, por forma a encontrar áreas do espaço de procura promissoras e seguidamente a componente de pesquisa local encarregar-se-á de intensificar a pesquisa nessas áreas, de maneira a encontrar soluções quase-ótimas. Este método vai ao encontro do que é pretendido com a redução do espaço de procura (subsecção 2.3.2). Um exemplo gráfico, hipotético e representativo das considerações expostas é apresentado no Gráfico 2.2.

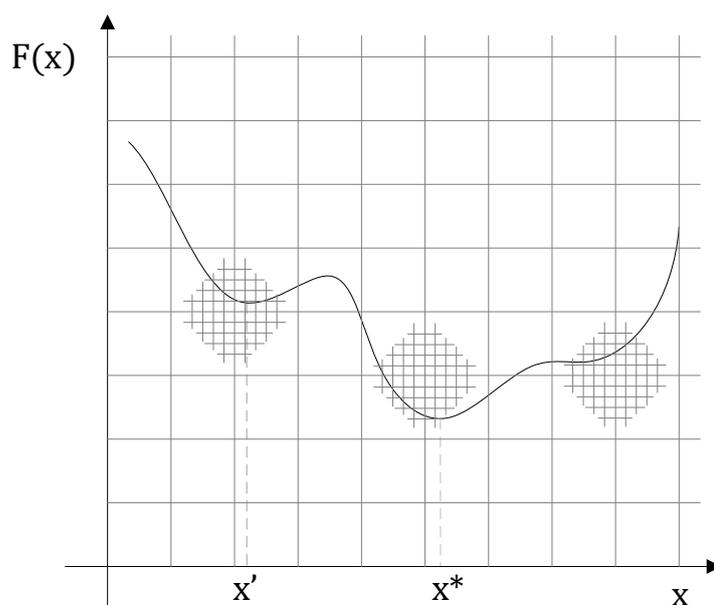


Gráfico 2.2 - Exemplo hipotético do espaço de procura de um algoritmo híbrido obtido por adição de um método de pesquisa local

Focando naquela que é a principal metaheurística utilizada nesta dissertação, podem ser referidos diversos exemplos onde os conceitos do SA foram combinados com outros métodos.

Lenin *et al.* (2016) utilizaram os conceitos do SA e da TS para criar um algoritmo que resolvia o problema de minimização da potência reativa em sistemas elétricos, não só obtiveram melhores resultados, como verificaram que o algoritmo híbrido convergia mais rapidamente que os algoritmos originais.

Xiao *et al.* (2014) propõem uma variação à VNS, utilizando o critério de Metropolis do SA para determinar se o algoritmo deve aceitar uma nova solução do problema de roteamento de veículos de transporte de mercadorias, aí é constatado que o algoritmo é especialmente eficaz na resolução de problemas de grandes dimensões quando comparados com outros algoritmos especializados na resolução de problemas de roteamento de veículos.

Niknam *et al.* (2013) combinaram a PSO (Otimização por Enxame de Partículas) com o SA para tentar ultrapassar o facto de que o problema do fluxo de carga dinâmico ótimo, novamente em sistemas elétricos, tem um número elevado de ótimos locais, os resultados obtidos validam a aplicabilidade do algoritmo.

Xiao *et al.* (2015) utilizaram o SA para intensificar a pesquisa na solução obtida pela técnica de relaxação Lagrangeana, o problema estudado é o de sequenciamento e dimensionamento de lotes no planeamento da produção industrial, os resultados do algoritmo composto superam os obtidos pelos métodos individualmente.

Bahrami *et al.* (2016) utilizaram o SA como forma de ajudar a treinar uma ANN (Rede Neural Artificial) a prever os caudais de extração de água subterrânea num poço de uma mina a céu aberto. Neste caso, o objetivo era criar uma alternativa aos métodos numéricos existentes que envolvem demasiados parâmetros a ser estimados. Os resultados indicam que o método híbrido aproximado apresenta erros bastante pequenos relativamente ao método numérico utilizado e pode, por isso, ser uma alternativa viável aos métodos numéricos existentes quando há falta de dados hidrogeológicos ou quando são necessárias intervenções rápidas.

Note-se a versatilidade do SA no que diz respeito a métodos hibridizados, tanto pode ser usado como medida de intensificação do espaço de procura, como medida de diversificação por aceitar soluções de pior custo.

Os limites são vastos no que diz respeito à hibridização de metaheurísticas, para além dos exemplos fornecidos, muitos outros poderiam ser dados. Atualmente, chegámos a um patamar em que, eventualmente, são testadas hibridizações de métodos híbridos.

Devido à liberdade na seleção das inúmeras opções de hibridização disponíveis, torna-se imperativo categorizar as mesmas por forma a simplificar o processo de seleção do algoritmo híbrido adequado a um determinado problema. Várias classificações já foram propostas por diversos autores (algumas vezes em discordância, como seria de esperar). A classificação aqui exposta é uma adaptação da classificação inicialmente proposta por Raidl (2006). Segundo o mesmo autor, as metaheurísticas híbridas podem ser classificadas de acordo com quatro critérios. Por sua vez, cada critério está dividido em grupos que são resumidos seguidamente:

- Tipo de algoritmos hibridizados:
 - Metaheurísticas com (meta-)heurísticas (algoritmo proposto);
 - Metaheurísticas com modelos de simulação (*e.g.* Yan & Wang, 2007);
 - Metaheurísticas com técnicas de Investigação Operacional/Inteligência Artificial (*e.g.* GA/GRG Tu *et al.*, 2005);
 - Metaheurísticas com interação humana (ver Klau *et al.*, 2010);
- Nível de hibridização:
 - Alto-nível (acoplamento fraco), ou seja, os algoritmos hibridizados trabalham quase independentemente;
 - Baixo-nível (acoplamento forte), ou seja, existe uma forte dependência entre os algoritmos hibridizados (algoritmo proposto);
- Ordem de execução:
 - Sequencial, ou seja, os algoritmos hibridizados são ativados sequencialmente (os resultados de um algoritmo são os dados de entrada de outro);
 - Entrelaçada, ou seja, nem completamente sequencial nem completamente paralela;
 - Paralela, ou seja, há informação constantemente a ser trocada entre os algoritmos hibridizados (algoritmo proposto);
- Estratégia de controlo:
 - Integrativa, ou seja, um dos algoritmos controla o outro (algoritmo proposto);
 - Colaborativa, ou seja, os algoritmos trocam informação, mas nenhum deles é dominado por qualquer outro.

As metaheurísticas híbridas têm as suas vantagens e desvantagens, sendo que a mais comum das desvantagens é a, inevitável, complexidade adicional. Esta complexidade adicional permite, porém, na generalidade dos casos, efetuar um *trade-off* entre a eficiência e a eficácia do algoritmo, ou seja, estes métodos devem ser considerados quando muito boas soluções são requeridas e não há problemas de se perder um pouco mais de tempo para as encontrar. Quando o tempo é limitante, um método “puro” deverá ser suficiente para encontrar soluções razoavelmente boas (Blum *et al.*, 2011). Contudo, demonstrou-se que a utilização destas hibridizações é muitas vezes a melhor opção e a tendência é a de que serão cada vez mais utilizadas no futuro.

Apesar de não ir diretamente ao encontro da temática das metaheurísticas híbridas, é interessante referir uma metodologia recente que tem crescido e progredido na área da otimização e da ciência computacional. As hiper-heurísticas são estratégias de otimização que trabalham não com um espaço de procura de soluções de um problema, mas sim com um espaço de procura de (meta-)heurísticas. É uma nova abordagem que tem por objetivo generalizar o processo de aplicação de uma (meta-)heurística adequada a um determinado problema.

3 FORMULAÇÃO DO MODELO DE OTIMIZAÇÃO

A conceção de um SDA completo normalmente envolve a realização de 4 passos sequencialmente, cada um visto como um subproblema específico. São eles os seguintes:

- Configuração (*layout*). Neste passo os elementos do SDA são dispostos no projeto tendo em conta a topografia do terreno em que se insere e tendo em conta as zonas que por ele devem ser servidas;
- Dimensionamento (*design*). Neste passo os elementos do SDA são dimensionados tendo em conta os níveis de qualidade do serviço pretendidos pelo utilizador final;
- Operação/Manutenção. Neste passo é definida a forma de operar os diversos elementos do SDA. Por exemplo, são definidos os períodos de funcionamento das estações elevatórias, é definida a periodicidade com que são efetuadas vistorias de controlo do SDA, *etc.*;
- Expansão/Reabilitação. Possivelmente, a imprevisibilidade do futuro fará com que seja necessário expandir a rede para ter em conta, por exemplo, novos focos populacionais ou uma alteração dos consumos. Para além disso, o desgaste inevitável de alguns elementos do SDA pode levar a uma intervenção nos mesmos, havendo diversas formas para proceder à sua reabilitação.

Apesar das diferenças, todos eles partilham dependências entre si, pelo que, preferencialmente, o problema de otimização de um SDA deveria ser resolvido conjuntamente, todavia, tal seria impraticável em grande parte dos casos devido às enormes dimensões que o problema de otimização único teria. Pode, ainda, ter-se em conta estas dependências através de um conjunto limitado de cenários.

No âmbito desta dissertação, apenas serão analisados problemas de dimensionamento de SDA, mais concretamente, em que só será necessário dimensionar as condutas da rede. Considera-se que a configuração da rede foi definida previamente. O objetivo será a minimização dos custos associados às condutas (subsecção 3.1) e deverão ser verificadas todas as restrições necessárias (imposições físicas – subsecções 3.2 e 3.3) (imposições técnicas – subsecções 3.4 a 3.7).

Considere-se uma rede de abastecimento de água composta por n nós e m condutas de comprimentos l e diâmetros x (as variáveis de decisão) com consumo nos nós, d . Para além disso, o abastecimento é feito a partir de rr reservatórios de cota fixa.

3.1 Função Objetivo

O objetivo será minimizar os custos com condutas. Admite-se que estes são diretamente proporcionais ao comprimento das condutas e relacionam-se de forma não linear com o diâmetro das condutas, $f(l, x)$. O fator de proporcionalidade c representa o custo unitário das condutas medido em unidades monetárias por unidade de comprimento e depende do diâmetro considerado. A função objetivo toma, então, a seguinte forma:

$$\min \sum_{k=1}^m c(x_k) \cdot l_k.$$

3.2 Restrições de Conservação da Massa

Seja N_i o conjunto de pontos que estão diretamente ligados ao nó i e q_{ij} o caudal orientado do nó i para o nó j . Considere-se, ainda, que a água é um líquido incompressível. Então, uma das imposições físicas do sistema indica que o caudal a montante de uma secção terá de ser igual ao caudal a jusante da mesma secção, o que se traduz pelo seguinte conjunto de restrições:

$$\sum_{j \in N_i} q_{ij} + d_i = 0, \quad \forall i \in \{1, \dots, n\}.$$

3.3 Restrições de Conservação da Energia

Seja M_p o conjunto de condutas que pertencem à malha p e ΔH_k a perda de carga na conduta k . Como resultado do princípio da conservação da energia, a perda de carga total ao longo das condutas de uma malha natural é, necessariamente, nula. Tal pode ser traduzido pelo seguinte conjunto de restrições:

$$\sum_{k \in M_p} \Delta H_k = 0, \quad \forall p \in \{1, \dots, nl\}.$$

Onde nl é o número de malhas naturais e pode ser obtido através do número de nós e troços da rede pela seguinte expressão:

$$nl = m - n + 1.$$

Caso exista mais de um reservatório a abastecer a rede ($rr > 1$) então formar-se-ão malhas fictícias entre os reservatórios e o princípio da conservação da energia toma a seguinte forma:

$$\sum_{k \in M_p} \Delta H_k = \Delta z_p, \quad \forall p \in \{nl + 1, \dots, nl + pl\}.$$

Onde Δz_p é a diferença de cotas entre os reservatórios que formam a malha fictícia p e pl é o número de malhas fictícias e pode ser obtido através do número de reservatórios de cota fixa pela seguinte expressão:

$$pl = rr - 1.$$

A perda de carga deve-se às forças de atrito desenvolvidas entre as partículas fluídas de água em movimento e entre as partículas de água e as paredes das condutas e depende essencialmente da viscosidade da água, da rugosidade, do raio hidráulico¹⁵, do comprimento e da velocidade de escoamento nas condutas. Existem várias leis de resistência propostas, porém, as mais utilizadas são a de Hazen-Williams e a de Darcy-Weisbach que se transcrevem de seguida:

$$\Delta H_k = H_{1,k} - H_{2,k} = \frac{A \cdot q_k^B}{C_k^B \cdot x_k^C} \cdot l_k,$$

$$\Delta H_k = H_{1,k} - H_{2,k} = \frac{8 \cdot f_k \cdot q_k^2}{\pi^2 \cdot g \cdot x_k^5} \cdot l_k.$$

Onde $H_{1,k}$ e $H_{2,k}$ correspondem à energia total do sistema no nó a montante e a jusante da conduta k , respetivamente, e C_k e f_k são os coeficientes de resistência da conduta k , respetivamente da fórmula de Hazen-Williams e de Darcy-Weisbach. O coeficiente A e os expoentes B e C da fórmula de Hazen-Williams não são unívocos, havendo pequenas discrepâncias entre valores utilizados por diversos autores. Contudo, o simulador hidráulico utilizado nesta dissertação (EPANET, v2.0) usa os seguintes valores para unidades SI (Manual do utilizador EPANET, 2002):

$$A = 10.674, \quad B = 1.852, \quad C = 4.871.$$

3.4 Restrições de Pressão Mínima / Máxima

A pressão na rede é um dos fatores mais influentes na qualidade do serviço prestado. Pressões demasiado baixas levam a baixos caudais à saída dos equipamentos do consumidor. Por outro lado, pressões demasiado elevadas contribuem para o aparecimento de fugas.

¹⁵ O raio hidráulico é dado pelo quociente entre a área e o perímetro molhado da secção. Para condutas circulares sobre pressão o raio hidráulico depende apenas do diâmetro da conduta.

Seja h_i a pressão no nó i e $h_{max,i}$ e $h_{min,i}$ as pressões máxima e mínima para o mesmo nó, respetivamente. As restrições de pressão máxima e mínima traduzem-se da seguinte forma:

$$h_{min,i} < h_i < h_{max,i}, \quad \forall i \in \{1, \dots, n\}.$$

Para determinar as pressões nos nós o teorema de Bernoulli pode ser usado na seguinte forma¹⁶:

$$h_i = H_i - z_i.$$

Onde H_i é a energia total do sistema no nó i e z_i é a cota geométrica do nó i , relativamente a um plano horizontal de referência.

3.5 Restrições de Velocidade Máxima / Mínima

A velocidade também influencia a qualidade do serviço prestado na medida em que, velocidades demasiado elevadas provocam ruídos e aceleram o processo de desgaste das condutas. Por outro lado, velocidades demasiado baixas podem provocar a sedimentação de algumas partículas sólidas em suspensão e, conseqüentemente, podem obstruir parcialmente a secção das condutas. Para além disso, tempos de retenção muito prolongados nas condutas da rede contribuem para um declínio na qualidade da água.

Seja v_k a velocidade do fluxo de água na conduta k e $v_{min,k}$ e $v_{max,k}$ a velocidade mínima e máxima permitida para a mesma conduta, respetivamente. As restrições de velocidade mínima e máxima traduzem-se da seguinte forma:

$$v_{min,k} < v_k < v_{max,k}, \quad v_k = \frac{4 \cdot q_k}{\pi \cdot x_k^2}, \quad \forall k \in \{1, \dots, m\},$$

3.6 Restrições de Diâmetro Mínimo

Outra gama de imposições técnicas pode passar por limitar inferiormente o diâmetro utilizado nas condutas (*e.g.* para medidas de combate a incêndios).

As restrições de diâmetro mínimo traduzem-se da seguinte forma:

$$x_k > D_{min,k}, \quad \forall k \in \{1, \dots, m\}.$$

¹⁶ É possível desprezar a componente energética referente à velocidade $(v^2/2 \cdot g)$ uma vez que, para este tipo de problemas, essa mesma componente é de uma ordem de grandeza bastante inferior quando comparada com a componente da pressão.

3.7 Restrições de Domínio das Variáveis de Decisão

O que caracteriza um problema de otimização combinatoria é o facto das variáveis de decisão não poderem tomar qualquer valor real, mas sim, apenas valores pertencentes a um conjunto finito de elementos, que no presente caso, corresponde a uma lista de diâmetros retirada de um catálogo de um fabricante de condutas.

Seja D um conjunto de diâmetros comerciais. As restrições de domínio das variáveis de decisão traduzem-se da seguinte forma:

$$x_k \in D, \quad D = \{\phi_1, \phi_2, \dots, \phi_{last}\}, \quad \forall k \in \{1, \dots, m\}.$$

3.8 Modelo Matemático Completo

Aglomerando a informação recolhida nas subsecções anteriores obtém-se o seguinte problema de otimização, onde, para além dos diâmetros das condutas, foi necessário introduzir como variáveis dependentes as pressões nos nós e os caudais nas condutas (adaptado de Cunha & Sousa, 1999 e Marques & Sousa, 2011):

$$\min \sum_{k=1}^m c(x_k) \cdot l_k$$

s. a.

$$\sum_{j \in N_i} q_{ij} + d_i = 0, \quad \forall i \in \{1, \dots, n\}$$

$$\sum_{k \in M_p} \frac{A \cdot q_k^B}{C_k^B \cdot x_k^C} \cdot l_k \cdot \text{sgn}(q_k) = 0, \quad \forall p \in \{1, \dots, nl\}$$

$$\sum_{k \in M_p} \frac{A \cdot q_k^B}{C_k^B \cdot x_k^C} \cdot l_k \cdot \text{sgn}(q_k) = \Delta z_p, \quad \forall p \in \{nl + 1, \dots, nl + pl\}$$

$$h_{min,i} < h_i < h_{max,i}, \quad \forall i \in \{1, \dots, n\}$$

$$v_{min,k} < \left| \frac{4 \cdot q_k}{\pi \cdot x_k^2} \right| < v_{max,k}, \quad \forall k \in \{1, \dots, m\}$$

$$x_k > D_{min,k}, \quad \forall k \in \{1, \dots, m\}$$

$$x_k \in D, \quad \forall k \in \{1, \dots, m\}$$

$$q_k \in \mathbb{R}, \quad \forall k \in \{1, \dots, m\}$$

$$h_i \in \mathbb{R}^+, \quad \forall i \in \{1, \dots, n\}.$$

4 DESCRIÇÃO DO ALGORITMO

Para o desenvolvimento deste trabalho foi utilizado, como ferramenta de apoio, um algoritmo que tem por base os conceitos fundamentais do SA, apresentado em Cunha (1999) e Cunha & Sousa (1999) e codificado em Visual Basic tal como descrito em Marques *et al.* (2015a). No âmbito desta dissertação, o algoritmo foi modificado com o intuito de melhorar a sua eficiência. Todas as alterações realizadas foram codificadas em Visual Basic utilizando o *software* Microsoft® Visual Studio® Community 2017, v15.1, para o efeito.

Apesar de se ter revisto alguns aspetos essenciais do SA na subsecção 2.4.2, devido à importância do mesmo para a presente dissertação, será efetuada uma análise pormenorizada do modo de funcionamento do mesmo.

No Gráfico 4.1 está representado o fluxograma típico de um algoritmo que assenta nos conceitos básicos do SA. Uma explicação detalhada de cada passo é dada na respetiva secção.

4.1 Parâmetros de Controlo / Solução Inicial

Entende-se por parâmetros de controlo, os valores introduzidos pelo utilizador que têm influência na trajetória seguida pelo algoritmo. No presente caso, são eles os seguintes (adaptado de Cunha & Sousa, 1999):

- Temperatura Inicial ou Elasticidade de Aceitação (t_0 ou ea). Para a temperatura inicial, o próprio nome explica o seu intento. É de salientar que temperaturas iniciais muito altas levam a uma perda de eficiência do algoritmo e temperaturas iniciais muito baixas levam a que o algoritmo convirja rapidamente para uma solução subótima. Para além do mais, a temperatura inicial não é um parâmetro adimensional (tem as mesmas unidades da função objetivo), o que dificulta ainda mais a escolha de um valor adequado, pelo que, na incerteza de um bom valor a utilizar, recomenda-se a utilização da elasticidade de aceitação para determinar a temperatura inicial, proposta por Cunha & Sousa (2001);
- Taxa de Arrefecimento (α). É utilizado o critério de arrefecimento geométrico pelo que as considerações estabelecidas na subsecção 2.4.2 são aqui aplicáveis;
- Parâmetros associados aos critérios de saída do ciclo interno. Existe apenas um parâmetro a definir (μ_1) que está diretamente associado ao número de iterações que são realizadas dentro do ciclo interno do algoritmo.

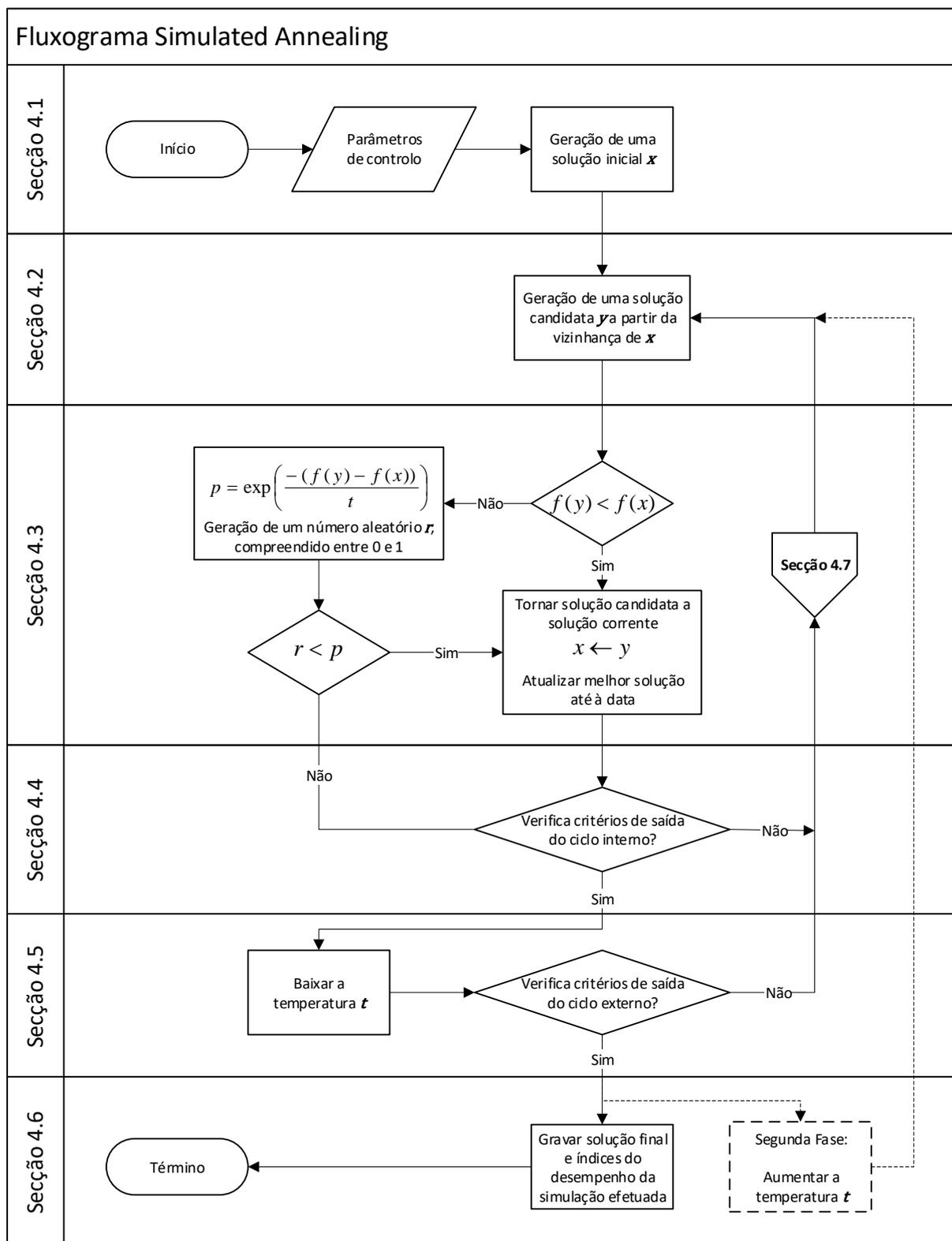


Gráfico 4.1 - Fluxograma do algoritmo utilizado (adaptado de Cunha, 1999 e Cunha & Sousa, 1999, 2001)

- Parâmetros associados aos critérios de saída do ciclo externo. Existe um parâmetro que define um limite à quantidade de vezes que a temperatura é diminuída sem melhoria da melhor solução até então encontrada (μ_2) e um parâmetro que limita a probabilidade de aceitação (pa_{lim}).

Gerar uma solução inicial pode ser um processo mais ou menos complicado dependendo essencialmente da forma do espaço das soluções admissíveis do problema em causa. Caso não sejam definidos limites à pressão máxima admissível e à velocidade mínima admissível, uma solução que é quase sempre admissível corresponde a atribuir o diâmetro máximo disponível a todas as condutas a dimensionar do problema. Para os casos mais gerais, deve obter-se uma solução inicial manualmente ou usar uma solução admissível do problema já conhecida.

4.2 Geração de Soluções Candidatas

A geração de soluções candidatas é um passo comum a qualquer metaheurística e funciona como o coração do algoritmo. A sua função é a de dar soluções ao algoritmo para o “manter vivo”. O fluxograma do modo de gerar soluções candidatas é exposto no Gráfico 4.2.

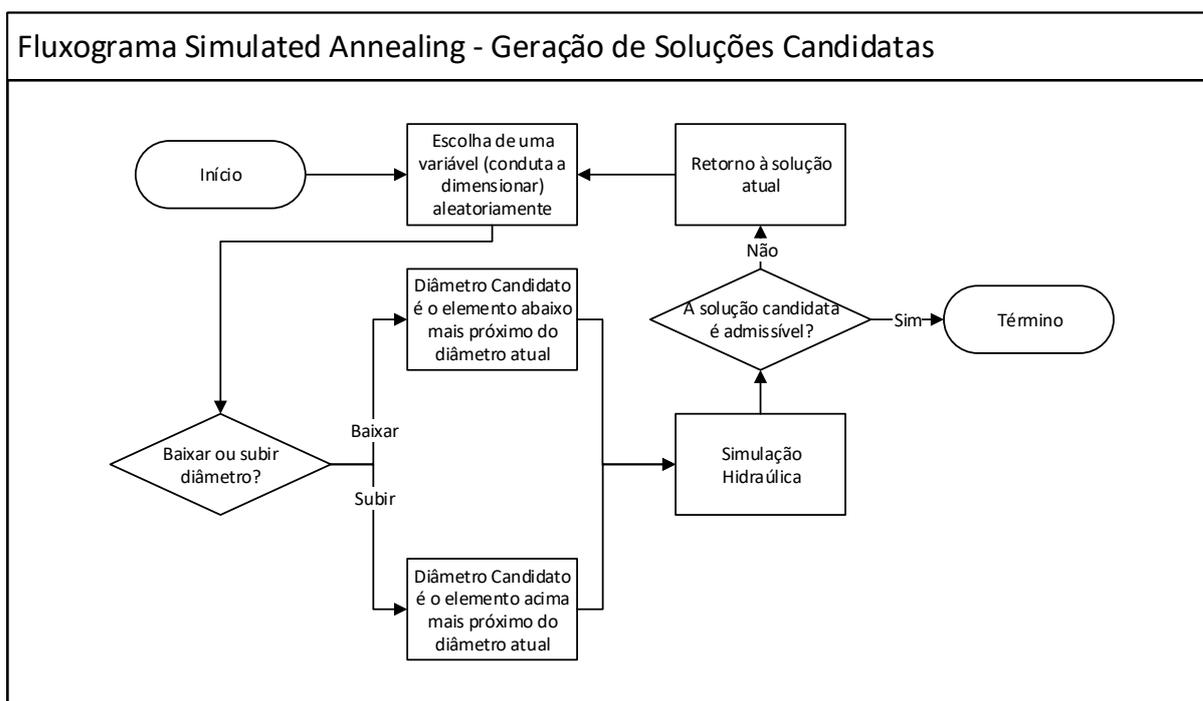


Gráfico 4.2 - Fluxograma da geração de soluções candidatas (adaptado Cunha & Sousa, 2001)

Começa-se por escolher, de forma aleatória, um troço da rede que necessite de ser dimensionado e verifica-se qual é o seu diâmetro corrente. De seguida, é escolhido um diâmetro vizinho do diâmetro corrente tendo em conta um critério adequado. Com a alteração do diâmetro de uma conduta tem-se uma nova solução candidata, que deverá ser aceite caso cumpra com as

restrições do estudo de caso ou descartada caso contrário, para tal, é realizada uma simulação hidráulica com o apoio do *software* EPANET, v2.0.

Atente-se que se apenas forem permitidos movimentos de descida de diâmetro, o algoritmo não fará mais do que uma pesquisa local e a solução obtida será, certamente, subótima. Porém, igualar as probabilidades de descida e subida tornaria o algoritmo ineficiente, principalmente nas iterações iniciais. Um compromisso entre os dois limites apresentados deve ser estudado para que se obtenha o melhor desempenho do algoritmo possível. Pretende-se que o algoritmo tenha uma ligeira tendência para baixar os diâmetros das condutas e, conseqüentemente, uma ligeira tendência para reduzir o custo das soluções.

4.3 Critério de Metropolis

A solução candidata passa a solução atual se for verificada uma melhoria no custo ou com uma probabilidade obtida através da distribuição de Boltzmann, tal como já descrito na subsecção 2.4.2, caso contrário, a solução candidata é descartada. É também neste ponto que se atualiza a melhor solução até então encontrada.

4.4 Saída do Ciclo Interno

Para que o algoritmo termine o ciclo interno é necessário que sejam verificadas algumas condições. Para além da condição fundamental, que corresponde a definir um número mínimo de iterações que devem ser realizadas durante o ciclo interno¹⁷, o algoritmo deve tomar outros aspetos em consideração, tais como, verificar se a melhor solução está a ser frequentemente atualizada. Este tipo de condições tem por objetivo impedir que o processo de saída do ciclo interno seja demasiado mecanizado, ou seja, o algoritmo deve ter a “inteligência” de sair do ciclo interno apenas quando necessitar de alterar algum parâmetro para conseguir melhores resultados (neste caso, a temperatura).

4.5 Saída do Ciclo Externo

Depois do algoritmo sair do ciclo interno, a temperatura é reduzida de acordo com o critério geométrico, à taxa de α , até que sejam verificados um conjunto de critérios. Esses critérios consistem em verificar se a temperatura desceu μ_2 vezes sem que se tenham registado melhorias na melhor solução até então encontrada, verificar se a probabilidade de aceitação é menor do que pa_{lim} e ainda, verificar se as diferenças relativas (de custos médios e/ou ótimos) são pequenas o suficiente para admitir que o algoritmo convergiu.

¹⁷ No algoritmo, este número depende do fator μ_1 e da quantidade de variáveis de decisão.

Mais uma vez, poderia simplesmente ter-se definido um critério que terminava o ciclo externo assim que a temperatura atingisse um limite mínimo, porém, dessa forma, tornar-se-ia muito mais difícil controlar a qualidade da solução final produzida pelo algoritmo.

4.6 Solução Final e Índices de Desempenho / Segunda Fase

Dado por terminado o ciclo externo, é possível tentar aprimorar a melhor solução até então obtida por aplicação do que é descrito na literatura como a Segunda Fase (Cunha & Sousa, 2001). Tal, não é mais do que voltar a elevar a temperatura final do algoritmo e continuar com a procura de melhores soluções normalmente. Foi notado, em alguns casos, que a Segunda Fase trouxe uma mais-valia para o algoritmo.

Assim que todos os processos estiverem completos, a melhor solução obtida pelo algoritmo é enviada para o utilizador (para os estudos de caso, são os diâmetros das condutas a dimensionar) e são fornecidos, juntamente, alguns indicadores do desempenho da simulação efetuada, tais como:

- O custo da melhor solução;
- O número total de iterações realizadas até ao término do algoritmo;
- O tempo de execução da simulação (dependente da máquina utilizada);
- Uma tabela com o custo atualizado da melhor solução, o número de iterações total e a probabilidade de aceitação por temperatura;
- Um gráfico de dispersão com o custo de cada solução por iteração.

4.7 Alterações Propostas

Nesta secção são propostas algumas alterações no algoritmo original que têm como principal objetivo, melhorar a eficiência do algoritmo, ou seja, torná-lo mais rápido sem perder a qualidade da solução final.

Como e onde se deve alterar, é a grande questão que se coloca. Analisando a metodologia do SA, nota-se que todos os processos, com exceção do critério de Metropolis (uma vez que é deste critério que parte a analogia para com o processo físico), são passíveis de sofrer alterações. Isto leva a que seja necessário identificar qual ou quais os processos que têm uma maior influência no tempo despendido pelo algoritmo para achar a solução final. Excluindo o mais óbvio, que será, garantidamente, a simulação hidráulica devido ao tipo de equações que regem o transporte de água numa rede, coloca-se como hipótese o processo de geração de soluções candidatas, que de facto, foi onde se concentrou o foco das modificações.

Até este ponto, o processo de geração de soluções candidatas era constante no decorrer do algoritmo, o que significa que não interessava quão avançado o algoritmo se encontrava num determinado momento que a probabilidade de mudar um determinado diâmetro só dependia da solução atual. No âmbito desta dissertação, propõe-se a mudança deste sistema para outro em que o passado do algoritmo influencie a forma de gerar soluções candidatas futuras. Tal como acontece com outras metaheurísticas, pretende-se adicionar uma componente de aprendizagem no algoritmo. Isso pode ser conseguido de várias formas, contudo, propõe-se a utilização de um algoritmo híbrido por adição de algumas propriedades naturais da CE (subsecção 2.4.5). Uma das propriedades mais interessantes é o facto de se poder armazenar informação estatística referente ao progresso do algoritmo na forma de uma matriz de probabilidades. Essa matriz de probabilidades deve armazenar informação relativa à probabilidade de um diâmetro comercial i estar associado ao troço j tal como se exemplifica no Quadro 4.1 (Perelman & Ostfeld, 2007). A informação deve obter-se a partir de uma lista de possíveis soluções dadas pelo SA, que deverá ser periodicamente atualizada com o decorrer do algoritmo.

Quadro 4.1 - Exemplo hipotético de matriz de probabilidades

<i>Troços</i>						<i>{Nº de variáveis}</i>
<i>Diâmetros</i>	1	2	...	j	...	
Φ_1	0.15	0.04	...	0.2	...	0
Φ_2	0.07	0.25	...	0	...	0
\vdots	\vdots	\vdots	\ddots	\vdots	\ddots	\vdots
Φ_i	0.1	0.08	...	0.2	...	0.7
\vdots	\vdots	\vdots	\ddots	\vdots	\ddots	\vdots
$\Phi_{\{n^\circ \text{ de diâmetros}\}}$	0.22	0.12	...	0.15	...	0.05
Total	1	1	...	1	...	1

As alterações propostas estão esquematizadas no Gráfico 4.3 e na subsecção 4.7.5. Alguns esclarecimentos adicionais são dados nas respetivas subsecções.

4.7.1 Novos Parâmetros de Controlo

Com as modificações efetuadas no algoritmo serão necessários mais 4 parâmetros definidos pelo utilizador, são eles:

- Dimensão da lista (N). Este parâmetro define a dimensão do vetor de soluções que deve ser tido em conta para gerar a matriz de probabilidades. Atente-se que uma lista muito extensa leva a um aumento do tempo despendido pelo algoritmo para encontrar soluções quase-ótimas, enquanto que uma lista demasiado curta faz com que o algoritmo perca o efeito de aprendizagem desejado e pode levar, inclusivamente, à estagnação precoce do mesmo. Preferencialmente, a dimensão da lista deverá depender do número de variáveis envolvidas no problema;

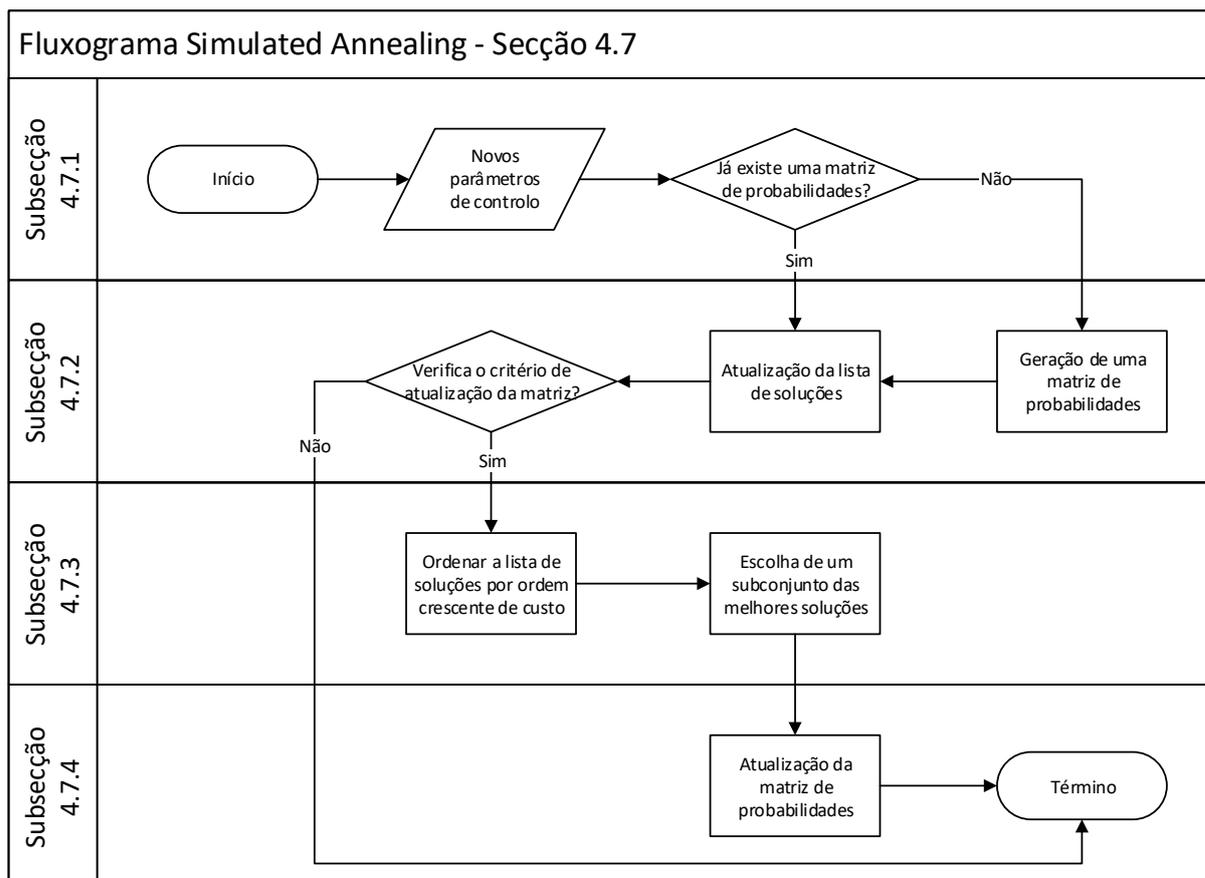


Gráfico 4.3 - Fluxograma das alterações propostas

- Frequência de atualização da matriz de probabilidades (γ). Este parâmetro define o intervalo entre iterações após o qual deve ser atualizada a matriz de probabilidades com informação relativa às entradas mais recentes no subconjunto de melhores soluções da lista;
- Percentagem de melhores soluções da lista (ρ). Este parâmetro define um subconjunto (de dimensão $\rho \cdot N$) das melhores soluções pertencentes à lista de dimensão N ;
- Coeficiente de suavização (β). Este parâmetro está diretamente relacionado com a diversificação do espaço de procura. Deve ser usado para evitar que o algoritmo convirja para ótimos locais prematuramente.

4.7.2 Geração da Matriz de Probabilidades Inicial / Atualização da Lista de Soluções / Critério de Atualização da Matriz de Probabilidades

Assim que o subalgoritmo (Gráfico 4.3) é ativado pela primeira vez deve ser definida uma matriz de probabilidades inicial (ver subsecção 2.4.5) e uma lista de N soluções inicial. Essa lista de soluções é gerada de forma completamente aleatória (devido à distribuição uniforme de probabilidades utilizada) e a cada elemento da lista é dado um valor da função objetivo muito

elevado por forma a que no decorrer da atualização da lista as soluções não admissíveis sejam rapidamente trocadas por admissíveis.

Assim que haja uma matriz de probabilidades e uma lista de soluções é verificado se a solução atual do algoritmo deve ser embutida na lista de soluções, para tal, a única restrição que se impõe é a de que as soluções a incluir na lista tenham verificado o critério de Metropolis previamente. Por forma a atualizar a lista de soluções com a nova entrada é necessário retirar uma solução da lista, para que a mesma mantenha a sua dimensão inicial, para tal, a solução da lista que retornar um pior custo da função objetivo é trocada pela nova entrada. Desta forma, a lista de soluções está constantemente a ser melhorada com base na informação recolhida e consequentemente melhores matrizes de probabilidades serão geradas tal como é suposto pelo método CE original.

Uma diferença relativamente ao método CE original é a introdução de um parâmetro γ que indica quando é que a matriz de probabilidades deve ser atualizada. Tal, deve-se ao facto de que no método CE original todas as soluções da lista são atualizadas de uma só vez em cada iteração enquanto que por este procedimento apenas uma solução da lista seria atualizada por iteração, na melhor das hipóteses¹⁸. Como consequência, o esforço computacional utilizado seria demasiado elevado sem que fossem verificados benefícios concretos no progresso do algoritmo. Desta forma a introdução do parâmetro γ visa atualizar a matriz de probabilidades apenas quando estiver reunida informação suficiente para que as alterações produzidas no progresso do algoritmo sejam visíveis.

4.7.3 Escolha de um Subconjunto de Melhores Soluções

Caso seja necessário atualizar a matriz de probabilidades, então a lista de soluções deve ser ordenada por ordem crescente de custo e um subconjunto de melhores soluções deve ser escolhido por forma a atualizar a matriz de probabilidades. A escolha é geralmente feita através do parâmetro ρ que funciona como uma percentagem de soluções da lista ordenada a utilizar, tal como já descrito na subsecção 2.4.5.

4.7.4 Atualização da Matriz de Probabilidades

Com as $\rho \cdot N$ melhores soluções da lista, é calculado um estimador da nova matriz de probabilidades que deverá ser sujeito à aplicação do parâmetro β , tal como descrito na subsecção 2.4.5.

¹⁸ Este facto retrata uma das principais diferenças entre metaheurísticas baseadas numa população e metaheurísticas baseadas numa trajetória.

No entanto, foi notado que a utilização da fórmula (que envolve o parâmetro β) tal como originalmente prevista não garantia que o algoritmo “escapasse” de ótimos locais como pressuposto. Para contornar esta situação, a fórmula original foi modificada de modo a que seja incentivada a diversificação do espaço de procura. Uma troca da parcela referente à matriz de probabilidades atual pela matriz de probabilidades inicial revelou ser bastante eficaz.

$$p_{ij}^{t+1} = \beta \cdot \hat{p}_{ij}^{t+1} + (1 - \beta) \cdot p_{ij}^0.$$

4.7.5 Nova Geração de Soluções Candidatas

O principal objetivo de implementar estes novos conceitos no algoritmo é o de alterar a rapidez de convergência para a solução ótima. Para que isso seja possível, a forma como são geradas as soluções candidatas deve adaptar-se a tal.

A grande diferença em relação ao processo de geração de soluções candidatas original é o facto de se atribuir uma classificação aos diâmetros comerciais com base na matriz de probabilidades. Essa classificação deve ser atribuída tendo em conta o valor e a disposição dos elementos que compõem o vetor probabilístico da conduta escolhida. Por exemplo, se a probabilidade de um determinado diâmetro comercial estar associado a uma determinada conduta é elevada, então significa que as soluções que continham essa opção eram boas soluções, e por esse motivo, na geração de soluções candidatas essa opção deve ser utilizada mais vezes. Se o contrário acontecer, essa opção deverá ser utilizada menos vezes, mas garantindo sempre que essa opção tem possibilidades de voltar a ser escolhida. Analogamente, caso seja notável que um grupo de diâmetros fornece melhores soluções em comparação com outros diâmetros, então esse grupo deve pertencer a uma classe que tenha uma maior probabilidade de ser escolhida.

Depois de classificados os diâmetros comerciais, deve ser escolhida uma classe de acordo com um critério adequado (*e.g.* probabilístico). A nova solução candidata será igual à atual em todas as condutas com a exceção da conduta selecionada onde estará instalado um diâmetro comercial pertencente à classe escolhida. O resto do procedimento é semelhante ao que já acontecia na geração de soluções original.

Como é possível depreender, com a nova geração de soluções candidatas, o conceito de vizinhança da solução atual é completamente alterado uma vez que se possibilita a escolha de qualquer diâmetro ao invés de, somente, os diâmetros imediatamente mais próximos do diâmetro atual, como acontecia na geração de soluções original. Desta forma, é esperado que a eficiência do algoritmo seja efetivamente melhorada.

5 ESTUDOS DE CASO

5.1 Two-Loop Network

A *Two-loop network* (TLN) (Figura 5.1) (Alperovits & Shamir, 1977) é composta por um reservatório, 6 nós de consumo e 8 condutas dispostas em 2 malhas. O reservatório está fixado à cota constante de 210 m e todas as condutas têm 1000 m de comprimento e coeficiente de Hazen-Williams igual a 130. Deve garantir-se a pressão mínima de 30 m.c.a. em todos os nós de consumo. Estão disponíveis 14 diâmetros comerciais (Quadro 5.1). Este problema é considerado de pequena dimensão com $14^8 \approx 10^{9.17}$ soluções possíveis.

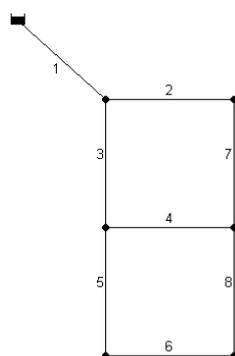


Figura 5.1 - Esquema da *Two-Loop Network*

5.2 Hanoi Network

A rede de Hanoi (HAN) (Figura 5.2) (Fujiwara & Khang, 1990) é composta por um reservatório à cota fixa de 100 m, 34 condutas dispostas em 3 malhas, com o coeficiente de Hazen-Williams igual a 130, e 31 nós de consumo. Deve garantir-se a pressão mínima de 30 m.c.a. em todos os nós de consumo. Estão disponíveis 6 diâmetros comerciais (Quadro 5.1). Este problema é considerado de média dimensão com $6^{34} \approx 10^{26.46}$ soluções possíveis.

5.4 Balerma Irrigation Network

A rede de irrigação de Balerma (BIN) (Figura 5.4) (Reca & Martínez, 2006) é composta por 4 reservatórios, com cotas fixadas entre 112 m e 127 m, 454 condutas de pequeno comprimento, com um coeficiente de rugosidade de Darcy-Weisbach igual a 0.0025 mm (PVC), e 443 nós de consumo. Deve garantir-se a pressão mínima de 20 m.c.a. em todos os nós de consumo. Estão disponíveis 10 diâmetros comerciais (Quadro 5.1). Este problema é considerado de muito grande dimensão com 10^{454} soluções possíveis.



Figura 5.4 - Esquema da rede de Balerma

Quadro 5.1 - Diâmetros comerciais disponíveis e respectivos custos unitários

<i>TLN</i>		<i>HAN</i>		<i>FOS</i>		<i>BIN</i>	
Diâmetros Comerciais [mm]	Custo Unitário [\$ /m]	Diâmetros Comerciais [mm]	Custo Unitário [\$ /m]	Diâmetros Comerciais [mm]	Custo Unitário [€ /m]	Diâmetros Comerciais [mm]	Custo Unitário [€ /m]
25.4	2	304.8	45.726	16	0.38	113	7.22
50.8	5	406.4	70.4	20.4	0.56	126.6	9.1
76.2	8	508	98.387	26	0.88	144.6	11.92
101.6	11	609.6	129.333	32.6	1.35	162.8	14.84
152.4	16	762	180.748	40.8	2.02	180.8	18.38
203.2	23	1016	278.28	51.4	3.21	226.2	28.6
254	32			61.4	4.44	285	45.39
304.8	50			73.6	6.45	361.8	76.32
355.6	60			90	9.59	452.2	124.64
406.4	90			102.2	11.98	581.8	215.85
457.2	130			114.6	14.93		
508	170			130.8	19.61		
558.8	300			147.2	24.78		
609.6	550			163.6	30.55		
				184	38.71		
				204.6	47.63		
				229.2	59.7		
				257.8	75.61		
				290.6	99.58		
				327.4	126.48		
				368.2	160.29		
				409.2	197.71		

6 RESULTADOS

Neste capítulo será realizada uma análise comparativa do desempenho de diferentes versões do algoritmo descrito no cap. 4 quando aplicadas aos estudos de caso referidos no cap. 5. As diferentes versões comparadas são as seguintes:

- Algoritmo original (SA). Algoritmo composto apenas pelo procedimento natural do Recozimento Simulado (Gráfico 4.1 sem Segunda Fase e sem a secção 4.7);
- Algoritmo com Segunda Fase (SAR). Algoritmo composto pelo procedimento natural do Recozimento Simulado com Segunda Fase no final (Gráfico 4.1 sem a secção 4.7);
- Algoritmo Híbrido (SACER). Algoritmo completo, descrito pelo Gráfico 4.1 na sua totalidade (incluindo Segunda Fase).

Efetuar-se-á, não só, uma comparação através dos indicadores descritos na secção 2.2 (eficácia, eficiência, fiabilidade e robustez), como também do progresso do algoritmo. Para o efeito, uma amostra de 20 soluções (constituídas por diferentes sementes de números aleatórios) foi utilizada para cada estudo de caso e para cada algoritmo¹⁹. Para comparar o progresso dos diferentes algoritmos e a influência que os seus parâmetros de controlo têm no mesmo, foram realizadas diversas simulações para certos intervalos dos parâmetros de controlo. Todos os resultados foram obtidos com uma CPU Intel® Core™ i5 de 2.53 GHz.

As melhores soluções encontradas ou conhecidas para os estudos de caso têm os seguintes custos:

- TLN - \$419 000 (De Corte & Sörensen, 2013);
- HAN - \$6 081 118 (De Corte & Sörensen, 2013);
- FOS - €28 218 (encontrada pelo algoritmo)²⁰;
- BIN - €1 921 400 (Cisty *et al.*, 2017).

É conveniente referir que para os estudos de caso FOS e BIN é conhecida uma solução de partida. Já para estudos de caso TLN e HAN, a pior solução possível é utilizada como solução inicial²¹.

¹⁹ Exceto para o estudo de caso BIN, onde se utilizou uma amostra de 10 soluções e $\mu_1 = 20$ por forma a aliviar o esforço computacional.

²⁰ Comparado com €29 000 encontrado por Bi *et al.* (2015).

²¹ Todas as redes, incluindo o catálogo das condutas, as restrições e as soluções iniciais, estão disponíveis em UECWS@ (2017).

6.1 Simulated Annealing

A amostra, para cada estudo de caso, foi gerada com os seguintes parâmetros¹⁹:

$$\alpha = 0.95, \quad ea = 0.10, \quad \mu_1 = 40, \quad \mu_2 = 5, \quad pa_{lim} = 5\%$$

Esta parametrização é o resultado de estudos efetuados pelo autor original do programa (João Marques) em conjunto com o autor desta dissertação que verificaram ser adequados a um grande conjunto de casos (e os resultados obtidos confirmam).

Alguns dados estatísticos referentes à amostra de cada estudo de caso são apresentados no Quadro 6.1.

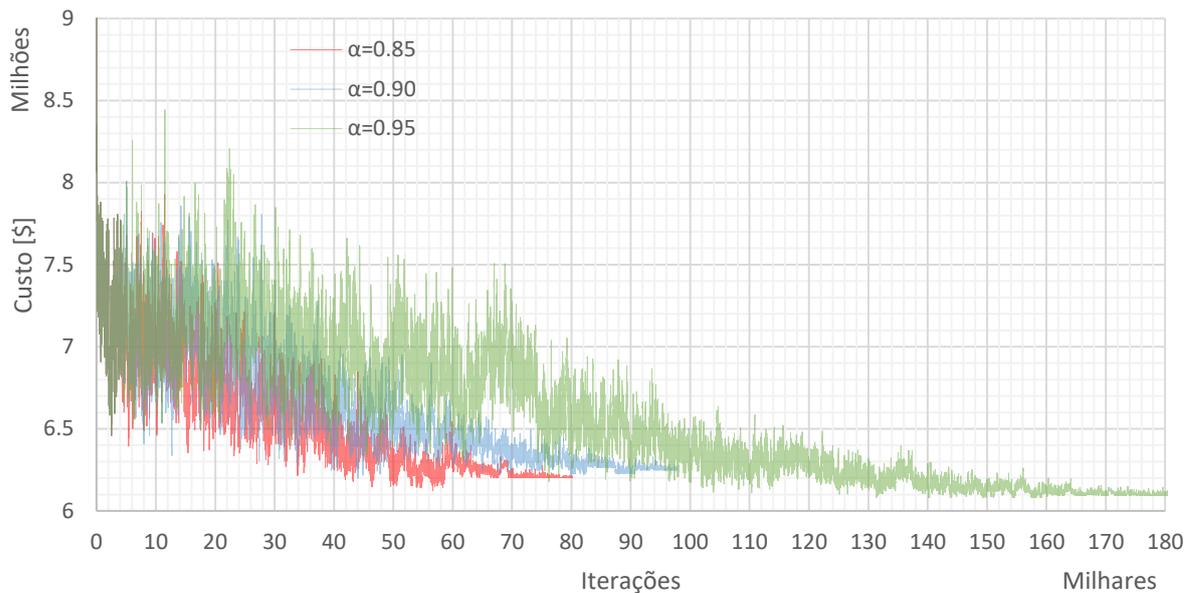
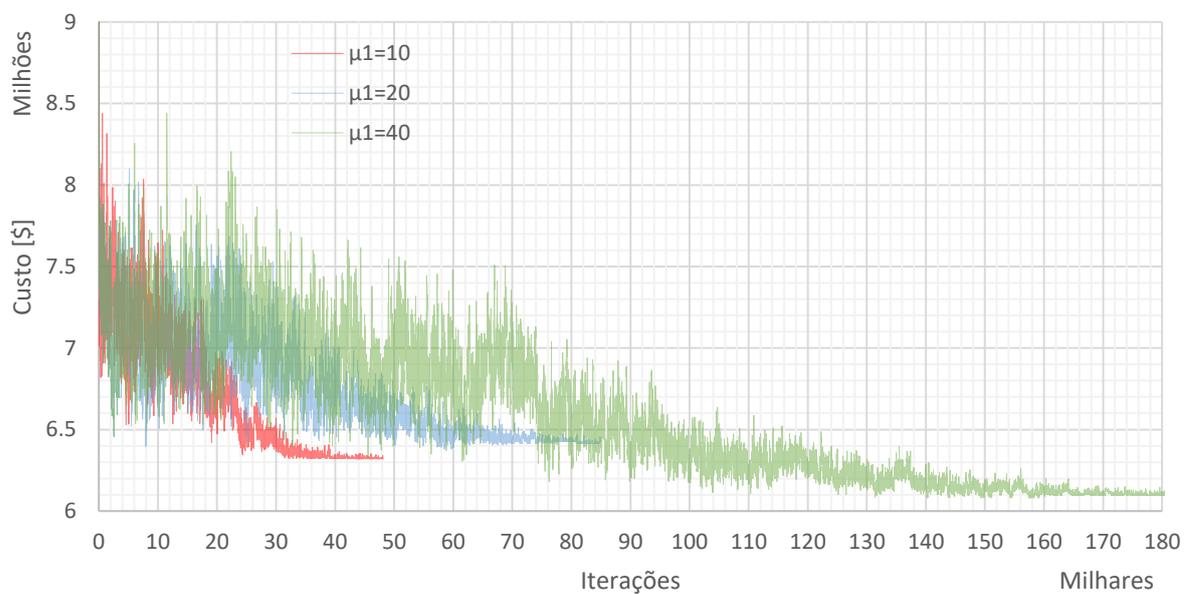
Quadro 6.1 - Resultados para o algoritmo SA

	<i>TLN</i>	<i>HAN</i>	<i>FOS</i>	<i>BIN</i>
Custo da Pior Solução [\$ €]	420 000	6 300 296	29 202	2 059 799
Custo Médio [\$ €]	419 050	6 104 646	28 682	2 031 643
Custo da Melhor Solução [\$ €]	419 000	6 081 118	28 218	1 945 551
Soluções “Ótimas” [%]	95	50	5	0
Desvio-Padrão [\$ €]	217.94	53 978.8	190.61	34 831.39
Média de Iterações por Execução [-]	36 335	161 246	263 092	2 962 479
Tempo de Execução Médio [s]	9.10	131.41	271.61	13 747.14

Ao observar o Quadro 6.1 percebe-se a relevância que a dimensão do problema tem na propensão para o algoritmo encontrar soluções quase-ótimas. Atente-se no caso BIN, a dimensão do problema é de tal forma grande que os tempos de cálculo tornar-se-iam intoleráveis caso fossem usados os mesmos parâmetros que para outros problemas de menor dimensão.

O Gráfico 6.1 e o Gráfico 6.2 representam, respetivamente, o andamento do algoritmo para diferentes valores do coeficiente de arrefecimento (α) e para diferentes valores do parâmetro μ_1 . Os restantes parâmetros (à exceção do parâmetro ea) não interferem diretamente no progresso do algoritmo pelo que não são aqui considerados. Ambos os gráficos foram obtidos para o estudo de caso HAN utilizando, quando aplicável, os parâmetros em cima definidos.

Como é possível notar, em ambos os casos, um aumento dos parâmetros leva a um maior número de iterações que é compensado por uma redução do custo mais ou menos significativa.

Gráfico 6.1 - Influência do parâmetro α no andamento do algoritmo SAGráfico 6.2 - Influência do parâmetro μ_1 no andamento do algoritmo SA

6.2 Simulated Annealing com Segunda Fase

No Quadro 6.2 estão representados os dados estatísticos obtidos pelo algoritmo SAR para a mesma amostra de soluções obtida pelo algoritmo SA. A percentagem de soluções melhoradas ao introduzir a Segunda Fase é a seguinte:

- TLN – 0%;
- HAN – 30%;
- FOS – 35%;
- BIN – 50%.

Quadro 6.2 - Resultados para o algoritmo SAR

	<i>TLN</i>	<i>HAN</i>	<i>FOS</i>	<i>BIN</i>
Custo da Pior Solução [\$ €]	420 000	6 300 296	28 872	2 059 799
Custo Médio [\$ €]	419 050	6 097 139	28 648	2 031 373
Custo da Melhor Solução [\$ €]	419 000	6 081 118	28 218	1 945 551
Soluções “Ótimas” [%]	95	70	5	0
Desvio-Padrão [\$ €]	217.94	47 534.38	145.9	34 809.9
Média de Iterações por Execução [-]	39 094	176 807	286 166	3 065 071
Tempo de Execução Médio [s]	11.49	153.35	359.00	14 241.30

O Gráfico 6.3 mostra um exemplo do que é esperado quando a componente da Segunda Fase é adicionada no algoritmo original. A simulação apresentada foi realizada para o estudo de caso HAN.

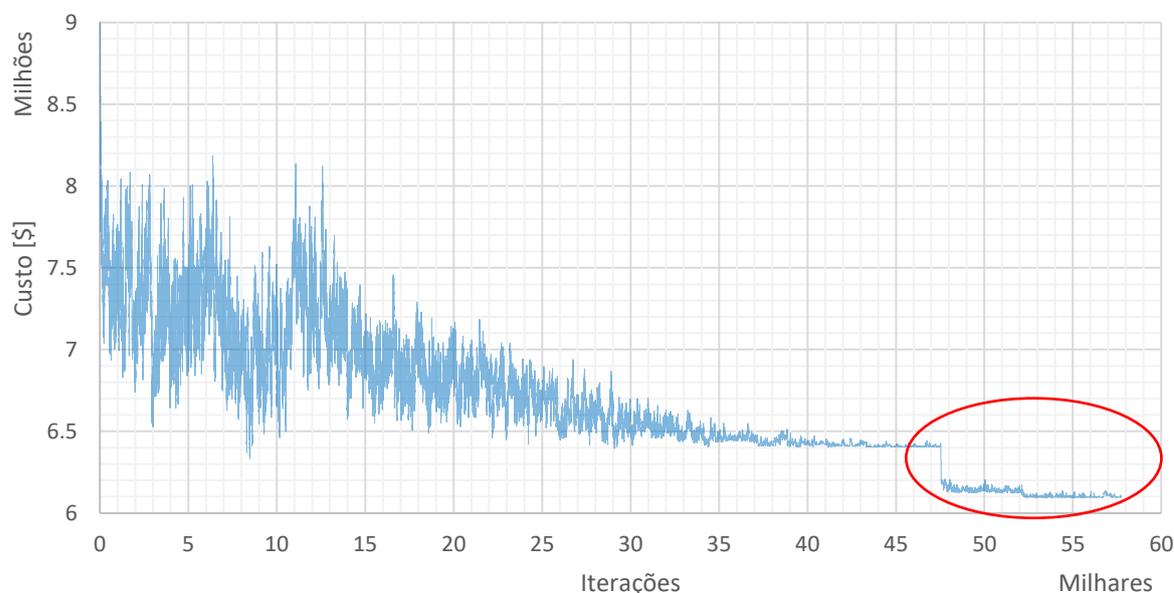


Gráfico 6.3 - Efeito da Segunda Fase

Apesar do inevitável aumento de iterações e tempo despendido para efetuar a simulação, a Segunda Fase permite desbloquear o algoritmo quando este se encontra com a liberdade de movimentos condicionada, possibilitando assim a eventual saída de ótimos locais. Para além disso, a Segunda Fase mostra ser especialmente eficaz em mitigar o efeito de uma convergência precoce, resultante de uma má atribuição de parâmetros.

6.3 Simulated Annealing Híbrido

A amostra, para cada estudo de caso, foi gerada com os parâmetros do Quadro 6.3. Para chegar a esta parametrização foram realizadas diversas simulações, de uma forma contínua, com diferentes valores dos parâmetros por forma a que o custo médio das soluções seja o mais próximo possível, ou eventualmente melhor do que o algoritmo SAR. Isto porque o principal objetivo do algoritmo SACER é o de melhorar a eficiência relativamente ao algoritmo SAR sob a forma de redução do número de soluções avaliadas (ou iterações realizadas) e do tempo despendido para executar cada simulação.

A propósito do tempo de execução refira-se que, apesar de o mesmo ser um indicador importante da eficiência do algoritmo (talvez o principal), ele é especialmente dependente do ambiente computacional em que são realizadas as simulações. Infelizmente, por ser muito difícil recriar as condições exatas em que foram efetuadas as simulações, este indicador deve ser julgado com bastante prudência.

Quadro 6.3 - Parâmetros de controlo usados para o algoritmo SACER

			<i>TLN</i>	<i>HAN</i>	<i>FOS</i>	<i>BIN</i>
Parâmetros de Controlo	SA	α	0.90	0.90	0.90	0.90
		ea	0.10	0.10	0.10	0.10
		μ_1	25	25	25	12
		μ_2	5	5	5	5
		pa_{lim}	5	5	5	5
	CE	N	250	750	1000	2500
		ρ	20	10	10	4
		β	0.85	0.85	0.85	0.85
		γ	500	1500	2000	5000

Os dados estatísticos obtidos para este algoritmo estão representados no Quadro 6.4.

Quadro 6.4 - Resultados para o algoritmo SACER

	<i>TLN</i>	<i>HAN</i>	<i>FOS</i>	<i>BIN</i>
Custo da Pior Solução [\$ €]	420 000	6 300 296	28 766	2 057 494
Custo Médio [\$ €]	419 150	6 093 575	28 606	1 967 592
Custo da Melhor Solução [\$ €]	419 000	6 081 118	28 497	1 926 660
Soluções “Ótimas” [%]	85	85	0	0
Desvio-Padrão [\$ €]	357.07	47 636.17	79.15	43 377.15
Média de Iterações por Execução [-]	13 633	92 627	132 729	1 244 096
Tempo de Execução Médio [s]	9.09	152.59	293.33	11 659.40

Por comparação do Quadro 6.4 com o Quadro 6.2 é possível retirar algumas conclusões relativamente ao desempenho do algoritmo SACER, tais como:

- Para problemas de pequena dimensão um algoritmo simples já funciona suficientemente bem, não havendo a necessidade de adicionar complexidade no algoritmo por forma a melhorar o seu desempenho;
- Para problemas de grande dimensão é notável uma clara melhoria tanto na eficiência como na eficácia quando é proposta a resolução pelo algoritmo híbrido SACER. Atente-se, a título de exemplo, no caso BIN, o tempo de execução médio sofre uma redução de 18%, em relação ao algoritmo SAR, enquanto o custo médio também é reduzido em cerca de 3%;
- Contudo, na maior parte dos casos, verifica-se que o algoritmo SACER apresenta uma maior variação de resultados (desvio-padrão) comparando com o algoritmo SAR, o que indica que ainda há trabalho a ser realizado para tornar o algoritmo SACER mais fiável;

No Gráfico 6.4, Gráfico 6.5, Gráfico 6.6 e Gráfico 6.7 está representada (de uma forma simplificada) a influência que uma variação dos novos parâmetros tem na qualidade das soluções finais obtidas pelo algoritmo. Cada ponto nos gráficos representa uma amostra de 100 simulações efetuadas para o estudo de caso HAN, no eixo das ordenadas está representado o desvio percentual médio da amostra em relação à melhor solução conhecida e no eixo das abcissas o respetivo valor do parâmetro considerado.

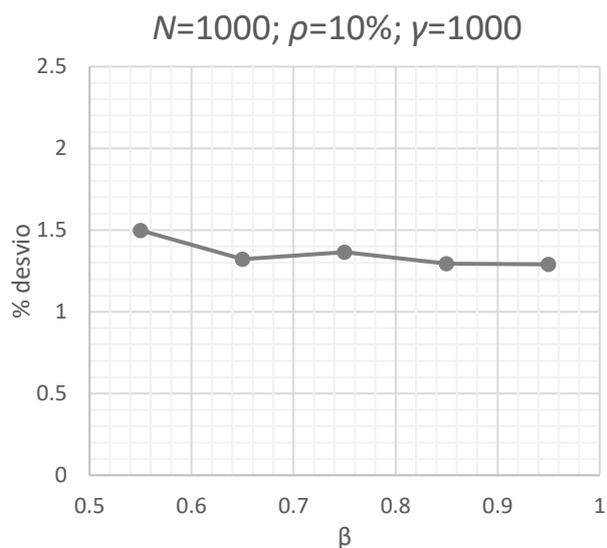


Gráfico 6.4 - Influência do parâmetro β

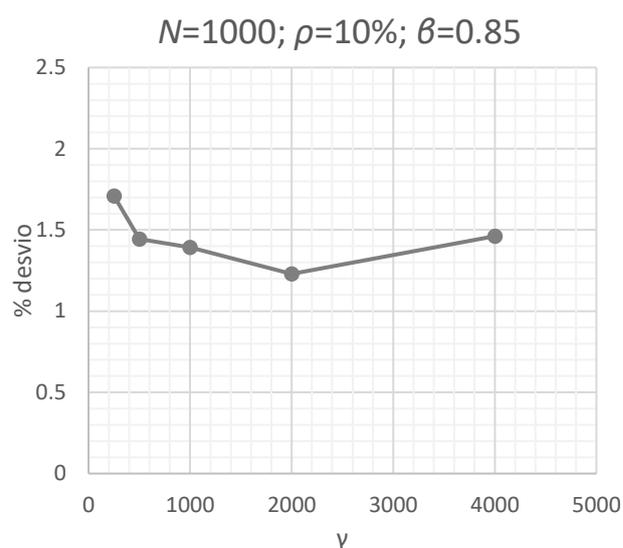


Gráfico 6.5 - Influência do parâmetro γ

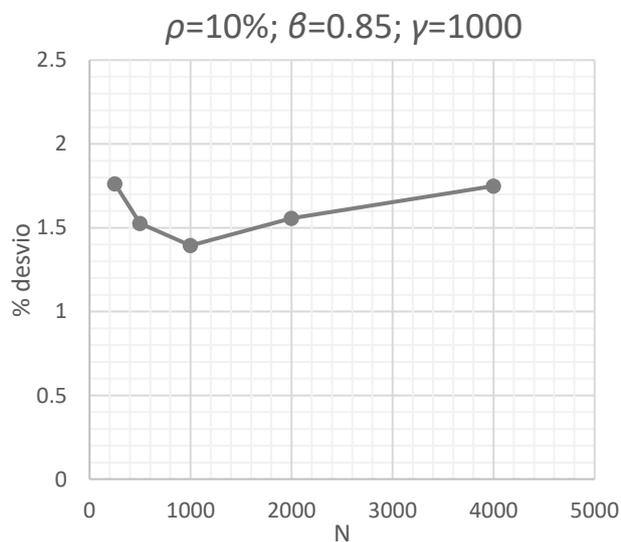


Gráfico 6.6 - Influência do parâmetro N

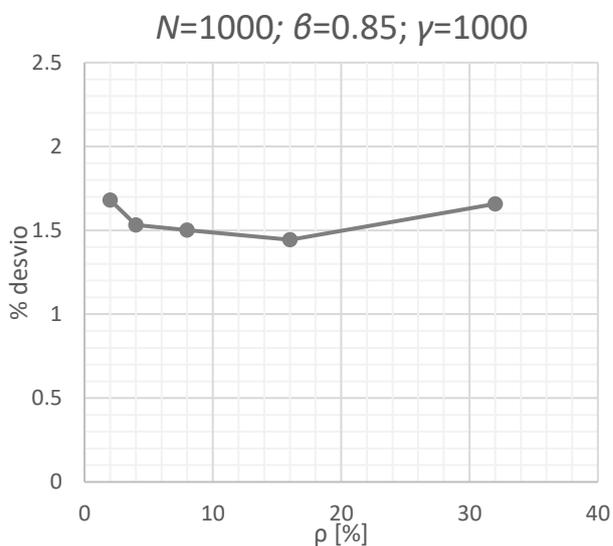


Gráfico 6.7 - Influência do parâmetro ρ

Como é possível notar, o formato da curva dos parâmetros γ , N e ρ é semelhante. Para os parâmetros N e ρ tal pode ser explicado pela ligação intrínseca com o número de soluções que é usado para atualizar a matriz de probabilidades ($\rho \cdot N$). Tal facto explicaria as semelhanças entre o Gráfico 6.6 e o Gráfico 6.7 na medida em que, se forem usadas poucas soluções para atualizar a matriz de probabilidades (independentemente de serem boas soluções ou não), não estará reunida informação suficiente para poder diversificar a procura, o que justificaria a quebra de eficácia no trecho inicial das curvas. Analogamente, usar demasiadas soluções para atualizar a matriz de probabilidades (desta vez, havendo soluções de diferentes qualidades), levará a que uma grande parte da informação transmitida ao algoritmo não seja relevante e acabará por impedir o algoritmo de fazer a intensificação desejada, tal justificaria a quebra de eficácia no trecho final das curvas.

Para o parâmetro γ é mais complicado estabelecer uma relação de causalidade, porém, aponta-se a seguinte hipótese que pode explicar a forma do Gráfico 6.5: para valores baixos do parâmetro γ o algoritmo não tem “tempo” suficiente para utilizar toda a informação da matriz de probabilidades e por esse motivo não consegue convergir para uma solução única (à semelhança do que acontece com o parâmetro μ_1 do SA). Para valores altos do parâmetro γ a atualização da matriz de probabilidades é efetuada menos vezes e é demasiado abrupta, o que pode levar à instabilização do algoritmo. É necessário um valor do parâmetro γ suficientemente alto para que o algoritmo utilize adequadamente a informação de cada matriz de probabilidades, mas não deve ser exageradamente alto sob o risco de “confundir” o algoritmo.

O parâmetro β aparenta ser o que menor relevância tem para garantir a qualidade das soluções finais, porém, é notada uma ligeira melhoria quando o parâmetro β é aumentado até valores próximos de 0.85 (para o caso HAN e a parametrização dada). Como já foi referenciado na subsecção 4.7.1, este parâmetro está diretamente relacionado com a diversificação do espaço de procura, sendo que valores mais baixos incentivam a diversificação e valores mais altos incentivam a intensificação.

O Gráfico 6.8 e o Gráfico 6.9 mostram o andamento do algoritmo (novamente para o estudo de caso HAN) para valores extremos dos parâmetros ρ e N respetivamente. Os restantes parâmetros não apresentaram mudanças visíveis no andamento do algoritmo, pelo que não são aqui expostos.

No Gráfico 6.8 é possível distinguir o andamento do algoritmo para valores extremos do parâmetro ρ . A zona vermelha, correspondente a um valor de ρ baixo, encontra-se ligeiramente abaixo da zona verde que corresponde a um valor de ρ alto, este facto é mais facilmente perceptível observando a média móvel (com 2500 pontos), disponibilizada no Gráfico 6.8, para ambos os valores de ρ . De igual modo, é possível observar no Gráfico 6.9 que a certa altura, o andamento do algoritmo para uma simulação realizada com um parâmetro N baixo é ultrapassado pelo andamento do algoritmo para uma simulação realizada com um parâmetro N

alto (próximo das 30 000 iterações). Tal facto é mais facilmente perceptível observando, novamente, as médias móveis disponibilizadas para o efeito. Este último caso salienta a importância que uma adaptação dos parâmetros, aquando do progresso do algoritmo, terá para melhorar o desempenho do mesmo.

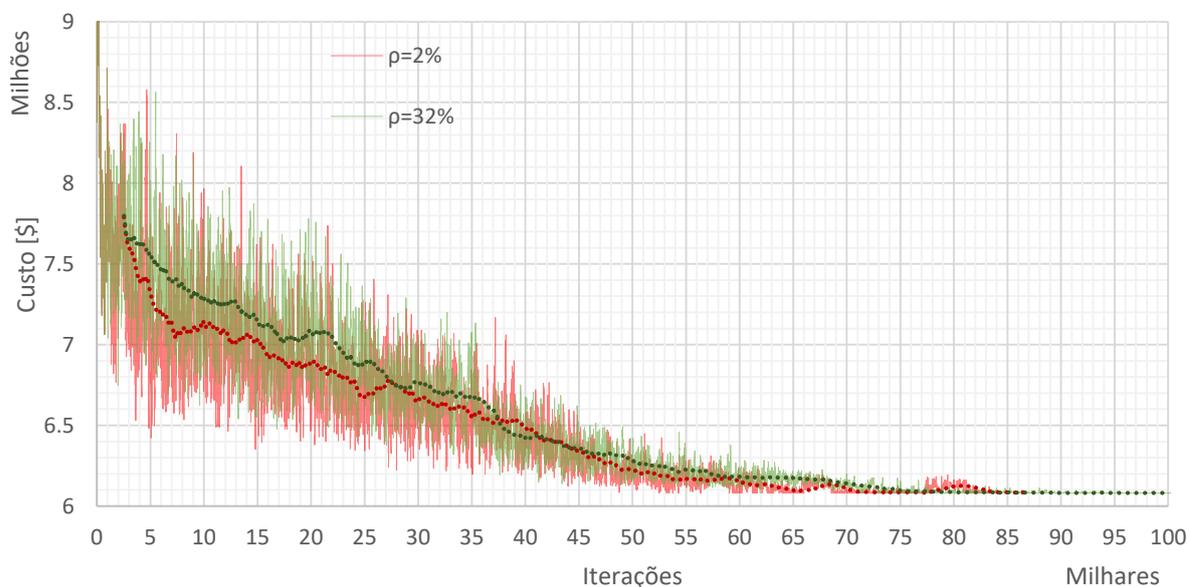


Gráfico 6.8 - Influência do parâmetro ρ no andamento do algoritmo SACER

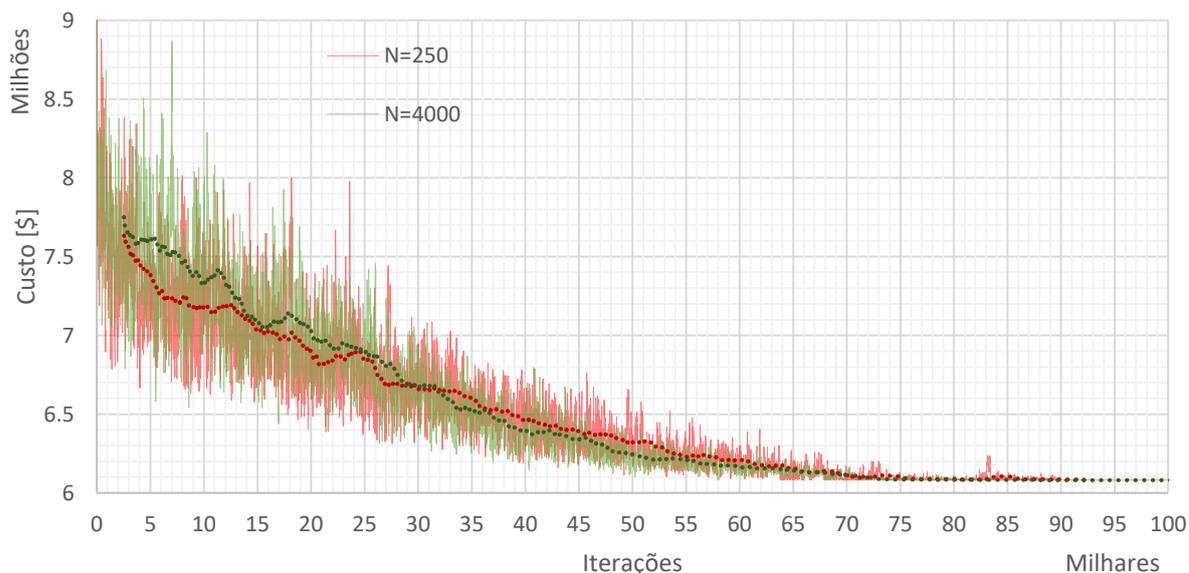


Gráfico 6.9 - Influência do parâmetro N no andamento do algoritmo SACER

No Gráfico 6.10 são sobrepostos o andamento do algoritmo SAR e SACER para o caso HAN. De igual forma, no Gráfico 6.11 são sobrepostos o andamento do algoritmo SAR e SACER para o caso FOS sem utilizar a solução inicial disponível (para uma melhor comparação).

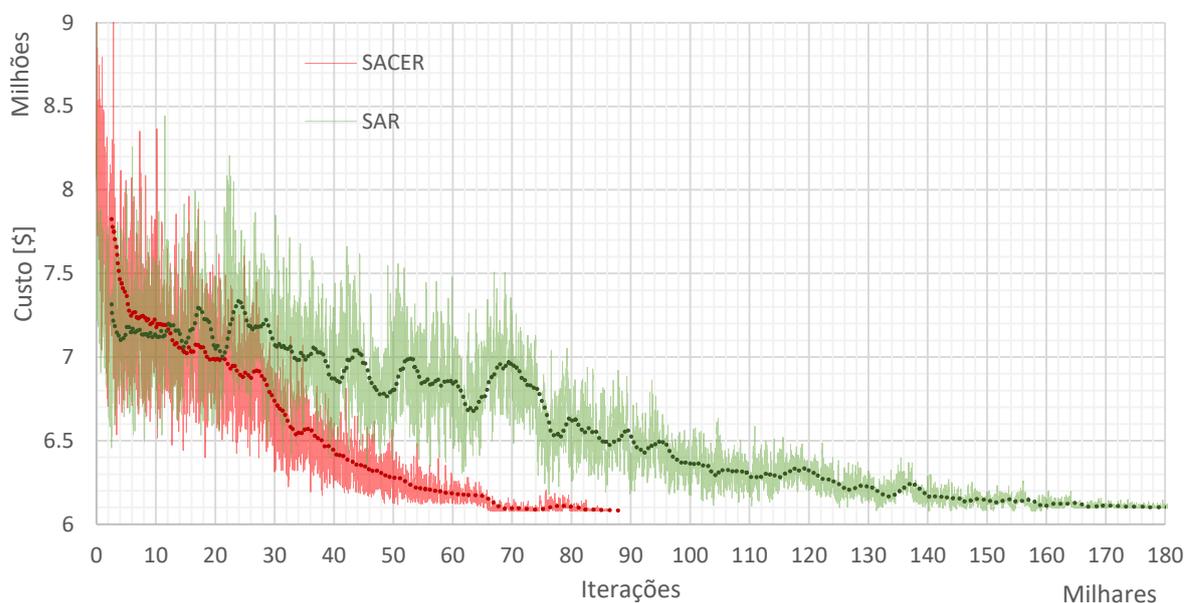


Gráfico 6.10 - Andamento dos algoritmos SACER e SAR para o caso HAN

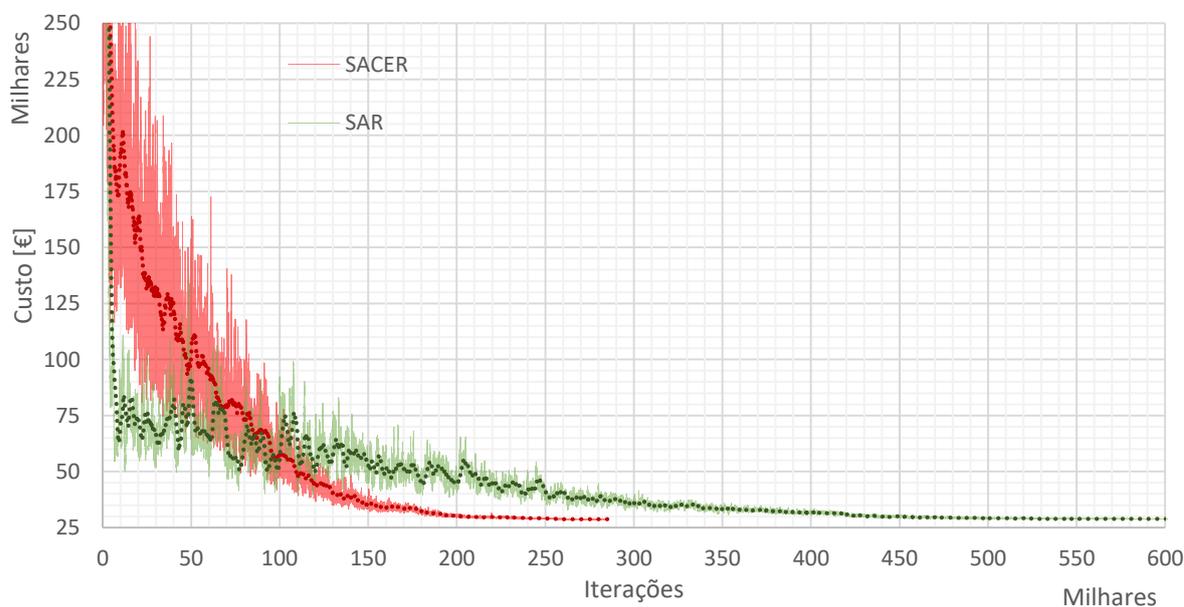


Gráfico 6.11 - Andamento dos algoritmos SACER e SAR para o caso FOS

A partir do Gráfico 6.10 e do Gráfico 6.11 é possível notar que o algoritmo SAR fornece melhores soluções nas iterações iniciais, mas a situação inverte-se rapidamente a favor do algoritmo SACER.

7 CONCLUSÃO

Nesta dissertação propôs-se o uso de uma metaheurística híbrida para resolver diversos problemas de dimensionamento de SDA bem conhecidos da literatura. Utilizou-se informação probabilística da CE para tentar reduzir o espaço de procura de cada problema e consequentemente melhorar a eficiência computacional. O objetivo principal era o de reduzir o número de soluções avaliadas pelo algoritmo e consequente tempo de cálculo sem perder a qualidade da solução final. Os resultados obtidos validam a aplicabilidade do novo método proposto, principalmente para problemas de grandes dimensões. Verificou-se existir uma redução significativa do espaço de procura dos problemas com uma mínima perturbação da qualidade da solução final.

7.1 SA vs. SAR

Como já tinha sido referido, a introdução da Segunda Fase no SA permite ao algoritmo “escapar” de eventuais ótimos locais no final da simulação. Os resultados obtidos vão ao encontro daquilo que era esperado, ou seja, a eficácia do algoritmo é melhorada a custo de uma perda na sua eficiência. Para problemas de pequena dimensão a melhoria acaba por ser insignificante, porém, o mesmo não pode ser dito quando estão em causa problemas de grande dimensão. Desta forma, cabe ao analista decidir em que situações compensa, ou não, utilizar a Segunda Fase.

7.2 SAR vs. SACER

Os resultados obtidos no capítulo anterior permitem concluir que o algoritmo híbrido proposto é realmente mais eficiente, principalmente em problemas de grande dimensão. Em problemas de pequena dimensão o algoritmo SACER perde um pouco a sua utilidade, uma vez que os métodos existentes já permitem resolver esse tipo de problemas de uma forma bastante eficiente. Não obstante ao disposto, o algoritmo SACER pode ainda ser aplicado a esses problemas sem prejuízos avultados.

No geral, o algoritmo híbrido proposto apresenta resultados que justificam a continuação do desenvolvimento do mesmo. Estudos mais específicos acerca dos novos parâmetros introduzidos no algoritmo devem ser realizados futuramente.

7.3 Trabalhos Futuros

Em relação à forma como o algoritmo foi implementado:

- O algoritmo híbrido apresentado nesta dissertação foi desenvolvido mantendo o foco na problemática da redução do espaço de procura (subsecção 2.3.2). Apesar de se ter conseguido alguns bons resultados, os mesmos foram conseguidos à custa de uma parametrização efetuada por tentativa e erro que se mantém constante no decorrer do algoritmo (adaptação *off-line*). Como já foi referido anteriormente (subsecção 2.3.4), os parâmetros de um algoritmo devem ser afinados durante o progresso da própria simulação de acordo com a informação recolhida pelo algoritmo (adaptação *on-line*), desta forma o algoritmo para além de ganhar uma maior aptidão para encontrar soluções ótimas, tornar-se-á efetivamente mais robusto e fiável. Como trabalho futuro, sugere-se que seja realizado um estudo mais aprofundado acerca da influência que cada parâmetro tem no progresso do algoritmo por forma a que sejam implementadas regras de adaptação dos novos parâmetros no mesmo;
- Devido à falta de experiência do autor na área da programação, crê-se que as alterações efetuadas no algoritmo possam ter sido programadas de um modo um pouco forçado. Tal como é referido na subsecção 2.3.8, o tempo de execução do algoritmo depende da facilidade com que o algoritmo é implementado. Desta forma, sugere-se uma revisão e eventuais alterações ao código do algoritmo como trabalho futuro;
- A hibridização através da CE é apenas uma de inúmeras opções disponíveis. Novas hibridizações (fundamentadas) deverão ser propostas e testadas.

Em relação ao tipo de problemas analisados:

- Nesta dissertação foram analisados problemas de dimensionamento otimizado de SDA onde apenas era necessário definir os diâmetros das condutas, contudo, num SDA existem mais elementos que podem necessitar de dimensionamento tais como estações elevatórias e/ou reservatórios e que contribuem significativamente para o custo da solução final. Sugere-se, como trabalho futuro, que o modelo de otimização seja modificado para ter em conta os outros elementos participantes num SDA;
- O modelo de otimização utilizado nesta dissertação é uniobjetivo. Como trabalho futuro, pondere-se alterar o modelo para ter em conta outros objetivos que permitam aos SDA tornarem-se menos vulneráveis às incertezas que os rodeiam e mais eficientes do ponto de vista ambiental.

De forma mais genérica, num futuro próximo, a comunidade científica que está diretamente ligada à otimização de problemas de gestão de recursos hídricos deve preocupar-se em criar um repositório (*on-line*) dos principais problemas estudados, categorizados por área de aplicação (*e.g.* dimensionamento de SDA; configuração de SDA; exploração de sistemas aquíferos; ...) e por dificuldade. Esses problemas devem (Maier *et al.*, 2014):

- Representar situações reais ou muito próximas da realidade;

- Considerar os diversos interesses de todas as partes envolvidas;
- Considerar as principais fontes de incerteza;
- Ser avaliados de forma consistente.

Em relação ao último ponto, um dos problemas encontrados no desenvolvimento desta dissertação está relacionado com a incerteza proveniente das leis de resistência. Como foi referenciado no cap. 5, os primeiros 3 estudos de caso (TLN, HAN, FOS) utilizam a fórmula empírica de Hazen-Williams para descrever a resistência ao escoamento provocada pelo atrito entre as partículas fluídas de água e entre as partículas de água e as paredes das condutas. Devido à utilização de diferentes coeficientes e expoentes da fórmula de Hazen-Williams por parte de diversos autores, tem-se como consequência que, a melhor solução encontrada por alguns não satisfará as restrições do modelo de outros (facto salientado em Cunha & Sousa, 2001). Por esse motivo, é difícil comparar diretamente as melhores soluções encontradas na literatura. Um esforço deve ser efetuado para que juntamente com a definição do problema sejam definidos os coeficientes e expoentes de fórmulas empíricas a utilizar na resolução dos mesmos.

REFERÊNCIAS BIBLIOGRÁFICAS

- Aarts, E. H. L., Korst, J. H. M., & van Laarhoven, P. J. M. (1997). Simulated Annealing. In E. Aarts & J. K. Lenstra (Eds.), *Local search in combinatorial optimization* (1st ed., pp. 91–120). New York, NY, USA: John Wiley & Sons, Ltd.
- Alperovits, E., & Shamir, U. (1977). Design of optimal water distribution systems. *Water Resources Research*, *13*(6), 885–900. <https://doi.org/10.1029/WR013i006p00885>
- Babbar-Sebens, M., & Minsker, B. (2008). Standard Interactive Genetic Algorithm—Comprehensive Optimization Framework for Groundwater Monitoring Design. *Journal of Water Resources Planning and Management*, *134*(6), 538–547. [https://doi.org/10.1061/\(ASCE\)0733-9496\(2008\)134:6\(538\)](https://doi.org/10.1061/(ASCE)0733-9496(2008)134:6(538))
- Babbar-Sebens, M., & Minsker, B. (2010). A Case-Based Micro Interactive Genetic Algorithm (CBMIGA) for interactive learning and search: Methodology and application to groundwater monitoring design. *Environmental Modelling & Software*, *25*(10), 1176–1187. <https://doi.org/10.1016/j.envsoft.2010.03.027>
- Babbar-Sebens, M., & Minsker, B. S. (2012). Interactive Genetic Algorithm with Mixed Initiative Interaction for multi-criteria ground water monitoring design. *Applied Soft Computing Journal*, *12*(1), 182–195. <https://doi.org/10.1016/j.asoc.2011.08.054>
- Bahrami, S., Doulati Ardejani, F., & Baafi, E. (2016). Application of artificial neural network coupled with genetic algorithm and simulated annealing to solve groundwater inflow problem to an advancing open pit mine. *Journal of Hydrology*, *536*, 471–484. <https://doi.org/10.1016/j.jhydrol.2016.03.002>
- Bellman, R. (1957). *Dynamic Programming*. Princeton University Press.
- Bi, W., Dandy, G. C., & Maier, H. R. (2015). Improved genetic algorithm optimization of water distribution system design by incorporating domain knowledge. *Environmental Modelling & Software*, *69*, 370–381. <https://doi.org/10.1016/j.envsoft.2014.09.010>
- Blum, C., Puchinger, J., Raidl, G. R., & Roli, A. (2011). Hybrid metaheuristics in combinatorial optimization: A survey. *Applied Soft Computing*, *11*(6), 4135–4151. <https://doi.org/10.1016/j.asoc.2011.02.032>
- Blum, C., & Roli, A. (2003). Metaheuristics in combinatorial optimization: overview and conceptual comparison. *ACM Computing Surveys*, *35*(3), 189–213. <https://doi.org/10.1007/s10479-005-3971-7>
- Bragalli, C., D’Ambrosio, C., Lee, J., Lodi, A., & Toth, P. (2008). Water Network Design by MINLP. *IBM Research*, RC24495.

- Broad, D. R., Dandy, G. C., & Maier, H. R. (2005). Water Distribution System Optimization Using Metamodels. *Journal of Water Resources Planning and Management*, 131(3), 172–180. [https://doi.org/10.1061/\(ASCE\)0733-9496\(2005\)131:3\(172\)](https://doi.org/10.1061/(ASCE)0733-9496(2005)131:3(172))
- Castelletti, A., Galelli, S., Restelli, M., & Soncini-Sessa, R. (2012). Data-driven dynamic emulation modelling for the optimal management of environmental systems. *Environmental Modelling & Software*, 34, 30–43. <https://doi.org/10.1016/j.envsoft.2011.09.003>
- Černý, V. (1985). Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm. *Journal of Optimization Theory and Applications*, 45(1), 41–51. <https://doi.org/10.1007/BF00940812>
- Cisty, M., Bajtek, Z., & Celar, L. (2017). A two-stage evolutionary optimization approach for an irrigation system design. *Journal of Hydroinformatics*, 19(1), 115–122. <https://doi.org/10.2166/hydro.2016.032>
- Collette, Y., & Siarry, P. (2005). Three new metrics to measure the convergence of metaheuristics towards the Pareto frontier and the aesthetic of a set of solutions in biobjective optimization. *Computers and Operations Research*, 32(4), 773–792. <https://doi.org/10.1016/j.cor.2003.08.017>
- Coloni, A., Dorigo, M., Maffioli, F., Maniezzo, V., Righini, G., & Trubian, M. (1996). Heuristics from Nature for Hard Combinatorial Optimization Problems. *International Transactions in Operational Research*. <https://doi.org/10.1111/j.1475-3995.1996.tb00032.x>
- Cunha, M. C. (1999). On solving aquifer management problems with simulated annealing algorithms. *Water Resources Management*, 13(3), 153–169. <https://doi.org/10.1023/A:1008149626428>
- Cunha, M. C., & Ribeiro, L. (2004). Tabu search algorithms for water network optimization. *European Journal of Operational Research*, 157(3), 746–758. [https://doi.org/10.1016/S0377-2217\(03\)00242-X](https://doi.org/10.1016/S0377-2217(03)00242-X)
- Cunha, M. C., & Sousa, J. (1999). Water Distribution Network Design Optimization: Simulated Annealing Approach. *Journal of Water Resources Planning and Management*, 125(4), 215–221. [https://doi.org/10.1061/\(ASCE\)0733-9496\(1999\)125:4\(215\)](https://doi.org/10.1061/(ASCE)0733-9496(1999)125:4(215))
- Cunha, M. C., & Sousa, J. (2001). Hydraulic Infrastructures Design Using Simulated Annealing. *Journal of Infrastructure Systems*, 7(1), 32–39. [https://doi.org/10.1061/\(ASCE\)1076-0342\(2001\)7:1\(32\)](https://doi.org/10.1061/(ASCE)1076-0342(2001)7:1(32))
- Cunha, M. C., & Sousa, J. J. O. (2010). Robust Design of Water Distribution Networks for a Proactive Risk Management. *Journal of Water Resources Planning and Management*, 136(2), 227–236. [https://doi.org/10.1061/\(ASCE\)WR.1943-5452.0000029](https://doi.org/10.1061/(ASCE)WR.1943-5452.0000029)
- de Boer, P.-T., Kroese, D. P., Mannor, S., & Rubinstein, R. Y. (2005). A Tutorial on the Cross-Entropy Method. *Annals of Operations Research*, 134(1), 19–67. <https://doi.org/10.1007/s10479-005-5724-z>
- De Corte, A., & Sörensen, K. (2013). Optimisation of gravity-fed water distribution network design: A critical review. *European Journal of Operational Research*, 228(1), 1–10. <https://doi.org/10.1016/j.ejor.2012.11.046>

- Deb, K., & Goldberg, D. E. (1994). Sufficient conditions for deceptive and easy binary functions. *Annals of Mathematics and Artificial Intelligence*, 10(4), 385–408. <https://doi.org/10.1007/BF01531277>
- Deb, K., & Jain, S. (2002). Running performance Metrics for evolutionary multi-objective optimization. *Indian Institute of Technology Kanpur*, 13–20.
- Dougherty, D. E., & Marryott, R. A. (1991). Optimal Groundwater Management: 1. Simulated Annealing. *Water Resources Research*, 27(10), 2493–2508. <https://doi.org/10.1029/91WR01468>
- ERSAR. (2016). *Relatório Anual dos Serviços de Águas e Resíduos em Portugal 2016* (Vol. 1). Entidade Reguladora dos Serviços de Águas e Resíduos.
- Fanni, A., Liberatore, S., Sechi, G. M., Soro, M., & Zuddas, P. (2000). Optimization of water distribution systems by a Tabu Search metaheuristic. In *Computing tools for modeling, optimization and simulation* (pp. 279–298). Springer US. https://doi.org/10.1007/978-1-4615-4567-5_16
- Franssen, M. (2005). Arrow’s theorem, multi-criteria decision problems and multi-attribute preferences in engineering design. *Research in Engineering Design*, 16(1–2), 42–56. <https://doi.org/10.1007/s00163-004-0057-5>
- Fujiwara, O., & Khang, D. B. (1990). A two-phase decomposition method for optimal design of looped water distribution networks. *Water Resources Research*, 26(4 Apr.), 539–549. <https://doi.org/10.1029/WR026i004p00539>
- Gibbs, M. S., Maier, H. R., & Dandy, G. C. (2010). Comparison of Genetic Algorithm Parameter Setting Methods for Chlorine Injection Optimization. *Journal of Water Resources Planning and Management*, 136(2), 288–291. [https://doi.org/10.1061/\(ASCE\)WR.1943-5452.0000033](https://doi.org/10.1061/(ASCE)WR.1943-5452.0000033)
- Gibbs, M. S., Maier, H. R., & Dandy, G. C. (2011). Relationship between problem characteristics and the optimal number of genetic algorithm generations. *Engineering Optimization*, 43(4), 349–376. <https://doi.org/10.1080/0305215X.2010.491547>
- Glover, F. (1986). Future paths for integer programming and links to artificial intelligence. *Computers and Operations Research*, 13(5), 533–549. [https://doi.org/10.1016/0305-0548\(86\)90048-1](https://doi.org/10.1016/0305-0548(86)90048-1)
- Glover, F. (1989). Tabu Search—Part I. *ORSA Journal on Computing*, 1(3), 190–206. <https://doi.org/10.1287/ijoc.1.3.190>
- Glover, F. (1990). Tabu Search—Part II. *ORSA Journal on Computing*, 2(1), 4–32. <https://doi.org/10.1287/ijoc.2.1.4>
- Goldberg, D. E., Korb, B., & Deb, K. (1989). Messy Genetic Algorithms : Motivation , Analysis , and First Results. *Engineering*, 3, 493–530.
- Hadka, D., & Reed, P. (2013). Borg: An Auto-Adaptive Many-Objective Evolutionary Computing Framework. *Evolutionary Computation*, 21(2), 231–259. https://doi.org/10.1162/EVCO_a_00075

- Hansen, P., & Mladenovi, N. (2001). Variable neighborhood search: Principles and applications. *European Journal of Operational Research*, 130(3), 449–467. [https://doi.org/10.1016/S0377-2217\(00\)00100-4](https://doi.org/10.1016/S0377-2217(00)00100-4)
- Holland, J. H. (1975). *Adaptation in Natural and Artificial Systems*. University of Michigan Press.
- Hu, J., & Hu, P. (2011). Annealing adaptive search, cross-entropy, and stochastic approximation in global optimization. *Naval Research Logistics*, 58(5), 457–477. <https://doi.org/10.1002/nav.20462>
- Kallel, L. (1998). Inside {GA} Dynamics: Ground Basis for Comparison. In A. E. Eiben, T. Bäck, M. Schoenauer, & H.-P. Schwefel (Eds.), *{P}arallel {P}roblem {S}olving from {N}ature -- {PPSN V}* (pp. 57–66). Berlin, Heidelberg: Springer Berlin Heidelberg. <https://doi.org/10.1007/BFb0056849>
- Katada, Y., & Ohkura, K. (2006). Estimating the degree of neutrality in fitness landscapes by the Nei's standard genetic distance - An application to evolutionary robotics. In *Proceedings of the 2006 IEEE World Congress on Computational Intelligence* (pp. 1590–1597). <https://doi.org/10.1109/CEC.2006.1688349>
- Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983). Optimization by Simulated Annealing. *Science*, 220(4598), 671–680. <https://doi.org/10.1126/science.220.4598.671>
- Klau, G. W., Lesh, N., Marks, J., & Mitzenmacher, M. (2010). Human-guided search. *Journal of Heuristics*, 16(3), 289–310. <https://doi.org/10.1007/s10732-009-9107-5>
- Knowles, J., & Corne, D. (2002). On metrics for comparing nondominated sets. In *Proceedings of the 2002 Congress on Evolutionary Computation. CEC'02 (Cat. No.02TH8600)* (Vol. 1, pp. 711–716). IEEE. <https://doi.org/10.1109/CEC.2002.1007013>
- Kollat, J. B., & Reed, P. (2007). A framework for Visually Interactive Decision-making and Design using Evolutionary Multi-objective Optimization (VIDEO). *Environmental Modelling & Software*, 22(12), 1691–1704. <https://doi.org/10.1016/j.envsoft.2007.02.001>
- Kollat, J. B., Reed, P. M., & Kasprzyk, J. R. (2008). A new epsilon-dominance hierarchical Bayesian optimization algorithm for large multiobjective monitoring network design problems. *Advances in Water Resources*, 31(5), 828–845. <https://doi.org/10.1016/j.advwatres.2008.01.017>
- Lenin, K., Ravindhranath Reddy, B., & Suryakalavathi, M. (2016). Hybrid Tabu search-simulated annealing method to solve optimal reactive power problem. *International Journal of Electrical Power & Energy Systems*, 82, 87–91. <https://doi.org/10.1016/j.ijepes.2016.03.007>
- López-Ibáñez, M., Paquete, L., & Stützle, T. (2010). Exploratory Analysis of Stochastic Local Search Algorithms in Biobjective Optimization. In T. Bartz-Beielstein, M. Chiarandini, L. Paquete, & M. Preuss (Eds.), *Experimental Methods for the Analysis of Optimization Algorithms* (pp. 209–222). Berlin, Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-02538-9_9
- Loucks, D. P. (2012). Water resource management modeling in 2050. In *Toward a Sustainable Water Future: Visions for 2050* (pp. 341–349).

- Maier, H. R., Kapelan, Z., Kasprzyk, J., Kollat, J., Matott, L. S., Cunha, M. C., ... Reed, P. M. (2014). Evolutionary algorithms and other metaheuristics in water resources: Current status, research challenges and future directions. *Environmental Modelling and Software*, 62, 271–299. <https://doi.org/10.1016/j.envsoft.2014.09.013>
- Manual do utilizador EPANET. (2002). Manual do utilizador EPANET - Versão 2.0 traduzida e adaptada para língua portuguesa. LNEC. Lisboa, Portugal.
- Marques, J. A. A. de S., & Sousa, J. J. de O. (2011). *Hidráulica Urbana - Sistemas de Abastecimento de Água e de Drenagem de Águas Residuais* (3ª ed.). Coimbra: Imprensa da Universidade de Coimbra.
- Marques, J., Cunha, M., & Savić, D. (2015a). Using Real Options in the Optimal Design of Water Distribution Networks. *Journal of Water Resources Planning and Management*, 141(2), 4014052. [https://doi.org/10.1061/\(ASCE\)WR.1943-5452.0000448](https://doi.org/10.1061/(ASCE)WR.1943-5452.0000448)
- Marques, J., Cunha, M., & Savić, D. A. (2015b). Using real options for an eco-friendly design of water distribution systems. *Journal of Hydroinformatics*, 17(1), 20. <https://doi.org/10.2166/hydro.2014.122>
- Mays, L. W. (1996). The Role of Risk Analysis in Water Resources Engineering. *Journal of Contemporary Water Research and Education*, 103(1), 8–12.
- McPhee, J., & Yeh, W. W.-G. (2008). Groundwater Management Using Model Reduction via Empirical Orthogonal Functions. *Journal of Water Resources Planning and Management*, 134(2), 161–170. [https://doi.org/10.1061/\(ASCE\)0733-9496\(2008\)134:2\(161\)](https://doi.org/10.1061/(ASCE)0733-9496(2008)134:2(161))
- Merz, P. (2004). Advanced fitness landscape analysis and the performance of memetic algorithms. *Evolutionary Computation*, 12(3), 303–325. <https://doi.org/10.1162/1063656041774956>
- Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., & Teller, E. (1953). Equation of state calculations by fast computing machines. *Journal Chemical Physics*, 21(6), 1087–1092. <https://doi.org/10.1063/1.1699114>
- Mladenović, N., & Hansen, P. (1997). Variable neighborhood search. *Computers & Operations Research*, 24(11), 1097–1100. [https://doi.org/10.1016/S0305-0548\(97\)00031-2](https://doi.org/10.1016/S0305-0548(97)00031-2)
- Moscato, P. (1989). On evolution, search, optimization, genetic algorithms and martial arts: Towards memetic algorithms. *Caltech Concurrent Computation Program, C3P Report*, 826.
- Niknam, T., Narimani, M. R., & Jabbari, M. (2013). Dynamic optimal power flow using hybrid particle swarm optimization and simulated annealing. *International Transactions on Electrical Energy Systems*, 23(7), 975–1001. <https://doi.org/10.1002/etep.1633>
- Perelman, L., & Ostfeld, A. (2007). An adaptive heuristic cross-entropy algorithm for optimal design of water distribution systems. *Engineering Optimization*, 39(4), 413–428. <https://doi.org/10.1080/03052150601154671>
- Pirlot, M. (1996). General local search methods. *European Journal of Operational Research*, 92(3), 493–511. [https://doi.org/10.1016/0377-2217\(96\)00007-0](https://doi.org/10.1016/0377-2217(96)00007-0)

- Raidl, G. R. (2006). A Unified View on Hybrid Metaheuristics. In F. Almeida, M. J. Blesa Aguilera, C. Blum, J. M. Moreno Vega, M. Pérez Pérez, A. Roli, & M. Sampels (Eds.), *Hybrid Metaheuristics: Third International Workshop, HM 2006 Gran Canaria, Spain, October 13-14, 2006 Proceedings* (pp. 1–12). Berlin, Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/11890584_1
- Razavi, S., Tolson, B. A., & Burn, D. H. (2012). Review of surrogate modeling in water resources. *Water Resources Research*, *48*(7). <https://doi.org/10.1029/2011WR011527>
- Reca, J., & Martínez, J. (2006). Genetic algorithms for the design of looped irrigation water distribution networks. *Water Resources Research*, *42*(5). <https://doi.org/10.1029/2005WR004383>
- Rubinstein, R. Y. (1997). Optimization of computer simulation models with rare events. *European Journal of Operational Research*, *99*(1), 89–112. [https://doi.org/10.1016/S0377-2217\(96\)00385-2](https://doi.org/10.1016/S0377-2217(96)00385-2)
- Savic, D. A., & Walters, G. A. (1997). Genetic Algorithms for Least-Cost Design of Water Distribution Networks. *Journal of Water Resources Planning and Management*, *123*(2), 67–77. [https://doi.org/10.1061/\(ASCE\)0733-9496\(1997\)123:2\(67\)](https://doi.org/10.1061/(ASCE)0733-9496(1997)123:2(67))
- Schaake, J., & Lai, F. H. (1969). Linear programming and dynamic programming application to water distribution network design. *M.I.T. -Dept Civ Eng-Hydrodynamics Laboratory-Report 116*, 222.
- Schraudolph, N. N., & Belew, R. K. (1992). Dynamic Parameter Encoding for Genetic Algorithms. *Machine Learning*, *9*(1), 9–21. <https://doi.org/10.1023/A:1022624728869>
- Sivanandam, S. N., & Deepa, S. N. (2008). *Introduction to Genetic Algorithms*. Vasa. https://doi.org/10.1007/978-3-540-73190-0_2
- Srinivas, N., & Deb, K. (1994). Multiobjective Optimization Using Nondominated Sorting in Genetic Algorithms. *Evolutionary Computation*, *2*(3), 221–248. <https://doi.org/10.1162/evco.1994.2.3.221>
- Tu, M., Tsai, F., & Yeh, W. (2005). Optimization of Water Distribution and Water Quality by Hybrid Genetic Algorithm. *Journal of Water Resources Planning and Management*, *131*(6), 431–440. [https://doi.org/doi:10.1061/\(ASCE\)0733-9496\(2005\)131:6\(431\)](https://doi.org/doi:10.1061/(ASCE)0733-9496(2005)131:6(431))
- UECWS@. (2017). University of Exeter, Centre for Water Systems. Retrieved July 4, 2017, from <http://emps.exeter.ac.uk/engineering/research/cws/resources/benchmarks/>
- Van Veldhuizen, D. A. (1999). *Multiobjective Evolutionary Algorithms: Classifications, Analyses, and New Innovations*. Ph.D. Thesis. Air Force Institute of Technology, Wright Patterson AFB, OH, USA.
- Van Veldhuizen, D. a, & Lamont, G. B. (1998). Evolutionary Computation and Convergence to a Pareto Front. In *Late Breaking Papers at the Genetic Programming 1998 Conference* (pp. 221–228).
- Wolpert, D. H., & Macready, W. G. (1997). No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, *1*(1), 67–82. <https://doi.org/10.1109/4235.585893>

- WRI@. (2015). Ranking the World's Most Water-Stressed Countries in 2040 | World Resources Institute. Retrieved June 27, 2017, from <http://www.wri.org/blog/2015/08/ranking-world's-most-water-stressed-countries-2040>
- Xiao, J., Yang, H., Zhang, C., Zheng, L., & Gupta, J. N. D. (2015). A hybrid Lagrangian-simulated annealing-based heuristic for the parallel-machine capacitated lot-sizing and scheduling problem with sequence-dependent setup times. *Computers & Operations Research*, *63*, 72–82. <https://doi.org/10.1016/j.cor.2015.04.010>
- Xiao, Y., Zhao, Q., Kaku, I., & Mladenovic, N. (2014). Variable neighbourhood simulated annealing algorithm for capacitated vehicle routing problems. *Engineering Optimization*, *46*(4), 562–579. <https://doi.org/10.1080/0305215X.2013.791813>
- Yan, Y., & Wang, G. (2007). A job shop scheduling approach based on simulation optimization. In *2007 IEEE International Conference on Industrial Engineering and Engineering Management* (pp. 1816–1822). IEEE. <https://doi.org/10.1109/IEEM.2007.4419506>
- Zeferino, J. A., Antunes, A. P., & Cunha, M. C. (2009). An Efficient Simulated Annealing Algorithm for Regional Wastewater System Planning. *Computer-Aided Civil and Infrastructure Engineering*, *24*(5), 359–370. <https://doi.org/10.1111/j.1467-8667.2009.00594.x>
- Zielinski, K., Peters, D., & Laur, R. (2005). Stopping criteria for single-objective optimization. In *Proceedings of the Third International Conference on Computational Intelligence, Robotics and Autonomous Systems*.
- Zitzler, E. (1999). *Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications*. Ph.D. Thesis. Swiss Federal Institute of Technology, Zurich, Switzerland.
- Zitzler, E., Deb, K., & Thiele, L. (2000). Comparison of Multiobjective Evolutionary Algorithms: Empirical Results. *Evolutionary Computation*, *8*(2), 173–195. <https://doi.org/10.1162/106365600568202>
- Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C. M., & da Fonseca, V. G. (2003). Performance assessment of multiobjective optimizers: an analysis and review. *IEEE Transactions on Evolutionary Computation*, *7*(2), 117–132. <https://doi.org/10.1109/TEVC.2003.810758>