

**UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA CIVIL E AMBIENTAL**

**ESTUDO DE UM MODELO DE OTIMIZAÇÃO DOS
CUSTOS DE ENERGIA ELÉTRICA DO BOMBEAMENTO
EM SISTEMA DE ABASTECIMENTO DE ÁGUA**

ANGÉLICA LUCIANA BARROS DE CAMPOS

ORIENTADOR: SÉRGIO KOIDE

CO-ORIENTADOR: WELITOM TTATOM PEREIRA DA SILVA

**DISSERTAÇÃO DE MESTRADO EM TECNOLOGIA AMBIENTAL E
RECURSOS HÍDRICOS**

BRASÍLIA/DF: AGOSTO – 2017

UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA CIVIL

ESTUDO DE UM MODELO DE OTIMIZAÇÃO DOS CUSTOS DE
ENERGIA ELÉTRICA DO BOMBEAMENTO EM SISTEMA DE
ABASTECIMENTO DE ÁGUA

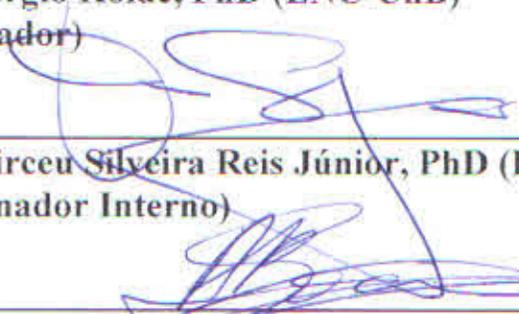
ANGÉLICA LUCIANA BARROS DE CAMPOS

DISSERTAÇÃO SUBMETIDA AO DEPARTAMENTO DE
ENGENHARIA CIVIL E AMBIENTAL DA FACULDADE DE
TECNOLOGIA DA UNIVERSIDADE DE BRASÍLIA COMO PARTE
DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU
DE MESTRE EM TECNOLOGIA AMBIENTAL E RECURSOS
HÍDRICOS.

APROVADA POR:



Prof. Sérgio Koide, PhD (ENC-UnB)
(Orientador)



Prof. Dirceu Silveira Reis Júnior, PhD (ENC-UnB)
(Examinador Interno)

Prof. Alexandre Kepler Soares, Doutor (UFG)
(Examinador Externo)

BRASÍLIA/DF, 31 DE AGOSTO DE 2017

FICHA CATALOGRÁFICA

CAMPOS, ANGÉLICA LUCIANA BARROS DE

Estudo de um modelo de otimização dos custos de energia elétrica do bombeamento em sistema de abastecimento de água [Distrito Federal] 2017.

xiv, 152p., 210 x 297 mm (ENC/FT/UnB, Mestre, Tecnologia Ambiental e Recursos Hídricos, 2017).

Dissertação de Mestrado – Universidade de Brasília. Faculdade de Tecnologia.

Departamento de Engenharia Civil e Ambiental.

1. Abastecimento de água

2. Otimização operacional

3. Descoberto

I. ENC/FT/UnB

II. Título (série)

REFERÊNCIA BIBLIOGRÁFICA

CAMPOS., A. L. B. (2017). Estudo de um modelo de otimização dos custos de energia elétrica do bombeamento em sistema de abastecimento de água. Dissertação de Mestrado em Tecnologia Ambiental e Recursos Hídricos, Publicação PTARH.DM-205/2017, Departamento de Engenharia Civil e Ambiental, Universidade de Brasília, Brasília, DF, 152p.

CESSÃO DE DIREITOS

AUTOR: Angélica Luciana Barros de Campos.

TÍTULO: Estudo de um modelo de otimização dos custos de energia elétrica do bombeamento em sistema de abastecimento de água.

GRAU: Mestre

ANO: 2017

É concedida à Universidade de Brasília permissão para reproduzir cópias desta dissertação de mestrado e para emprestar ou vender tais cópias somente para propósitos acadêmicos e científicos. O autor reserva outros direitos de publicação e nenhuma parte dessa dissertação de mestrado pode ser reproduzida sem autorização por escrito do autor.

Angélica Luciana Barros de Campos
angelbarrosdecampos@hotmail.com

DEDICATÓRIA

A minha mãe, que apesar de tudo nunca me deixou desistir dos meus sonhos.

AGRADECIMENTOS

Agradeço primeiramente a DEUS, pois sem ele nada disso seria possível.

A minha mãe Maria Lúcia, meu pai Saturnino Avelino, meus irmãos Anderson e Angélison, minha avó Simiana Brizida, meu avô Manoel de Barros, meu namorado Cristhian Froes e toda a minha família pelo apoio e incentivo e, por sempre acreditarem em mim.

Aos meus queridos amigos Fernanda Lemos, Renei Carvalho e Greison Santos membros da “República do PTARH”, pelos bons momentos, companheirismo, amizade e as boas risadas.

Ao meu orientador Sergio Koide pela paciência e conversas durante todo o período de mestrado.

Ao professor Welitom Ttatom pelas conversas encorajadoras e pelo incentivo desde a época da graduação.

Aos professores do PTARH por todos os ensinamentos durante o mestrado, ao pessoal do administrativo do PTARH, sempre dispostos a ajudar e a banca do mestrado pelas boas sugestões.

A Junia Porto pela amizade e pela imensa ajuda na reta final do mestrado.

Aos meus colegas e amigos do PTARH pelo auxílio nas disciplinas.

A equipe da “Produção de água” e “Gestão Operacional” da Caesb que contribuíram para a realização deste trabalho, especialmente ao Diogo Gebrim, pela grande ajuda no desenvolvimento do código computacional e Luiz Carlos Itonaga pelas dicas valiosas e pela participação na banca do seminário de dissertação.

A CAPES pela concessão da bolsa de mestrado e a FAPDF pelos auxílios financeiros.

E a todos que de alguma forma contribuíram para a concretização dessa dissertação.

RESUMO

ESTUDO DE UM MODELO DE OTIMIZAÇÃO DOS CUSTOS DE ENERGIA ELÉTRICA DO BOMBEAMENTO EM SISTEMA DE ABASTECIMENTO DE ÁGUA

Cerca de 90% das despesas com energia elétrica das empresas de saneamento são provenientes de motores elétricos utilizados em estações elevatórias de água. Assim, existe a necessidade de otimizar os custos operacionais do bombeamento em SAAs. Devido à complexidade da operação desses sistemas, a busca da regra operacional otimizada, deve ser feita com o auxílio de rotinas computacionais. No entanto, essa otimização necessita de grande tempo de processamento computacional, tornando-se um impasse na utilização do modelo para a operação real, como no caso do modelo desenvolvido por Gebrim (2013). Assim, o presente trabalho visou desenvolver um modelo de otimização tendo como base o modelo de Gebrim (2013), para definir regras operacionais que minimizem os custos operacionais de energia elétrica provenientes do bombeamento do sistema de abastecimento do Rio Descoberto, localizado no Distrito Federal. As soluções do modelo foram avaliadas a partir da função aptidão que considera além dos custos de energia, o atendimento às restrições operacionais expressas em termos de penalidade. Empregou-se no modelo estratégias como a utilização dos algoritmos de otimização *Dynamically Dimensioned Search*, Evolução Diferencial e Algoritmo Genético; simplificação do modelo hidráulico; tratamento das penalidades por meio de uma abordagem baseada na lógica *fuzzy*; e a técnica *seeding*, com a inserção de uma população inicial conhecida no algoritmo. Os resultados obtidos demonstraram que as estratégias utilizadas foram eficientes para encontrar soluções de menor custo e tempo de processamento se comparados ao modelo de otimização original, no entanto, o número de acionamentos das bombas e válvulas foi excessivo. Além disso, os níveis dos reservatórios gerados apresentaram elevada segurança em sua operação, o que pode encorajar a utilização do modelo pelos operadores do sistema. Portanto, com algumas adaptações como a redução do número de acionamentos e desenvolvimento de um previsor de demanda, o modelo poderá ser utilizado para a operação real.

Palavras-chaves: abastecimento de água, otimização operacional, Descoberto.

ABSTRACT

STUDY OF MODEL OPTIMIZATION OF ELECTRIC ENERGY COST OF PUMPING IN WATER SUPPLY SYSTEM

About 90% of the electricity costs from water companies are from pumps used in water pumping stations. Thus, there is a need to optimize the operating costs of pumping in water supply systems. Due to the complexity of the operation of water supply systems, the search for optimized operating rules must be accomplished with the aid of computational routines. However, this optimization requires a great amount of computational time, becoming an impasse in the use of the model for the real operation. Thus, this study aimed to develop an optimization model, to define operational rules that minimize the operational costs of electricity from the pumping of the Descoberto river water supply system, located in the Federal District – Brazil. Were evaluated the solutions of the model, from the fitness function that considers besides the energy costs, compliance with the operational constraints expressed in terms of penalty. Were used the model strategies such as Genetic Algorithm, Dynamically Dimensioned Search and Differential Evolution optimization algorithms; simplification of the hydraulic model; handling of penalties, through an approach based on fuzzy logic; and the seeding technique, with the insertion of an initial population known in the algorithm. The obtained results demonstrated that the strategies used were efficient to find solutions of lower cost and computational time when compared to the original optimization model, however, leading to excessive pump and valve switches. In addition, the reservoirs levels generated presented high safety in the operation of the reservoirs, which can encourage the use of the model by the system operators. Therefore, with some adaptations as the reduction of the number of pump and valve switches and the development of a demand forecast, the model can be used in the real operation.

Keywords: water supply systems, operational optimization, Descoberto river.

SUMÁRIO

1 - INTRODUÇÃO	1
2 - OBJETIVOS	5
2.1 - OBJETIVO GERAL	5
2.2 - OBJETIVOS ESPECÍFICOS.....	5
3 - FUNDAMENTAÇÃO TEÓRICA E REVISÃO BIBLIOGRÁFICA	6
3.1 - OPERAÇÃO DE SAAs	6
3.2 - OTIMIZAÇÃO DA OPERAÇÃO DE SAAs	7
3.2.1 - Regras operacionais de SAAs	9
3.2.2 - Técnicas de otimização aplicadas aos SAAs	11
3.2.3 - Planejamento operacional em nível estratégico e a operação em tempo real	29
3.2.4 - Representação da programação de bombas.....	35
3.2.5 - Método das penalidades	37
3.2.6 – Lógica <i>Fuzzy</i>	41
4 - METODOLOGIA	46
4.1 - CARACTERIZAÇÃO DO SAA DO RIO DISCOBERTO	46
4.2 - DESCRIÇÃO E IMPLEMENTAÇÃO DO MODELO DE OTIMIZAÇÃO SELECIONADO	54
4.3 - IMPLEMENTAÇÃO E SIMPLIFICAÇÃO DO MODELO HIDRÁULICO	63
4.4 - UTILIZAÇÃO DE ALGORITMOS DE OTIMIZAÇÃO	67
4.4.1 - Algoritmo genético	67
4.4.2 - Algoritmo DDS	70
4.4.3 - Algoritmo DE	73
4.4.4 - Parâmetros dos algoritmos	75
4.5 - TRATAMENTO DAS PENALIDADES	80
4.5.1 - Tratamento 1	80
4.5.2 - Tratamento 2	85
4.5.3 - Tratamento 3	87
4.6 - EMPREGO DA TÉCNICA <i>SEEDING</i>	89
5 - RESULTADOS E DISCUSSÕES	91
5.1 - UTILIZAÇÃO DOS ALGORITMOS DE OTIMIZAÇÃO	91
5.2 - SIMPLIFICAÇÃO DO MODELO HIDRÁULICO.....	99
5.3 - TRATAMENTO DAS PENALIDADES	105

5.4 - TÉCNICA <i>SEEDING</i>	112
5.4.1 - Regras operacionais	115
5.4.2 - Níveis dos reservatórios	120
6 - CONCLUSÕES E RECOMENDAÇÕES	125
REFERÊNCIAS BIBLIOGRÁFICAS	129

LISTA DE TABELAS

Tabela 3.1 - Otimização da operação de SAA.....	8
Tabela 3.2 – Publicações que utilizaram AEs para a otimização de SAAs (Zheng <i>et al.</i> , 2012, modificado).....	14
Tabela 3.3 – Soluções para a rede de 2 malhas (Suribabu e Neelakantan, 2006)	22
Tabela 3.4 – Soluções para a rede Hanoi (Suribabu e Neelakantan, 2006).....	23
Tabela 3.5 - Trabalhos desenvolvidos sobre operação em tempo real	30
Tabela 3.6 – Variáveis hidráulicas e suas respectivas funções de pertinência	43
Tabela 4.1 - Código das unidades operacionais do sistema Descoberto (Gebrim, 2013) ...	51
Tabela 4.2 – Tarifa de energia elétrica por elevatória analisada (Gebrim, 2013)	52
Tabela 4.3 - Variáveis de decisão do problema (Gebrim, 2013).....	55
Tabela 4.4 - Divisão dos equipamentos para aplicação da Penalidade 4	60
Tabela 4.5 – Funções da biblioteca EPANET- <i>Matlab-Toolkit</i> (Eliades e Kyriakou, 2016).....	64
Tabela 4.6 – Funções do MATLAB empregadas no algoritmo DDS	70
Tabela 4.7 – Coeficientes de penalidades adotados	76
Tabela 4.8 – Operadores e parâmetros genéticos adotados	77
Tabela 4.9 – Valores dos parâmetros DDS utilizados na calibração.....	77
Tabela 4.10 – Valores sugeridos dos parâmetros DE.....	78
Tabela 4.11 – Valores dos parâmetros DE utilizados na calibração	79
Tabela 4.12 – Cenários simulados para o tratamento 1	85
Tabela 4.13 – Grupos de equipamentos, limites e coeficientes de penalidades	87
Tabela 5.1 – Resultados dos testes de calibração dos parâmetros do DDS.....	91
Tabela 5.2 – Resultados dos testes de calibração dos parâmetros do DE	92
Tabela 5.3 – Comparação do desempenho dos algoritmos de otimização	94
Tabela 5.4 – Comparação entre os componentes do modelo completo e simplificado.....	99
Tabela 5.5 – Cálculo do erro médio absoluto	100
Tabela 5.6 – Comparação entre o modelo hidráulico completo e simplificado	102
Tabela 5.7 – Cenário sem tratamento e cenários simulados com o tratamento 1.....	105
Tabela 5.8 – Cenários sem tratamento das penalidades e com tratamento 1, 2 e 3.....	109
Tabela 5.9 – Simulações empregando diferentes soluções iniciais e a regra de referência	112
Tabela 5.10 – Simulação da solução 4 sem a penalidade 7	119

LISTA DE FIGURAS

Figura 3.1 – Fluxograma das regras operacionais em SAAs (Lacerda <i>et al.</i> , 2008, modificado).....	10
Figura 3.2 – Funcionamento básico do AG simples.....	15
Figura 3.3 – Esquema gráfico do cruzamento de um ponto (Pedrosa, 2006).....	16
Figura 3.4 – Esquema gráfico de mutação simples (Pedrosa, 2006).....	17
Figura 3.5 – Frentes de Pareto dos algoritmos multiobjetivo (Carrijo, 2004).....	18
Figura 3.6 – Funcionamento dos operadores mutação, cruzamento e seleção do DE (Storn e Price, 1997 <i>apud</i> Dandy <i>et al.</i> , 2010).....	24
Figura 3.7 – Ilustração do processo de cruzamento para 7 parâmetros (variáveis de decisão - D) (Storn e Price, 1997).....	25
Figura 3.8 – Metodologia de comparação de AEs (Dandy <i>et al.</i> , 2010).....	27
Figura 3.9 – Esquema de definição de políticas operacionais (Ribeiro, 2005).....	30
Figura 3.10 - (a) Sistema completo (b) Sistema simplificado (Shamir e Salomons, 2008).....	32
Figura 3.11 – Comparação do comportamento do nível de água em um reservatório do sistema (Shamir e Salomons, 2008).....	33
Figura 3.12 – Representação binária da programação de bombas (López-Ibáñez, 2009).....	36
Figura 3.13 - RPB no tempo (Odan, 2013).....	36
Figura 3.14 – Exemplo da representação das variáveis de decisão segundo a RPB “ <i>time-controlled trigger</i> ” (Odan, 2013).....	37
Figura 3.15 – Diagrama de blocos LF (Bezerra, 2009).....	41
Figura 3.16 - Funções de pertinência (a) triangular; (b) trapezoidal; (c) gaussiana; e d) singleton (Camboim, 2012, adaptado).....	42
Figura 3.17 – Função de pertinência trapezoidal L (Jameel e Radhi, 2014).....	43
Figura 4.1 - Sistemas Produtores de Água do Distrito Federal (DF) (ANA, 2010).....	46
Figura 4.2 - Sistemas de produção de água do DF (CAESB, 2012).....	47
Figura 4.3 - SAA do Rio Descoberto e suas principais unidades (Gebrim, 2013).....	50
Figura 4.4 - Fluxograma de adução do sistema de abastecimento do Rio Descoberto (Gebrim, 2013).....	53
Figura 4.5 – Vetor solução com três bombas, duas válvulas e período de otimização T (Gebrim, 2013).....	56
Figura 4.6 - SAA do Rio Descoberto no ambiente EPANET2 (Gebrim, 2013, adaptado).....	65
Figura 4.7 – Exemplo de esqueletonização de um trecho em série (Odan, 2013, modificado).....	66
Figura 4.8 - Funcionamento do AG.....	67
Figura 4.9 – Integração do modelo otimizador ao modelo de simulação (simulador hidráulico).....	69
Figura 4.10 - Funcionamento do algoritmo DDS.....	71
Figura 4.11 – Funcionamento do algoritmo DE.....	74
Figura 4.12 – Aplicação do conceito <i>fuzzy</i> à penalidade P6 (antiga P2 e P3).....	81
Figura 4.13 – Aplicação do conceito <i>fuzzy</i> à penalidade P7 (antiga P5).....	83
Figura 4.14 – Aplicação do conceito <i>fuzzy</i> à penalidade P4.....	86
Figura 4.15 – Aplicação do conceito <i>fuzzy</i> à penalidade P1.....	88
Figura 5.1 – Comparação da convergência de três algoritmos de otimização.....	93
Figura 5.2 - Regras operacionais obtidas pela simulação do AG, DDS e DE.....	98

Figura 5.3 – Trechos do modelo hidráulico antes e após a simplificação	99
Figura 5.4 – Comparação entre os níveis do REQ.GA1 do modelo hidráulico completo e simplificado	101
Figura 5.5 – Comparação entre os níveis do RAP.SA2 do modelo hidráulico completo e simplificado	101
Figura 5.6 - Regras operacionais obtidas pela simulação do modelo completo e simplificado	104
Figura 5.7 – Pressões nos nós de consumo do sistema Descoberto	111
Figura 5.8 – Regras operacionais para bombas de potência igual a 5.500 e 11.000 cv	116
Figura 5.9 – Regras operacionais para bombas de potência igual a 550 e 800 cv	116
Figura 5.10 – Regras operacionais para bombas de potência igual a 375 cv	117
Figura 5.11 – Regras operacionais para bombas de potência igual a 150 e 200 cv	117
Figura 5.12 – Regras operacionais para bombas de potência igual a 100 e 150 cv	118
Figura 5.13 – Regras operacionais para válvulas 1	118
Figura 5.14 – Regras operacionais para válvulas 2	119
Figura 5.15 – Trajetória dos níveis do reservatório Riacho Fundo	121
Figura 5.16 – Trajetória dos níveis do reservatório M Norte 1	121
Figura 5.17 – Trajetória dos níveis do reservatório Gama 2	122
Figura 5.18 – Trajetória dos níveis do reservatório Samambaia	122
Figura 5.19 – Trajetória dos níveis do reservatório M Norte 2	123

LISTA DE SÍMBOLOS, NOMENCLATURAS E ABREVIACÕES

ABNT	Associação Brasileira de Normas Técnicas
ACO	<i>Ant Colony Optimization</i>
AE	Algoritmo Evolucionário
AG	Algoritmo Genético
AMALGAM	<i>A Multialgorithm Genetically Adaptive Method</i>
AMS	<i>Adaptive Metropolis Search</i>
ANA	Agência Nacional de Águas
CAESB	Companhia de Saneamento Ambiental do Distrito Federal
CAPES	Coordenação de aperfeiçoamento de Pessoal de Nível Superior
CAPs	Captações Superficiais e Subterrâneas
CECOP	Centro de controle operacional
CLPs	Controladores lógicos programáveis
CSV	Calculador de solução viável
DAN2	<i>Dynamic Architecture for Artificial Neural Networks</i>
DDDS	<i>Discrete Dynamically Dimensioned Search</i>
DDS	<i>Dynamically Dimensioned Search</i>
DE	<i>Differential Evolution</i>
DF	Distrito Federal
EABs	Elevatórias de Água Bruta
EATs	Elevatórias de Água Tratada
EBOs	Elevatórias tipo <i>Booster</i>
EE	Estação Elevatória
EPOs	Elevatórias de Poço Profundo
EREs	Elevatórias de Reservatórios Elevados
ETA	Estação de Tratamento de Água
FAPDF	Fundação de Amparo à Pesquisa do Distrito Federal
GALib	<i>Genetic Algorithms Library</i>
GO	Goiás
GRG	Gradiente Reduzido Generalizado
HS	<i>Harmony Search</i>
IBM	<i>International Business Machines</i>
LF	Lógica <i>Fuzzy</i>
MAE	<i>Mean Absolute Error</i>
MIT	<i>Massachusetts Institute of Technology</i>
NBR	Norma Brasileira
NSGA	<i>Nondominated Sorting Genetic Algorithm</i>
NSGA II	<i>Nondominated Sorting Genetic Algorithm II</i>
PB	Paraíba
PD	Programação Dinâmica
PL	Programação Linear
PNL	Programação Não-Linear

PSO	<i>Particle Swarm Optimization</i>
RAM	<i>Random Access Memory</i>
RAPs	Reservatórios Apoiados
RELS	Reservatórios Elevados
REQ	Reservatório de Equalização
RNA	Rede Neural Artificial
RPB	Representação da Programação de Bombas
SP	São Paulo
SA	<i>Simulated Annealing</i>
SAA	Sistema de Abastecimento de Água
SCA	<i>Shuffled Complex Algorithm</i>
SCADA	<i>Supervisory Control and Data Acquisition</i>
SDK	<i>Software Development Kit</i>
SFLA	<i>Shuffled Frog Leaping Algorithm</i>
SGA	<i>Simple Genetic Algorithm</i> ou <i>Standard Genetic Algorithm</i>
SMPW	Setor de Mansões Park Way
SPA	Sistema Produtor de Água para Atendimento Urbano
SPEA	<i>Strength Pareto Evolutionary Algorithm</i>
THS	Tarifa Horo-Sazonal
UCPs	Unidades de Cloração de Poço
USEPA	<i>United States Environmental Protection Agency</i>
UTSs	Unidades de Tratamento Simplificado
WADISO	<i>Water Distribution Simulation and Optimization</i>

1 - INTRODUÇÃO

É crescente a preocupação por parte das companhias de distribuição de água em relação à conservação de energia, especialmente a de energia elétrica, pois ela está intimamente ligada aos custos operacionais, sendo um dos itens de maior importância (Camboim *et al.*, 2014).

Os sistemas de abastecimento e esgotamento sanitário utilizam cerca de 3% da energia consumida no mundo. Entre 2 e 3% do consumo total de energia elétrica no Brasil, correspondente a 10 bilhões de KWh/ano, são consumidos por prestadoras de serviços de abastecimento de água e esgotamento sanitário (Gomes e Carvalho, 2012). Cerca de 90% das despesas com energia elétrica das empresas de saneamento são provenientes de motores elétricos utilizados em estações elevatórias de água (Tsutiya, 2006).

Na Companhia de Saneamento Ambiental do Distrito Federal (Caesb), os custos de energia elétrica representam 29,56% do custo operacional total para a produção de água, sendo o segundo maior custo. Somente o sistema Descoberto utiliza 41,64% desse custo total (CAESB, 2014).

O consumo de energia na maior parte dos sistemas de abastecimento do mundo todo poderia ser reduzido em pelo menos 25%, se os sistemas fossem mais eficientes nos quesitos hidráulicos e energéticos e 10% desse consumo poderia ser economizado se métodos de otimização adequados fossem utilizados na otimização da operação de SAAs (Mackle *et al.*, 1995; James *et al.*, 2002).

A operação de sistemas de abastecimento de água envolve diversos fatores como a necessidade de confiabilidade no atendimento às demandas, economia no uso de equipamentos (energia e manutenção) e o planejamento de investimento para expansões futuras. Para a operação desses sistemas, a equipe de planejamento normalmente define regras fixas ou variáveis de controle dos sistemas. Três condições básicas são necessárias para alcançar a melhor regra de operação: o conhecimento do sistema, a definição clara dos objetivos a serem alcançados e a disponibilidade de modelos de análise hidráulica e otimização (Carrijo, 2004).

A obtenção de regras operacionais otimizadas é imprescindível e pode ser realizada com a utilização de modelos matemáticos, para que as companhias atendam ao objetivo de suprir a demanda dos consumidores com o abastecimento de água de qualidade com menores custos, sem depender exclusivamente da experiência dos operadores (Odan, 2013).

A política para a otimização da operação de bombas é definida pela programação do bombeamento que resultará no menor custo operacional, para um determinado conjunto de condições de funcionamento, como a demanda do sistema, a tarifa de energia elétrica e os níveis desejados de água nos reservatórios, associados ao risco de não atendimento dos requisitos de funcionamento do sistema e das demandas (Ormsbee *et al.*, 1989).

Em decorrência da elevada complexidade da operação de sistemas de abastecimento de água, a busca da decisão mais apropriada em termos de política operacional, deve ser realizada com o auxílio de processos otimizados, obtidos por meio de rotinas computacionais adequadas (Pedrosa, 2006).

Algoritmos evolucionários têm provado ser mais adequados e eficazes do que as abordagens tradicionais como a Programação Linear, Programação Não-Linear, Programação Dinâmica entre outras (Savic e Walters, 1997). No entanto, esses métodos determinísticos possuem desvantagens como a necessidade do cálculo de derivadas, dificuldade de adaptação a projetos com variáveis discretas e complexidade de implementação em trabalhos práticos de engenharia (Wu e Simpson, 2001).

Nos últimos anos, foram utilizados um grande número de algoritmos evolucionários para a otimização da operação de sistemas de distribuição de água como o Algoritmo Genético (Savic *et al.*, 1997; Simpson *et al.*, 1994; Cheung, 2004; Cunha, 2009; Gebrim, 2013; Sousa e Soares, 2014); Algoritmo *Shuffled Frog Leaping* (Eusuff e Lansey, 2003); Busca Harmônica - *Harmony Search* (Geem, 2006); Otimização por Enxame de Partículas - *Particle Swarm Optimization* (Suribabu e Neelakantan, 2006; Montalvo *et al.*, 2010); Otimização da Colônia de Formiga - *Ant Colony Optimization* (Ostfeld e Tubaltzev, 2008); Algoritmo *Discrete Dynamically Dimensioned Search* – DDDS (Tolson *et al.*, 2008); Algoritmo *Honey-Bee Mating* (Mohan e Babu, 2010); Evolução Diferencial - *Differential Evolution* (Vasan e Simonovic, 2010).

Esses algoritmos de otimização são utilizados em conjunto com o simulador hidráulico para a otimização da operação de SAAs. No entanto, segundo Odan (2013) os simuladores hidráulicos utilizam boa parte do tempo de processamento computacional gasto na otimização da operação, e podem inviabilizar seu uso para operação em tempo real e até mesmo em nível estratégico. Uma alternativa para solucionar o problema do tempo de processamento computacional é a utilização de modelos esqueletonizados, que consiste em simplificar o sistema, reduzindo a quantidade de nós e trechos a serem representados no modelo de simulação.

Outra técnica muito empregada na literatura para a redução do tempo de processamento do modelo de otimização é a denominada *seeding*, que consiste na introdução de uma ou mais soluções já conhecidas na população inicial do algoritmo (Machado *et al.*, 2008).

A aceleração da convergência do algoritmo também colabora na redução do tempo computacional, assim a definição das penalidades através da lógica *fuzzy* (teoria dos conjuntos *fuzzy*) integrada ao método das penalidades se mostra uma alternativa interessante (Geem, 2015), pois a técnica permite considerar que os valores das variáveis de controle (variáveis referentes as penalidades) estão se aproximando de valores críticos, a partir da *fuzzificação* das penalidades, evitando descontinuidades na busca de soluções ótimas e que soluções boas sejam descartadas.

Gebrim (2013) tinha como principal objetivo o desenvolvimento de um modelo de otimização para a redução dos custos de energia elétrica no sistema Descoberto, localizado no DF, a partir da utilização de Algoritmo Genético, no entanto, o programa desenvolvido pelo autor demandou elevado tempo de processamento, entre 3 e 73h.

Assim, o presente trabalho se configura como uma sequência da dissertação do autor e tem o propósito de melhorar o desempenho da otimização, em termos de tempo de processamento do modelo e qualidade da solução, a partir da utilização de outra linguagem de programação, diferentes algoritmos de otimização, simplificação do modelo hidráulico, emprego de outra abordagem de penalização (tratamento das penalidades) e utilização da técnica *seeding*, como forma de melhorar o desempenho do modelo em diversos aspectos, além do tempo de processamento.

No próximo capítulo serão mostrados os objetivos gerais e específicos do trabalho; no capítulo 3, a fundamentação teórica e revisão bibliográfica; o capítulo 4 apresentará a metodologia utilizada no estudo; o capítulo 5, os resultados e discussão; e o capítulo 6 as conclusões e recomendações.

2 - OBJETIVOS

2.1 - OBJETIVO GERAL

O objetivo principal do trabalho é desenvolver um modelo de otimização que defina regras da operação que minimizem os custos operacionais de energia elétrica provenientes do bombeamento, tendo como base o modelo desenvolvido por Gebrim (2013) e mantendo como estudo de caso o sistema do Rio Descoberto, localizado no Distrito Federal.

2.2 - OBJETIVOS ESPECÍFICOS

- Melhorar o desempenho do modelo de otimização de Gebrim (2013), por meio da implementação de outros algoritmos de otimização como o Algoritmo Genético, o *Dynamically Dimensioned Search* e o Evolução Diferencial;
- Aperfeiçoar a qualidade das soluções do modelo de otimização, a partir da utilização de técnicas que levem em consideração que os valores das variáveis de controle estão se aproximando de valores críticos e assim evitar descontinuidades nas buscas de soluções ótimas;
- Reduzir o tempo de processamento do modelo de otimização, pela simplificação do modelo hidráulico e melhoria nas técnicas de inicialização da busca.

3 - FUNDAMENTAÇÃO TEÓRICA E REVISÃO BIBLIOGRÁFICA

Este capítulo apresenta a fundamentação teórica e revisão bibliográfica referente à operação de Sistemas de Abastecimento de Água (SAAs); a otimização da operação de SAAs; as técnicas utilizadas para a otimização; o planejamento operacional em nível estratégico e a operação em tempo real; as estratégias para reduzir o tempo de processamento de algoritmos; os métodos das penalidades e a Lógica *Fuzzy*.

3.1 - OPERAÇÃO DE SAAs

O SAA é composto pelo manancial, captação, estação elevatória, adutora, estação de tratamento de água, reservatório e pela rede de abastecimento de água e tem por principais objetivos o fornecimento ao usuário de água de boa qualidade, quantidade adequada e pressão suficiente (Tsutiya, 2006). São raros os sistemas de distribuição de água que operam somente por gravidade. Em geral os sistemas incluem conjuntos motor-bomba para recalcar a água (Bezerra, 2009).

Assim, a operação dos SAAs é fundamentada no controle do estado de funcionamento de bombas (ligado/desligado/alteração da velocidade de rotação) e válvulas (fechada/aberta/alteração do ângulo de abertura), realizadas para suprir a demanda de água dos usuários. A decisão sobre a melhor combinação dos estados dos elementos é complexa, visto que eles estão propensos a alterações de seus estados a cada momento (Odan, 2013).

Segundo Carrijo (2004), as maiores dificuldades para uma operação apropriada de SAA são a obtenção de informações referentes aos procedimentos operacionais, à metodologia de cobrança de tarifa de energia elétrica, à variação da topologia e às demandas. Um SAA pode ser operado de várias formas que dependem de fatores como a dimensão e complexidade do sistema, a experiência dos operadores, a disponibilidade de equipamentos adequados para comunicação (comando de estruturas de controle e realização de cálculos), e a disponibilidade de modelos matemáticos para análises das informações, além da segurança, custos da operação e a confiabilidade. Segundo Odan (2013) a confiabilidade consiste na probabilidade de suprir a demanda dos usuários, com a distribuição de água em quantidade, qualidade e pressões adequadas, nos locais e horários apropriados.

A operação de um SAA também depende da demanda que pode variar de forma aleatória. Possíveis falhas ou retirada do serviço de um ou mais componentes eletromecânicos do sistema, como tubulações, bombas, válvulas, junções, a quantidade de água nos reservatórios para compensar o aumento diário ou semanal da demanda e a qualidade da água fornecida aos consumidores podem afetar a operação (Gargano e Pianese, 2000).

O controle das diversas unidades que compõem um SAA pode ser efetuado isoladamente (controle local) ou por meio de uma central (controle remoto/central) (Carrijo, 2004). Segundo Tsutiya (2006), o controle local pode ser realizado de forma manual ou automática. Na manual, existe a necessidade da presença do operador para controlar o processo, já na automática, um equipamento ou um conjunto de equipamentos realiza esse controle.

Segundo Carrijo (2004), no controle central, as decisões são emitidas por um centro de controle específico. Assim, a forma mais simples do controle centralizado é aquela em que o operador de uma unidade do sistema obedece aos comandos do centro de controle e transmite os dados operacionais a este centro. No entanto, para aumentar a confiabilidade e a segurança operacionais, o centro de controle pode dispensar o operador local, nesse caso o controle é chamado de SCADA (*Supervisory Control and Data Acquisition*). De acordo com Tsutiya (2006), o SCADA realiza o controle remoto de dispositivos como bombas e válvulas, e a obtém dados operacionais como vazão e pressão.

O estágio mais avançado da operação de um SAA de grande porte exige a alteração de uma forma SCADA para uma outra, designada controle automático por computador, que necessita de algoritmos de otimização da operação (Carrijo, 2004).

3.2 - OTIMIZAÇÃO DA OPERAÇÃO DE SAAs

O problema do controle ótimo é encontrar decisões para operação de bombas e válvulas ao longo de um período determinado, em que as demandas sejam atendidas, condições de pressão sejam satisfeitas e o custo total minimizado (Zessler e Shamir, 1989).

Segundo Carrijo (2004), o problema de otimização de SAA, de forma geral, pode ser descrito como um problema de otimização com estrutura clássica, apresentadas na Equação 3.1 e 3.2.

Com finalidade de maximizar ou minimizar uma função objetivo, obedecendo um conjunto de uma ou mais equações ou inequações denominadas restrições.

$$\text{Minimize: } FO(X) \quad (3.1)$$

$$\text{sujeito a: } P_r(X) \geq 0, r = 1, 2, \dots, n_r \quad (3.2)$$

em que: X é o vetor de variáveis de decisão $\{x_1, x_2, \dots, x_n\}$; n é o número total de variáveis de decisão; e n_r é o número de restrições.

De acordo com Ormsbee e Lansey (1994), as restrições associadas a problemas de otimização operacional consistem em limitações físicas do sistema, leis físicas e requisitos definidos externamente. As limitações físicas do sistema compreendem volume de água que pode ser armazenado em reservatórios, quantidade de água que pode ser fornecida pelo manancial e as configurações das válvulas e bombas. As leis físicas incluem a conservação de massa nos nós e a conservação da energia nos anéis. Normalmente os requisitos externos são apenas para satisfazer as demandas e manter cargas de pressão aceitáveis no sistema.

A otimização operacional de SAAs tem se tornado alvo de muitas pesquisas práticas e teóricas. Pode-se citar os estudos de Zessler e Shamir (1989); Lansey e Awumah (1994); Sakarya e Mays (2000); Righetto (2002); Pedrosa (2006); Sousa e Soares (2014); mostrados na Tabela 3.1.

Tabela 3.1 - Otimização da operação de SAA

AUTORES/ ANO	VARIÁVEIS DE DECISÃO	OBJETIVOS	RESTRIÇÕES	MÉTODO/ALG ORITMO DE OTIMIZAÇÃO
Zessler e Shamir (1989)	Vazões das estações de bombeamento, volumes dos reservatórios	Minimizar custo de operação	Volumes máximos e mínimos dos reservatórios	Programação dinâmica
Lansey e Awumah (1994)	Estado das bombas	Minimizar custos de bombeamento	Número de bombas em funcionamento, acionamentos das bombas, níveis de água nos reservatórios, máximo consumo de energia	Programação dinâmica

Tabela 3.1 - Continuação

AUTORES/ ANO	VARIÁVEIS DE DECISÃO	OBJETIVOS	RESTRICÇÕES	MÉTODO/ALG ORITMO DE OTIMIZAÇÃO
Sakarya e Mays (2000)	Concentração de substância no tubo, período de operação de uma bomba, nível de água, carga de pressão, vazão	Minimizar o desvio da concentração atual de uma substância, o tempo de operação total das bombas, o custo total de energia	Níveis dos reservatórios, pressões, tempo de operação das bombas	Programação não-linear (GRG2)
Righetto (2002)	Número de bombas em operação, velocidade rotacional das bombas, condição de abertura das válvulas (aberta/fechada)	Minimizar custos operacionais	Vazões, pressões, níveis de água nos reservatórios, quantidade de acionamentos das bombas e válvulas	AG
Pedrosa (2006)	Estado das bombas e válvulas	Minimizar custos energéticos	Quantidade de acionamentos, recuperação do estado inicial, pressão não negativa, níveis dos reservatórios	AG
Sousa e Soares (2014)	Estado das bombas	Minimizar perdas por vazamentos, custo de energia elétrica, e maximizar a confiabilidade em atender às demandas	Atendimento às demandas e pressões	SPEA

3.2.1 - Regras operacionais de SAAs

A operação ótima de um SAA baseia-se na determinação de regras operacionais que proporcionam a maior eficiência hidroenergética do sistema (Sousa e Soares, 2014). As regras operacionais recomendam uma adequação na rotina operacional executada diariamente pelos operadores, a fim de aumentar a eficiência hidráulica e energética. Os procedimentos realizados pelos operadores normalmente são variados em todo sistema e não obedecem a um padrão. A construção de regras tem a finalidade de reproduzir e aprimorar a forma de pensar do operador, uniformizando os procedimentos para que se estabeleça um roteiro genérico e robusto das variações hidráulicas do sistema (Lacerda *et al.* 2008).

Ainda que a operação de um SAA seja entendida como uma sequência de comandos exercidos sobre os equipamentos, que têm como finalidade o atendimento da demanda, na realidade, o problema é muito mais complexo e abrange aspectos de planejamento, controle e supervisão, serviços de infraestrutura de apoio e de atendimento ao consumidor, todos considerados simultaneamente e interdependentes entre si. A operação normalmente é feita através da definição de regras (fixas ou variáveis) pela equipe de operação, a partir de informações e experiências anteriores, além do conhecimento do estado atual do sistema. As regras definidas são repassadas à equipe que controla o sistema, que por sua vez implementa as regras e comunica os resultados das operações realizadas ao setor de planejamento, para avaliação da performance e as adequações necessárias (Carrijo, 2004).

De acordo com Lacerda *et al.* (2008), as regras atuam ligando e desligando bombas, abrindo e fechando válvulas, respeitando os limites de pressão, velocidade e os níveis operacionais para os reservatórios. Geralmente, as regras operacionais em SAA possui estrutura apresentada na Figura 3.1.

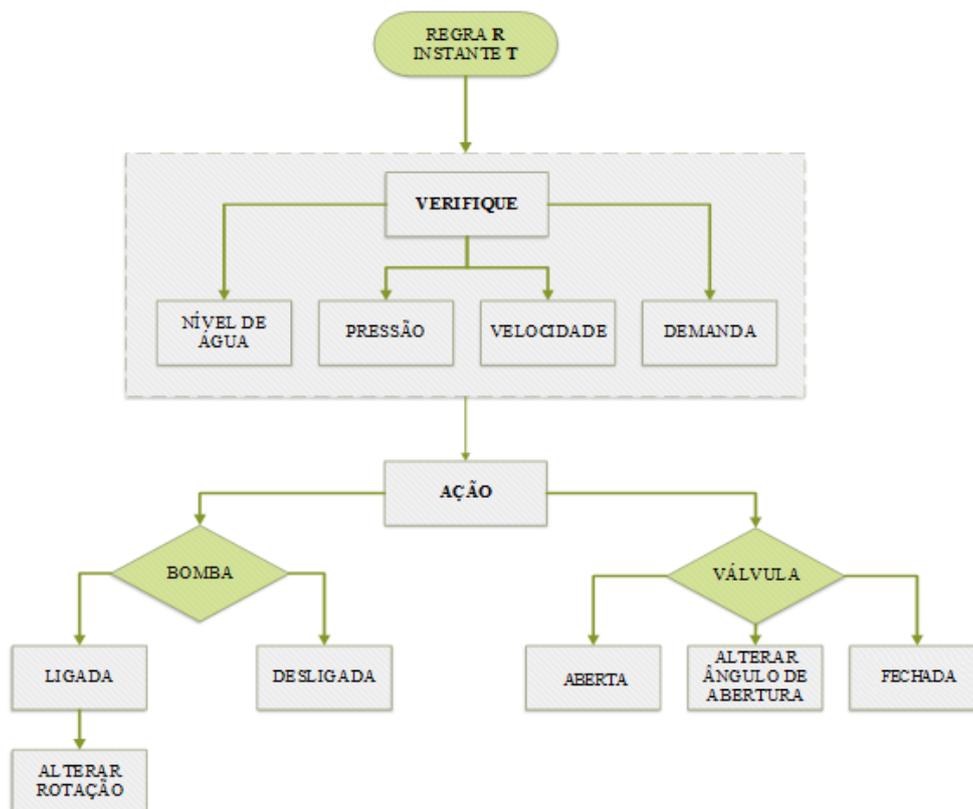


Figura 3.1 – Fluxograma das regras operacionais em SAAs (Lacerda *et al.*, 2008, modificado)

Segundo Carrijo e Reis (2006), quatro condições são indispensáveis para que o planejamento determine a melhor regra operacional: (1) a definição dos objetivos; (2) a disponibilidade de modelos matemáticos de análise; (3) os equipamentos para o processamento; e, (4) o conhecimento do sistema. A definição dos objetivos é uma das etapas mais difíceis do processo de obtenção da melhor regra de operação, pois vários aspectos não técnicos estão envolvidos. Após definidos os objetivos da operação, devem ser estabelecidas as políticas de operação (Carrijo, 2004).

A busca da decisão mais apropriada no que se refere à política operacional só pode ser realizada com o auxílio de processos otimizados a serem alcançados através de rotinas computacionais adequadas. Sem um planejamento otimizado, os SAAs continuam perdendo uma quantidade razoável de energia elétrica (Pedrosa, 2006).

3.2.2 - Técnicas de otimização aplicadas aos SAAs

Até a década de 1990, os problemas de otimização da operação de SAAs eram solucionados por meio de métodos como programação linear - PL (Jowitt e Germanopoulos, 1992), não-linear - PNL (Shamir, 1974; Brion e Mays, 1991; Pezeshk *et al.*, 1994; Yu *et al.*, 1994) e dinâmica - PD (Zessler e Shamir, 1989; Lansey e Awumah, 1994).

Segundo Lansey (2006), as primeiras aplicações bem sucedidas da PL, na otimização de SAAs, foram os trabalhos Karmeli *et al.* (1968) e Gupta (1969), que determinaram os diâmetros de tubos em um sistema ramificado. As equações hidráulicas de um sistema ramificado eram descritas por um conjunto de equações lineares independentes, em vez de um sistema de equações no caso de um sistema malhado.

Sistemas ramificados foram o foco das pesquisas até meados dos anos 70, além da PL a PD também foi eficaz na otimização de sistemas mais simples. Porém, segundo Brion e Mays (1991), as abordagens da programação dinâmica limitavam-se ao tamanho e a complexidade do sistema (grande quantidade de bombas e reservatórios), dessa forma, a PD só era aplicável a sistemas muito pequenos.

Shamir (1974) formulou um problema de PNL para otimizar projeto e operação de SAAs. A otimização foi resolvida pela combinação do Gradiente Reduzido Generalizado e método de

Penalidade. Segundo Lansey (2006), naquele ano e nos 10 anos seguintes, programas de domínio comercial ou público que resolviam problemas de PNL não estavam disponíveis, em contrapartida, havia grande quantidade de programas que solucionavam problemas de PL em computadores de grande porte. Dessa forma, a PL continuou a ser utilizada para a otimização de SAAs.

Morgan e Goulter (1985) desenvolveram uma abordagem de PL sucessiva que foi um dos primeiros programas a acoplar uma técnica que simulava o comportamento hidráulico da rede (Hardy-Cross) com uma rotina de otimização para selecionar diâmetros ótimos em sistemas malhados.

O desenvolvimento de modelos de simulação hidráulica eficientes foram um avanço na aplicação da otimização em SAAs, pois eles simplificavam a modelagem e reduziam o tempo de processamento dos modelos. Os simuladores hidráulicos WADISO e KYPipe foram os primeiros a serem acoplados aos otimizadores (Lansey, 2006). Lansey e Awumah (1994) utilizaram o simulador hidráulico KYPipe ligado a PD com o objetivo de minimizar custos de bombeamento em um SAA.

Em meados da década de 1980, os algoritmos de PNL estavam se tornando mais robustos e vários códigos estavam disponíveis, incluindo o método Gradiente Reduzido Generalizado (GRG), programação quadrática sucessiva e programação linear sucessiva. Com esses códigos e o advento dos simuladores hidráulicos eficientes, observou-se uma mudança nas formulações dos modelos de otimização e nos tamanhos dos sistemas que poderiam ser resolvidos. A abordagem geral da PNL também permitiu sua utilização em outros tipos de problemas, tais como calibração de modelos, reabilitação e operações de SAAs (Lansey, 2006). Brion e Mays (1991) desenvolveram uma metodologia que utilizava a PNL ligada ao KYPipe, para otimizar a operação de estações elevatórias do SAA de Austin no Texas.

Segundo Lansey (2006), no início da década de 1990 ocorreram 3 importantes mudanças que resultaram em um aumento significativo das publicações relacionadas à otimização de SAAs. A primeira mudança refere-se à introdução do processador Pentium, que aumentou significativamente a velocidade computacional de computadores *desktop*, tornando os computadores de grande porte menos necessários.

A segunda inovação foi a introdução do simulador hidráulico EPANET em 1993. No entanto, a maior inovação foi o lançamento do EPANET2 *Toolkit* em 1999, que permitiu ao programador extrair resultados hidráulicos do EPANET2 e vinculá-los a outros programas de forma facilitada.

A terceira mudança foi o surgimento de algoritmos estocásticos aplicados aos SAAs. De acordo com Luke (2013) a otimização estocástica é uma classe geral de algoritmos e técnicas que empregam algum grau de aleatoriedade para encontrar soluções ideais ou próximas destas. O principal subcampo dentro da otimização estocástica são as chamadas metaheurísticas.

As metaheurísticas, em sua definição original, são métodos de soluções que combinam procedimentos de busca local e estratégias de alto nível para criar um processo capaz de escapar de mínimos locais e realizar uma busca robusta no espaço de soluções do problema (Glover e Kochenberger, 2003). Segundo Coello (2006), a otimização utilizando metaheurísticas tornou-se um tema de pesquisa muito popular nos últimos anos. As técnicas metaheurísticas mais aplicadas para a resolução de problemas de recursos hídricos são os Algoritmos Evolucionários (Maier *et al.*, 2014).

Os Algoritmos Evolucionários (AEs) constituem uma classe de algoritmos que se baseiam em mecanismos da seleção natural (evolução biológica) como reprodução, mutação, cruzamento e seleção, e utilizam uma abordagem baseada na população, em que mais de uma solução participa da iteração e evolui para uma nova população de soluções a cada iteração. Os algoritmos evolucionários são muito utilizados na otimização de um único objetivo, mas se tornaram comuns na otimização multiobjetivo (Coello, 2006; Nicklow *et al.*, 2010; Deb, 2011).

Segundo Coello (1999), AEs são adequados para a resolução de problemas de otimização multiobjetivo, pois eles lidam com um conjunto de soluções possíveis (população), que possibilitam encontrar todo o conjunto de soluções ótimas de Pareto em uma única execução do algoritmo.

De acordo com Formiga e Chaudhry (2002), a fundamentação teórica para os AEs foi proposta por John Holland na década de 1960. Porém, somente a partir da década de 1990,

houve aumento no desenvolvimento e aplicação de AEs para a otimização de SAAs (Dandy *et al.*, 2010). Um grande número de AEs tem sido desenvolvidos para a otimização de SAAs, sendo assim, a primeira publicação significativa de cada um é mostrado na Tabela 3.2.

Tabela 3.2 – Publicações que utilizaram AEs para a otimização de SAAs (Zheng *et al.*, 2012, modificado)

ALGORITMO	PRIMEIRA REFERÊNCIA
Algoritmo Genético	Simpson <i>et al.</i> (1994)
Busca Harmônica (<i>Harmony search</i>)	Geem <i>et al.</i> (2002)
Algoritmo <i>Shuffled frog leaping</i>	Eusuff e Lansey (2003)
Otimização da colônia de formigas (<i>Ant colony optimization</i>)	Maier <i>et al.</i> (2003)
Otimização por enxame de partículas (<i>Particle swarm optimization</i>)	Suribabu e Neelakantan (2006)
Busca dispersa (<i>Scatter search</i>)	Lin <i>et al.</i> (2007)
Entropia cruzada (<i>Cross-entropy</i>)	Perelman e Ostfeld (2007)
Evolução diferencial (<i>Differential evolution</i>)	Suribabu (2010)
Otimização <i>Honey-Bee Mating</i>	Mohan e Babu (2010)

Dentre os AEs, o AG foi uns dos primeiros algoritmos aplicados na otimização de SAAs. Segundo Pedrosa (2006) os AGs são métodos computacionais de busca e otimização de problemas complexos, baseados numa analogia com os mecanismos de seleção natural e sobrevivência dos mais aptos, princípios mostrados na Teoria da Evolução proposta em 1859 por Charles Darwin.

O AG foi desenvolvido em 1975, por John Holland, nos Estados Unidos, através do livro denominado *Adaptation in Natural and Artificial Systems*. O algoritmo proposto por Holland, é conhecido na literatura como *Simple Genetic Algorithm* ou *Standard Genetic Algorithm* ou, simplesmente, SGA.

O ciclo básico de funcionamento do algoritmo genético simples (Figura 3.2) apresenta, inicialmente, a criação da população inicial, que é um conjunto de vetores de soluções iniciais (valores das variáveis de decisão). O conjunto de vetores pode ser criado de forma aleatória ou pode ser definido pelo usuário. Com esses valores das variáveis de decisão é calculada a função aptidão (*fitness*) para cada vetor. Em seguida, é verificado se o critério

de parada é atendido, nesse caso é o número de gerações do AG. Se o critério for atendido o algoritmo termina, se não, ocorre os processos de seleção, cruzamento (recombinação) e mutação que irão gerar uma nova população para nova avaliação da função aptidão. Esse processo continua até o atendimento do critério de parada.

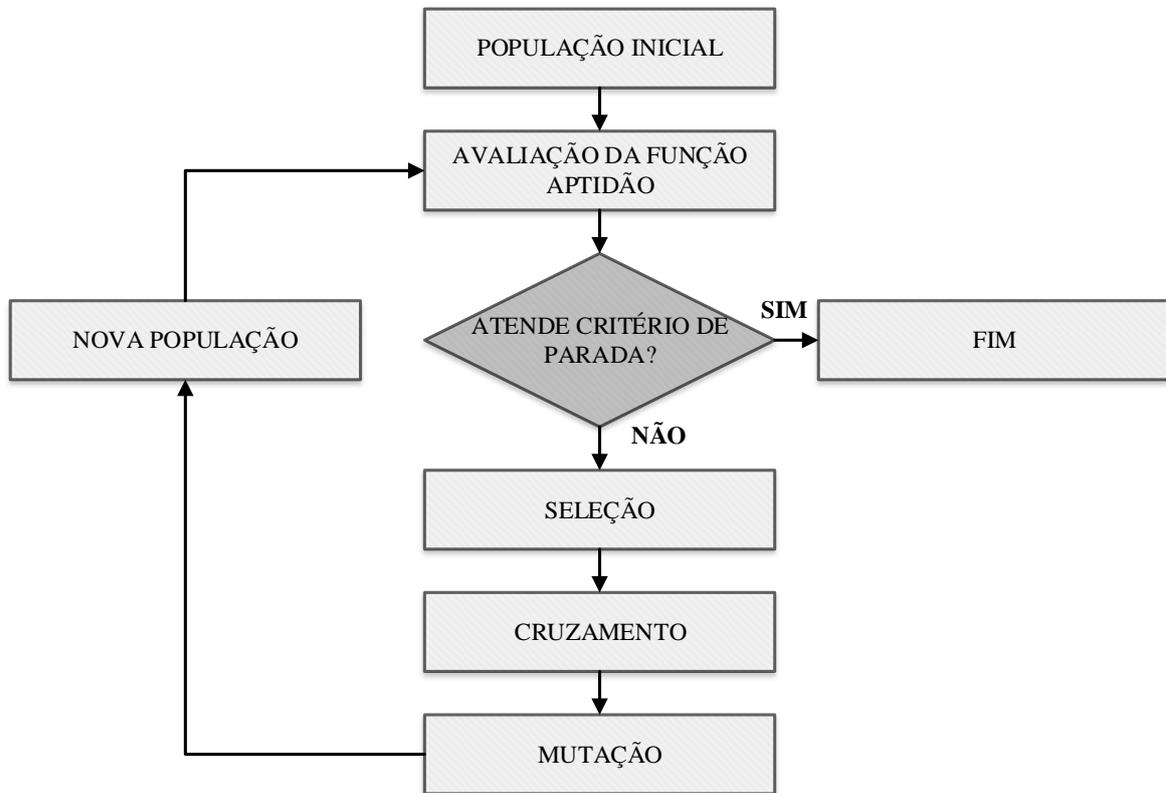


Figura 3.2 – Funcionamento básico do AG simples

O processo de seleção desempenha a função da seleção natural na evolução, selecionando, para sobreviver e reproduzir, os organismos melhor adaptados ao meio, ou seja, as soluções que geram melhores valores da função aptidão (Carrijo, 2004). A seleção pode se apresentar dos seguintes tipos: roda da roleta, torneio, ordenamento, uniforme e amostragem determinística, sendo os métodos da roda da roleta e do torneio amplamente empregados na literatura.

O processo de seleção proposto por Holland é conhecido por roda da roleta. Neste, os indivíduos de uma geração são escolhidos para a próxima geração, através de um sorteio de roleta. Os indivíduos são representados na roleta proporcionalmente ao seu índice de aptidão. A roleta é girada um determinado número de vezes, dependendo do tamanho da população, de modo que, aqueles sorteados na roleta são escolhidos como indivíduos que irão participar

da próxima geração. Na seleção, denominada torneio encontra-se o cromossomo (vetor de soluções) melhor adaptado numa série de n elementos, comparando aleatoriamente pares desse subconjunto e escolhendo os melhores para serem cromossomos pais na próxima geração (Pedrosa, 2006).

Uma adição aos métodos de seleção é chamada de Elitismo. Segundo Carrijo (2004), o elitismo é uma estratégia em que ocorre a transferência do melhor cromossomo de uma geração para a outra, sem alterações, já que o melhor cromossomo pode ser perdido em decorrência da aplicação dos operadores de cruzamento e mutação. Normalmente, esta estratégia melhora o desempenho do AG, pois encontra a solução mais rapidamente se comparado ao AG sem elitismo.

O cruzamento ou recombinação é o mecanismo de criação de indivíduos filhos, ou seja, novas soluções, por meio da recombinação de indivíduos pais (soluções atuais ou corrente). O cruzamento pode ser realizado de diversas formas, chamadas operadores convencionais de um ponto, dois pontos e uniforme (Wardlaw e Sharif, 1999).

No cruzamento em um ou dois pontos (Figura 3.3), cada um dos cromossomos pais tem sua cadeia “cortada” em uma (Um Ponto) ou duas posições (Dois Pontos) aleatórias. As partes são recombinadas, originando dois novos cromossomos filhos, com uma dada probabilidade pré-estabelecida, normalmente na faixa entre 0,6 e 1,0 (Carrijo, 2004; Cunha, 2009).

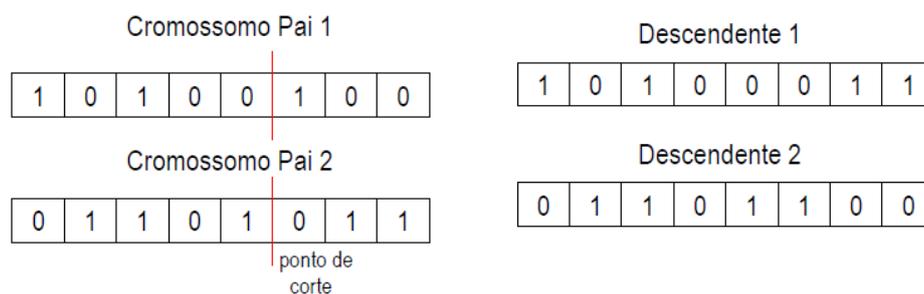


Figura 3.3 – Esquema gráfico do cruzamento de um ponto (Pedrosa, 2006)

O cruzamento uniforme opera nos genes (elementos do vetor de soluções) individuais dos cromossomos selecionados ao invés de agir nos blocos de material genético, e cada gene é considerado por sua vez para cruzamento ou troca (Wardlaw e Sharif, 1999).

A mutação é um processo que permite a introdução de características genéticas novas às soluções existentes, através da substituição de genes dos indivíduos. Geralmente é realizada *bit a bit* (representação binária) ou variável por variável (representação real) possibilitando que cada *bit* ou gene seja modificado com uma pequena probabilidade de mutação (P_m) que varia de 0,001 e 0,1 (Reis e Akutsu, 2002). Na mutação um número randômico é gerado entre 0 e 1 que comparado ao valor de P_m definirá ou não a ocorrência da mutação, que consiste na troca do valor do *bit* de 0 por 1 ou vice-versa, no caso da representação binária (Pedrosa, 2006).

Em resumo, o objetivo do operador mutação é “perturbar” cada solução da população pai (população corrente) para criar uma nova população de soluções descendentes (nova população) (Carrijo, 2004). A Figura 3.4 apresenta um tipo de mutação simples, onde o valor do *bit* é trocado de 0 para 1 no ponto de mutação.

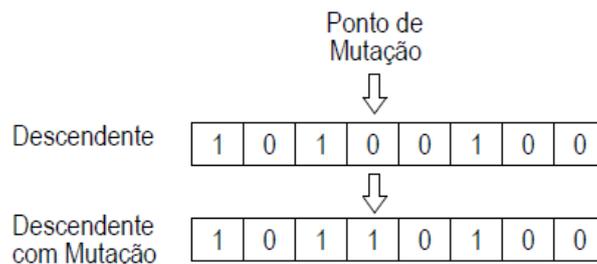


Figura 3.4 – Esquema grfico de mutaç o simples (Pedrosa, 2006)

Segundo Lansey (2006), os primeiros a aplicar o AG para a otimizaç o de SAAs foram Walters e Lohbeck (1993); e Simpson *et al.* (1994), seguidos por Dandy *et al.* (1996); Savic e Walters (1997). No Brasil, um nmero considervel de trabalhos fizeram uso dos AGs para a otimizaç o da operaç o de SAAs, como os trabalhos de Righetto (2002); Pedrosa (2006); Carrijo (2004); Cunha (2009); Odan (2013); Gebrim (2013); entre outros.

Carrijo (2004) desenvolveu um modelo computacional para o controle operacional timo de sistemas de distribuiç o de gua potvel localizado em Goinia/GO, utilizando um mdulo para a simulaç o hidrulica (EPANET2), um mdulo para otimizaç o (considerando a minimizaç o dos custos de energia eltrica e a maximizaç o dos benefcios hidrulicos) e um mdulo que empregou algoritmo de aprendizado de mquina para extraç o de regras

operacionais. O módulo de otimização empregou Algoritmos Genéticos multiobjetivo entre eles o NSGA, o NSGA II e o SPEA.

Para a avaliação do desempenho dos algoritmos o autor realizou simulações considerando diferentes populações iniciais (100, 200, 300, 400 e 500) e pela análise visual das soluções geradas, verificou que o SPEA obteve uma frente Pareto com soluções melhores distribuídas (menores custos e maiores benefícios) do que o NSGA e NSGA II (Figura 3.5).

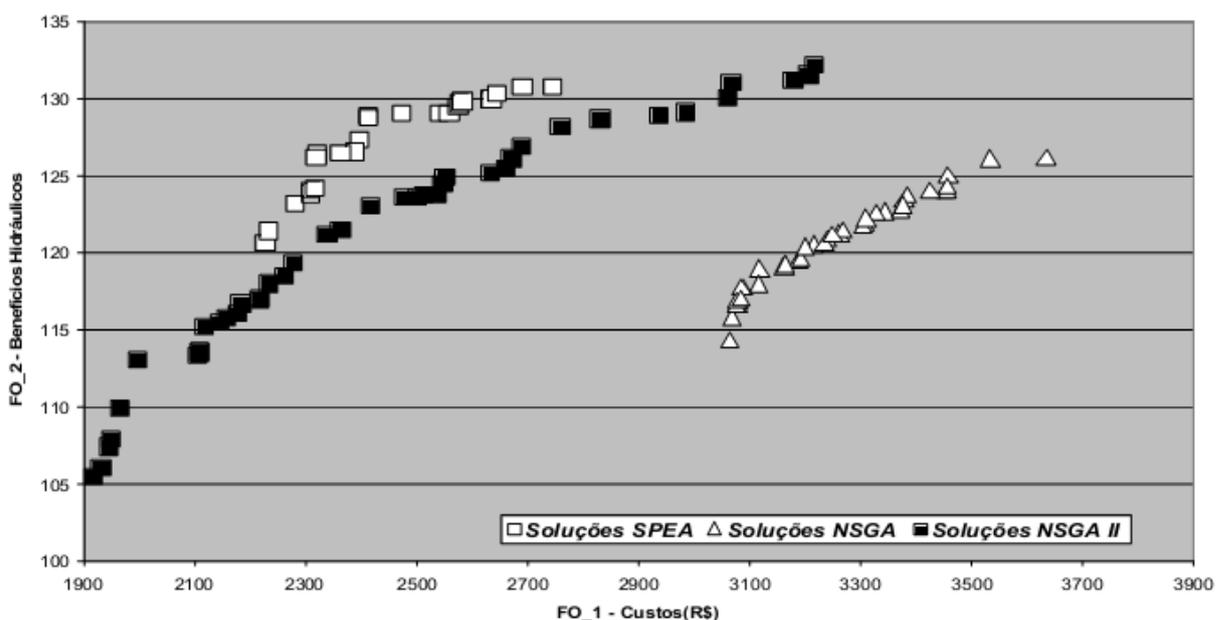


Figura 3.5 – Frontes de Pareto dos algoritmos multiobjetivo (Carrijo, 2004)

O tempo de processamento computacional utilizado pelo SPEA foi menor que o gasto pelos outros dois algoritmos, assim ele foi aplicado para a definição das regras operacionais do sistema.

Além de Carrijo (2004), Pedrosa (2006) também utilizou AG em seu estudo. Ele realizou a otimização da operação do Sistema Adutor Marés, localizado em João Pessoa/PB, com o objetivo de definir estratégias diárias dos estados de bombas e válvulas que sejam operacionalmente viáveis, a fim de minimizar os custos de energia elétrica. Para isso, ele desenvolveu o programa OTIMIZE, que integra o simulador hidráulico EPANET2 ao AG. As restrições do problema de otimização foram tratadas pelo método das Penalidades, que são acrescentadas à função objetivo em termos de penalidades, transformando o problema inicialmente restrito em um problema irrestrito. As funções de penalidades consideradas no

modelo foram: custo de manutenção das bombas, custo de recuperação do estado inicial do sistema, custo de garantia da demanda.

Segundo o autor, a qualidade da resposta do AG mostrou-se intensamente influenciada pelos valores dos coeficientes de penalidade aplicados às funções de penalidades, podendo gerar soluções inviáveis em termos operacionais se valores inadequados de coeficientes forem considerados. No entanto, o programa OTIMIZE se mostrou eficiente na otimização dos custos de energia elétrica, visto que acarretou em uma economia diária de 21,73% em relação aos custos da operação real do sistema estudado.

Cunha (2009) desenvolveu um modelo para a otimização energética de SAA em tempo real, empregando um módulo para simulação hidráulica (EPANET2), um módulo para previsão de demandas e históricos de demanda, e um módulo otimizador que utiliza AG. Para a otimização foi utilizada uma função objetivo que minimizava os custos de energia elétrica do sistema, em conjunto com penalidades referentes à manutenção das pressões mínimas nos nós de consumo, *diferença entre níveis iniciais e finais dos reservatórios*, quantidade de acionamentos, e ocorrências de interrupção no abastecimento. Cada penalidade possuía um multiplicador, chamado de coeficiente de penalidade.

O modelo desenvolvido por Cunha (2009) foi aplicado ao sistema estudado por Pedrosa (2006) e o histórico de consumos provenientes do trabalho de Moreira (2007). Dessa forma, não foi possível a comparação dos resultados do modelo, que se trata de um sistema hipotético, com os resultados da operação real. No entanto, em comparação com o modelo desenvolvido por Pedrosa (2006), o autor verificou uma redução de 8,5% nos custos diários de energia elétrica do sistema. O tempo de processamento do modelo variou de 15 minutos a 46 horas para a simular 24 horas de operação, tempo excessivamente longo para a otimização em tempo real. O autor também concluiu que a qualidade dos resultados produzidos pelo AG foi fortemente influenciada pelos valores dos coeficientes de penalidade e, pelos operadores e parâmetros genéticos usados no processo de otimização.

Gebrim (2013) desenvolveu um modelo de otimização baseado no modelo proposto por Cunha (2009), com foco na redução dos custos de energia elétrica, para aplicação no planejamento operacional do SAA do Rio Descoberto localizado no Distrito Federal. Assim

como realizado por Cunha (2009), o autor utilizou o EPANET2 para a simulação hidráulica e AG para a otimização.

A função objetivo era minimizar os custos de energia elétrica com 5 restrições explícitas: pressão positiva nos nós de demanda, nível de água nos reservatórios acima do limite mínimo operacional, nível de água nos reservatórios abaixo do limite de extravasamento, número de acionamento de equipamentos menor que o limite máximo tolerado e variável conforme o porte dos equipamentos, e *diferença entre os níveis iniciais e finais dos reservatórios menor que o limite máximo tolerado*.

Para cada uma das restrições foi atribuída uma penalidade referente ao não atendimento às demandas, nível dos reservatórios abaixo do limite mínimo, nível dos reservatórios acima do limite máximo (extravasamento), excesso de acionamentos e *diferença entre os níveis iniciais e finais dos reservatórios*. Cada penalidade foi multiplicada por um coeficiente de penalidade. Os resultados obtidos foram um conjunto de regras operacionais otimizadas, cujos resultados foram comparados com uma operação de referência do sistema e, com isto, verificou-se que a operação otimizada levou a menores custos energéticos que a operação de referência. Porém o tempo computacional exigido pelo modelo foi muito grande.

Segundo Balla e Lingireddy (2000), embora, os AGs se caracterizam como técnicas robustas e poderosas na obtenção de uma solução ótima e soluções não-dominadas, a necessidade de excessivo tempo computacional (característica dos AGs) tem sido um obstáculo para a aplicação desse método em problemas reais de grande escala e na otimização da operação em tempo real.

Geem *et al.* (2001) desenvolveram o algoritmo Busca Harmônica (*Harmony Search - HS*) pela analogia entre a improvisação de músicos e o processo de otimização. Segundo Geem (2006), o HS imita o comportamento de músicos no processo de improvisação, onde eles aperfeiçoam suas músicas, a fim de obter melhores harmonias, ou seja, melhores soluções no problema de otimização.

Geem (2006) utilizou o algoritmo HS para otimizar diâmetros de tubulações, com isto, minimizar custos de projeto de expansão de redes de distribuição de água. O algoritmo de Busca Harmônica foi combinado com o simulador hidráulico EPANET2 para o

desenvolvimento do modelo, que foi aplicado a uma rede da literatura, denominada rede *New York Tunnels*, com 20 nós e 21 trechos. Os resultados obtidos foram comparados com outros algoritmos metaheurísticos como AG, *Simulated Annealing*, Busca Tabu (*Tabu Search*), Otimização da Colônia de Formiga (*Ant Colony Optimization*), Algoritmo *Shuffled Frog Leaping* e Entropia Cruzada (*Cross Entropy*). A comparação foi realizada pela análise dos custos obtidos e pelo número de iterações, o HS encontrou a solução de menor custo com o menor número de iterações.

Segundo Ostfeld e Tubaltzev (2008), o ACO (*Ant Colony Optimization*) é um algoritmo baseado no comportamento de formigas em busca de alimentos proposto por Dorigo (1992). Foi utilizado pela primeira vez na otimização de projeto de SAA por Maier *et al.* (2003) e aplicado a dois problemas de otimização de referência em SAA, os resultados foram comparados com aqueles obtidos por meio de AGs. O ACO obteve melhores resultados em termos de eficiência computacional e capacidade de encontrar a solução de menor custo nos dois problemas de otimização empregados.

Já Ostfeld e Tubaltzev (2008) utilizaram o ACO acoplado ao EPANET2 para otimizar os custos de projeto e operação de sistemas bombeados. O modelo de otimização foi parametrizado seguindo as orientações de Zecchin *et al.* (2005), desse modo, apresentou melhores resultados do que os mostrados no trabalho de Maier *et al.* (2003), a partir da simulação dos mesmos sistemas de referência.

A otimização por enxame de partículas (*Particle swarm optimization* - PSO) foi originalmente concebido por Kennedy e Eberhart (1995). Nesse algoritmo uma partícula pode representar um pássaro, uma abelha, um peixe, ou qualquer outro tipo de agente natural que exibe comportamento do tipo enxame. Segundo Suribabu e Neelakantan (2006), a técnica simula o comportamento social entre os indivíduos (partículas) "voando" através de um espaço de busca multidimensional (possíveis soluções).

Para Odan (2013), o PSO é um método de busca populacional que trabalha com um grupo de soluções (partículas), realizando a busca continuamente através das gerações. Primeiramente são atribuídas aleatoriamente posição e velocidade às partículas em um espaço n-dimensional. Depois da inicialização, cada partícula ajusta iterativamente a posição conforme sua própria experiência e também das demais partículas, por meio da utilização da

melhor posição encontrada por si mesma, e de toda a população. Assim, o PSO combina princípios de busca local e global, na evolução das soluções em direção à frente de Pareto.

Suribabu e Neelakantan (2006) utilizaram o PSO para otimização de custos de projeto de rede de distribuição de água. O algoritmo foi acoplado ao simulador hidráulico EPANET2 e as simulações foram realizadas em duas redes de referência, a rede de 2 malhas e a rede Hanoi. Foram realizadas comparações com os resultados obtidos em trabalhos anteriores que utilizaram o AG, o Algoritmo *Simulated Annealing* (SA) e o Algoritmo *Non-Equilibrium Simulated Annealing*. As soluções obtidas para a rede de distribuição de água de 2 malhas e para a rede Hanoi são mostrados na Tabela 3.3 e Tabela 3.4.

Tabela 3.3 – Soluções para a rede de 2 malhas (Suribabu e Neelakantan, 2006)

AUTORES	ALGORITMO DE OTIMIZAÇÃO	CUSTO ÓTIMO (Unid.)	NÚMERO DE AVALIAÇÕES	TEMPO COMPUTACIONAL (s)
Savic e Walters (1997)	AG	419.000	65.000	600
Cunha e Sousa (1999)	SA	419.000	25.000	40
Eusuff e Lansey (2003)	SFLA	419.000	11.155	Não disponível
Liong e Atiquzzaman (2004)	SCA	419.000	1.019	18
Suribabu e Neelakantan (2006)	PSO	419.000	760	2
Suribabu e Neelakantan (2006)	AG	420.000	58.380	86
Suribabu e Neelakantan (2006)	SA	419.000	268.200	43
Suribabu e Neelakantan (2006)	<i>Non-Equilibrium</i> SA	419.000	109.957	25

SFLA - *Shuffled Frog Leaping Algorithm*; SCA - *Shuffled Complex Algorithm*

Para a rede de 2 malhas, todos os algoritmos exceto o AG, obtiveram o menor custo (419.000), porém, o PSO necessitou de menor número de avaliações para encontrar a solução ótima, em menor tempo computacional.

Tabela 3.4 – Soluções para a rede Hanoi (Suribabu e Neelakantan, 2006)

AUTORES	ALGORITMO DE OTIMIZAÇÃO	CUSTO ÓTIMO (Un.)	NÚMERO DE AVALIAÇÕES	TEMPO COMPUTACIONAL (s)	COMENTÁRIO
Savic e Walter (1997)	AG	6.073.000	Não disponível	10.800	Violação da pressão no nó 2
Cunha e Sousa (1999)	SA	6.056.000	53.000	7.200	Violação da pressão no nó 6
Eusuff e Lansey (2003)	SFLA	6.073.000	26.987	Não disponível	Violação da pressão no nó 2
Liong e Atiquzzaman (2004)	SCA	6.220.000	25.402	660	-
Suribabu e Neelakantan (2006)	PSO	6.081.000	6.600	9	-
Suribabu e Neelakantan (2006)	AG	6.081.000	176.700	160	-
Suribabu e Neelakantan (2006)	SA	6.081.000	1.234.340	1.800	-
Suribabu e Neelakantan (2006)	<i>Non-Equilibrium SA</i>	6.081.000	18.752	47	-

SFLA - *Shuffled Frog Leaping Algorithm*

SCA - *Shuffled Complex Algorithm*

Já para o caso da rede Hanoi, os menores custos foram alcançados pelo algoritmo SA. Entretanto, o PSO apresentou o menor número de avaliações e tempo de processamento, com um custo muito próximo da obtida pelo SA.

O algoritmo Evolução Diferencial (*Differential evolution* - DE) foi proposto por Storn e Price (1995) e possui abordagem semelhante ao AG, pois emprega uma população de soluções e utiliza operadores de cruzamento (Odan, 2013). Os operadores envolvidos na DE durante a otimização são mutação, cruzamento e seleção. Os nomes dos operadores são semelhantes aos utilizados nos AGs, no entanto, existem diferenças significativas na ordem de aplicação e forma deles como no processo de mutação (Zheng *et al.* 2012).

O DE é um algoritmo de otimização global, para variáveis contínuas e foi desenvolvido por Storn e Price (1995). De acordo com Storn e Price (1997), a otimização realizada pelo DE é bastante semelhante ao do AG, com os operadores de mutação, cruzamento e seleção (Figura 3.6). Os nomes dos operadores são semelhantes aos utilizados nos AGs, no entanto, existem diferenças significativas na ordem de aplicação e forma deles como no processo de mutação, que envolve a criação de uma solução mutante através da seleção de três soluções aleatórias (x_{r0} , x_{r1} , e x_{r2}) da geração atual (Dandy *et al.*, 2010; Zheng *et al.*, 2012).

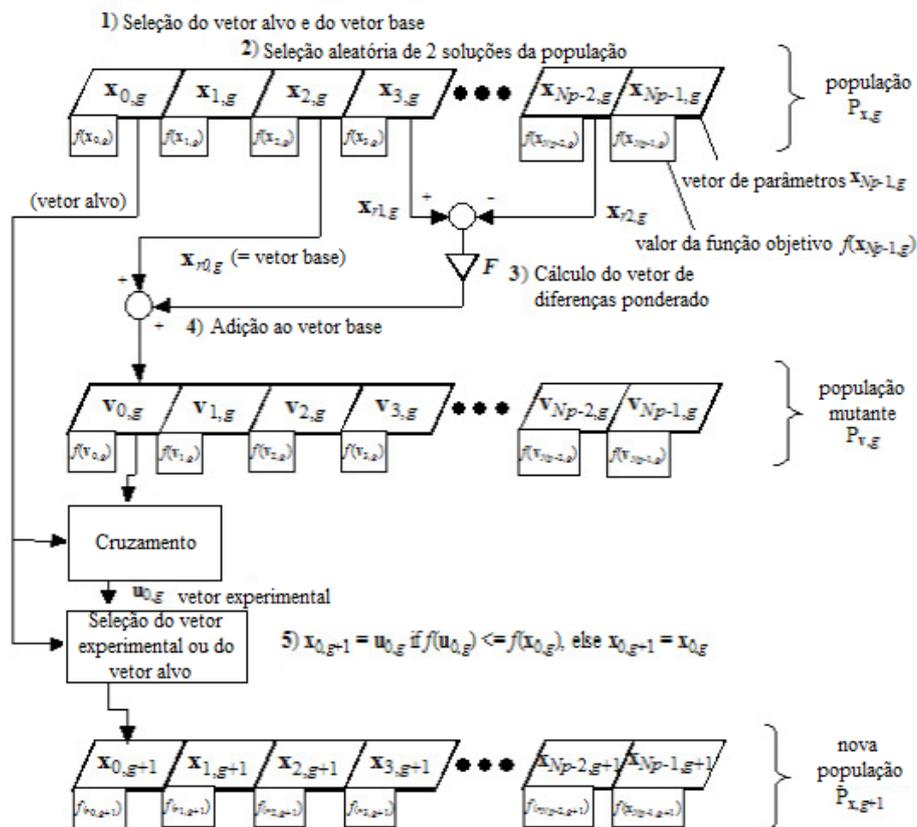


Figura 3.6 – Funcionamento dos operadores mutação, cruzamento e seleção do DE (Storn e Price, 1997 *apud* Dandy *et al.*, 2010)

Segundo Odan (2013), o algoritmo DE inicia com a seleção de uma solução aleatória para ser substituída (vetor alvo), juntamente com outra solução, denominada principal, $x_{r0,g}$ (vetor base).

Em seguida, são selecionadas aleatoriamente mais duas soluções ($x_{r1,g}$ e $x_{r2,g}$), calcula-se então, a diferença entre os vetores $x_{r1,g}$ e $x_{r2,g}$, a qual é ponderada por um fator F , de valor aleatório entre 0 e 1. A mutação de $x_{r0,g}$, consiste em realizar a adição do vetor de diferenças ponderado ao vetor base, que resulta em um vetor mutante, Equação 3.3, esse tipo de mutação representa a estratégia de mutação clássica do DE.

$$v_{0,g} = x_{r,0} + F \times (x_{r,1} - x_{r,2}) \quad (3.3)$$

em que:

$x_{r0,g}$ é a solução principal; $x_{r1,g}$ e $x_{r2,g}$ são as soluções aleatórias; $v_{0,g}$ é a solução mutante gerada; F é a constante de mutação.

O cruzamento entre o vetor alvo e o vetor mutante, que consiste em substituir os parâmetros do vetor alvo pelos parâmetros do vetor mutante, em que cada parâmetro tem uma probabilidade CR (probabilidade de cruzamento) de ser substituído, desse cruzamento resulta o vetor experimental $u_{0,g}$. O processo de cruzamento está ilustrado na Figura 3.7.

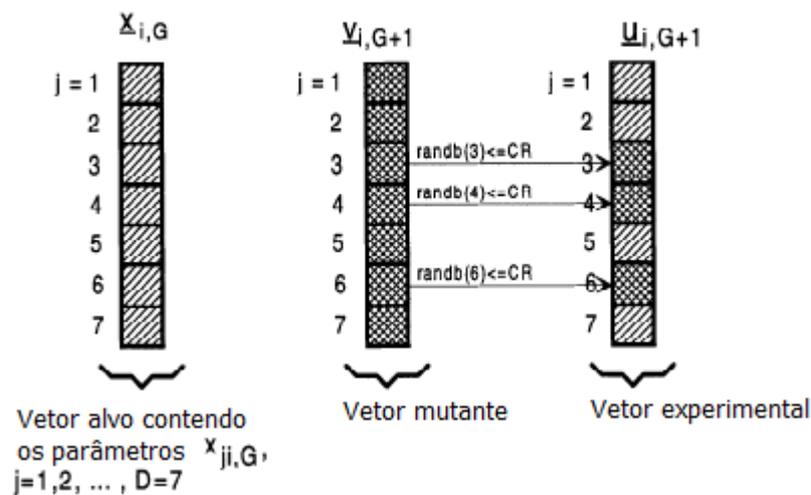


Figura 3.7 – Ilustração do processo de cruzamento para 7 parâmetros (variáveis de decisão - D) (Storn e Price, 1997)

Após o cruzamento, realiza-se o processo de seleção entre o vetor alvo e o vetor experimental, que depende da função aptidão (*fitness*).

Além do tamanho da população e do número máximo de gerações, o DE necessita da definição de dois parâmetros adicionais: a constante de mutação F e a constante de cruzamento CR , ambos no intervalo entre 0 e 1 (Marchi *et al.*, 2014).

Suribabu (2010) utilizou o algoritmo DE para otimizar projetos de SAA, ele aplicou o DE a 4 redes de referência e comparou os resultados com outros algoritmos de otimização, a partir desse estudo ele concluiu que a DE tem melhor ou igual desempenho se comparado ao AG, para a otimização de sistemas de distribuição de água. Em contrapartida, Dandy *et al.* (2010) relataram que o AG obteve desempenho geral melhor do que a DE e o PSO baseado em simulações realizadas em duas redes de referência. De acordo com Zheng *et al.* (2012), essa contradição pode ser explicada pela utilização de valores diferentes dos parâmetros nos dois estudos, além disso, essas comparações não são razoáveis, pois os valores dos parâmetros da DE não foram selecionados de forma adequada.

Zheng *et al.* (2012) utilizaram variantes da DE e do AG, e os aplicou em duas redes de referência. Mas antes eles determinaram valores apropriados para os parâmetros dos algoritmos, assim, concluíram baseados no estudo, que os algoritmos de evolução diferencial são mais adequados para a otimização de redes de distribuição de água, quando comparados aos algoritmos genéticos.

Tolson *et al.* (2008) apresentaram um algoritmo para a otimização de sistemas de distribuição de água, o DDDS (*Discrete Dynamically Dimensioned Search*), que é uma extensão discreta gerada a partir da modificação do DDS (*Dynamically Dimensioned Search*) proposto por Tolson e Shoemaker (2007), desenvolvido originalmente para resolver problemas de calibração automática de modelos de simulação de bacias hidrográficas.

Segundo Tolson *et al.* (2008), o DDS é um algoritmo estocástico de solução única baseado na busca heurística global, que foi desenvolvido com a finalidade de encontrar boas soluções globais, dentro de um limite de avaliação máximo da função objetivo, sendo este o critério de parada do algoritmo. Em outras palavras, o algoritmo realiza a busca em nível global no

início da otimização e a busca se torna mais local quando o número de iterações se aproxima do limite máximo de avaliações da função.

Tolson *et al.* (2008) adaptaram o DDS para resolver especificamente o problema de otimização de sistema de distribuição de água de um único objetivo (minimização dos custos de projeto de rede), sujeito a restrições como pressão mínima em todos os nós da rede, onde as variáveis de decisão são diâmetros de tubos (variáveis discretas) e comparou o desempenho do DDDS com o ACO e o AG, e os aplicou a duas redes de referência (Hanoi e *New York Tunnels*). O DDDS e o ACO apresentaram o menor custo para a rede *New York Tunnels* e para a rede Hanoi gerou o segundo menor custo, no geral o DDDS obteve bom desempenho nas duas aplicações, isso mostra o potencial de aplicação do algoritmo em problemas complexos de SAAs.

Dandy *et al.* (2010) desenvolveram uma metodologia para comparar o desempenho de AEs. A metodologia consiste de 4 etapas apresentadas na Figura 3.8.

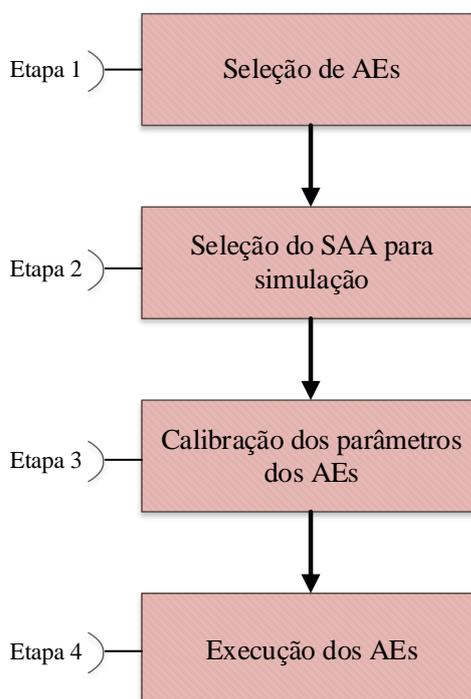


Figura 3.8 – Metodologia de comparação de AEs (Dandy *et al.*, 2010)

- (1) Seleção de algoritmos evolucionários para serem comparados - é uma parte subjetiva, pois depende do conhecimento e experiência dos pesquisadores. Segundo os autores, a literatura sugere que se utilize pelo menos um desses algoritmos na comparação:

AG, DE e ACO. Para o estudo, os autores, selecionaram os dois primeiros algoritmos e o PSO;

- (2) Seleção de problemas apropriados para a simulação - os autores selecionaram a rede *Nova York Tunnels* e a rede Hanoi;
- (3) Calibração/ajuste dos algoritmos selecionados - de acordo com os autores, na comparação de algoritmos pode ocorrer problemas referentes ao ajuste dos parâmetros, pois, defensores de um algoritmo em particular, podem passar mais tempo ajustando esse algoritmo que os outros. Dessa forma, os autores empregaram um procedimento de ajuste objetivo dos parâmetros, em que é identificada, a partir da literatura, uma faixa típica de valores dos parâmetros e realizadas simulações preliminares para cada AE com a combinação desses parâmetros. Assim, enquanto um parâmetro é testado os outros são mantidos constantes. O melhor conjunto de parâmetros para cada AE foi identificado, com base nessas simulações, pela melhor solução obtida.
- (4) Execução dos algoritmos evolucionários - às execuções finais foram realizadas para as duas redes com os parâmetros obtidos no passo 3; além disso, todas as técnicas foram executadas com o mesmo número de avaliações.

Os algoritmos foram comparados com base na solução ótima obtida, velocidade de convergência, e consistência das soluções obtidas ao longo do número de avaliações. Com as simulações observou-se que o AG levou a melhores resultados no geral. Já para um menor número de avaliações o PSO obteve os melhores resultados. No entanto, seu desempenho não melhorou significativamente com o aumento do número de avaliações. O DE apresentou bom desempenho para um grande número de avaliações. Como esse algoritmo é relativamente recente, espera-se que resultados melhores sejam encontrados a partir da experiência adquirida e sua aplicação em SAAs.

Marchi *et al.* (2014) utilizaram a metodologia e os algoritmos empregados por Dandy *et al.* (2010) para a otimização de 3 redes de referência, sendo 2 empregadas anteriormente no estudo de Dandy *et al.* (2010) e a uma rede denominada rede rural com 476 trechos de tubulação. Para a primeira rede, o AG e o DE apresentaram os melhores resultados, porém o DE encontrou a solução ótima em um menor tempo. Para a rede Hanoi, todos os algoritmos encontraram a solução ótima, entretanto, o DE novamente foi o mais rápido. Também para a rede rural, o DE encontrou a solução ótima em um tempo computacional menor.

De acordo com Lansey (2006), aplicações de otimização na área de SAAs têm sido geralmente bem sucedidas e progrediram ao longo do tempo, devido aos avanços na área computacional, na modelagem de SAAs e nos métodos de otimização. A maioria dos grandes problemas de otimização em SAA foram resolvidos, embora, em alguns casos, não foram solucionados da forma tão eficiente quanto necessário ou desejado.

Outras melhorias são necessárias como formulação de modelos de otimização de projetos considerando os transientes (Jung e Karney, 2006), estudo da confiabilidade, e especialmente, otimização da operação em tempo real. Os esforços iniciais na operação em tempo real foram realizados por Alonso *et al.* (2000), no processamento paralelo e Broad *et al.* (2005), na utilização de metamodelos em substituição aos simuladores tradicionais.

3.2.3 - Planejamento operacional em nível estratégico e a operação em tempo real

Segundo Cunha (2009), a otimização da operação de SAA pode ser abordada através do planejamento operacional em nível estratégico e da operação em tempo real, apesar de ambas as abordagens terem o objetivo de determinar estratégias operacionais otimizadas. O planejamento operacional em nível estratégico utiliza curvas de demanda típicas (médias) conhecidas e níveis d'água iniciais empregados para os reservatórios com a finalidade de determinar as trajetórias operacionais ótimas, de forma que os níveis d'água dos reservatórios no final do horizonte de planejamento (normalmente de 24 horas), coincidam com os níveis iniciais. Pode-se dizer que as estratégias obtidas não são diretamente aplicáveis à operação real, pois utilizam curvas de demanda médias, porém são ferramentas importantes para a avaliação da capacidade de armazenamento e transporte, adequação das bombas, atendimento das demandas e proporcionam subsídios para decisões relativas à contratação de tarifas de energia elétrica.

A operação em tempo real transforma uma estratégia operacional estática em uma estratégia operacional dinâmica, ou seja, um plano estratégico para uma estratégia constantemente em construção. Ela fornece para a operação, a capacidade de ajustar-se continuamente, apresentando decisões mais adequadas, mesmo quando o que se deve fazer, não se mostra tão óbvio (Cunha, 2009).

A otimização em tempo real para a definição de políticas de operação, normalmente, é realizada através de três componentes (Figura 3.9): um simulador hidráulico; um modelo de otimização; e, uma abordagem para a demanda, que pode ser um modelo de previsão de demanda e/ou históricos de demanda.

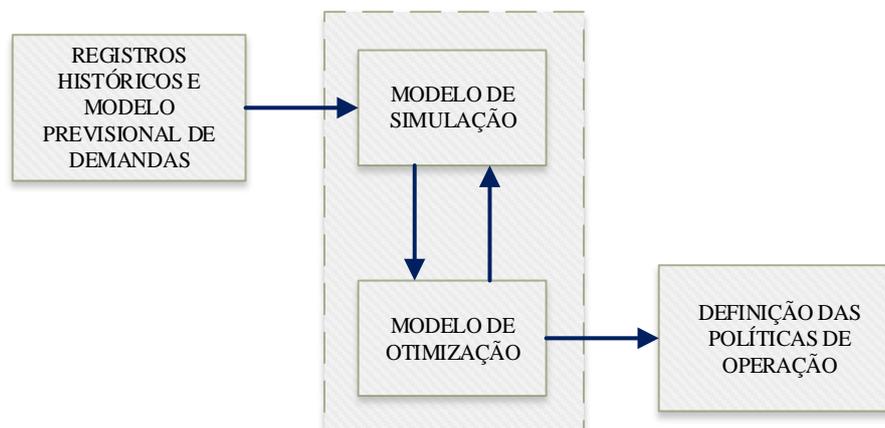


Figura 3.9 – Esquema de definição de políticas operacionais (Ribeiro, 2005)

Dentre os trabalhos que abordaram a operação em tempo real estão os estudos de: Lansey e Awumah (1994), Nitivattananon *et al.* (1996), Jamieson *et al.* (2007), Shamir e Salomons (2008), Boulos *et al.* (2014). No Brasil destacam-se os trabalhos de Cunha (2009), Odan (2013), Odan *et al.* (2014), Odan *et al.* (2015). A Tabela 3.5 mostra alguns desses trabalhos e os componentes citados anteriormente.

Tabela 3.5 - Trabalhos desenvolvidos sobre operação em tempo real

AUTORES/ANO	SIMULADOR HIDRÁULICO	MÉTODO/ALGORITMO DE OTIMIZAÇÃO	ABORDAGEM UTILIZADA PARA DEMANDA
Lansey e Awumah (1994)	KYPIPE	Programação dinâmica	Registros históricos de demanda
Nitivattananon <i>et al.</i> (1996)	Métodos heurísticos	Programação dinâmica	Registros históricos de demanda
Jamieson <i>et al.</i> (2007)	RNA	AG	Perfis históricos de demanda
Shamir e Salomons (2008)	EPANET2	AG	Registros históricos de demanda

Tabela 3.5 – Continuação

AUTORES/ANO	SIMULADOR HIDRÁULICO	MÉTODO/ALGORITMO DE OTIMIZAÇÃO	ABORDAGEM UTILIZADA PARA DEMANDA
Cunha (2009)	EPANET2	AG	Registros históricos e Previsor de demanda (Modelo híbrido DAN2)
Odan <i>et al.</i> (2014)	EPANET2 e Metamodelo baseado em RNA	AMALGAM (NSGA-II, DE, PSO e AMS)	Previsor de demanda (Modelo híbrido DAN2)

Além dos trabalhos mostrados na Tabela 3.5, também está disponível na literatura nacional, o trabalho de Vicente (2005) e Ribeiro (2005), em que ambas desenvolveram um modelo para otimização da operação em tempo real do Sistema Adutor Metropolitano Leste de São Paulo, com o objetivo de minimizar os custos de energia elétrica no sistema. Vicente (2005) utilizou a PL como módulo de otimização, o simulador hidráulico *WaterCad* e um previsor de demanda baseado em série de Fourier. O tempo de processamento do modelo para a simulação de 24 horas de operação foi de 45 min e 12 min para a previsão de demanda, utilizando um microcomputador Pentium 4 (1GHz) com 512 Mb RAM.

Ribeiro (2005) utilizou o AG para a otimização, o simulador hidráulico EPANET2 e registros históricos de demanda, ao invés de um previsor de demanda. O tempo computacional demandado pelo programa desenvolvido por Ribeiro (2005) foi de 3h20 min, para simular 24h de operação, utilizando um computador pessoal com microprocessador tipo Pentium 4 com 2,2 GHz de velocidade de processamento e 358 MB de memória RAM. Somente o simulador hidráulico utilizou 2 horas do tempo de processamento, segundo o autor, a natureza complexa do sistema colaborou para a alta demanda computacional. De acordo com Lansey e Awumah (1994), o tempo computacional aumenta linearmente com a quantidade de demandas de água e níveis de reservatórios, e aumenta exponencialmente com o número de bombas.

3.2.3.1 - Estratégias para reduzir o tempo de processamento do algoritmo

Normalmente, a utilização do simulador hidráulico tradicional (EPANET2) necessita de considerável tempo de processamento computacional para otimizar a operação de SAAs,

podendo tornar a otimização em tempo real inviável, embora no presente estudo seja realizado o planejamento operacional em nível estratégico.

Uma alternativa para solucionar o problema do tempo de processamento computacional dos simuladores hidráulicos tradicionais seria a utilização de modelos esqueletonizados, que consiste em simplificar o sistema, reduzindo a quantidade de nós e trechos a serem representados no modelo de simulação (Odan, 2013). Segundo Walski *et al.* (2003) a esqueletonização permite a geração de resultados confiáveis e precisos sem a utilização desnecessária de tempo.

Shamir e Salomons (2008), com o objetivo de realizar a otimização da operação de SAA em tempo real, compararam os resultados de um sistema representado por um modelo completo com 867 nós e 987 trechos (Figura 3.10a); e um modelo simplificado do mesmo sistema com 77 nós e 92 trechos (Figura 3.10b).

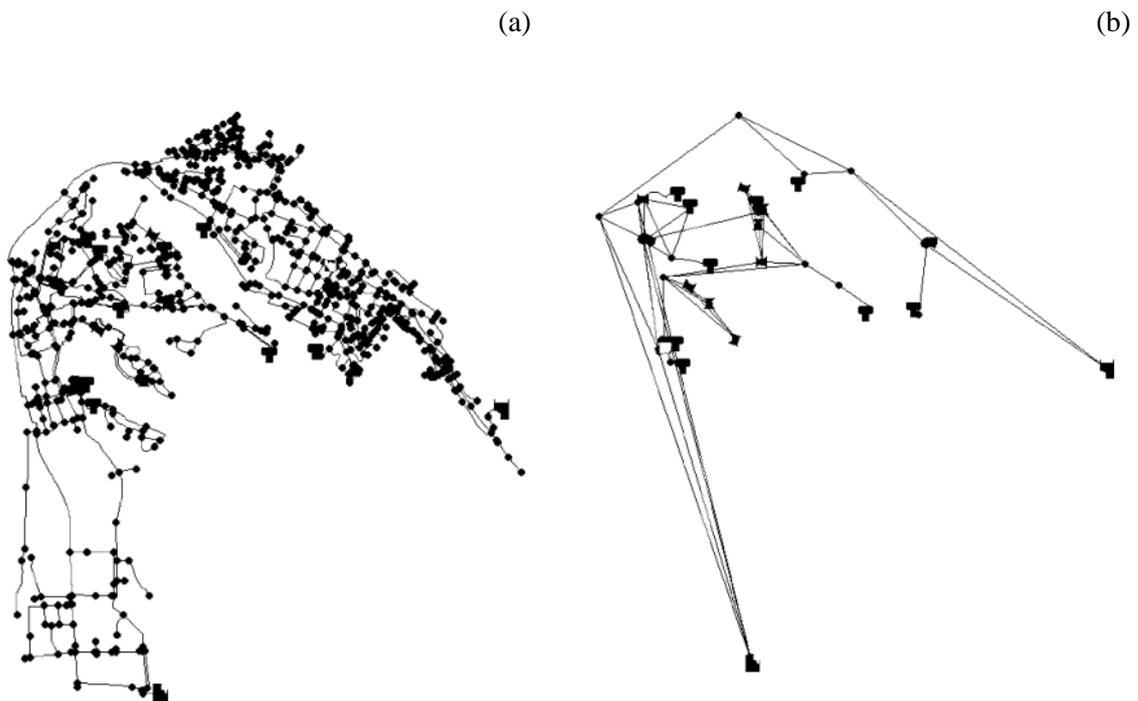


Figura 3.10 - (a) Sistema completo (b) Sistema simplificado (Shamir e Salomons, 2008)

O estudo mostrou que o modelo reduzido reproduziu com elevada precisão os resultados do modelo completo (Figura 3.11), além disso, foi verificado que a simplificação do modelo reduziu significativamente o tempo de processamento do programa, que foi de 8 horas para

o modelo completo e de 40 segundos para o modelo reduzido, ambos processados em um Pentium 4 (2 GHz) IBM ThinkPad com 1,5 GB RAM, considerando 24 horas de simulação.

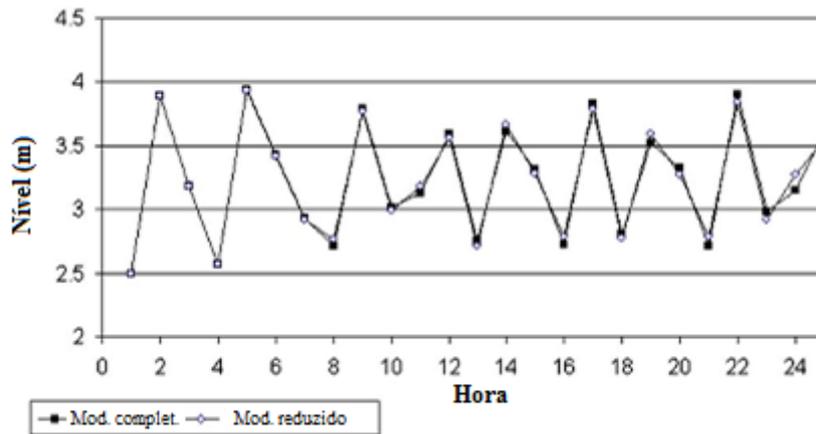


Figura 3.11 – Comparação do comportamento do nível de água em um reservatório do sistema (Shamir e Salomons, 2008)

Odan (2013) desenvolveu um modelo para a otimização da operação de SAAs, em tempo real, com o objetivo de minimizar custos de bombeamento e maximizar a confiabilidade hidráulica do sistema. Para a otimização ele utilizou um algoritmo multimétodo (ou hiperheurística) chamado AMALGAM, que emprega simultaneamente diferentes algoritmos de busca. Os algoritmos utilizados pelo AMALGAM foram: NSGA-II, PSO, AMS e DE. Além dos simuladores hidráulico (EPANET2 e metamodelo) e o previsor de demanda baseado na Rede Neural Dinâmica (DAN2). O modelo foi aplicado à 3 redes de distribuição de água (Eliana, Iguatemi e Martinez) da cidade de Araraquara/SP.

A estratégia utilizada por Odan (2013), para reduzir o tempo de processamento do modelo e possibilitar a otimização em tempo real, foi a simplificação dos sistemas, a aplicação do metamodelo, porém não foi obtido sucesso na implementação deste e a utilização de solução inicial proveniente do planejamento estratégico. O autor verificou que houve redução de 14%, 13% e 30%, respectivamente, nos custos de energia elétrica dos sistemas Eliana, Iguatemi e Martinez. A otimização mono objetivo (minimizar custos de bombeamento), com as redes simplificadas simularam 24h de operação em cerca de 1,5 min, já o tempo de processamento para otimização multiobjetivo, utilizando a rede completa, foi de 1,8, 16,7 e 2,5 minutos.

De acordo com Poles *et al.* (2009), a população inicial desempenha um papel importante na convergência e, conseqüentemente, no tempo de processamento de AGs independentemente do tipo do algoritmo e do problema. A estratégia mais comum para geração da população inicial de um algoritmo genético é a randômica. No entanto, algumas técnicas podem ser aplicadas com o propósito de diversificar a população inicial e representar pontos em todos os espaços de busca, ou ainda com a finalidade de conduzir a busca inicial do AG para uma determinada região conhecidamente factível, para acelerar a convergência do algoritmo (Haubelt *et al.*, 2005).

A técnica *seeding* consiste na introdução de uma ou mais soluções já conhecidas na população inicial do algoritmo, essas soluções são previamente definidas por outros métodos ou através do conhecimento do operador (Machado *et al.*, 2008).

Segundo Gandibleux *et al.* (2001) utilizar soluções iniciais boas permite a propagação da informação genética superior a outros indivíduos durante o processo de evolução. Além disso, quando essas soluções são utilizadas na população inicial, o tempo de processamento computacional e o número de gerações necessárias para encontrar as soluções ótimas são reduzidos consideravelmente.

Savic *et al.* (1997) realizaram alterações no código desenvolvido por Mäckle *et al.* (1995), que otimiza os custos de bombeamento de SAA utilizando AG. Sendo uma das mudanças a inserção da técnica *seeding* na população inicial com as soluções obtidas em simulações anteriores. Os resultados obtidos mostraram que a técnica trouxe melhorias na velocidade e qualidade das soluções encontradas, se comparado com a utilização da população aleatória, mesmo no caso em que, as alterações nos níveis iniciais dos reservatórios do sistema foram grandes.

Machado *et al.* (2008) realizaram a otimização multiobjetivo em SAAs utilizando um AG. Os autores tinham como objetivos minimizar o custo de energia elétrica referente ao bombeamento, minimizar o custo de manutenção das bombas e minimizar a variação do nível do reservatório, sem violar as restrições operacionais. Com o objetivo de guiar a busca das soluções por regiões mais promissoras do espaço de busca, eles empregaram o conceito de otimalidade de Pareto, o conhecimento nos operadores de reprodução e a técnica de *seeding*.

Os autores realizaram experimentos considerando 2 suposições, em que a primeira considera que a solução de referência é uma solução ótima do problema previamente conhecida. A suposição 2 considerou que nenhuma solução ótima é previamente conhecida sendo a população inicial do AG, a solução gerada pelo calculador de solução viável (CSV).

Os resultados mostraram que a metodologia foi adequada na otimização de SAAs para as duas suposições, no entanto, é mais recomendada para casos que já possuem uma operação aplicável ao sistema (solução conhecida), a qual pode ser a operação usual dos operadores, e se deseja otimizar essa operação ou mesmo encontrar outras soluções alternativas tão boas quanto a usual. Para casos em que essas regras não estão disponíveis (suposição 2), também foram encontradas soluções ótimas, embora pouco usuais.

É importante mencionar que a utilização de uma população bem distribuída em vez de uma população inicial aleatória, aumenta a robustez e a convergência de um AG, em contrapartida se a informação genética presente na população não for suficiente, um AG pode sofrer convergência prematura (acontece quando a informação genética armazenada nos indivíduos de uma população não contém a informação genética necessária para melhorar a qualidade da solução) e ficar preso em soluções ótimas locais (Affenzeller e Wagner, 2004; Poles *et al.*, 2009).

3.2.4 - Representação da programação de bombas

De acordo com Ormsbee e Reddy (1995), a programação da operação diária de bombas é uma área potencial para redução de custos em SAA. A representação da programação de bombas (RPB), isto é, o estado em que as bombas são representadas durante o processo de otimização e o algoritmo de otimização, interferem na qualidade e velocidade do processo de otimização (Odan, 2013).

Segundo López-Ibáñez (2009), a RPB pode ser explícita, pela especificação direta do estado de cada bomba, ou implícita, definindo a operação de bombas em termos da propriedade de outros elementos do sistema. As representações explícita e implícita mais utilizadas são respectivamente, a binária e a representação baseada nos níveis de reservatório. A representação binária divide o período de programação em pequenos intervalos de tempo e

codifica a programação da bomba em uma cadeia de *bits*, cada *bit* representa o estado da bomba (ligado/desligado) durante o intervalo de tempo (Figura 3.12). Por outro lado, a representação baseada no nível do reservatório provoca alteração dos estados das bombas dependendo do nível dentro do reservatório. Assim, a bomba é desligada quando o nível de água exceder um nível máximo e é ligada quando o nível de água atingir um nível mínimo.

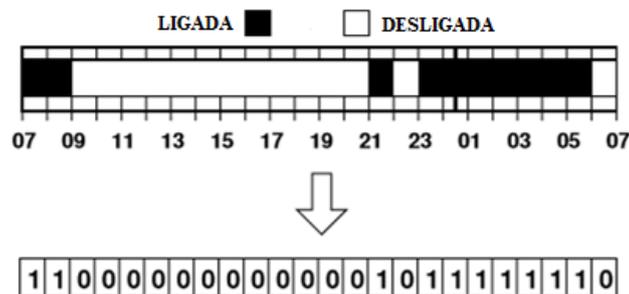


Figura 3.12 – Representação binária da programação de bombas (López-Ibáñez, 2009)

Em contraste com a representação binária, que codifica o estado de uma bomba durante cada intervalo de tempo, a RPB denominada “*time-controlled trigger*” (representação de acionamentos controlados) proposta por López-Ibáñez (2009) possibilita o controle do estado da bomba por meio do tempo e limita o número de acionamentos de forma explícita. O estado de cada bomba durante a análise é representado por pares de variáveis de decisão t_i e t'_i , apresentados na Figura 3.13. Cada par representa o tempo em que a bomba deve permanecer desligada (t_i) e ligada (t'_i).

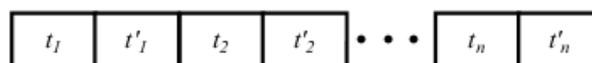


Figura 3.13 - RPB no tempo (Odan, 2013)

A Figura 3.14 exemplifica a RPB proposta por López-Ibáñez (2009), onde a soma de todos os pares é igual a 24h (período de simulação), a quantidade máxima de acionamentos da bomba é 3; t_1 é igual a 3, ou seja, a bomba deve estar desligada durante 3h desde o início do período; t'_1 igual a 4 significa que a bomba deve ficar ligada por 4h após o intervalo de t_1 e assim sucessivamente até chegar em t'_3 igual a 7, onde a bomba ficará ligada por 7h após o intervalo t_3 .

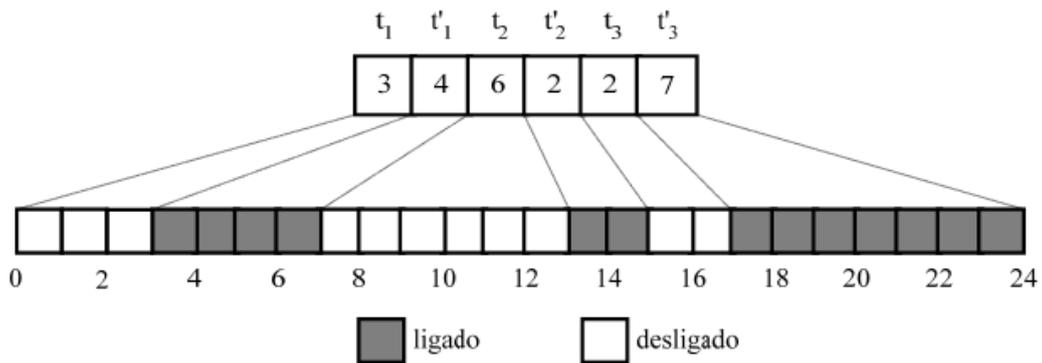


Figura 3.14 – Exemplo da representação das variáveis de decisão segundo a RPB “*time-controlled trigger*” (Odan, 2013)

Segundo Lansey e Awumah (1994), regras operacionais na qual bombas são ligadas e desligadas muitas vezes podem reduzir o consumo de energia elétrica. No entanto, essas regras podem desgastar as bombas e aumentar os custos de manutenção. Estes custos não foram quantificados, mas pode-se supor que aumentam à medida que aumentam os acionamentos. Para lidar com essa dificuldade, alguns trabalhos têm utilizado o método das penalidades, como os estudos realizados por Sakarya e Mays (2000), Pedrosa (2006), Cunha (2009) e Gebrim (2013).

Segundo Berhe (2012), o método das penalidades é uma das melhores técnicas para lidar com problemas de otimização não linear, pois o método converge para pelo menos, um mínimo local na maioria dos casos, independentemente, das características de convexidade da função objetivo e das restrições.

3.2.5 - Método das penalidades

As funções de penalidade foram originalmente propostas por Courant na década de 1940. A ideia do método é transformar um problema de otimização restrito em um problema irrestrito pela adição (ou subtração) de um certo valor da função objetivo baseado nas violações da restrição presente em determinada solução (Coello, 2002). Segundo Wu e Simpson (2002), o método das penalidades tem como objetivo penalizar soluções inviáveis e forçar a busca para uma região de soluções viáveis.

No método das penalidades cada restrição é inserida na função objetivo ligada a um peso. Os pesos têm a função de ponderar as penalidades, inclusive definindo as mais importantes

para a otimização. O resultado da avaliação da função depende diretamente da quantidade das restrições violadas (Oliveira, 2001).

Os pesos das penalidades devem ser ajustados com muito cuidado, pois uma pequena penalidade (penalidade pouco severa) pode levar a soluções inviáveis, ou seja, uma região muito grande será explorada e grande parte do tempo utilizado na busca será usado para explorar regiões distantes da região viável. Por outro lado, os altos níveis de penalidade (penalidade muito severa) restringem a busca dentro da região viável, proibindo qualquer atalho para a região inviável e, assim, eventualmente, a população não consegue convergir para a solução ótima (Smith e Coit, 1997; Kusakci e Can, 2012).

Vários métodos foram desenvolvidos para evitar os problemas mencionados. Segundo Coello (2002), os métodos baseados nas funções de penalidades podem ser classificados como penalidade de morte, estáticas, dinâmicas, adaptativas, *annealing*, co-evolucionárias.

3.2.5.1 - Penalidades de morte

A penalidade de morte é considerada a maneira mais fácil de lidar com restrições e é computacionalmente eficiente (Coello, 2002). Segundo Smith e Coit (1997) esse método penaliza as soluções inviáveis aplicando uma penalidade constante às soluções que violam às restrições. A função objetivo penalizada será, então, a função objetivo não-penalizada somada as penalidades, para casos de minimização da função. A função de penalidade para um problema com m restrições para um problema de minimização é mostrado na Equação 3.4.

$$f_p(x) = f(x) + \sum_{i=1}^m C_i \lambda_i \quad (3.4)$$

$$\lambda_i = \begin{cases} 1, & \text{se a restrição } i \text{ é violada} \\ 0, & \text{se a restrição } i \text{ é satisfeita} \end{cases}$$

em que:

$f_p(x)$ = função objetivo penalizada (função aptidão); $f(x)$ = função objetivo sem penalização;

C_i = constante aplicada se a restrição i for violada.

O método de eliminar soluções inviáveis de uma população pode funcionar, razoavelmente bem, quando o espaço de busca viável é convexo e constitui uma parte considerável do espaço de busca. Caso contrário, essa abordagem possui limitações, como no caso de problemas de busca em que os indivíduos inviáveis, de uma população inicial, que, provavelmente, serão descartados pelo método, podem ser essenciais para melhorar a solução otimizada. Além disso, muitas vezes o sistema pode obter a solução otimizada mais facilmente se for possível atravessar uma região inviável, especialmente em espaços de busca viáveis não convexos (Michalewicz *et al.*, 1996).

3.2.5.2 - Penalidades estáticas

O método de penalidade estática inclui uma métrica de distância para cada restrição e adiciona uma penalidade que se torna mais grave com a distância da região viável. O problema desta abordagem é a suposição de que a distância métrica escolhida fornece informações adequadas sobre a proximidade da solução para a viabilidade, e a suposição ainda mais implícita de que a proximidade da viabilidade tem a mesma importância para a aptidão da solução (Smith e Coit, 1997).

Segundo os autores, esse método de penalidade é mais eficaz que o método de penalidade de morte, embora a penalização de acordo com a distância estar grosseiramente presente nas funções de penalidade constante (penalidade de morte, assumindo que a distância pode ser declarada apenas pelo número de restrições violadas). As métricas de distância podem ser contínuas (Juliff, 1993) ou discretas (Patton *et al.*, 1995) e podem ser lineares ou não lineares (Le Riche *et al.*, 1995).

A Equação 3.5 representa a formulação geral deste método para problemas de minimização.

$$f_p(x) = f(x) + \sum_{i=1}^m C_i d_i^k \quad (3.5)$$

$$d_i = \begin{cases} \lambda_i g_i(x), & \text{para } i = 1, \dots, q \\ |h_i(x)|, & \text{para } i = q + 1, \dots, m \end{cases}$$

em que:

d_i é a métrica de distância da restrição i aplicada à solução x ; κ é um expoente definido pelo usuário, com valores de κ frequentemente usados de 1 ou 2; C_i = constante aplicada se a restrição i for violada. As restrições de 1 a q são restrições de desigualdade, então a penalidade só será ativada quando a restrição for violada, enquanto as restrições $q + 1$ a m são restrições de igualdade que ativarão a penalidade se houver distância entre o valor da solução e o valor da restrição.

Homaifar *et al.* (1994) propôs a primeira abordagem para o uso de penalização estática, onde o usuário estabelece múltiplos níveis de violação (graus de penalização) para cada restrição, e para cada grau de violação é definido um coeficiente de penalização, de modo que o coeficiente aumenta à medida que os níveis de violação também aumentam. A Equação 3.6 apresenta a função de penalização estática para um problema com m restrições, onde restrições de igualdade são transformadas em restrições de desigualdade (Yeniay, 2005; Carvalho, 2014).

$$F(x) = f(x) + \sum_{i=1}^m R_{k,i} \cdot \max[0, g_i(x)^2] \quad (3.6)$$

em que:

$F(x)$ é a função objetivo penalizada, $f(x)$ é a função objetivo sem penalização, $R_{k,i}$ são os coeficientes de penalização da restrição i e $k = 1, 2, \dots, l$, sendo l o número de níveis de violação das restrições definidas.

Descrições e formulações de métodos de penalidades mais sofisticados como as abordagens dinâmicas, adaptativas, *annealing*, co-evolucionárias podem ser encontradas nos trabalhos de Michalewicz (1995), Smith e Coit (1997) e Coello (2002).

Michalewicz e Schoenauer (1996) concluíram que o método da função de penalidade estática é uma abordagem mais robusta do que os métodos sofisticados. Isso ocorre porque os métodos sofisticados podem não funcionar tão bem para um determinado problema de otimização (Deb, 2000). Segundo Smith e Coit (1997) esta abordagem tem a desvantagem de exigir ajuste iterativo através da experimentação de um grande número de parâmetros.

Além do método das penalidades há outras formas de lidar com as restrições, uma delas é a Lógica *Fuzzy* (Coello, 2002).

3.2.6 – Lógica *Fuzzy*

De acordo com Camboim *et al.* (2014), a LF trabalha com conceitos incertos, sendo uma técnica de caracterização de classes que não define limites rígidos entre elas. O emprego da LF é indicado para problemas com ambiguidade, abstração e ambivalência em modelos matemáticos ou conceituais de fenômenos empíricos. Por suas características intrínsecas, a LF é capaz de incorporar tanto o conhecimento objetivo relativo aos dados numéricos, quanto o conhecimento subjetivo, referentes às informações linguísticas.

Segundo Camboim (2012), o processo de tomada de decisão através da LF segue o diagrama de dados mostrado na Figura 3.15, composto pelos seguintes blocos funcionais: *Fuzzificação*, *Inferência Fuzzy*, *Defuzzificação*.

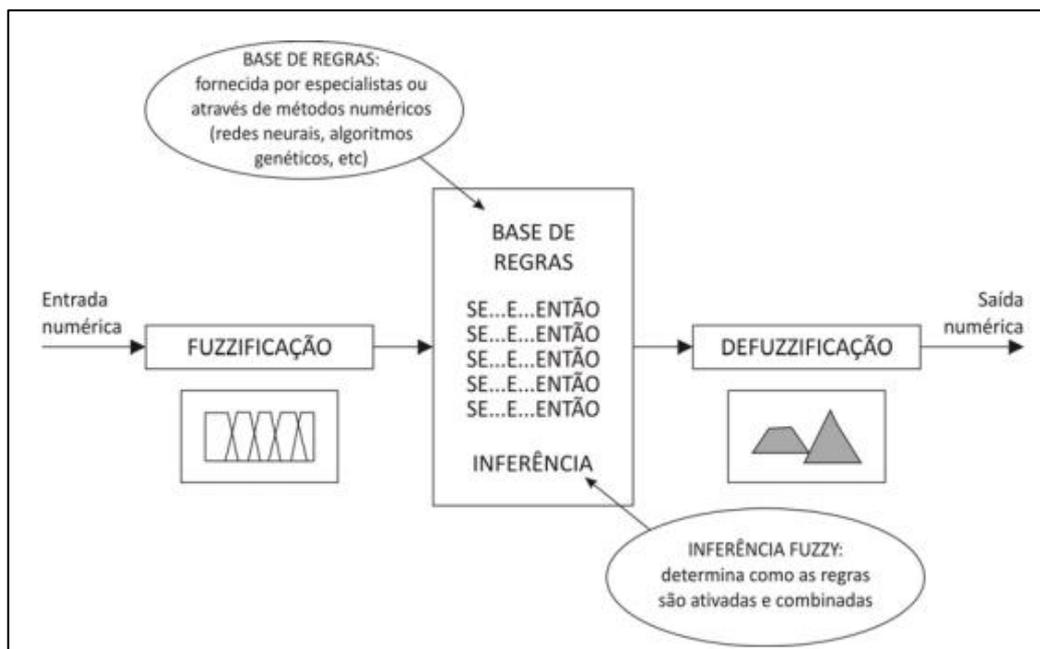


Figura 3.15 – Diagrama de blocos LF (Bezerra, 2009)

Segundo Camboim (2012), antes de iniciar o processo *Fuzzy*, deve-se determinar as variáveis que estarão envolvidas no processo. As variáveis de entrada e saída são as grandezas que darão apoio para a tomada de decisão ou as grandezas a serem controladas. Para cada

variável linguística deve ser empregado um universo de discurso, que serão todos os valores que a variável poderá adquirir. Assim, para cada variável de entrada, são criadas funções de pertinência dentro desse universo de discurso, que através do grau de pertinência conferido à função definirá o valor da entrada.

Uma função de pertinência é uma função numérica gráfica ou tabulada que concede valores de pertinência *Fuzzy* para valores discretos de uma variável, em seu universo de discurso. O universo de discurso de uma variável representa o intervalo numérico de todos os possíveis valores reais que uma variável específica pode adquirir (Simões e Shaw, 2007).

As curvas das funções de pertinência podem assumir diversas formas como triangular, trapezoidal, gaussiana (forma de parábola), singleton (grau de pertinência 1 para um único valor) (Camboim, 2012). A Figura 3.16 ilustra essas funções de pertinência.

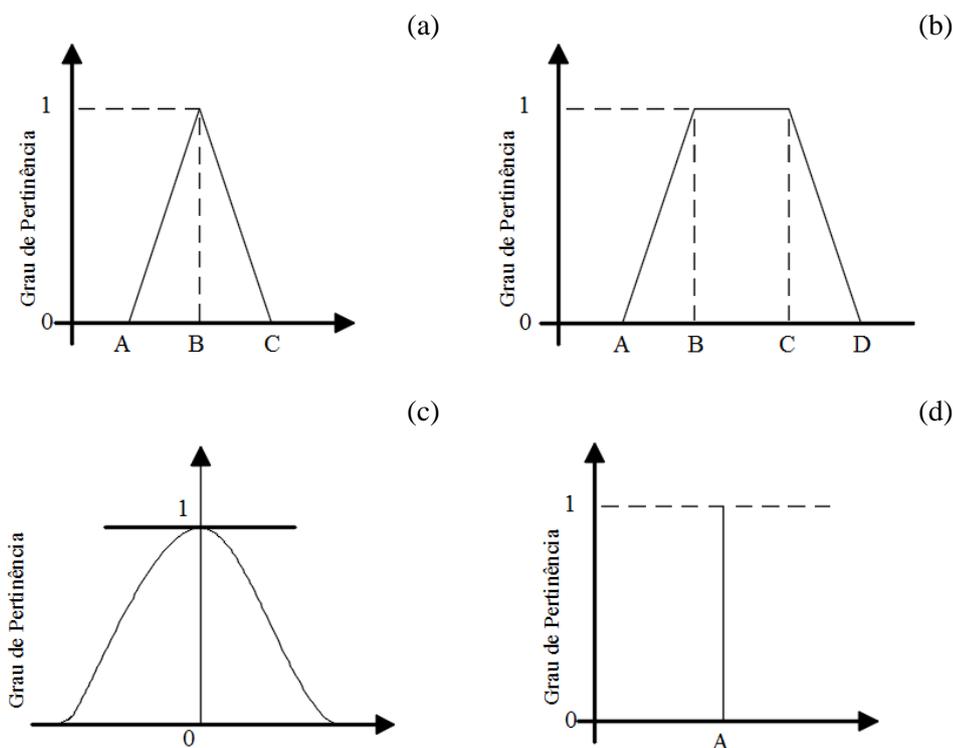


Figura 3.16 - Funções de pertinência (a) triangular; (b) trapezoidal; (c) gaussiana; e d) singleton (Camboim, 2012, adaptado)

Muitos autores utilizaram funções de pertinência *fuzzy* referentes a variáveis hidráulicas em seus trabalhos tais como: Revelli e Ridolfi (2002), Bhave e Gupta (2004), Sivakumar *et al.* (2014), Lacerda *et al.* (2015). Esses autores, as variáveis hidráulicas e as respectivas funções de pertinência estão apresentadas na Tabela 3.6.

Tabela 3.6 – Variáveis hidráulicas e suas respectivas funções de pertinência

AUTOR/ANO	VARIÁVEIS	FUNÇÃO DE PERTINÊNCIA
Revelli e Ridolfi (2002)	Rugosidade	Triangular e Trapezoidal
Bhave e Gupta (2004)	Demandas e Cargas	Trapezoidal
Sivakumar <i>et al.</i> (2014)	Rugosidade	Triangular
	Instante do dia	Trapezoidal
Lacerda <i>et al.</i> (2015)	Nível do reservatório	Trapezoidal
	Estado da bomba	Triangular

A teoria dos conjuntos *Fuzzy* foi introduzido por Zadeh (1965). Segundo o autor, o conjunto é caracterizado por funções de pertinência (características), que atribui a cada objeto um grau de pertinência que varia entre 0 e 1.

Na teoria de conjuntos clássica, um elemento pertence ou não a um dado conjunto. Dado um universo U e um elemento particular $x \in U$, o grau de pertinência $\mu_A(x)$ com respeito a um conjunto $A \subseteq U$ é dado por:

$$\mu_A(x) = \begin{cases} 1 & \text{se } x \in A \\ 0 & \text{se } x \notin A \end{cases} \quad (3.7)$$

Segundo Drinkov *et al.* (1996) *apud* Jameel e Radhi (2014), o conjunto *Fuzzy* referente a funções de pertinência trapezoidais lineares (Figura 3.17) é representado pela Equação 3.8.

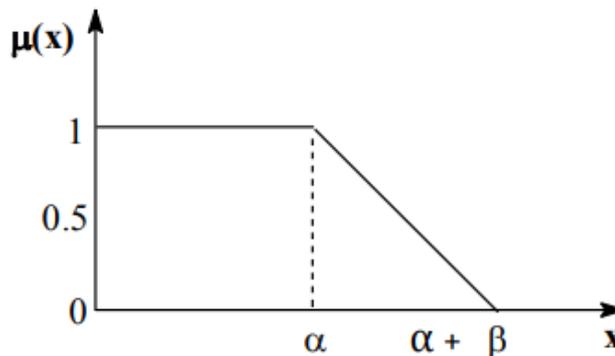


Figura 3.17 – Função de pertinência trapezoidal L (Jameel e Radhi, 2014)

A função $L: X \rightarrow [0,1]$ com 2 parâmetros é definida como:

$$L(x; \alpha + \beta) = \begin{cases} 1, & \text{if } x < \alpha \\ \frac{\alpha + \beta - x}{\beta}, & \text{if } \alpha \leq x \leq \alpha + \beta \\ 0, & \text{if } x > \alpha + \beta \end{cases} \quad (3.8)$$

em que:

L é a função de pertinência trapezoidal, α e β são parâmetros, x é o universo de discurso.

A utilização da LF para lidar com as restrições dos problemas de otimização ocorre pela substituição das restrições, pelo conjunto *Fuzzy*, como mostrado no trabalho de Van Le (1995), que propôs uma combinação da LF e da programação evolucionária para lidar com restrições no processo de otimização. A ideia principal foi substituir as restrições na forma da Equação 3.9 pelo conjunto de restrições *fuzzy* $C_1, \dots, C_m, i = 1, \dots, m$ definido pela Equação 3.11, a partir da *fuzzificação* das restrições, considerando um problema onde as variáveis de decisão são: $\vec{x} = (x_1, \dots, x_n)$.

$$g_i(\vec{x}) \leq b_i \quad i = 1, \dots, m \quad (3.9)$$

$$\mu_{C_i}(\vec{x}) = \mu_{\sigma(b_i, \varepsilon_i)}(g_i(\vec{x})) \quad i = 1, \dots, m \quad (3.10)$$

$$\mu_{C_i}(\vec{x}) = \begin{cases} 1 & \text{se } g_i(\vec{x}) \leq b_i \\ \frac{e^{-\left(\frac{g_i(\vec{x}) - b_i}{\varepsilon_i}\right)^2} - e^{-1}}{1 - e^{-1}} & \text{se } b_i < g_i(\vec{x}) \leq b_i + \varepsilon_i \\ 0 & \text{se } g_i(\vec{x}) > b_i + \varepsilon_i \end{cases} \quad (3.11)$$

em que:

g_i = função que calcula a restrição i ; b_i = limites de violação da restrição; \vec{x} = vetor das variáveis de decisão; n = número de variáveis de decisão; ε_i = tolerância de violação da restrição (acréscimo ao limite de violação); i = restrições; m = número de restrições; C_i = conjunto de restrições *fuzzy*.

A função objetivo é então mostrada na Equação 3.12:

$$f(\vec{x}) = \min(\mu_{C_1}(\vec{x}), \dots, \mu_{C_m}(\vec{x})) \quad (3.12)$$

Segundo Coello (2002) a ideia de usar graus de restrição como fatores de peso na aptidão de soluções potenciais é interessante e o uso de LF para definir a aceitabilidade de uma determinada solução parece ser uma forma natural lidar com restrições. No entanto, o principal problema dessa abordagem é que ela requer a definição da violação tolerável das restrições (limites máximos e mínimos) e do peso para cada problema particular. Além disso, Van Le (1995) fornece pouca evidência empírica do desempenho de sua técnica, embora este seja certamente um caminho de pesquisa que vale a pena explorar.

Saha *et al.* (2016) propuseram uma nova abordagem de função de penalidade baseada em regras *Fuzzy* para resolver problemas de otimização mono objetivo utilizando o algoritmo DE. Eles desenvolveram um sistema de inferência *Fuzzy* baseado em regras IF-THEN do sistema Mamdani. Os resultados das simulações mostraram que as penalidades *Fuzzy* não só superam as penalidades auto-adaptativas, mas também permaneceram competitivas em relação a várias estratégias padrões, bem como as estratégias complexas desenvolvidas atualmente.

4 - METODOLOGIA

A metodologia é apresentada nas seguintes etapas: (1) Caracterização do SAA do Rio Descoberto; (2) Descrição e implementação do modelo de otimização selecionado; (3) Implementação e simplificação do modelo hidráulico; (4) Utilização de algoritmos de otimização; (5) Tratamento das penalidades; e (6) Emprego da técnica *seeding*.

4.1 - CARACTERIZAÇÃO DO SAA DO RIO DESCOBERTO

Segundo CAESB (2014), o Sistema Produtor de Água para Atendimento Urbano (SPA) está dividido em cinco grandes sistemas produtores: Torto/Santa Maria, Descoberto, Brazlândia, Sobradinho/Planaltina e São Sebastião (Figura 4.1 e Figura 4.2).

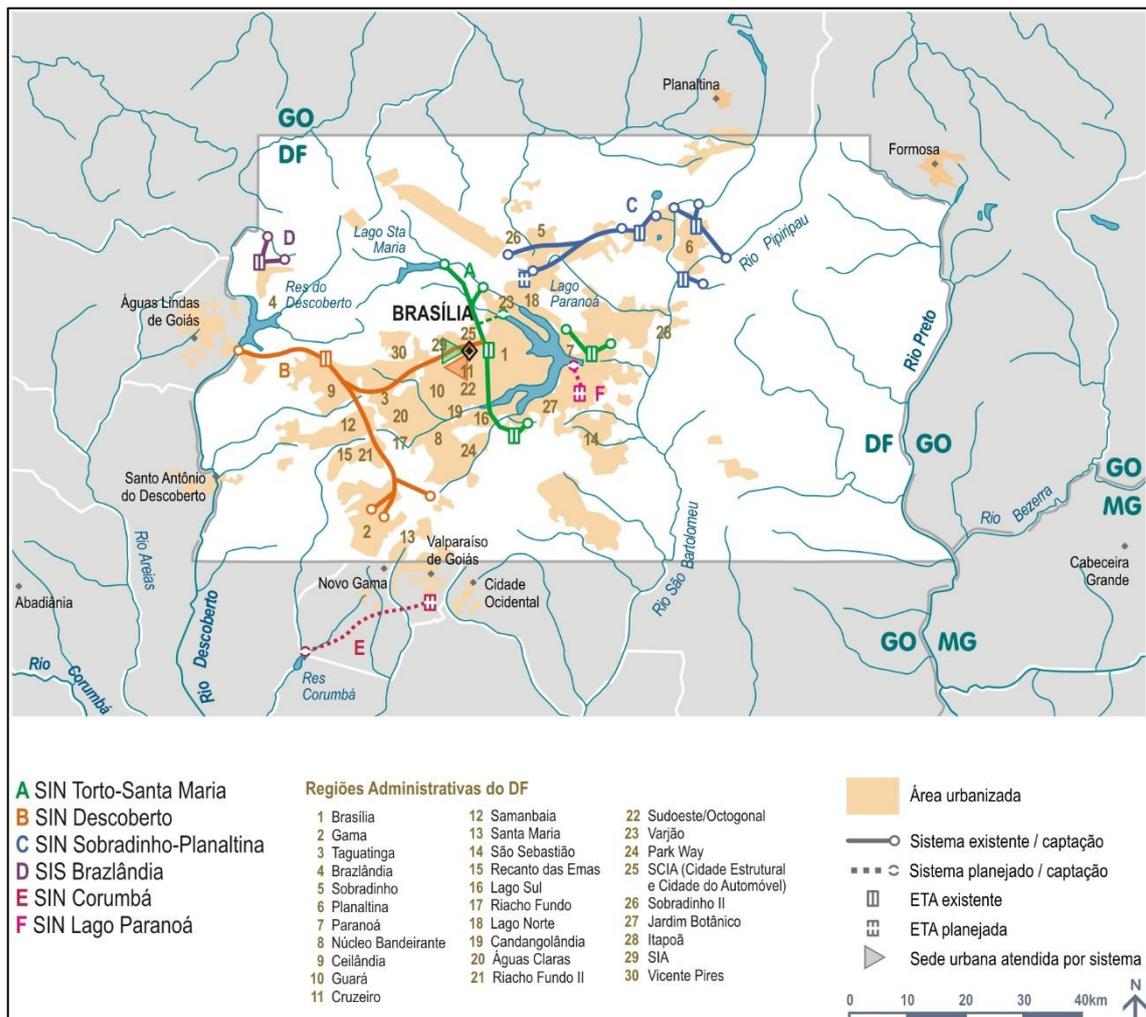


Figura 4.1 - Sistemas Produtores de Água do Distrito Federal (DF) (ANA, 2010)

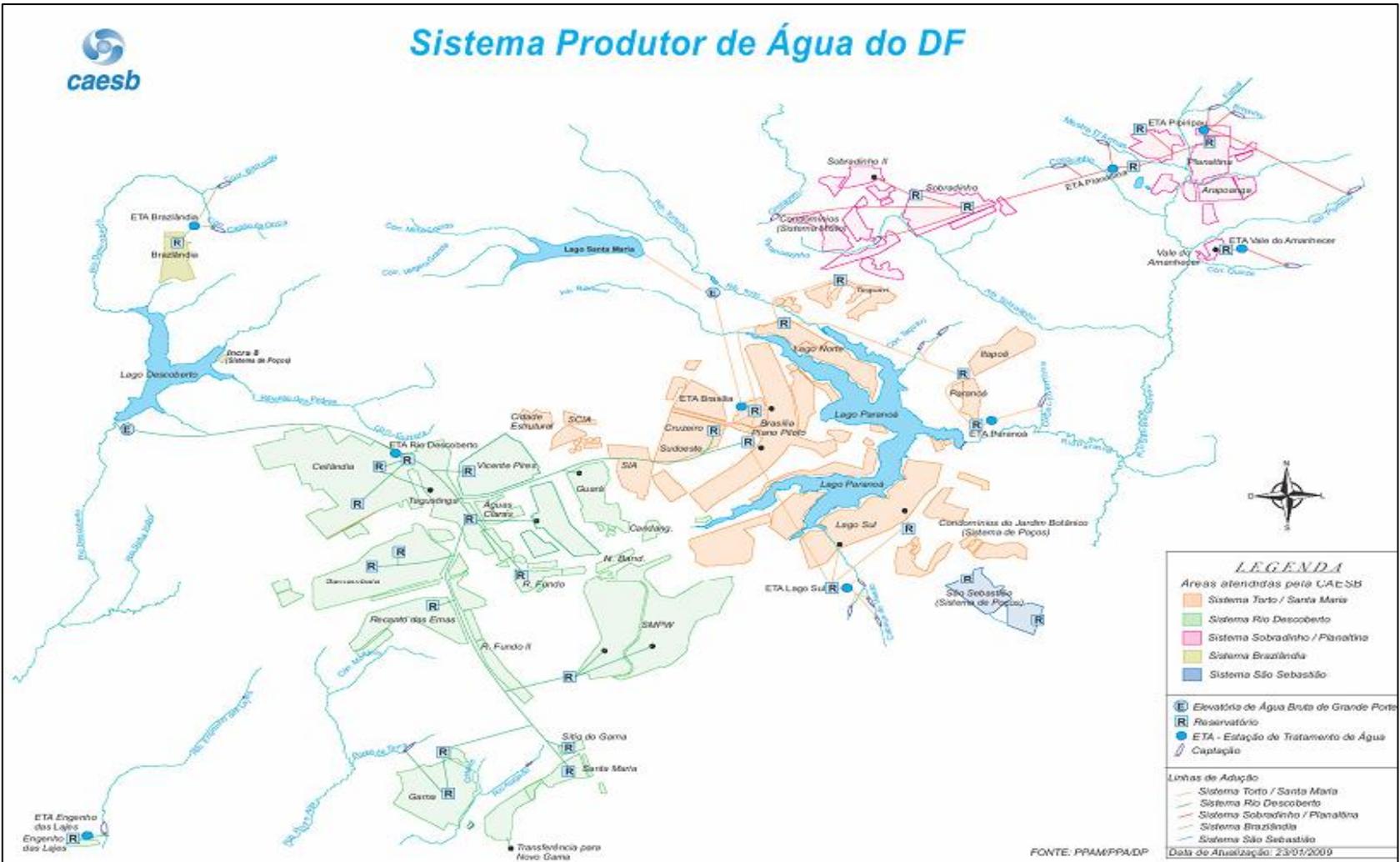


Figura 4.2 - Sistemas de produção de água do DF (CAESB, 2012)

O SPA possui capacidade de produção instantânea de 9.565,2 l/s e é composto por 468 unidades operacionais, além de adutoras de água bruta e tratada. No entanto, apenas 389 unidades estavam em operação em dezembro de 2013. Naquele ano, o SPA atendeu 2.690.404 habitantes na área urbana do Distrito Federal e produziu um volume de 245.166.512 m³, tendo valores de perdas de água inferiores a 20% (CAESB, 2014).

O segundo maior custo operacional no processo de produção de água da CAESB é a energia elétrica, menor apenas que os custos com pessoal. O consumo de energia elétrica das unidades operacionais do SPA, em 2013, chegou a 225.820.841 KWh, sendo que as unidades operacionais do sistema do Rio Descoberto foram responsáveis por 71,16% do consumo de energia elétrica, cerca de 160 milhões de KWh, totalizando 32 milhões de reais (Gebrim, 2013; CAESB, 2014). Dessa forma, esse sistema foi escolhido como estudo de caso para o presente trabalho.

O sistema do Rio Descoberto abastece as seguintes regiões administrativas: Gama, Taguatinga, Núcleo Bandeirante, SMPW (Setor de Mansões Park Way), Ceilândia, Guará, Samambaia, Santa Maria, Recanto das Emas, Riacho Fundo I e II, Candangolândia, Vicente Pires, Águas Claras, Arniqueiras, Novo Gama (Saneago) (CAESB, 2014). Além disso, o sistema Descoberto transfere parte da produção para o Sistema Torto/Santa Maria (atualmente em torno de 200 l/s) (CAESB, comunicação pessoal).

O SAA do Rio Descoberto começou sua operação em 1979, com a conclusão das obras de construção da barragem de acumulação e da elevatória de água bruta. O sistema foi concebido para uma vazão de final de plano de 6.000 l/s (Gebrim, 2013). Em 2013, o sistema Descoberto produziu uma vazão média de 4.579 l/s, fornecendo água para aproximadamente 61,5% da população atendida do Distrito Federal (CAESB, 2014).

Em decorrência de alterações ocorridas no sistema do Rio Descoberto desde sua concepção, estima-se que sua produção chegou a 5.300 l/s (vazão máxima instantânea) nos dias de maior consumo do ano, ficando muito próxima da capacidade de produção para qual o sistema foi projetado. Vazões muito próximas da capacidade de produção do sistema, podem sugerir que há poucas possibilidades para a implantação da otimização, visto que, quanto maior o nível de utilização dos equipamentos, mais tempo eles devem permanecer em

funcionamento, restringindo as condições para que mudanças sejam adotadas nas regras de operação (Gebrim, 2013).

Segundo CAESB (2014), o sistema do Rio Descoberto possui as seguintes unidades: 11 CAPs (Captações Superficiais e Subterrâneas), 2 EABs (Elevatórias de Água Bruta), 2 ETAs (Estações de Tratamento de Água), 3 UTSs (Unidades de Tratamento Simplificado), 3 UCPs (Unidades de Cloração de Poço), 7 EATs (Elevatórias de Água Tratada), 6 EREs (Elevatórias de Reservatórios Elevados), 5 EBOs (Elevatórias tipo *Booster*), 9 EPOs (Elevatórias de Poço Profundo), 12 RELs (Reservatórios Elevados), 14 RAPs (Reservatórios Apoiados) e 1 REQ (Reservatório de Equalização). Apesar das 11 CAPs existentes no sistema, apenas a captação do Rio Descoberto (maior manancial de abastecimento do DF), contribuiu com 98,8% da produção desse sistema, que corresponde a cerca de 58,2% do total da água de abastecimento produzida pela Caesb. Somente a EAB do Descoberto foi responsável por 99,89% da vazão total elevada. Além disso, apenas a ETA Descoberto tratou cerca de 98,99% da vazão total.

Sendo assim, as principais unidades operacionais do sistema Descoberto são: a captação do Rio Descoberto, a estação elevatória de água bruta do Rio Descoberto, a ETA Rio Descoberto, quinze reservatórios apoiados, seis reservatórios elevados, sete estações elevatórias de água tratada e nove estações elevatórias tipo *booster* (Gebrim, 2013). A Figura 4.3 apresenta as principais unidades do sistema, já a Tabela 4.1 mostra os códigos dessas unidades.

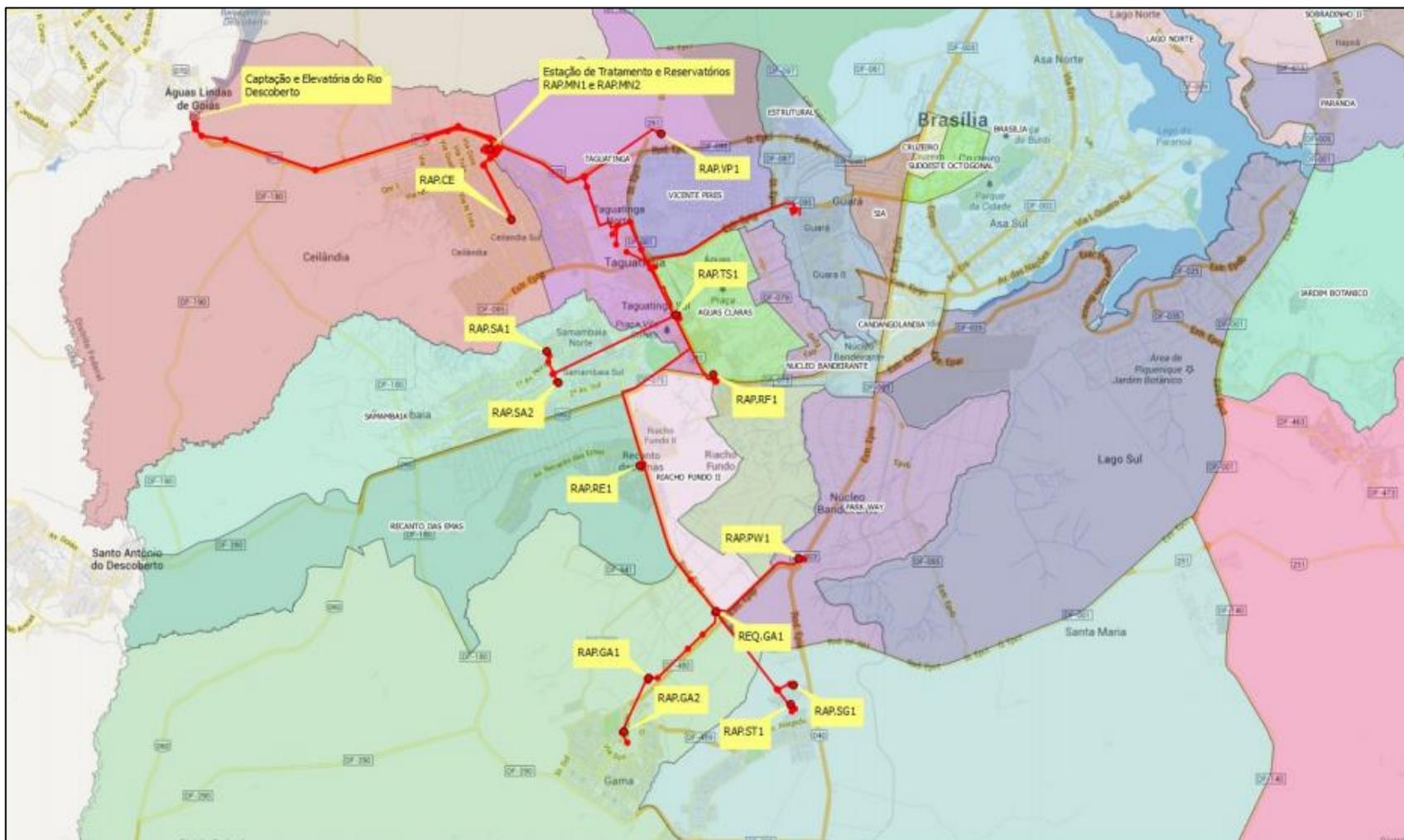


Figura 4.3 - SAA do Rio Descoberto e suas principais unidades (Gebirim, 2013)

Tabela 4.1 - Código das unidades operacionais do sistema Descoberto (Gebrim, 2013)

CÓDIGO	UNIDADES
EAB.RD1	Elevatória de Água Bruta do Rio Descoberto
ETA.RDE.001	Estação de Tratamento de Água do Rio Descoberto
RAP.MN1	Reservatório Apoiado M Norte 1
RAP.MN2	Reservatório Apoiado M Norte 2
EAT.MN1	Elevatória de Água Tratada M Norte 1
EAT.MN2	Elevatória de Água Tratada M Norte 2
EAT.MN3	Elevatória de Água Tratada M Norte 3
EAT.MN4	Elevatória de Água Tratada M Norte 4
EAT.MN5	Elevatória de Água Tratada M Norte 5
EAT.MN6	Elevatória de Água Tratada M Norte 6
RAP.CE1	Reservatório Apoiado de Ceilândia
ERE.CE1	Elevatória do Reservatório Elevado de Ceilândia
REL.CE1	Reservatório Elevado de Ceilândia
EAT.VP1	Elevatória de Água Tratada de Vicente Pires
RAP.VP1	Reservatório Apoiado de Vicente Pires
RAP.TS1	Reservatório Apoiado de Taguatinga Sul
TCV- TS1	Válvula Controladora do Reservatório de Taguatinga Sul
EAT.TS1	Elevatória de Água Tratada Taguatinga Sul 1
EAT.TS2	Elevatória de Água Tratada Taguatinga Sul 2
ERE.TS1	Elevatória do Reservatório Elevado de Taguatinga Sul
REL.TS1	Reservatório Elevado de Taguatinga Sul
VCN- RF1	Válvula Controladora do Reservatório do Riacho Fundo
RAP.RF1	Reservatório Apoiado do Riacho Fundo
ERE.RF1	Elevatória do Reservatório Elevado do Riacho Fundo
REL.RF1	Reservatório Elevado do Riacho Fundo
TCV- SG10	Válvula Controladora de Vazão de Samambaia
RAP.SA1	Reservatório Apoiado de Samambaia 1
ERE.SA1	Elevatória do Reservatório Elevado de Samambaia
REL.SA1	Reservatório Elevado de Samambaia
RAP.SA2	Reservatório Apoiado de Samambaia 2
TCV- SA2	Válvula Controladora do Reservatório de Samambaia 2
RAP.RE1	Reservatório Apoiado do Recanto das Emas
TCV- RE1	Válvula Controladora do Reservatório do Recanto das Emas
ERE.RE1	Elevatória do Reservatório Elevado do Recanto das Emas
REL.RE1	Reservatório Elevado do Recanto das Emas
REQ.GA1	Reservatório de Equalização do Gama
TCV- 11	Válvula Controladora do Reservatório de Equalização do Gama
RAP.PW1	Reservatório Apoiado do Park Way
TCV- PW1	Válvula Controladora do Reservatório do Park Way
RAP.GA1	Reservatório Apoiado do Gama 1
TCV- GA1	Válvula Controladora do Reservatório do Gama 1
RAP.GA2	Reservatório Apoiado do Gama 2
VCN- GA2	Válvula Controladora do Reservatório do Gama 2
RAP.SG1	Reservatório Apoiado do Sítio do Gama
TCV- S G1	Válvula Controladora do Reservatório do Sítio do Gama
ERE.SG1	Elevatória do Reservatório Elevado do Sítio do Gama
REL.SG1	Reservatório Elevado do Sítio do Gama
RAP.GA2	Reservatório Apoiado de Santa Maria
VCN- GA2	Válvula Controladora do Reservatório de Santa Maria
V - Transf	Válvula Controladora da transferência para área central

A captação de água para abastecimento é realizada no Rio Descoberto, junto ao município de Águas Lindas de Goiás, na fronteira do DF com o Goiás. A EE de água bruta do Rio Descoberto, possui cinco conjuntos de bombeamento, que totalizam 44.000 cv de potência instalada, sendo 27.500 cv em operação e 16.500 cv em reserva (CAESB, comunicação

peçoal). A água bruta é bombeada pela EE para a ETA do Rio Descoberto, após o tratamento, toda a água tratada é encaminhada para o Reservatório Apoiado M Norte 1 (RAP.MN1) (Gebrim, 2013). A partir desse reservatório, a água é distribuída para as outras unidades operacionais do sistema que abastecem as regiões administrativas como apresenta a Figura 4.4.

Além das unidades operacionais, o sistema Descoberto possui 5 unidades consumidoras de energia elétrica, onde estão instaladas as 10 estações elevatórias que compõem o problema de otimização. Cada unidade possui o seu próprio contrato de fornecimento, com características que podem ou não ser comuns a mais de uma unidade (Gebrim, 2013). As unidades consumidoras, suas respectivas elevatórias e as características do fornecimento de energia elétrica estão detalhadas na Tabela 4.2.

Tabela 4.2 – Tarifa de energia elétrica por elevatória analisada (Gebrim, 2013)

Unidade Consumidora	Elevatória	Tipo de Contrato	Tarifa Consumo (R\$/kWh)		Tarifa Demanda (R\$/kWh)	
			Fora de Ponta	Ponta	Fora de Ponta	Ponta
Elevatória Rio Descoberto	EAB.RD1	THS Azul ²	0,16325	0,25640	2,5022	17,7837
	EAT.MN1					
Complexo M Norte	EAT.MN2	THS Azul	0,16325	0,25640	7,0471	28,2395
	EAT.MN3					
	EAT.MN4					
	EAT.MN6					
Antiga ETA Taguatinga	EAT.MN5	THS Verde ³	0,16325	0,91231	7,0471	
Reservatório de Taguatinga Sul	EAT.TS1	THS Azul	0,16325	0,25640	7,0471	28,2395
	EAT.TS2					
Elevatória de Vicente Pires	EAT.TS3	THS Verde	0,16325	0,91231	7,0471	

² Tarifa horo-sazonal azul conforme resolução 456/2000 Aneel

³ Tarifa horo-sazonal verde conforme resolução 456/2000 Aneel

A CAESB possui um sistema SCADA que recebe informações em tempo real, permitindo que o sistema seja operado à distância por meio de um centro de controle operacional (CECOP), que monitora as informações de campo e age no acionamento de bombas e válvulas.

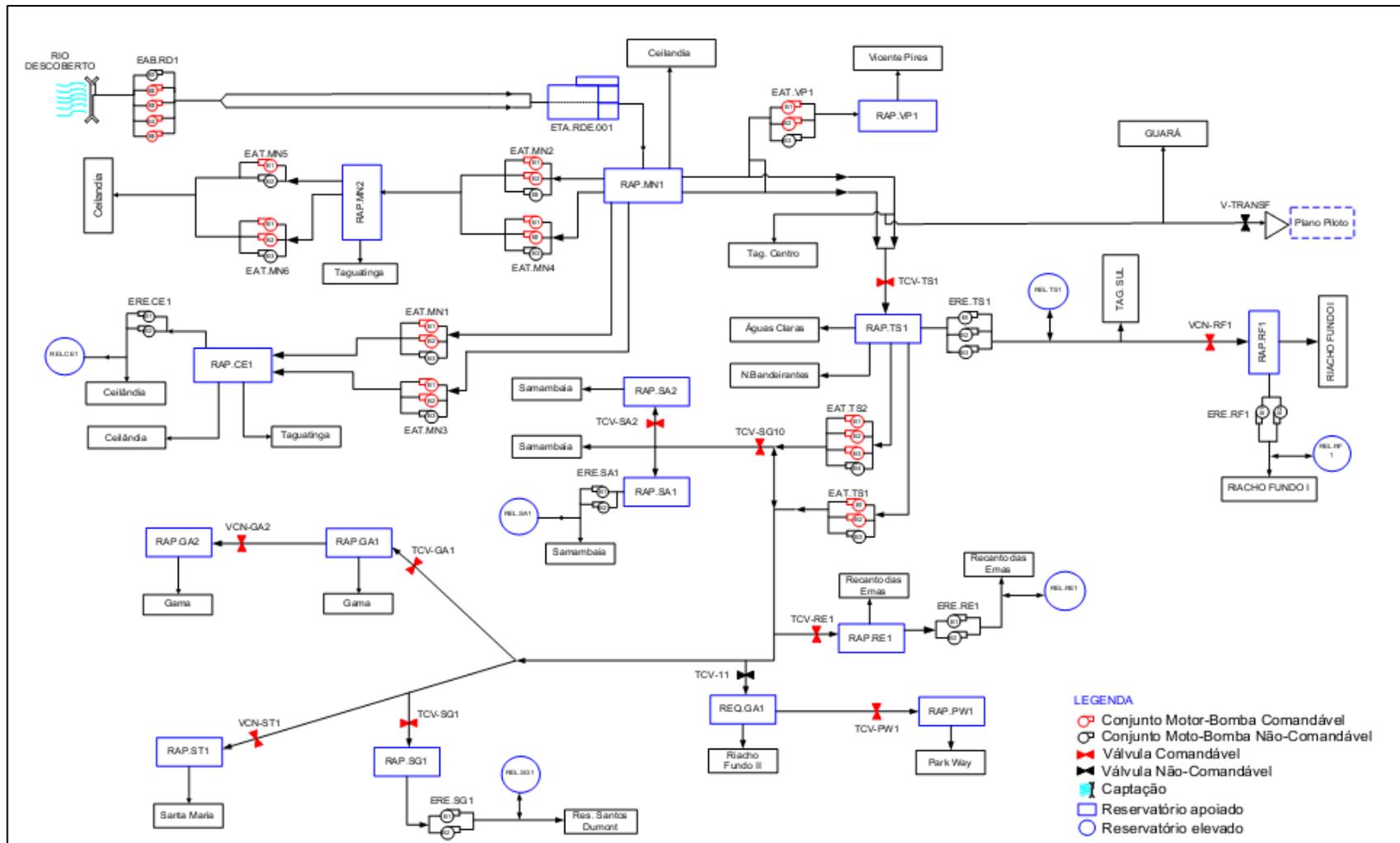


Figura 4.4 - Fluxograma de adução do sistema de abastecimento do Rio Descoberto (Gebrim, 2013)

O sistema SCADA registra os dados de forma sequencial, própria de sistemas supervisórios, e armazena de forma estruturada no banco de dados de um sistema historiador, utilizado para geração de relatórios analíticos e auditoria de dados. Atualmente, o sistema historiador vem sendo integrado ao banco de dados do sistema de informática para permitir sua compatibilização com outros sistemas de informações da Caesb (CAESB, comunicação pessoal).

O SCADA recebe informações de níveis de reservatórios e estado de bombas e válvulas, porém, em relação ao monitoramento das vazões, vários pontos do sistema são medidos, mas poucos transmitem dados ao CECOP (Gebrim, 2013). Alguns medidores transmitem os dados remotamente e outros são lidos em campo semanalmente (CAESB, comunicação pessoal).

A disponibilidade de dados semanais dificulta a modelagem hidráulica e a otimização operacional, visto que, o programa deverá simular 24 horas de operação, necessitando de dados horários de vazão. Além disso, dificulta o desenvolvimento de modelos de otimização para geração de regras operacionais em tempo real, já que praticamente inviabiliza o funcionamento de modelos de previsão de demanda (Gebrim, 2013).

Entretanto, em alguns locais, a área de monitoramento da CAESB possui registros horários de vazões, que foram utilizados para obter o padrão típico de consumo como mostra o trabalho de Gebrim (2013).

4.2 - DESCRIÇÃO E IMPLEMENTAÇÃO DO MODELO DE OTIMIZAÇÃO SELECIONADO

Para o presente estudo foi utilizado como base inicial o programa desenvolvido por Gebrim (2013). O programa emprega o Algoritmo Genético simples para a otimização, através da *GALib*, biblioteca de algoritmos genéticos do MIT, ligados ao simulador hidráulico EPANET2, por meio dos códigos disponíveis na *Toolkit Library*. O algoritmo de otimização foi desenvolvido com base no trabalho de Cunha (2009) utilizando linguagem de programação C++ com o uso do compilador *Microsoft Visual Studio Express 2012*.

As variáveis de decisão do modelo de otimização são de natureza discreta e totalizam 32, sendo 22 bombas e 10 válvulas. Nem todas as bombas e válvulas do sistema são consideradas variáveis de decisão. A Tabela 4.3 mostra as variáveis de decisão do problema de otimização.

Tabela 4.3 - Variáveis de decisão do problema (Gebrim, 2013)

Item	Unidade	Variáveis de decisão	Tipo	Potência/ Diâmetro
1		EAB.RD1_B2	Bomba	5.500 cv
2	Elevatória de Água Bruta do Rio Descoberto (EAB.RD1)	EAB.RD1_B3	Bomba	11.000 cv
3		EAB.RD1_B4	Bomba	5.500 cv
4		EAB.RD1_B5	Bomba	11.000 cv
5	Elevatória de Água Tratada M.Norte 1 (EAT.MN1)	EAT.MN1_B1	Bomba	375 cv
6		EAT.MN1_B2	Bomba	375 cv
7	Elevatória de Água Tratada M.Norte 2 (EAT.MN2)	EAT.MN2_B1	Bomba	200 cv
8		EAT.MN2_B2	Bomba	200 cv
9	Elevatória de Água Tratada M.Norte 3 (EAT.MN3)	EAT.MN3_B1	Bomba	375 cv
10		EAT.MN3_B2	Bomba	375 cv
11	Elevatória de Água Tratada M.Norte 4 (EAT.MN4)	EAT.MN4_B1	Bomba	150 cv
12		EAT.MN4_B2	Bomba	150 cv
13	Elevatória de Água Tratada M.Norte 5 (EAT.MN5)	EAT.MN5_B1	Bomba	100 cv
14		EAT.MN6_B1	Bomba	100 cv
15	Elevatória de Água Tratada M.Norte 6 (EAT.MN6)	EAT.MN6_B2	Bomba	100 cv
16		EAT.VP1_B1	Bomba	150 cv
17	Elevatória de Água Tratada de Vicente Pires (EAT.VP1)	EAT.VP1_B2	Bomba	150 cv
18		EAT.TS1_B1	Bomba	550 cv
19	Elevatória de Água Tratada de Taguatinga Sul 1 (EAT.TS1)	EAT.TS1_B2	Bomba	550 cv
20		EAT.TS2_B1	Bomba	800 cv
21	Elevatória de Água Tratada de Taguatinga Sul 2 (EAT.TS2)	EAT.TS2_B2	Bomba	800 cv
22		EAT.TS2_B3	Bomba	800 cv
23	Válvula de Entrada do Reservatório de Taguatinga Sul	TCV-TS1	Válvula	2x 800 mm
24	Válvula Controladora de Vazão para Samambaia	TCV-SG10	Válvula	1000 mm
25	Válvula de Entrada do Reservatório Samambaia 2	TCV-SA2	Válvula	400 mm
26	Válvula de Entrada do Reservatório Riacho Fundo	VCN-RF1	Válvula	250 mm
27	Válvula de Entrada do Reservatório Recanto das Emas	TCV-RE1	Válvula	400 mm
28	Válvula de Entrada do Reservatório Park Way	TCV-PW1	Válvula	250 mm
29	Válvula de Entrada do Reservatório Sítio do Gama	TCV-SG1	Válvula	150 mm
30	Válvula de Entrada do Reservatório Santa Maria	VCN-ST1	Válvula	350 mm
31	Válvula de Entrada do Reservatório Gama 1	VCN-GA1	Válvula	250 mm
32	Válvula de Entrada do Reservatório Gama 2	VCN-GA2	Válvula	250 mm

A codificação das soluções foi feita por meio de uma representação vetorial, de uma dimensão, composta por valores binários, onde todos os equipamentos representados no problema podem assumir somente as posições de ligado/desligado (no caso de bombas), ou aberto/fechado (no caso de válvulas).

Para codificar as soluções do problema utilizou-se um vetor de 768 posições, que é produto das 32 variáveis de decisão e do tempo de simulação de 24h, onde cada posição representa a condição de uma variável em uma determinada hora, podendo assumir o valor de 0 ou 1. Vale mencionar que bombas de rotação variável não foram consideradas como variáveis de decisão, no entanto, as válvulas TCV-TS1 e TCV-SG10 (variáveis de decisão), podem assumir posições intermediárias no sistema real visando a regulação de vazão, foram consideradas somente nas posições totalmente aberta ou fechada como uma forma de simplificação. A Figura 4.5 apresenta um exemplo de codificação binária para um problema contendo três bombas e duas válvulas em um período de otimização T .

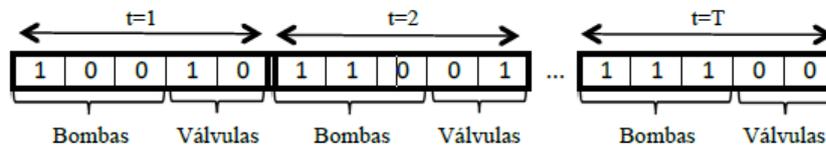


Figura 4.5 – Vetor solução com três bombas, duas válvulas e período de otimização T (Gebrim, 2013)

O objetivo do modelo de otimização foi minimizar os custos de energia elétrica decorrente do bombeamento do SAA do Descoberto, composto pelo custo de consumo de energia elétrica de todas as bombas mostrado na Equação 4.1 e o custo de demanda de energia elétrica de todas as unidades consumidoras para as tarifas horo-sazonais azul, Equação 4.2, e verde, Equação 4.3, separando os horários de ponta (entre 18 e 21 horas) e fora de ponta.

$$CC = Consumo_{ponta} \times TC_{ponta} + Consumo_{fponta} \times TC_{fponta} \quad (4.1)$$

$$CD_{azul} = Demanda_{ponta} \times TD_{ponta} + Demanda_{fponta} \times TD_{fponta} \quad (4.2)$$

$$CD_{verde} = Demanda \times TD_{verde} \quad (4.3)$$

em que: CC = custo de consumo de energia (tarifas verde e azul); CD_{azul} = custo de demanda de energia na tarifa horo-sazonal azul; CD_{verde} = custo de demanda de energia na tarifa horo-sazonal verde; $Consumo_{ponta}$ = energia consumida no horário de ponta; $Consumo_{fponta}$ = energia consumida no horário fora de ponta; $Demanda_{ponta}$ = potência máxima demandada no horário de ponta; $Demanda_{fponta}$ = potência máxima demandada no horário fora de ponta; $Demanda$ = potência máxima demandada em todo o período de medição; TC_{ponta} = tarifa de

consumo no horário de ponta; TC_{fponta} = tarifa de consumo no horário fora de ponta; TD_{verde} = tarifa de demanda da modalidade tarifária verde; TD_{ponta} = tarifa de demanda no horário de ponta na modalidade tarifária azul; TD_{fponta} = tarifa de demanda no horário fora de ponta na modalidade tarifária azul.

O custo total de energia elétrica utilizada no bombeamento representa a função objetivo e foi calculado pela Equação 4.4, em que o custo de demanda é dividido por 30 para representar o custo diário, pois o $CD_{(u)}$ corresponde ao valor mensal. Além disso, para o cálculo da demanda deve ser encontrado o maior valor de potência utilizada na unidade consumidora, em cada período tarifário do dia (horário de ponta e fora de ponta).

$$FO(X; Y) = Cb = \sum_{b=1}^B \sum_{t=1}^T CC_{(b,t)} + \sum_{u=1}^U \left(\frac{CD_{(u)}}{30} \right) \quad (4.4)$$

$$CD_{(u)} = \sum_{d=1}^D Demanda_{(u,d)} \times TD_{(u,d)} \quad (4.5)$$

em que: $FO(X; Y)$ = função objetivo; X = matriz das bombas que são variáveis do problema; Y = matriz das válvulas que são variáveis do problema; Cb = custo total de energia elétrica gasto com bombeamento; $CC_{(b, t)}$ = custo de consumo de energia elétrica da bomba b no período $t=1, \dots, T$; u = índice da unidade consumidora em análise; U = total de unidades consumidoras (elevatórias) do problema; $CD_{(u)}$ = custo total de demanda de energia elétrica da unidade consumidora u no período $t=1, \dots, T$; b = índice que representa cada bomba do problema; B = número total de bombas do problema; t = instante da simulação em análise; T = período total de análise; $Demanda_{(u, d)}$ = demanda de energia elétrica da unidade consumidora u no período tarifário d ; $TD_{(u, d)}$ = tarifa de demanda de energia elétrica da unidade consumidora u no período tarifário d (R\$/W); d = índice que indica o período tarifário do dia (ponta ou fora de ponta); D = número de períodos tarifário do dia.

Para o problema, foram consideradas 2 restrições implícitas (resolvidas pelo simulador hidráulico): conservação de massa em cada nó e conservação de energia no sistema; e 5 restrições explícitas: pressão positiva nos nós de demanda, nível de água nos reservatórios acima do limite mínimo operacional, nível de água nos reservatórios abaixo do limite de

extravasamento, número de acionamentos de equipamentos menor que o limite máximo considerado tolerável e variável conforme o porte dos equipamentos, e *diferença entre os níveis iniciais e finais dos reservatórios* menor que um limite admissível para a continuidade do sistema.

Para cada uma dessas restrições foi atribuída uma penalidade que foi aplicada à função objetivo, conforme o método das Penalidades. No cálculo das penalidades foram considerados coeficientes de penalidades (pesos), para atribuir diferentes graus de penalização para cada restrição avaliada.

As penalidades são: (1) não atendimento das demandas, (2) nível dos reservatórios abaixo do limite mínimo, (3) nível dos reservatórios acima do limite máximo (extravasamento), (4) excesso de acionamentos, (5) *diferença entre os níveis inicial e final nos reservatórios acima dos limites*.

A penalidade 1 visa a garantir a continuidade do fornecimento de água, eliminando as soluções que provoquem desabastecimento. Para o cálculo da penalidade, o algoritmo de otimização verifica a cada passo de tempo a pressão nos nós com demanda positiva e penaliza as pressões negativas ou iguais a zero. A Equação 4.6 calcula a P1.

$$P1 = \sum_{t=1}^T \sum_{i=1}^G (D_{(i,t)} \times \lambda_1 \times C1_{(i,t)}) \quad (4.6)$$

$$C1_{(i,t)} = 0 \text{ se } P_{(i,t)} > 0 \quad (4.7)$$

$$C1_{(i,t)} = 1 \text{ se } P_{(i,t)} \leq 0 \quad (4.8)$$

em que: t = instante da simulação em análise; T = período total de análise; i = índice do nó de consumo em análise; G = número total de nós de consumo analisados; $D_{(i,t)}$ = Demanda do nó i no instante t; $\lambda_1 = 100$ (coeficiente aplicado à Penalidade 1); $C1_{(i,t)}$ = condição para aplicação da Penalidade 1 em cada nó i a cada passo de tempo t; $P_{(i,t)}$ = pressão do nó i no instante t.

Segundo Gebrim (2013) a opção de utilizar o valor da demanda instantânea do nó de consumo pode ser útil nos casos em que se admita a ocorrência de interrupção do abastecimento, pois é possível diferenciar o valor das ocorrências de interrupção em horários e locais de alto e baixo consumo.

A penalidade 2 tem o objetivo de evitar que as soluções gerem níveis de reservatórios menores do que os níveis mínimos operacionais para garantir a continuidade do abastecimento. De acordo com as condições de operação normalmente observadas, o limite mínimo considerado para a aplicação dessa penalidade foi arbitrado em 10% do nível máximo de cada reservatório.

$$P2 = \sum_{t=1}^T \sum_{j=1}^R (\lambda_2 \times C2_{(j,t)}) \quad (4.9)$$

$$C2_{(j,t)} = 0 \text{ se } N_{(j,t)} \geq 0,1 \times N_{j \text{ max}} \quad (4.10)$$

$$C2_{(j,t)} = 1 \text{ se } N_{(j,t)} < 0,1 \times N_{j \text{ max}} \quad (4.11)$$

em que: t = instante da simulação em análise; T = período total de análise; j = índice do reservatório em análise; R = número total de reservatórios analisados; $\lambda_2 = 10.000$ (coeficiente aplicado à Penalidade 2); $C2_{(j,t)}$ = condição para aplicação da Penalidade 2 em cada reservatório j a cada passo de tempo t ; $N_{(j,t)}$ = nível do reservatório j no instante t ; $N_{j \text{ max}}$ = nível máximo do reservatório j .

A penalidade 3 busca eliminar as soluções que levem os reservatórios a atingirem níveis de extravasamento. O limite máximo para a aplicação da penalidade foi de 98% do nível máximo do reservatório, esse limite foi estabelecido a partir dos registros operacionais da CAESB.

$$P3 = \sum_{t=1}^T \sum_{j=1}^R (\lambda_3 \times C3_{(j,t)}) \quad (4.12)$$

$$C3_{(j,t)} = 0 \text{ se } N_{(j,t)} < 0,98 \times N_{j \text{ max}} \quad (4.13)$$

$$C3_{(j,t)} = 1 \text{ se } N_{(j,t)} \geq 0,98 \times N_{j \text{ max}} \quad (4.14)$$

em que: $\lambda_3 = 10.000$ (coeficiente aplicado à Penalidade 3); $C3_{(j,t)}$ = condição para aplicação da Penalidade 3 em cada reservatório j a cada passo de tempo t .

A penalidade 4 tem como objetivo a redução do número de acionamentos de bombas e válvulas que são variáveis de decisão, já que o excesso de acionamentos afeta a durabilidade, comprometendo o bom funcionamento e a vida útil desses equipamentos.

A penalidade 4 possui dois componentes, o primeiro componente é aplicado apenas às bombas e penaliza todo e qualquer desligamento, atribuindo à penalidade o valor de 1,5 vezes o gasto com consumo de energia daquela bomba na hora anterior ao desligamento, assim, o desligamento gera maior impacto na função aptidão que a conservação do funcionamento da bomba por uma hora. Em termos de custo de energia elétrica, o benefício de desligar uma bomba por apenas uma hora é inferior ao custo do próprio desligamento, o que, de maneira indireta, força o modelo de otimização a evitar soluções em que as bombas são desligadas por uma hora e religadas em seguida.

O segundo componente foi aplicado tanto para bombas quanto para válvulas e visa eliminar as soluções em que os equipamentos são acionados mais vezes do que um determinado número. Para isso, os equipamentos foram divididos em quatro grupos e para cada grupo foi estabelecido um limite máximo de acionamentos (Tabela 4.4).

Tabela 4.4 - Divisão dos equipamentos para aplicação da Penalidade 4

Grupos	Equipamentos	Limite de acionamentos (NA_b lim)	Coefficiente de penalidade
1	Bombas com potência de 11.000 cv	1	$\lambda_{4-g_1} = 10.000$
2	Bombas com potência de 5.500 cv	2	$\lambda_{4-g_2} = 10.000$
3	Bombas com potência entre 5.500 cv e 200 cv	3	$\lambda_{4-g_3} = 2.000$
4	Bombas com potência igual ou menor a 200 cv e válvulas	4	$\lambda_{4-g_4} = 1.000$

Além de estabelecer valores diferentes para os limites de acionamentos em cada grupo, também foram definidos coeficientes de penalidade diferentes em caso de violação desse limite, onde o maior coeficiente foi aplicado ao grupo 1 e o menor aplicado ao grupo 4. Essa diferenciação teve o objetivo de tornar o problema menos restritivo.

O acionamento é configurado quando o estado na hora analisada for igual a 0 e o estado na hora anterior for igual a 1. Para bombas o número de acionamentos é calculado pela Equação 4.15 e válvulas pela Equação 4.18.

$$NA_b = \sum_{t=1}^T A_{(b,t)} \quad (4.15)$$

$$C4_b = 1 \text{ se } NA_b > NA_b \text{ lim} \quad (4.16)$$

$$C4_b = 0 \text{ se } NA_b \leq NA_b \text{ lim} \quad (4.17)$$

$$NA_v = \sum_{t=1}^T A_{(v,t)} \quad (4.18)$$

$$C4_v = 1 \text{ se } NA_b > NA_b \text{ lim} \quad (4.19)$$

$$C4_v = 0 \text{ se } NA_v \leq NA_v \text{ lim} \quad (4.20)$$

A penalidade 4 foi calculada pela Equação 4.21.

$$P4 = \sum_{b=1}^B \left[\left(\sum_{t=1}^T 1,5 \times CC_{(b,t-1)} \times Ac_{(b,t)} \right) + (\lambda_4 - g_n \times C4_b) \right] + \sum_{v=1}^V (\lambda_4 - g_n \times C4_v) \quad (4.21)$$

em que: b = índice que representa cada bomba do problema; B = número total de bombas do problema; v = índice que representa cada válvula do problema; B = número total de válvulas do problema; t = instante da simulação em análise; T = período total de análise; $CC_{(b, t)}$ = custo de consumo de energia elétrica da bomba b no período t-1 até t; $\lambda_4 - g_n$ = coeficiente aplicado à Penalidade 4 referente ao grupo de equipamento n; n = grupo a qual o equipamento analisado (bomba/válvula) pertence; $C4_b$ = condição para aplicação do coeficiente de penalidade $\lambda_4 - g_n$ em cada bomba b; $C4_v$ = condição para aplicação do

coeficiente de penalidade λ_4-g_n em cada válvula v ; $A_{(b, t)}$ = verificação da ocorrência de acionamento da bomba b no instante t ; $A_{(v, t)}$ = verificação da ocorrência de acionamento da válvula v no instante t ; NA_b = quantidade total de acionamentos da bomba b ; $NA_{b \text{ lim}}$ = quantidade limite de acionamentos tolerados para a bomba b ; NA_v = quantidade total de acionamentos da válvula v ; $NA_{v \text{ lim}}$ = quantidade limite de acionamentos tolerados para a válvula v .

A penalidade 5 visa à preservação das condições hidráulicas iniciais no sistema ao final do ciclo diário e possibilita a viabilidade operacional dos períodos subsequentes ao simulado. A penalidade pune soluções em que a *diferença entre o nível inicial e final* é maior que o limite estabelecido de 5% do nível máximo do reservatório. O limite foi definido para evitar que o problema fosse excessivamente restritivo.

$$P5 = \sum_{t=1}^T |N_{j \text{ inicial}} - N_{j \text{ final}}| \times \lambda_5 \times C5_j \quad (4.22)$$

$$C5_j = 0 \text{ se } |N_{j \text{ inicial}} - N_{j \text{ final}}| \leq 0,05 \times N_{j \text{ max}} \quad (4.23)$$

$$C5_j = 1 \text{ se } |N_{j \text{ inicial}} - N_{j \text{ final}}| > 0,05 \times N_{j \text{ max}} \quad (4.24)$$

em que: j = índice do reservatório em análise; R = número total de reservatórios analisados; $N_{j \text{ final}}$ = nível do reservatório j no instante final de simulação; $N_{j \text{ inicial}}$ = nível do reservatório j no início da simulação; $N_{j \text{ max}}$ = nível máximo do reservatório j ; $\lambda_5 = 10.000$ (coeficiente aplicado à Penalidade 5); $C5_j$ = condição para aplicação da Penalidade 5 em cada reservatório j .

A adição das penalidades transformou a Equação 4.4 na Equação 4.25, denominada função aptidão, convertendo o problema inicialmente irrestrito em um problema restrito.

$$\text{apt } F(X; Y; Rr) = FO(X; Y) + P1 + P2 + P3 + P4 + P5 \quad (4.25)$$

em que: FO = função objetivo; X = matriz das bombas que são variáveis do problema; Y = matriz das válvulas que são variáveis do problema; Rr = restrições do problema; $P1$ = não atendimento das demandas; $P2$ = nível dos reservatórios abaixo do limite mínimo; $P3$ = nível

dos reservatórios acima do limite máximo (extravasamento); P4 = excesso de acionamentos; P5 = *diferença entre os níveis inicial e final nos reservatórios acima dos limites.*

O modelo hidráulico utilizado por Gebrim (2013) foi elaborado pela área de planejamento operacional e controle da Caesb, com algumas atualizações e adequações feitas pelo autor. O modelo usa valores de consumo médios mensais, e assim as regras operacionais utilizadas pelo programa correspondem a uma condição média de funcionamento do sistema.

O modelo foi calibrado utilizando uma simulação de tempo estendido (24h), aplicando-se nessa simulação uma regra operacional conhecida que já tinha sido utilizada no sistema real. A calibração foi realizada por meio do processo iterativo, variando-se as perdas de cargas, curvas de bombas e padrões de consumo, até que os níveis do reservatório do modelo estivessem próximos dos níveis reais do sistema. O autor não validou o modelo, pois este já foi utilizado em outras situações. O modelo simula a operação de um dia médio de funcionamento em um período de 24 horas de operação do sistema, de 0h a 24h. No presente trabalho utilizou-se o modelo empregado por Gebrim (2013), já que o sistema não sofreu alterações significativas.

O programa de Gebrim (2013) selecionado demandou grande tempo computacional para realizar simulações. Assim foram introduzidas simplificações no modelo e utilização de outros algoritmos de otimização além de tratamento das penalidades e soluções iniciais utilizadas com o objetivo de melhorar o desempenho do modelo de otimização.

No presente estudo, o código de Gebrim (2013) anteriormente implementado em C++ foi implementado na plataforma MATLAB versão 2013a, de forma a possibilitar facilidade de integração com diversas técnicas de otimização e manipulação dos resultados. O desempenho desse código utilizando o AG foi comparado com as alterações introduzindo outros algoritmos de otimização.

4.3 - IMPLEMENTAÇÃO E SIMPLIFICAÇÃO DO MODELO HIDRÁULICO

O simulador hidráulico empregado por Gebrim (2013) foi o EPANET2. O pacote computacional do EPANET2 é distribuído gratuitamente nas formas executável e código fonte. A forma executável é utilizada em estudos que envolvem somente simulações

hidráulicas. Já a forma de código fonte é recomendável para trabalhos que envolvem a vinculação de outros modelos com o EPANET2.

A USEPA disponibiliza também uma biblioteca de vínculo dinâmico (DLL), que possui funções que possibilitam aos programadores o desenvolvimento de aplicações específicas. Essas funções podem ser integradas às aplicações *Windows* nas linguagens de programação C/C++, Pascal (Delphi), *Visual Basic* ou outras linguagens que permitam chamar as funções por meio de DLLs. Essa biblioteca também admite abrir e fechar um arquivo que representa a rede (.inp), ler e alterar os diversos parâmetros de projeto e operação, simular períodos extensivos acessando resultados a cada passo de tempo e escrever os resultados das simulações nos arquivos de saída (.rtp) (Cheung, 2004).

De acordo com Lacerda *et al.* (2015), o simulador hidráulico EPANET2 possui a característica de simular operações dinâmicas e permitir a programação via regras. Além disso, possibilita fácil integração com diversas linguagens de programação, através de sua biblioteca de funções.

Empregou-se a biblioteca de funções chamada de *EPANET-Matlab-Toolkit* desenvolvida por Eliades e Kyriakou (2016) na plataforma MATLAB (<https://github.com/OpenWaterAnalytics/EPANET-Matlab-Toolkit>) que utiliza a versão não oficial EPANET 2.1, porém no presente estudo adotou-se a versão oficial EPANET2. A biblioteca exige o uso do compilador *Microsoft Windows SDK* para *Windows 7*. Algumas funções utilizadas no algoritmo são mostradas na Tabela 4.5.

Tabela 4.5 – Funções da biblioteca *EPANET-Matlab-Toolkit* (Eliades e Kyriakou, 2016)

FUNÇÕES	DESCRIÇÃO
<i>epanet</i>	Carrega o arquivo de entrada de extensão <i>.inp</i> e abre a biblioteca
<i>openHydraulicAnalysis</i>	Abre a análise hidráulica
<i>initializeHydraulicAnalysis</i>	Inicializa os níveis dos reservatórios, estados, o tempo de simulação e outras configurações antes de executar a análise hidráulica
<i>runHydraulicAnalysis</i>	Executa a análise hidráulica
<i>nextHydraulicAnalysisStep</i>	Determina o período de tempo até o próximo evento hidráulico ocorrer em uma simulação dinâmica
<i>closeHydraulicAnalysis</i>	Fecha a análise hidráulica, liberando a memória alocada

Tabela 4.5 - Continuação

FUNÇÕES	DESCRIÇÃO
<i>setLinkStatus</i>	Define os valores dos estados
<i>setNodeTankInitialLevel</i>	Define os valores dos níveis iniciais dos reservatórios
<i>getLinkStatus</i>	Recupera os valores dos estados (0 = fechado, 1 = aberto)
<i>getLinkPumpEnergy</i>	Recupera os valores das energias das bombas em kw
<i>getNodePressure</i>	Recupera os valores das pressões
<i>getLinkFlows</i>	Recupera os valores das vazões
<i>getNodeActualDemand</i>	Recupera os valores das demandas

O esquema hidráulico conceitual utilizado no estudo é o modelo dinâmico proposto por Gebrim (2013), com algumas simplificações comentadas a seguir. O esquema apresenta 273 nós, sendo: 23 reservatórios de nível variável, 1 reservatório de nível fixo e 35 nós de consumo. Possui também 331 trechos, sendo: 267 trechos de tubulação, 47 bombas e 17 válvulas. O modelo hidráulico do SAA do Rio Descoberto na interface gráfica do EPANET2 é ilustrado na Figura 4.6.

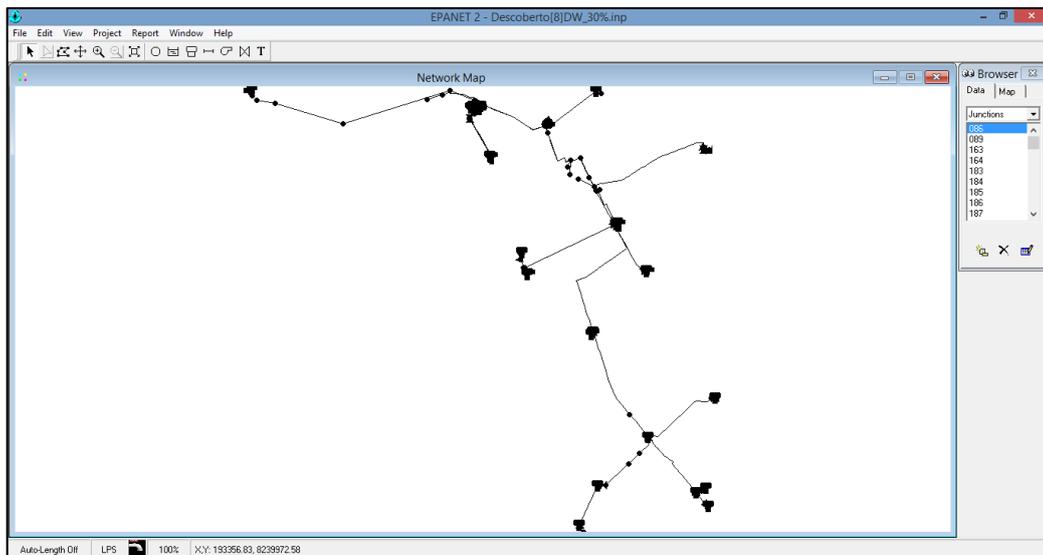


Figura 4.6 - SAA do Rio Descoberto no ambiente EPANET2 (Gebrim, 2013, adaptado)

Nota-se que o modelo utilizado por Gebrim (2013) possui muitos nós e trechos, que contribuem para o aumento do tempo de processamento do algoritmo de otimização, então, a simplificação (esqueletonização) do modelo hidráulico foi realizada, pela criação de tubos de comprimentos equivalentes, a partir da simplificação de trechos em série e paralelo. Vale mencionar que nenhuma unidade operacional foi simplificada, nem nós de consumo. A Figura 4.7 apresenta um exemplo de simplificação de tubos de mesmo diâmetro e material, dispostos em série.

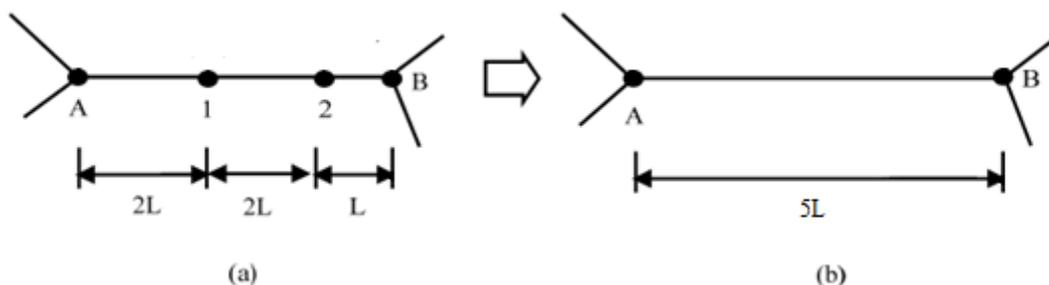


Figura 4.7 – Exemplo de esqueletonização de um trecho em série (Odan, 2013, modificado)

Para a simplificação dos trechos em série e paralelo foram utilizadas as Equações (4.26) e (4.27), respectivamente, mostradas em Porto (2006). Onde foram mantidos os valores do diâmetro (D) e fator de atrito (f) empregados no modelo hidráulico de Gebrim (2013) e encontrados os valores do comprimento equivalente (L). Não foi realizada a calibração do modelo hidráulico simplificado, pois espera-se que ele represente fielmente o modelo completo.

$$\frac{f \cdot L}{D^5} = \sum_{i=1}^n \frac{f_i \cdot L_i}{D_i^5} \quad (4.26)$$

em que:

f = fator de atrito do tubo simplificado; L = comprimento equivalente do tubo simplificado (m); D = diâmetro do tubo simplificado (m); n = quantidade de tubos; i = número do tubo; f_i = fator de atrito do tubo de número i ; L_i = comprimento do tubo de número i (m); D_i = diâmetro do tubo de número i (m).

$$\frac{D^{2,5}}{f^{0,5} \cdot L^{0,5}} = \frac{D_1^{2,5}}{f_1^{0,5} \cdot L_1^{0,5}} + \frac{D_2^{2,5}}{f_2^{0,5} \cdot L_2^{0,5}} \quad (4.27)$$

em que:

D = diâmetro do tubo simplificado (m); f = fator de atrito do tubo simplificado; L = comprimento equivalente do tubo simplificado (m); D_1 = diâmetro do tubo 1 (m); f_1 = fator de atrito do tubo 1; L_1 = comprimento do tubo 1 (m); D_2 = diâmetro do tubo 2 (m); f_2 = fator de atrito do tubo 2; L_2 = comprimento do tubo 2 (m).

Para avaliar a qualidade da simplificação utilizou-se o erro médio absoluto (MAE) e como parâmetro de comparação adotou-se os níveis dos reservatórios, com intervalo de uma hora, durante um dia. Portanto, a simplificação será considerada exitosa se o modelo hidráulico simplificado, apresentar níveis de reservatórios próximos aos níveis do modelo completo.

4.4 - UTILIZAÇÃO DE ALGORITMOS DE OTIMIZAÇÃO

Foram testados alguns algoritmos para a otimização da operação para a minimização dos custos operacionais de energia elétrica, provenientes do bombeamento do SAA do Rio Descoberto, a fim de melhorar o desempenho do processo de otimização. Os algoritmos testados foram o AG na linguagem C++, o DDS e o DE na plataforma MATLAB. Os desempenhos deles foram comparados com base na metodologia proposta por Dandy *et al.* (2010).

4.4.1 - Algoritmo genético

O AG utilizado foi o mesmo empregado por Gebrim (2013) na linguagem C++, com o uso do compilador *Microsoft Visual Studio Express 2012*. O funcionamento do AG é descrito na Figura 4.8.

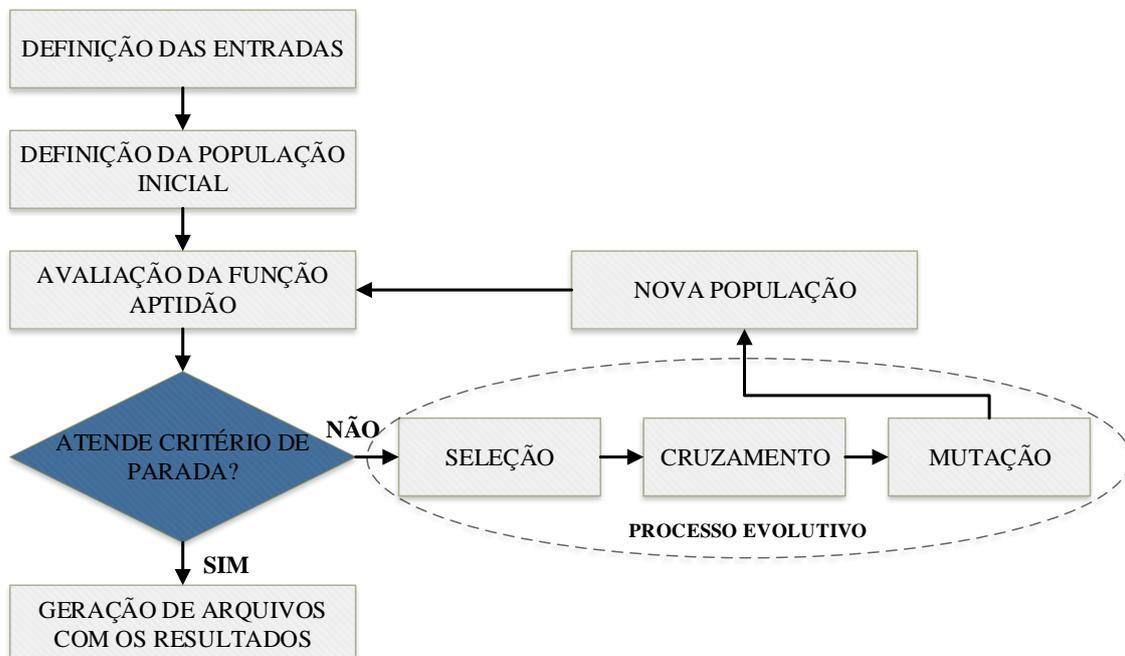


Figura 4.8 - Funcionamento do AG

A etapa 1 do AG, consiste na definição das entradas, ou seja, os dados gerais que serão utilizados no processo de otimização. Para isso, utilizou-se duas rotinas diferentes. A primeira abre o arquivo de entrada do EPANET2, contabiliza os números de nós, os trechos e os reservatórios.

A segunda rotina abre o arquivo texto (extensão *.txt*), que contém informações como: os parâmetros do algoritmo genético (tipo de seleção, tipo de cruzamento, probabilidade de cruzamento, tipo de mutação, probabilidade de mutação, tamanho da população, número de gerações, e semente aleatória); os coeficientes de penalidades; o número e identificação das variáveis do problema (bombas e válvulas); tempo de simulação (número de horas); e níveis iniciais dos reservatórios.

Na etapa 2, ocorre a definição da população inicial (conjunto de vetores de soluções iniciais) aleatória com aptidão de 1.980.000, sendo a mesma solução inicial (regra operacional) empregada em todas as simulações dos algoritmos de otimização. A definição das soluções iniciais iguais se deu pela necessidade de iniciar as simulações do mesmo ponto para comparar as alternativas, já que a solução inicial é um fator essencial no processo de otimização e afeta os resultados das simulações.

Na terceira etapa do AG, acontece a avaliação da função aptidão, que é calculada na primeira avaliação com o vetor de soluções iniciais. Nas avaliações seguintes o algoritmo recebe as soluções do AG e emprega o vetor de soluções (estados de bombas e válvulas) e o vetor com os níveis iniciais dos reservatórios no simulador hidráulico EPANET2, através das funções da *toolkit*, *ENsetlinkvalue* e *ENsetnodevalue*, respectivamente. Com esses vetores o EPANET2 realiza a simulação hidráulica.

A execução da simulação hidráulica pelo modelo otimizador é realizada pelo conjunto de funções, *ENopenH* - *ENinitH* - *ENrunH* - *ENnextH* - *EncloseH*, que integram a *toolkit* de programação do EPANET2. Essa forma de execução é indicada para os casos em que se pretende acessar dados parciais ao longo da simulação dinâmica e é comumente utilizada em estudos de otimização.

O acesso às variáveis no decorrer da simulação é feito pelas funções *ENgetlinkvalue* e *ENgetnodevalue*. A simulação hidráulica tem como produto as variáveis pressão, vazão,

energia, estado e demanda que são utilizadas para o cálculo da função aptidão que ocorre no modelo otimizador (Figura 4.9).

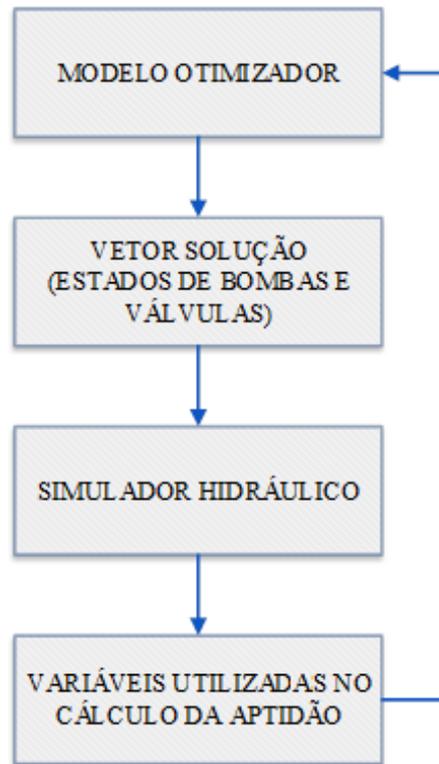


Figura 4.9 – Integração do modelo otimizador ao modelo de simulação (simulador hidráulico)

Na quarta etapa é avaliado o critério de parada, que nesse caso é o número de gerações. Se o critério de parada for atendido o algoritmo seleciona a melhor solução encontrada e gera arquivos *.txt* com informações detalhadas de consumo de energia, níveis de reservatórios, acionamentos, violações das restrições e custos.

No entanto, se o critério não for atendido, a população corrente passa pelo processo evolutivo do AG (etapa 5), que consiste na seleção, cruzamento e mutação para gerar uma nova população, que será utilizada para a avaliação da função aptidão.

Utilizou-se uma versão do AG que não gera sobreposição (*overlapping*) de populações, classificada na *GAlib* como “Algoritmo Genético Simples”. Nessa versão, o AG cria, a cada geração, uma população inteiramente nova que é descendente da população anterior. Além disso, optou-se pelo uso do recurso de elitismo, que mantém na população seguinte o melhor indivíduo da população anterior, essa estratégia melhora o desempenho do AG.

4.4.2 - Algoritmo DDS

Optou-se pela utilização do DDS (*Dynamically Dimensioned Search*) desenvolvido por Tolson (2005). Segundo Zheng *et al.* (2012), o DDS é um algoritmo mais simples que alguns algoritmos evolucionários e metaheurísticos, já que, foi projetado para encontrar boas soluções rapidamente e não é um algoritmo baseado na população. Além disso, o DDS não é impactado pela escala da função objetivo (Tolson *et al.* 2008).

O algoritmo possui apenas um parâmetro, chamado tamanho da vizinhança de perturbação (*scalar neighborhood size perturbation parameter - r*), pode ser utilizado para variáveis discretas, contínuas e/ou mistas (Tolson *et al.* 2007) e seu uso é recomendado para a otimização de funções objetivo computacionalmente caras, ou seja, que utilizam grande tempo computacional nas avaliações (Tolson, 2015).

O DDS está implementado e disponível gratuitamente para uso público no site (<http://www.civil.uwaterloo.ca/btolson/software.html>) na linguagem MATLAB, sendo utilizado no trabalho a versão 1.2 de 2015 do DDS, para funções de objetivo único.

O modelo otimizador que utiliza o DDS é composto por 9 funções do MATLAB (arquivos de extensão .m), onde a primeira é a função principal que “chama” as outras funções. A Tabela 4.6 apresenta as funções e a descrição delas, sendo que a última é a função do MATLAB que calcula a função aptidão.

Tabela 4.6 – Funções do MATLAB empregadas no algoritmo DDS

FUNÇÕES	DESCRIÇÃO
MainDDS.m	Função principal
read_DDS_inp.m	Lê o arquivo de entrada do programa
DDS.inout.m	Grava todas as entradas do DDS em um arquivo de saída
bounds.m	Lê os limites das variáveis de decisão
DDS.m	Código do algoritmo DDS
get_objfunc.m	Chama o arquivo .m da função aptidão
neigh_value_mixed.m	Determina se a variável de decisão selecionada para a variável de perturbação é discreta ou contínua
neigh_value_discrete.m	Perturba variável de decisão discreta
FADDS.m	Calcula a função aptidão

As etapas do funcionamento do algoritmo DDS são apresentadas na Figura 4.10. Segundo Tolson *et al.* (2007), a primeira etapa do algoritmo DDS é a definição das entradas que são: os coeficientes de penalidades, o número e identificação das variáveis do problema (bombas e válvulas); tempo de simulação (número de horas), os níveis iniciais dos reservatórios, o parâmetro que determina o tamanho da vizinhança de perturbação (r), o número máximo de avaliações da função objetivo (m), os vetores com limites superiores e inferiores das variáveis de decisão, x^{\max} e x^{\min} , respectivamente, e o vetor com as soluções iniciais, $x^0=[x_1,x_2,\dots,x_D]$, ou seja, os valores iniciais de cada variável de decisão.



Figura 4.10 - Funcionamento do algoritmo DDS

Na segunda etapa é realizada a avaliação da função aptidão com o vetor de soluções iniciais (x^0), que resulta na função $F(x^0)$ na iteração i . Por ser a primeira avaliação é atribuído F_{melhor} a esta solução e x^{melhor} ao vetor x^0 mostrado na Equação 4.28.

$$F_{\text{melhor}} = F(x^0) \text{ então } x^{\text{melhor}} = x^0 \text{ e } i=1 \quad (4.28)$$

Para o cálculo da aptidão é necessário a realização da simulação hidráulica utilizando o EPANET2, através da biblioteca de funções do EPANET2 (*toolkit*). O vetor de soluções iniciais que representa uma regra operacional de 24h é inserido no simulador pela função *setLinkStatus* e o vetor de níveis iniciais pela função *setNodeTankInitialLevel*.

A execução da simulação hidráulica pelo modelo de simulação segue a mesma lógica do AG, porém com funções da *toolkit* diferentes, devido, a linguagem de programação utilizada, que nesse caso é o MATLAB.

As funções empregadas para a simulação hidráulica dinâmica são *openHydraulicAnalysis*, *initializeHydraulicAnalysis*, *runHydraulicAnalysis*, *nextHydraulicAnalysisStep* e *closeHydraulicAnalysis*. O acesso as variáveis estado, energia, pressão, vazão, demanda é realizado, respectivamente, pelas funções *getLinkStatus*, *getLinkPumpEnergy*, *getNodePressure*, *getLinkFlows* e *getNodeActualDemand*. As variáveis obtidas pela simulação são utilizadas no cálculo da aptidão.

A etapa 3, consiste na seleção aleatória de um subconjunto J , entre as D variáveis de decisão, para inclusão na vizinhança $\{N\}$, para isso, primeiramente, calcula-se a probabilidade de incluir cada variável de decisão em $\{N\}$, como uma função de indexação da iteração corrente, mostrada na Equação 4.29. Então, para $d=1,2,\dots,D$ variáveis de decisão é adicionado d a $\{N\}$ com probabilidade P , se $\{N\}$ estiver vazio é selecionado um valor aleatório para d e adicionado a $\{N\}$.

$$P(i)=1-\ln(i)/\ln(m) \quad (4.29)$$

P = probabilidade de incluir cada variável em $\{N\}$;

i = iteração;

m = número máximo de avaliações da função objetivo.

Na próxima etapa ocorre a perturbação do vetor da melhor solução corrente (x^{melhor}) para todas as $j=1,2,\dots,J$ variáveis de decisão em $\{N\}$, usando variável aleatória normal padrão, $N(0,1)$, refletindo nos limites das variáveis de decisão, da seguinte forma:

$$x_j^{\text{nov}} = x_j^{\text{melhor}} + \sigma_j N(0,1), \text{ onde } \sigma_j = r(x_j^{\text{max}} - x_j^{\text{min}})$$

Se $x_j^{\text{nov}} < x_j^{\text{min}}$, então

$$x_j^{\text{nov}} = x_j^{\text{min}} + (x_j^{\text{min}} - x_j^{\text{nov}})$$

Se $x_j^{\text{nov}} > x_j^{\text{max}}$, define $x_j^{\text{nov}} = x_j^{\text{min}}$

Se $x_j^{\text{nov}} > x_j^{\text{max}}$, então

$$x_j^{\text{nov}} = x_j^{\text{max}} - (x_j^{\text{nov}} - x_j^{\text{max}})$$

Se $x_j^{\text{nov}} < x_j^{\text{min}}$, define $x_j^{\text{nov}} = x_j^{\text{max}}$

em que:

x_j^{nov} = vetor com soluções candidatas; x_j^{melhor} = vetor com soluções corrente; σ_j = desvio padrão; r = parâmetro tamanho da vizinhança de perturbação; x_j^{max} = vetor com limites superiores; x_j^{min} = vetor com limites inferiores.

As perturbações alteram os valores das variáveis de decisão aleatoriamente, estas perturbações são amostradas de uma distribuição normal com uma média de zero e desvio padrão especificado pelo usuário (parâmetro r) (Tolson e Schoemaker, 2007).

Com a perturbação da melhor solução corrente, surge um vetor com soluções candidatas (x^{nov}). Na etapa 5, ocorre a avaliação da função aptidão $F(x^{\text{nov}})$, que é calculada com o vetor de soluções candidatas (x^{nov}), e a melhor solução corrente é atualizada, se $F(x^{\text{nov}})$ for menor igual a F_{melhor} , então $F_{\text{melhor}} = F(x^{\text{nov}})$ e $x^{\text{melhor}} = x^{\text{nov}}$. Na etapa 6 é verificado se o critério de parada é satisfeito, se sim, o algoritmo imprime os resultados em arquivos *.txt*, se não, o algoritmo volta a etapa 3.

4.4.3 - Algoritmo DE

Segundo Dandy *et al.* (2010), o DE é mais rápido, em média, a convergir para uma solução ótima, do que outros algoritmos de otimização. O algoritmo DE também apresentou bons resultados para a obtenção da solução ótima com um tempo de processamento relativamente pequeno em problemas de otimização de SAAs, nos trabalhos de Dandy *et al.* (2010), Suribabu (2010), Zheng *et al.* (2012), Marchi *et al.* (2014), por isso, o algoritmo foi escolhido para minimizar os custos de energia elétrica do SAA do Descoberto.

O código original de Storn e Price (1997), criadores do DE, está disponível para uso público (<http://www1.icsi.berkeley.edu/~storn/code.html>) na linguagem MATLAB, esse algoritmo foi construído para variáveis contínuas, no entanto, as variáveis de decisão utilizadas no presente trabalho é de natureza discreta, assim foi empregado o algoritmo modificado por Buehren (2016) e encontra-se disponível na central MATLAB (<https://www.mathworks.com/matlabcentral/fileexchange/18593-differential-evolution>), que pode ser utilizado para variáveis contínuas e/ou discretas.

Assim como no DDS, o modelo otimizador DE incorpora várias funções no MATLAB, entre elas uma função principal. A função principal além de “chamar” as outras funções, como a função com o código do DE, a função que calcula a aptidão e funções auxiliares, ela também funciona como o local onde os dados de entrada são definidos, entre eles os coeficientes de penalidades, tempo de simulação (número de horas), níveis iniciais dos reservatórios, tamanho da população, número de gerações, a estratégia de mutação, o número e identificação das variáveis de decisão (768 variáveis) e seus limites superiores e inferiores, o número da população e gerações, a constante de mutação (F) e a constante de cruzamento (CR), sendo esta a primeira etapa do funcionamento do algoritmo (Figura 4.11).

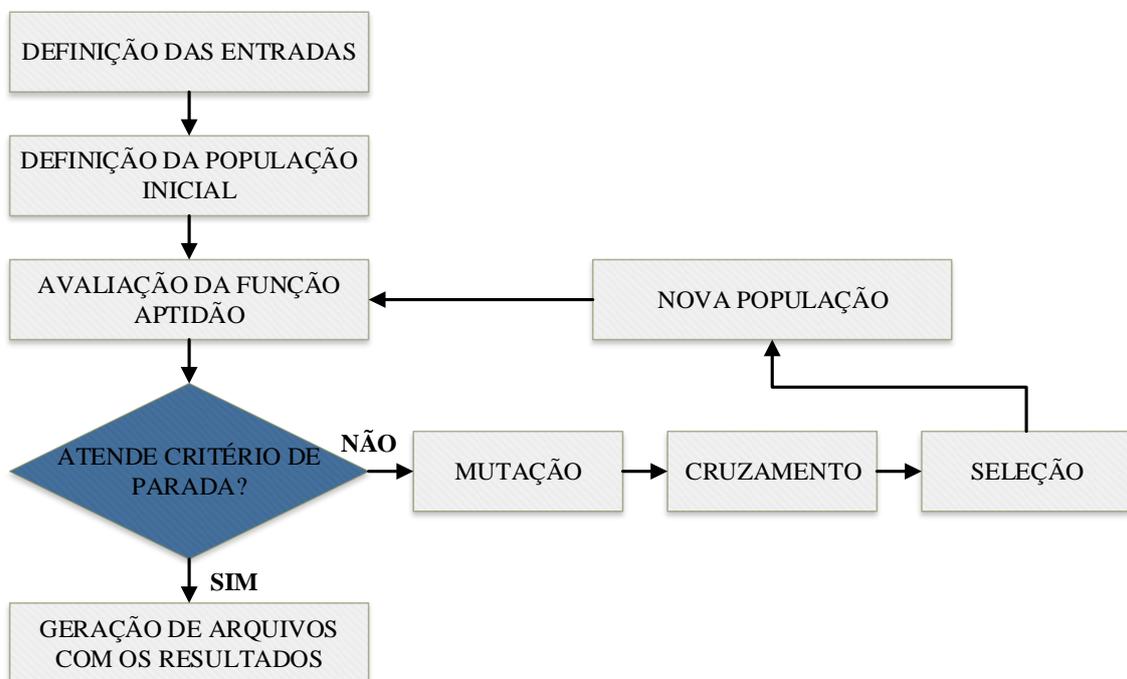


Figura 4.11 – Funcionamento do algoritmo DE

Na etapa 2 foi definido o vetor de solução inicial. Assim, todas as variáveis de decisão recebem um valor. Com esses valores foi realizada a avaliação da função aptidão, ou seja, o seu cálculo (etapa 3). O processo de simulação hidráulica para a obtenção das variáveis utilizadas no cálculo da aptidão, usam as mesmas funções empregadas pelo DDS, pois se trata da mesma linguagem de programação.

Na próxima etapa, foi verificado se o critério de avaliação foi atendido, que, nesse caso, é o número gerações. Se não foi atendido, ocorre os processos de mutação, cruzamento e seleção, para gerar uma nova população (conjunto de soluções) e calcular novamente a função aptidão. Se o critério de avaliação for atendido, o algoritmo simula a melhor solução, gera arquivos *.txt* com os resultados, e termina o processo.

O desempenho dos algoritmos testados foi analisado, a partir dos indicadores de desempenho: custo (função objetivo), aptidão e tempo de processamento.

4.4.4 - Parâmetros dos algoritmos

Garantir que um algoritmo está bem ajustado e que os algoritmos sejam ajustados igualmente é uma parte essencial de um estudo comparativo, caso contrário, é provável a obtenção de um desempenho inferior na análise comparativa de algoritmos (Dandy *et al.*, 2010).

A definição dos parâmetros utilizados para os algoritmos ocorreu a partir da metodologia proposta por Dandy *et al.* (2010), exceto os parâmetros do AG, a metodologia denominada calibração objetiva consiste, primeiramente, na revisão da literatura e seleção de diversos valores dos parâmetros dos algoritmos, seguido de diversas simulações onde altera-se um parâmetro e o outro é mantido constante, até que todos os parâmetros selecionados sejam testados.

Para as análises iniciais foi utilizado o valor de tamanho da população de 10, pois foi a população que gerou melhores soluções nas simulações de Gebrim (2013) e número de gerações de 1.000, totalizando 10.000 avaliações da função aptidão para todos os algoritmos. Esse número de avaliações foi definido, pois o tempo de processamento de cada avaliação é relativamente grande, assim, a definição de um número maior de avaliações demandaria excessivo tempo de processamento.

Os coeficientes de penalidades utilizados foram os empregados por Gebrim (2013), já que o problema estudado é o mesmo do autor e ele realizou a análise de sensibilidade desses coeficientes. A melhor combinação de coeficientes encontrados por Gebrim (2013) são apresentados na Tabela 4.7.

Tabela 4.7 – Coeficientes de penalidades adotados

COEFICIENTE DE PENALIDADE	VALOR
λ_1	100
λ_2	10.000
λ_3	10.000
λ_{4_g1}	10.000
λ_{4_g2}	10.000
λ_{4_g3}	2.000
λ_{4_g4}	1.000
λ_5	10.000

Segundo Carrijo (2004), a situação desejável, em termos de operação de sistemas de distribuição de água, é manter níveis de água próximos dos máximos nos reservatórios no final da noite.

De acordo com o autor, as soluções que consideram níveis iniciais máximos apresentaram maiores valores de benefício hidráulico (atendimento às pressões mínimas, níveis de água adequados nos reservatórios, garantia de atendimento das demandas) com, praticamente, os mesmos valores dos custos de energia elétrica que níveis médios.

A análise de sensibilidade de níveis iniciais dos reservatórios, realizada por Ribeiro (2005), indicou que os melhores resultados foram gerados para nível inicial igual a 90% do nível máximo, iniciando a operação às 06:00 horas.

Assim, optou-se por definir o nível inicial em todos os reservatórios em 98% do nível máximo (limite máximo), com o objetivo de gerar maiores benefícios hidráulicos e, consequentemente, aumentar a segurança operacional dos reservatórios.

Além disso, definiu-se o início da operação às 06:00 horas, já que, analisando Gebrim (2013), percebeu-se que nesse horário, os reservatórios atingiram o nível máximo, então, a

inicialização seria uma forma de acelerar a busca das soluções ótimas, e porque a operação real inicia nesse horário que é aproximadamente quando os operadores chegam à companhia.

4.4.4.1 - AG

Foram utilizados os operadores genéticos e parâmetros definidos por Gebrim (2013), já que se trata do mesmo problema de otimização (sistema de abastecimento do Descoberto) conforme Tabela 4.8.

Tabela 4.8 – Operadores e parâmetros genéticos adotados

OPERADORES GENÉTICOS			PARÂMETROS DE OTIMIZAÇÃO	
Tipo de Seleção	Tipo de Cruzamento	Mutação	Probabilidade de Cruzamento	Probabilidade de Mutação
Torneio	1 ponto	Simples	0,7	0,004

4.4.4.2 - DDS

Segundo Tolson *et al.* (2008), o único parâmetro de algoritmo do algoritmo DDS é o tamanho da vizinhança de perturbação (*scalar neighborhood size perturbation parameter - r*). Esse parâmetro define o desvio padrão do tamanho da perturbação aleatória, sendo 0,2 o valor padrão, que foi projetado para permitir que o algoritmo escape de regiões próximas aos mínimos locais.

Apesar do parâmetro variar de 0,1 a 1, Tolson (2015) não recomenda o experimento com diferentes r , pois à medida que o parâmetro aumenta em direção a 1, a amostragem torna-se cada vez mais espalhada, a partir do melhor valor atual da variável de decisão, porém se o usuário realizar a calibração do parâmetro, ele sugere a utilização de 0,1 a 0,3. Como o problema de otimização do presente estudo é complexo e os resultados dependem dos parâmetros utilizados, realizou-se a calibração do parâmetro r variando de 0,1 a 1, como mostrado na Tabela 4.9.

Tabela 4.9 – Valores dos parâmetros DDS utilizados na calibração

TESTE	R
1	0,1
2	0,2
3	0,3
4	0,4
5	0,5

Tabela 4.9 – Continuação

TESTE	R
6	0,6
7	0,7
8	0,8
9	0,9
10	1

4.4.4.3 - DE

A literatura apresenta ampla faixa para os parâmetros F (constante de mutação) e CR (constante de cruzamento) como mostra a Tabela 4.10, que também apresenta as estratégias de mutação empregadas em alguns trabalhos que utilizam o DE na otimização.

Tabela 4.10 – Valores sugeridos dos parâmetros DE

AUTOR	PARÂMETROS DE		
	Estratégia de mutação	F^*	CR^*
Storn e Price (1997)	DE/rand/1/bin	0,5	0,1 ou 0,9 a 1
Dandy <i>et al.</i> (2010)	DE/rand/1/bin	0,95	0,9
Suribabu (2010)	DE/rand/1/bin	0,6 a 0,9	0,3 a 0,6
Vasan e Simonovic (2010)	DE/rand/1/bin	0,5 a 1	0,5 a 1
Zheng <i>et al.</i> (2011)	DE/rand/1/bin; DE/best/1;	0,5 a 1	0,8 a 1
	DE/best/2;		
Marchi <i>et al.</i> (2014)	DE/current-to- best/2; DE/rand/2	0,1; 0,3; 0,5	0,1; 0,7
	DE/rand/1/bin		

* F : Constante de mutação; CR : Constante de cruzamento

A estratégia DE/rand/1/bin foi utilizada, pois é a estratégia clássica e empregada amplamente na literatura, como apresentada na Tabela 4.10.

Como os melhores valores dos parâmetros dependem do problema de otimização, foram testadas diferentes combinações dos parâmetros F e CR , seguindo a metodologia de Dandy *et al.* (2010). A escolha dos valores dos parâmetros testados foi baseada nos valores mais frequentes da Tabela 4.10. A Tabela 4.11 apresenta esses valores.

Tabela 4.11 – Valores dos parâmetros DE utilizados na calibração

TESTE	<i>F</i>	<i>CR</i>
1	0,5	0,5
2	0,5	0,6
3	0,5	0,7
4	0,5	0,8
5	0,5	0,9
6	0,5	1
7	0,6	0,5
8	0,6	0,6
9	0,6	0,7
10	0,6	0,8
11	0,6	0,9
12	0,6	1
13	0,7	0,5
14	0,7	0,6
15	0,7	0,7
16	0,7	0,8
17	0,7	0,9
18	0,7	1
19	0,8	0,5
20	0,8	0,6
21	0,8	0,7
22	0,8	0,8
23	0,8	0,9
24	0,8	1
25	0,9	0,5
26	0,9	0,6
27	0,9	0,7
28	0,9	0,8
29	0,9	0,9
30	0,9	1
31	1	0,5
32	1	0,6
33	1	0,7
34	1	0,8
35	1	0,9
36	1	1

Para a análise de sensibilidade dos parâmetros e utilização dos algoritmos de otimização, empregou-se o sistema completo e o mesmo vetor de soluções iniciais para todas as simulações. A melhor combinação dos parâmetros dos algoritmos foi avaliada através do valor da aptidão, custo (função objetivo) e tempo de processamento.

Para o ajuste dos parâmetros do AG e do DE, o tamanho da população e geração foram 10 e 1.000, respectivamente, totalizando 10.000 avaliações da função aptidão. O algoritmo DDS não possui o parâmetro população e nem o critério de parada (número de gerações), nesse caso o critério é o número de avaliações, que foi definido como 10.000.

4.5 - TRATAMENTO DAS PENALIDADES

Gebrim (2013) aplicou em seu trabalho o método das penalidades, onde a solução era fortemente penalizada pela ocorrência de determinados eventos. Como a estratégia de penalização da função aptidão interfere diretamente e consideravelmente nos resultados da otimização, foi implementada a utilização de uma nova estratégia de penalização.

No presente trabalho buscou-se penalizar gradualmente a função aptidão, baseado na metodologia de Homaifar *et al.* (1994), chamada de função de penalidade estática com vários níveis de violação (graus de penalizações) estabelecidos para cada restrição, e da metodologia proposta por Van Le (1995) que utiliza a lógica *fuzzy* para gerar as equações de penalização, a partir da *fuzzificação* das variáveis referentes às restrições.

Foram adotadas funções de pertinência de forma trapezoidal para definir os graus de penalização das restrições: pressão positiva nos nós de demanda; nível de água nos reservatórios no limite mínimo operacional; nível de água nos reservatórios no limite de extravasamento; número de acionamento de equipamentos no limite máximo tolerado; *diferença entre os níveis iniciais e finais dos reservatórios no limite tolerado.*

Foram aplicados 3 tratamentos nas penalidades utilizadas por Gebrim (2013), sendo o primeiro tratamento realizado nas penalidades 2 (níveis acima do limite máximo), 3 (níveis abaixo do limite mínimo) e 5 (*diferença entre o nível inicial e o final acima do limite estabelecido*). O segundo tratamento foi aplicado na penalidade 4 (número de acionamentos acima do limite) e o terceiro tratamento na penalidade 1 (não atendimento às demandas).

4.5.1 - Tratamento 1

O tratamento 1 foi realizado primeiramente na P2 e P3, que foram convertidas a P6. A penalidade 6 tem o objetivo de evitar que as soluções tenham níveis dos reservatórios

menores do que os limites mínimos e maiores que os limites máximos, garantindo assim, continuidade do abastecimento, a eliminação de soluções muito arriscadas sob o ponto de vista operacional e soluções que induzam a níveis de extravasamento.

Por isso, buscou-se penalizar gradualmente as soluções que geram níveis próximos a esses limites, de modo a obter soluções com níveis mais distantes dos limites proposto por Gebrim (2013), garantindo, assim, segurança à operação dos reservatórios. O limite mínimo e o máximo estabelecidos por Gebrim (2013) foram mantidos no trabalho.

A penalização proposta por Gebrim (2013) ocorria apenas nos casos em que o nível, no instante analisado, era menor ou maior que o limite estabelecido (L_{minmin} e L_{maxmax}). O tratamento 1, propõe penalizar a função aptidão de maneira gradual. Dessa forma os níveis (em %) entre o limite mínimo inferior (L_{minmin}) e o limite mínimo superior (L_{min}), bem como entre o limite máximo superior (L_{maxmax}) e o limite máximo inferior (L_{max}), foram penalizados. A Figura 4.12 apresenta duas funções de penalização de forma trapezoidal (analogia à função de pertinência), os limites dos níveis e seus respectivos graus de penalizações.

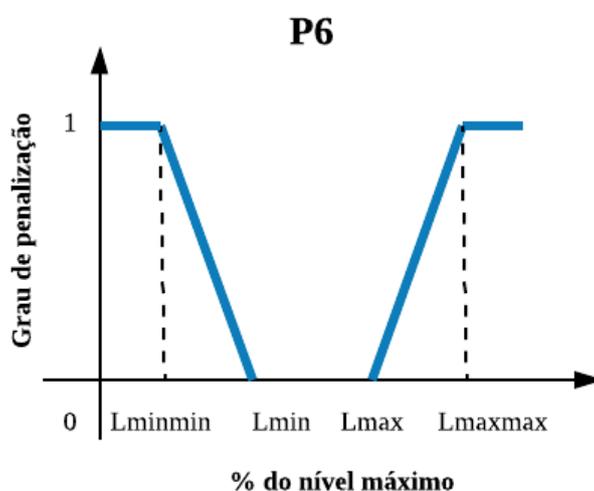


Figura 4.12 – Aplicação do conceito *fuzzy* à penalidade P6 (antiga P2 e P3)

Pela Figura 4.12, percebe-se que quanto mais próximo o nível (em %) for de L_{minmin} , maior a penalização aplicada e quanto mais próximo de L_{min} , menor a penalização (Equação 4.32), essa regra também vale para o caso do L_{maxmax} e L_{max} (Equação 4.34). A penalização gradual ($G6_g$) ocorreu entre L_{minmin} e L_{min} ; e L_{max} e L_{maxmax} ; e foi calculada por

interpolação linear conforme Equação 4.33 e 4.35. Níveis menores que L_{minmin} e maiores que L_{maxmax} receberam penalização completa igual a 1 ($G6_c$) (Equação 4.31) e níveis entre o L_{min} e L_{max} não foram penalizados (Equação 4.30).

$$\begin{aligned} \text{Se } L_{min} \leq \frac{N_{(j,t)} \times 100}{N_{jmax}} \leq L_{max} \\ G6_c = 0 \end{aligned} \quad (4.30)$$

$$\begin{aligned} \text{Se } \frac{N_{(j,t)} \times 100}{N_{jmax}} \leq L_{minmin} \text{ ou } \frac{N_{(j,t)} \times 100}{N_{jmax}} \geq L_{maxmax} \\ G6_c = 1 \end{aligned} \quad (4.31)$$

$$\begin{aligned} \text{Se } L_{minmin} \leq \frac{N_{(j,t)} \times 100}{N_{jmax}} \leq L_{min} \\ 0 < G6_g < 1 \end{aligned} \quad (4.32)$$

Por interpolação linear, temos que:

$$G6_g = \frac{L_{minmin} - \left(\frac{N_{(j,t)} \times 100}{N_{jmax}} \right)}{L_{minmin} - L_{min}} \quad (4.33)$$

$$\begin{aligned} \text{Se } L_{maxmax} \leq \frac{N_{(j,t)} \times 100}{N_{jmax}} \leq L_{max} \\ 0 < G6_g < 1 \end{aligned} \quad (4.34)$$

Por interpolação linear, temos que:

$$G6_g = \left[\frac{L_{max} - \left(\frac{N_{(j,t)} \times 100}{N_{jmax}} \right)}{L_{max} - L_{maxmax}} \right] \quad (4.35)$$

Para as penalizações graduais $G6_g$, o coeficiente de penalidade (γ_6) foi dividido por 2, a fim de não aumentar significativamente a penalidade pois, de acordo com Coello (2002), grandes

penalidades reduzem a possibilidade de exploração da região inviável desde o início da busca. Então a penalidade 6 será igual a:

$$P6 = \sum_{t=1}^T \sum_{j=1}^R (G6_c \times \gamma6) + \left(G6_g \times \frac{\gamma6}{2} \right) \quad (4.36)$$

em que: Lmin = limite mínimo superior (%); Lmax = limite máximo inferior (%); Lminmin = limite mínimo inferior (%); Lmaxmax = limite máximo superior (%); N(j,t) = nível do reservatório j no instante t (m); Njmax: nível máximo do reservatório j (m); G6_c = grau de penalização completa aplicado a P6; G6_g = grau de penalização gradual aplicado a P6; γ6 = coeficiente de penalidade aplicado a G6; P6 = penalidade referente aos limites mínimos e máximos dos níveis; t = instante da simulação em análise (h); T: período total de análise (h); j: índice do reservatório em análise; R: número total de reservatórios analisados.

Outra parcela do tratamento 1, refere-se à penalidade denominada *diferença entre o nível inicial e final acima do limite* (P5), que foi convertida a P7, no entanto, manteve o mesmo objetivo e o limite estabelecido por Gebrim (2013) de 5% do nível máximo do reservatório. A maior penalização nesse caso, ocorreu quando a *diferença entre nível inicial e final* em %, foi maior que Lmaxd (Equação 4.38). A penalização gradual foi aplicada quando a diferença esteve entre Lmind e Lmaxd (Equação 4.39), sendo que, quanto mais próxima a diferença foi de Lmaxd, maior a penalização empregada. As diferenças entre 0 e Lmind não foram penalizadas (Equação 4.37), conforme ilustrado pela Figura 4.13.

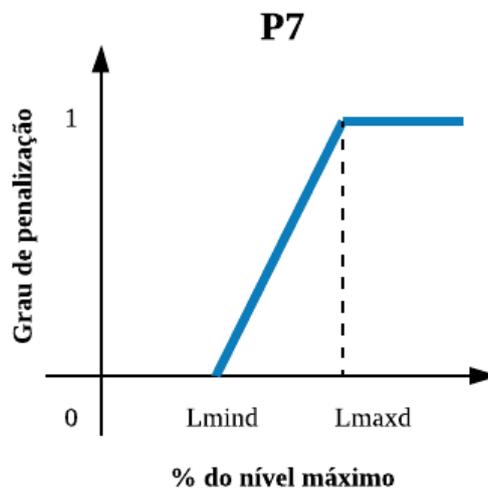


Figura 4.13 – Aplicação do conceito *fuzzy* à penalidade P7 (antiga P5)

As Equações retiradas da Figura 4.13 são mostradas a seguir.

$$\begin{aligned} Se \left(\frac{|N_{j\text{inicial}} - N_{j\text{final}}| \times 100}{N_{j\text{max}}} \right) \leq L_{\text{mind}} \\ G7_c = 0 \end{aligned} \quad (4.37)$$

Os G7 completos e graduais foram multiplicados pela *diferença entre níveis iniciais e finais*, já que esta estratégia gerou bons resultados nas simulações de Gebrim (2013).

$$\begin{aligned} Se \left(\frac{|N_{j\text{inicial}} - N_{j\text{final}}| \times 100}{N_{j\text{max}}} \right) \geq L_{\text{maxd}} \\ G7_c = 1 \times |N_{j\text{inicial}} - N_{j\text{final}}| \end{aligned} \quad (4.38)$$

$$\begin{aligned} Se L_{\text{mind}} \leq \left(\frac{|N_{j\text{inicial}} - N_{j\text{final}}| \times 100}{N_{j\text{max}}} \right) \leq L_{\text{maxd}} \\ 0 < G7_g < 1 \end{aligned} \quad (4.39)$$

Por interpolação linear, temos que:

$$G7_g = \left[\frac{L_{\text{mind}} - \left(\frac{|N_{j\text{inicial}} - N_{j\text{final}}| \times 100}{N_{j\text{max}}} \right)}{L_{\text{mind}} - L_{\text{maxd}}} \right] \times |N_{j\text{inicial}} - N_{j\text{final}}| \quad (4.40)$$

Para as penalizações graduais a $G7_g$, o coeficiente de penalidade (γ_7) foi dividido por 2, a fim de não aumentar significativamente a dimensão da penalidade, pelo mesmo motivo da divisão da γ_6 , exposto anteriormente. Então a penalidade 7 será igual a:

$$P7 = \sum_{t=1}^T \sum_{j=1}^G (G7_c \times \gamma_7) + \left(G7_g \times \frac{\gamma_7}{2} \right) \quad (4.41)$$

em que:

L_{mind} = limite mínimo (%); L_{maxd} = limite máximo (%); $N_{j\text{inicial}}$ = nível do reservatório j no início da simulação (m); $N_{j\text{final}}$ = nível do reservatório j no instante final de simulação

(m); N_{jmax} : nível máximo do reservatório j (m); $G7_c$ = grau de penalização completa aplicado a P7; $G7_g$ = grau de penalização gradual aplicado a P7; γ_7 = coeficiente de penalidade aplicado a G7; P7 = penalidade referente as *diferenças entre os níveis iniciais e finais*; t = instante da simulação em análise (h); T = período total de análise (h); j = índice do reservatório em análise; G = número total de reservatórios analisados.

Para o tratamento 1 foram testados os seguintes cenários, onde os limites apresentados na Tabela 4.12, representam as porcentagens dos níveis máximos nos reservatórios. Os limites intermediários (Lmin, Lmax e Lmind) definidos não foram muito distantes dos limites estabelecidos por Gebrim (2013) (Lminmin, Lmaxmax e Lmaxd) para não restringir ainda mais a busca pelas soluções na otimização.

Tabela 4.12 – Cenários simulados para o tratamento 1

CENÁRIO	Lminmin	Lmin	Lmax	Lmaxmax	Lmind	Lmaxd
1	10%	-	-	98%	-	5%
2	10%	13%	95%	98%	4%	5%
3	10%	12%	96%	98%	4%	5%
4	10%	14%	94%	98%	4%	5%
5	10%	15%	93%	98%	4%	5%
6	10%	13%	95%	98%	3%	5%
7	10%	12%	96%	98%	3%	5%
8	10%	14%	94%	98%	3%	5%
9	10%	15%	93%	98%	3%	5%

4.5.2 - Tratamento 2

O tratamento 2 foi aplicado à penalidade 4, que tem como objetivo a redução do número de acionamentos de bombas e válvulas. No tratamento foi mantido o primeiro componente da penalidade 4, assim o tratamento foi aplicado apenas ao segundo componente. Nesse caso manteve-se a divisão dos grupos realizada por Gebrim (2013), em que para cada grupo foi estabelecido um limite mínimo e máximo de acionamentos apresentados na Figura 4.14, onde o conceito *fuzzy* da função de pertinência de forma trapezoidal foi aplicado à penalidade 4 (P4).

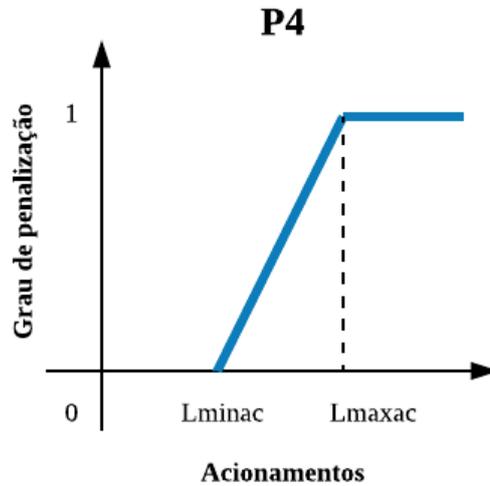


Figura 4.14 – Aplicação do conceito *fuzzy* à penalidade P4

Pela Figura 4.14, observa-se que os acionamentos entre o L_{minac} e L_{maxac} são penalizados gradualmente, onde a penalização aumenta com a proximidade do L_{maxac} , e acionamentos maiores que L_{maxac} recebem penalização completa. As Equações (4.43), (4.44) e (4.45) mostram o cálculo da penalização completa e gradual.

$$\begin{aligned} \text{Se } Ac_{(b,t)} \leq L_{minac} \\ G4_c = 0 \end{aligned} \quad (4.42)$$

$$\begin{aligned} \text{Se } Ac_{(b,t)} \geq L_{maxac} \\ G4_c = 1 \end{aligned} \quad (4.43)$$

$$\begin{aligned} \text{Se } L_{minac} < Ac_{(b,t)} < L_{maxac} \\ 0 \leq G4_g \leq 1 \end{aligned} \quad (4.44)$$

Por interpolação linear:

$$G4_g = \frac{Ac_{(b,t)} - L_{minac}}{L_{minac} + L_{maxac}} \quad (4.45)$$

As penalizações completas $G4_c$, foram multiplicadas pelo coeficiente de penalidade ($\lambda_4 - g_n$) e por 2, a fim de obter maior impacto na função aptidão de soluções que ultrapassam os limites máximos de acionamentos e, conseqüentemente, evitá-las.

A Equação 4.46 calcula a penalidade 4 para o período total de simulação (T = 24h).

$$P4 = \sum_{b=1}^B \left[\left(\sum_{t=1}^T 1,5 \times CC_{(b,t-1)} \times Ac_{(b,t)} \right) + (G4_c \times \lambda_{4-g_n} \times 2) + (G4_g \times \lambda_{4-g_n}) \right] + \sum_{v=1}^V [(G4_c \times \lambda_{4-g_n} \times 2) + (G4_g \times \lambda_{4-g_n})] \quad (4.46)$$

em que:

b = índice que representa cada bomba do problema; B = número total de bombas do problema; v = índice que representa cada válvula do problema; V = número total de válvulas do problema; t = instante da simulação em análise; T = período total de análise; $CC_{(b, t)}$ = custo de consumo de energia elétrica da bomba b no período t-1 até t; λ_{4-g_n} = coeficiente aplicado à Penalidade 4 referente ao grupo de equipamento n; n = grupo a qual o equipamento analisado (bomba/válvula) pertence; $G4_c$ = grau de penalização completa aplicado a P4; $G4_g$ = grau de penalização gradual aplicado a P4; $Ac_{(b, t)}$ = verificação da ocorrência de acionamento da bomba b no instante t; L_{minac} = Limite mínimo de acionamentos; L_{maxac} = limite máximo de acionamentos.

A Tabela 4.13 apresenta os grupos dos equipamentos com suas respectivas potências, limites e coeficiente de penalidades utilizados no tratamento 2.

Tabela 4.13 – Grupos de equipamentos, limites e coeficientes de penalidades

Grupos	Equipamentos	Lminac	Lmaxac	Coeficiente de penalidade (λ_{4-g_n})
1	Bombas com potência de 11.000 cv	0	2	10.000
2	Bombas com potência de 5.500 cv	0	3	10.000
3	Bombas com potência entre 5.500 cv e 200 cv	1	4	2.000
4	Bombas com potência igual ou menor a 200 cv e válvulas	2	5	1.000

4.5.3 - Tratamento 3

O tratamento 3 foi aplicado a P1, que visa garantir a continuidade do abastecimento de água no sistema, a partir da penalização das pressões. A penalização aplicada por Gebrim (2013)

penalizava somente as soluções em que as pressões eram menores do que zero. Logo o tratamento se propôs a aplicar a penalidade também às soluções com pressões entre 0 (Lminp) e 10 m.c.a (Lmaxp), ou seja, soluções que geram pressões menores que a pressão mínima recomendada pela ABNT NBR 12218/1994. Assim, foi definida penalização completa para pressões menores que zero e gradual para pressões entre 0 e 10 m.c.a, sendo que a penalidade decresce com a proximidade do limite de pressão de 10 m.c.a.

Gebrim (2013) utilizou para o cálculo da penalidade 1 a multiplicação do coeficiente de penalidade (λ_1) pela demanda no nó. Essa opção foi mantida no presente trabalho.

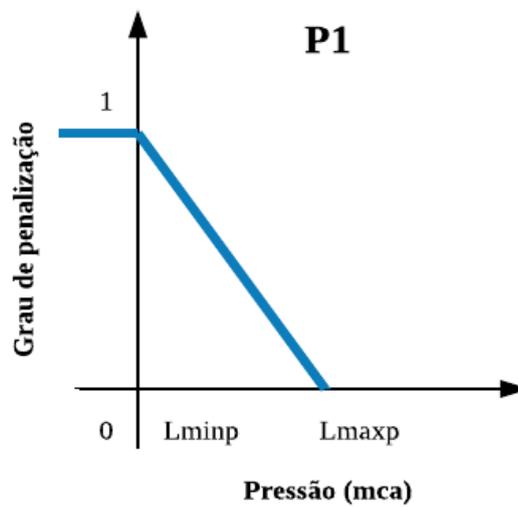


Figura 4.15 – Aplicação do conceito *fuzzy* à penalidade P1

As Equações representadas na Figura 4.15 são mostradas a seguir:

$$\begin{aligned} \text{Se } P_{(i,t)} \leq Lminp \\ G1_{c(i,t)} = 1 \end{aligned} \quad (4.47)$$

$$\begin{aligned} \text{Se } Lmaxp > P_{(i,t)} > Lminp \\ 0 \leq G1_{g(i,t)} \leq 1 \end{aligned} \quad (4.48)$$

Por interpolação linear:

$$G1_{g(i,t)} = \frac{Lmaxp - P_{(i,t)}}{Lmaxp - Lminp} \quad (4.49)$$

A Equação 4.50 calcula a penalidade 1 para o período total de simulação (T = 24h).

$$P1 = \sum_{t=1}^T \sum_{i=1}^G [(D_{(i,t)} \times \lambda_1 \times G1_{c(i,t)}) + (D_{(i,t)} \times \lambda_1 \times G1_{g(i,t)})] \quad (4.50)$$

em que:

t = instante da simulação em análise; T = período total de análise; i: índice do nó de consumo em análise; G = número total de nós de consumo analisados; $D_{(i,t)}$ = Demanda do nó i no instante t; λ_1 = coeficiente aplicado à Penalidade 1; $G1_{c(i,t)}$ = grau de penalização completa aplicado a P1 em cada nó i a cada instante t; $G1_{g(i,t)}$ = grau de penalização gradual aplicado a P1 em cada nó i a cada instante t; $P_{(i,t)}$ = pressão do nó i no instante t; L_{minp} = limite mínimo de pressão; L_{maxp} = limite máximo de pressão.

A primeira parcela da penalidade 1 foi aplicada somente nos nós com demanda, já a parte da penalidade que utilizava o grau de penalização gradual não foi aplicada a nós localizados nas saídas de reservatórios apoiados ou enterrados e na sucção de bombas, pois estes sempre terão pressões inferiores à mínima, devido a pressão no nó ser proveniente da carga do reservatório por exemplo.

Todas as simulações dos tratamentos, foram realizadas com a mesma solução inicial utilizada nos testes anteriores e 10.000 avaliações da função aptidão (tamanho de população igual a 10 e 1.000 gerações).

4.6 - EMPREGO DA TÉCNICA SEEDING

Gebrim (2013) e outros recomendaram a utilização de soluções iniciais boas para acelerar a convergência do algoritmo e, conseqüentemente, reduzir o tempo de processamento. A técnica de inicialização consiste na introdução de uma população inicial com alta aptidão no algoritmo e permite encontrar boas soluções rapidamente.

Assim, foi inserida uma solução inicial no algoritmo genético para a verificação da técnica. A solução de alta aptidão (Solução 3) foi obtida pela execução da solução média (Solução 2), resultante das simulações com os tratamentos das penalidades, como realizado no trabalho de Savic *et al.* (1997), com população igual a 10 e 10.000 gerações, totalizando

100.000 avaliações da função aptidão, pois a partir desse número de avaliações a solução não evolui significativamente.

O desempenho da solução inicial de alta aptidão foi comparado com uma solução aleatória com parâmetro semente aleatória igual a 100 (Solução 1), sendo esse valor do parâmetro utilizado no trabalho de Gebrim (2013); uma solução inicial considerada média, resultante das simulações com tratamentos (Solução 2); além da Solução 4, obtida a partir da simulação de 100.000 avaliações da função aptidão de uma regra operacional utilizada na operação real do sistema de abastecimento do Descoberto, nesse caso utilizou-se a mesma regra empregada por Gebrim (2013).

Todas as simulações do estudo foram realizadas em computador de mesa, Windows 10 com microprocessador Intel Core i7, com 2,2 GHz de velocidade e 8 GB de memória RAM.

5 - RESULTADOS E DISCUSSÕES

5.1 - UTILIZAÇÃO DOS ALGORITMOS DE OTIMIZAÇÃO

Para o ajuste dos parâmetros e comparação dos algoritmos utilizou-se o modelo hidráulico completo, com o mesmo vetor de soluções iniciais, com aptidão inicial de 1.980.000.

Os indicadores de desempenho utilizados para analisar os resultados foram custo (função objetivo), aptidão e tempo de processamento. A Tabela 5.1 apresenta os resultados dos testes de calibração dos parâmetros do DDS, com os resultados dos indicadores custo (função objetivo), aptidão e tempo.

Tabela 5.1 – Resultados dos testes de calibração dos parâmetros do DDS

TESTE	Custo (R\$)	Aptidão (Custo + Penalidades)	Tempo (s)
1	67271,3	957197,0	863,3
2	74710,0	1017062,9	816,3
3	79859,8	964528,1	668,5
4	76839,1	893892,6	829,3
5	85121,6	906283,3	826,3
6	75934,2	976334,8	830,2
7	79163,7	1011588,6	756,9
8	84696,3	859416,8	980,3
9	68173,6	1080551,1	892,2
10	80681,6	1004808,1	743,1

O menor valor da função objetivo (custo) foi verificado no teste 1, com 67.271,3 reais e a menor aptidão no teste 8, mas não considerou-se o custo como indicador decisivo na análise dos parâmetros, já que ele está agregado à aptidão. Assim, soluções de menor valor da função objetivo (custo) podem não representar necessariamente a melhor regra operacional.

O teste 8, com parâmetro igual a 0,8, obteve a menor aptidão (859.416,8), no entanto, o tempo de processamento foi o maior (980,3 segundos). Pela Tabela 5.1 é possível constatar que o menor tempo de processamento foi de 668,5 segundos, obtido no teste 3 ($r=0,3$), com uma aptidão relativamente boa 964.528,1, então escolheu-se esse parâmetro para o teste utilizando o DDS, o que atende as recomendações de Tolson (2015) que indica que o parâmetro r deve estar entre 0,1 e 0,3.

A Tabela 5.2 apresenta os resultados da calibração dos parâmetros do algoritmo DE.

Tabela 5.2 – Resultados dos testes de calibração dos parâmetros do DE

TESTE	Custo (R\$)	Aptidão (Custo + Penalidades)	Tempo (s)
1	85541,9	1256826,4	1311,6
2	72677,9	1166209,5	1075,3
3	70093,8	1237748,5	1021,8
4	71089,5	1176171,1	1423,8
5	74084,0	1305093,1	1350,8
6	79267,2	1226497,0	908,7
7	83331,1	1229919,3	1214,8
8	79422,7	1322710,6	979,6
9	73362,5	1175539,1	982,5
10	75210,6	962974,6	871,3
11	75457,3	1400371,9	1029,9
12	70976,1	1056292,5	1305,4
13	77019,6	1222291,6	1082,1
14	75757,4	1115500,8	1083,5
15	80200,0	1178979,6	1231,2
16	76830,2	1086395,3	1176,6
17	81896,5	1093104,4	898,5
18	74962,0	1362904,1	970,3
19	73051,5	1084012,6	1027,9
20	81359,5	1306292,5	1003,3
21	83379,7	1266164,5	893,5
22	68843,6	1307175,1	1154,7
23	77808,5	1347471,8	844,0
24	83407,1	1242598,8	838,4
25	72785,4	1170512,8	1078,3
26	76537,6	1221708,3	1449,4
27	77233,6	1188529,8	1088,7
28	78484,6	1385183,4	1135,1
29	65861,2	1186569,8	1042,0
30	74758,3	1124452,3	992,2
31	78435,6	1136855,6	1020,8
32	74185,9	1237241,6	1132,8
33	74459,3	1253016,9	1200,5
34	68172,2	1357624,4	1092,5
35	71602,5	1196857,3	1064,5
36	77662,1	1116514,6	1106,9

Analisando a Tabela 5.2, observa-se que o teste 10 ($F=0,6$ e $CR=0,8$) levou à menor aptidão (962.974,6), em um tempo de processamento de 871,3 segundos, valor muito próximo ao menor tempo, que foi de 838,4 segundos no teste 24. Então, os parâmetros do teste 10, foram utilizados para a simulação algoritmo DE. Como na análise dos parâmetros do DDS, nessa análise, não se considerou o custo (função objetivo) como indicador decisivo na avaliação do melhor desempenho do DE, pois ele está agregado a função aptidão (custo + penalidades). Isso pode ser constatado no teste 29, que levou ao menor valor da função objetivo (65.861,2 reais), no entanto, a aptidão não foi a menor, demonstrando que possivelmente a função objetivo desse teste apresentou o menor valor porque as penalidades relacionadas a ele foram elevadas.

O número de avaliações utilizadas para a comparação dos algoritmos foi de 10.000, com todos os algoritmos iniciando a simulação a partir da mesma solução inicial. Para comparação do desempenho dos algoritmos utilizou-se os indicadores: penalidades, custos, valores da aptidão e tempos, além da velocidade de convergência. A Figura 5.1 apresenta a comparação das propriedades de convergência dos três algoritmos de otimização (aptidão *versus* avaliações) quando aplicados ao SAA do Descoberto.

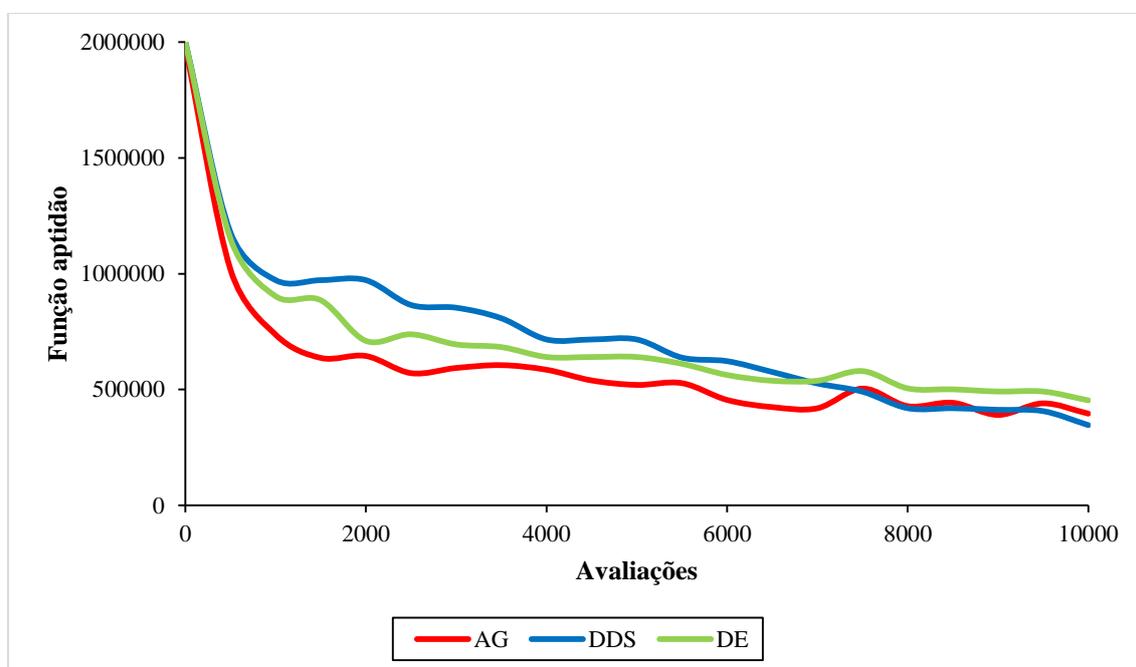


Figura 5.1 – Comparação da convergência de três algoritmos de otimização

Conforme Figura 5.1, o AG levou a velocidade de convergência mais rápida que o DDS e DE, no entanto, não converge para a melhor aptidão em 10.000 avaliações. O AG converge

rapidamente nas primeiras gerações, enquanto exibe uma convergência lenta nas gerações futuras, isso possivelmente ocorre porque o AG é um algoritmo de busca global, então em algumas avaliações, principalmente nas avaliações finais, ele possui mais dificuldade do que o DDS em melhorar a aptidão a partir do melhor valor, com isto é necessário a integração do algoritmo a um método de busca local ao final da otimização, para melhorar as soluções geradas pelo AG.

O DE apresentou convergência próxima ao AG nas primeiras avaliações, mas nas últimas avaliações a convergência se mostrou ruim, talvez pelo fato do DE segundo Dandy *et al.* (2010) gerar melhores resultados apenas para um grande número de avaliações ou porque o algoritmo possui o mesmo problema que o AG ao final da otimização, pois também realiza a busca global.

Apesar do DDS apresentar convergência mais lenta, do que o AG e o DE, o algoritmo levou à melhor aptidão, isso se deve provavelmente devido à estratégia de busca empregada pelo algoritmo, que realiza a busca em nível global no início da otimização e a busca se torna mais local quando o número de iterações se aproxima do número máximo de avaliações da função. O ajuste da busca global para a local é alcançado pela redução dinâmica e probabilística do número de dimensões na vizinhança, isto é, o conjunto de variáveis de decisão modificadas a partir de seu melhor valor (Tolson e Shoemaker, 2007). Porém, a avaliação global sofre com as penalidades, pois varre o espaço de busca com mais intensidade. Os resultados das simulações utilizando os algoritmos de otimização AG, DDS e DE com o mesmo número de avaliações (10.000) são apresentados na Tabela 5.3.

Tabela 5.3 – Comparação do desempenho dos algoritmos de otimização

Indicadores de Desempenho	AG	DDS	DE
Número de avaliações	10.000	10.000	10.000
P _{pressão} ($\lambda_1 = 100$)	0	0	0
P _{NAmin} ($\lambda_2 = 10.000$)	0	0	0
P _{NAmax} ($\lambda_3 = 10.000$)	60000	80000	90000
P _{acionamentos} ($\lambda_{4_g1} = 10.000$; $\lambda_{4_g2} = 10.000$; $\lambda_{4_g3} = 2.000$; $\lambda_{4_g4} = 1.000$)	118371	92109,1	110349,7
P _{diferençaentreNA} ($\lambda_5 = 10.000$)	106433	92495,2	172592,1
Pressão abaixo do limite ¹ (un.)	0	0	0
Nível abaixo do limite mínimo ² (un.)	0	0	0
Nível acima do limite máximo ³ (un.)	6	8	9

Tabela 5.3 - Continuação

Indicadores de Desempenho	AG	DDS	DE
Acionamentos acima do limite ⁴ (un.)	57	55	56
Diferença entre o nível inicial e final acima do limite ⁵ (un.)	9	8	13
Custo (R\$)	78348,6	81638,6	80207,9
Aptidão (Custo + Penalidades)	363153	346242,9	453149,7
Tempo EPANET2 (h)	4,08	13,3	13,26
Tempo Total (h)	4,1	13,58	13,54
% do tempo gasto com o EPANET2	99,1	98	97,9

*NA = nível de água no reservatório

¹ pressões entre 0 e 10 m.c.a;

² níveis abaixo de 10% do nível máximo;

³ níveis acima de 98% do nível máximo;

⁴ maior que 1 para bombas de 11000cv; maior que 2 para bombas de 5500 cv; maior que 3 para bombas entre 5500 e 200 cv; maior que 4 para bombas com potência igual ou inferior a 200 cv e todas as válvulas;

⁵ diferenças entre nível inicial e final maior que 5%.

O menor custo (função objetivo) foi de 78.348,6 reais, na otimização utilizando o AG. O custo de operação do SAA depende do custo da demanda das unidades consumidoras e do custo de consumo de energia elétrica das bombas, que é o produto do consumo de energia pelas bombas e do preço do consumo, que varia no horário de ponta e fora de ponta. Então, o menor valor da função objetivo pode ser explicado, pois a solução obtida possui menos bombas ligadas no horário de ponta e fora de ponta (Figura 5.2).

A menor aptidão encontrada foi no algoritmo DDS, valor de 346.242,9, se comparado aos algoritmos baseados na população (AG e DE), devido a estratégia de busca diferenciada empregada pelo DDS, mencionada anteriormente. A regra obtida pelo DDS não apresentou nenhuma pressão menor que 10 m.c.a e nenhum nível menor que 10% do nível máximo, porém obteve 8 níveis acima de 98% do nível máximo, 55 acionamentos acima do limite e 5 diferenças entre os níveis iniciais e finais maior que 5%.

As regras geradas pelos 3 algoritmos apresentaram número de acionamentos acima do limite suficientemente próximos, além disso demonstraram que as bombas de maior potência possuem menor número desse indicador, se comparadas às bombas de menor potência e as válvulas (Figura 5.2), pois possuem peso (coeficiente de penalidade) elevado, consequentemente maior impacto na função aptidão.

Já o maior valor de aptidão encontrado foi de 453.149,7, pelo algoritmo DE. Embora, a regra obtida pelo DE apresentou indicadores de desempenho próximos aos indicadores do AG e DE, exceto no caso do indicador *diferença entre o nível inicial e final acima do limite*. Segundo Dandy *et al.* (2010), o AG tem desempenho geral melhor que o DE, o que foi comprovado no presente estudo. No entanto, isso também pode ter sido causado pelo número de avaliações utilizado nas simulações.

Pela Tabela 5.3, nota-se que o tempo de processamento da simulação hidráulica (tempo EPANET2) é um elemento crítico na velocidade do processo de otimização, pois utiliza grande parte do tempo total, como demonstrado nos estudos de Ribeiro (2005) e Cunha (2009).

Percebeu-se que nas avaliações iniciais o tempo de processamento é maior do que nas avaliações subsequentes, onde o tempo de cada avaliação foi reduzido drasticamente. No entanto, essa redução não ocorre na mesma proporção para os diversos algoritmos, já que depende da estratégia de busca da solução ótima empregada por eles e, conseqüentemente, da simulação hidráulica das soluções geradas (regras operacionais) pelos algoritmos em cada avaliação.

Com o avanço do processo de otimização, à medida que soluções mais equilibradas são encontradas, o tempo de processamento é reduzido (Gebrim, 2013), pois soluções mais equilibradas ou mais estáveis hidraulicamente requerem menor número de iterações para a solução das equações de equilíbrio hidráulico do simulador.

Contudo, o AG apresentou o menor tempo de processamento (4,1h), provavelmente devido à linguagem de programação utilizada, já que, segundo a literatura, o AG necessita de grande tempo de processamento (Balla e Lingireddy, 2000) e, segundo os trabalhos de Suribabu e Neelakantan (2006); Zheng *et al.* (2012), o AG é mais lento que o DE. De acordo com Semeria (2007), o MATLAB é uma linguagem interpretada, isso significa que cada operação carrega uma sobrecarga adicional não encontrada em linguagens compiladas como C ou C++.

Pelo indicador % do tempo gasto com o EPANET2, no valor de 99,1%, percebe-se que o simulador hidráulico utiliza a maior parte do tempo de processamento total, o que significa

que o modelo otimizador usa muito pouco desse tempo, comprovando a rapidez do AG na linguagem C++. Em contrapartida, o DDS e DE utilizam mais tempo no modelo otimizador cerca de 2%, acredita-se que isso foi provocado pela linguagem de programação e pelo número de funções do MATLAB utilizadas nos algoritmos DDS e DE.

Dessa forma, aparentemente não é o algoritmo em si o responsável pelo maior ou menor tempo de execução, talvez o próprio EPANET2, ao ser executado dentro da plataforma MATLAB, tenha um desempenho inferior ao obtido quando executado diretamente dentro de um programa compilado em C++.

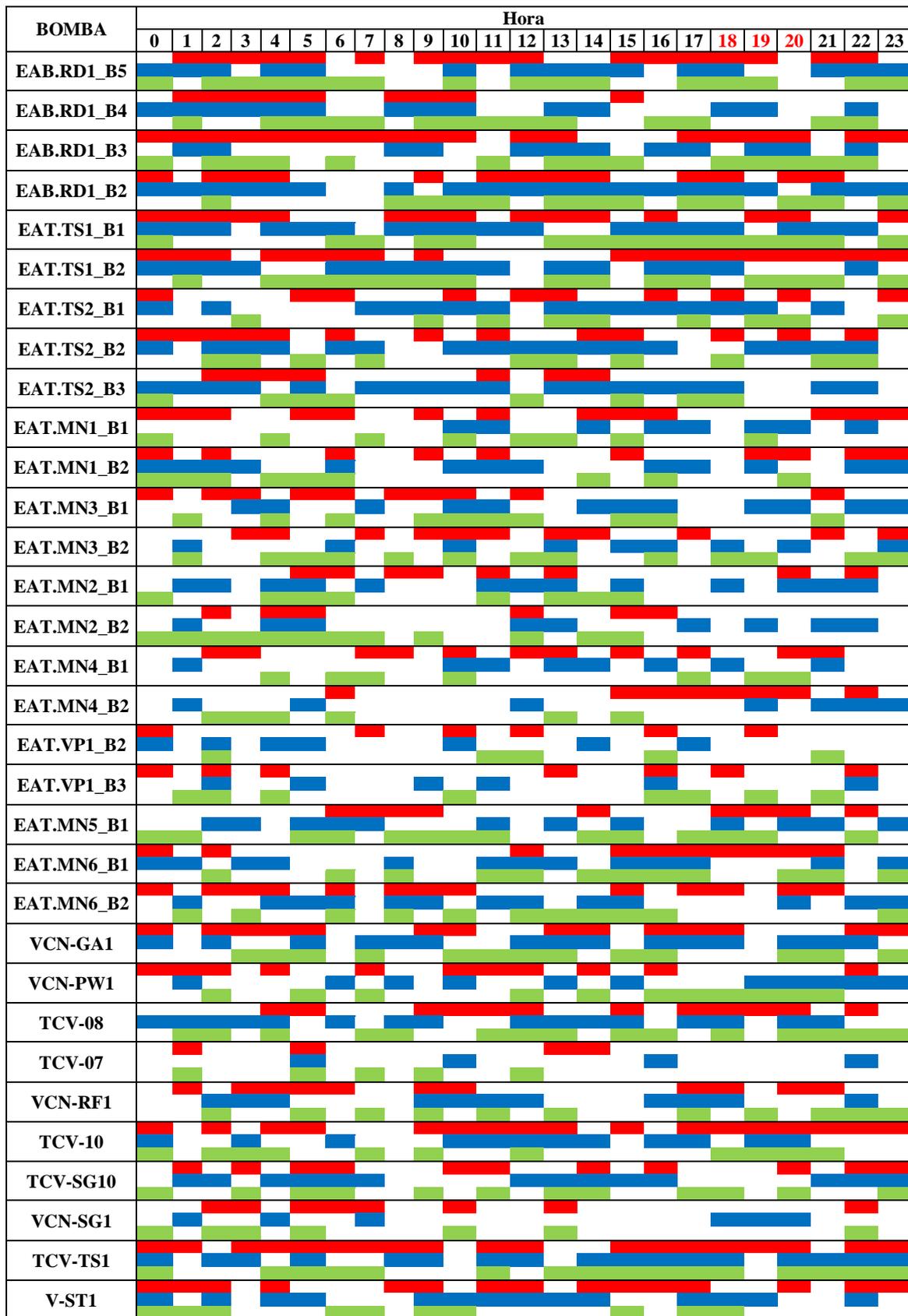
Embora, não seja possível a comparação direta dos tempos de processamento do AG, DDS e DE, pois, os algoritmos não foram simulados na mesma linguagem de programação (AG foi simulado em C++ e o DDS e DE na linguagem MATLAB), pode-se comparar o desempenho desse indicador entre o DDS e DE.

O DDS apresentou tempo de processamento ligeiramente maior que o DE, provavelmente isso ocorreu devido ao maior número de funções do MATLAB no DDS e/ou em decorrência da busca diferenciada empregada por cada algoritmo, que determina as soluções encontradas e, conseqüentemente, o tempo utilizado pelo simulador hidráulico.

Fatores como características do problema, número de avaliações da função objetivo, codificação da variável, natureza da função objetivo, operadores específicos dos algoritmos e valores dos parâmetros, podem influenciar os resultados finais dos algoritmos (Marchi *et al.* 2014).

Para as próximas simulações utilizou-se o AG em C++, pois ele apresentou menor valor da função objetivo e tempo de processamento, que foi pelo menos 3,3 vezes menor que dos outros algoritmos, ainda que a aptidão do AG não tenha sido a menor, ela se aproxima consideravelmente da menor aptidão encontrada.

Apesar das diferenças nos resultados, todos os algoritmos testados se mostraram bons na otimização do SAA do Descoberto, com destaque para DDS, que obteve resultados consistentes no quesito aptidão e é um algoritmo de fácil utilização, pois possui apenas um parâmetro.



- █ Equipamento ligado (Regra AG)
- █ Equipamento ligado (Regra DDS)
- █ Equipamento ligado (Regra DE)

Figura 5.2 - Regras operacionais obtidas pela simulação do AG, DDS e DE

5.2 - SIMPLIFICAÇÃO DO MODELO HIDRÁULICO

A etapa anterior mostrou a importância do tempo de processamento da simulação hidráulica e, portanto, da simplificação da representação da rede. A Tabela 5.4 apresenta uma comparação entre o modelo completo e simplificado. Observa-se que cerca de 10,3% dos trechos de tubulação foram removidos, juntamente com 12,1% das junções e 11% dos nós. Porém, as unidades operacionais como bombas, válvulas e reservatórios não foram simplificadas.

Tabela 5.4 – Comparação entre os componentes do modelo completo e simplificado

COMPONENTE	COMPLETO	SIMPLIFICADO
Tubulações	331	297
Nós	273	243
Junções	248	218
Reservatório de nível fixo	2	2
Reservatórios de nível variável	23	23
Bombas	47	47
Válvulas	17	17

Os trechos em série e paralelo foram simplificados como mostra Figura 5.3, que representa simplificações de trechos em paralelo.

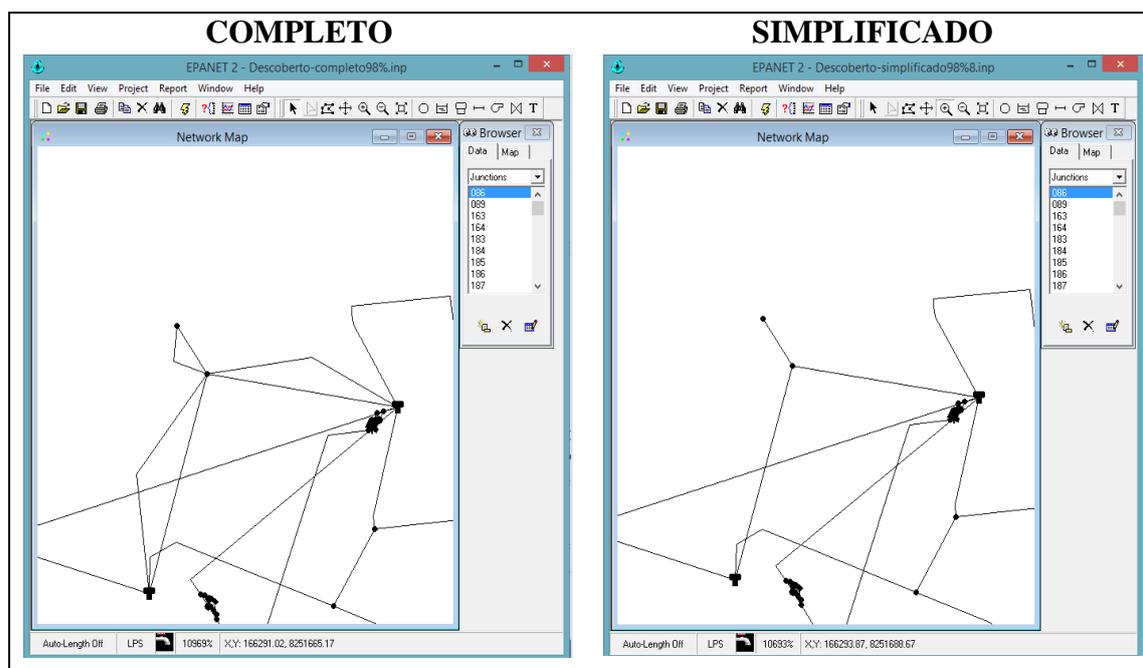


Figura 5.3 – Trechos do modelo hidráulico antes e após a simplificação

A Tabela 5.5 apresenta o cálculo do avaliador da qualidade da simplificação, o erro médio absoluto (MAE) entre o modelo completo e simplificado, com os reservatórios utilizados na calibração realizada por Gebrim (2013). Os níveis foram obtidos em intervalos de uma hora, no período de um dia, a partir da simulação da solução inicial (regra operacional) empregada nas simulações anteriores.

Tabela 5.5 – Cálculo do erro médio absoluto

Reservatório	Média dos níveis modelo completo (m)	Média dos níveis modelo simplificado (m)	Erro médio absoluto (m)
RAP.GA2	0,8096	0,8008	0,0088
RAP.GA1	4,2096	4,2260	0,0164
RAP.SG1	5,1668	5,1680	0,0012
RAP.MN2_C3C4	3,5232	3,5100	0,0132
RAP.SA1	1,2668	1,2532	0,0136
RAP.ST1	2,1812	2,1792	0,0020
RAP.TS1	2,7808	2,7960	0,0152
RAP.CE1	4,4628	4,5456	0,0828
REQ.GA1	3,0140	3,0136	0,0004
RAP.RE1	1,7880	1,7904	0,0024
RAP.SA2	4,2680	4,1656	0,1024
RAP.MN1_C1C2	3,9200	3,9200	0,0000
RAP.MN1_C3C4	1,5384	1,4724	0,0660
RAP.RF1	4,5800	4,5700	0,0100
RAP.PW1	3,9752	3,9744	0,0008
RAP.MN2_C1C2	3,9200	3,9200	0,0000
RAP.VP1	5,8812	5,8864	0,0052
<i>Média do erro médio absoluto</i>			<i>0,0200</i>

Pela Tabela 5.5 é possível observar que o menor erro foi encontrado no REQ.GA1 e o maior no RAP.SA2, sendo o erro médio absoluto no valor de 0,02 m, valor considerado pequeno. Pode-se, portanto, assumir que o modelo simplificado representa bem o modelo completo. As comparações entre os níveis desses reservatórios do modelo completo e simplificado são apresentadas na Figura 5.4 e Figura 5.5.

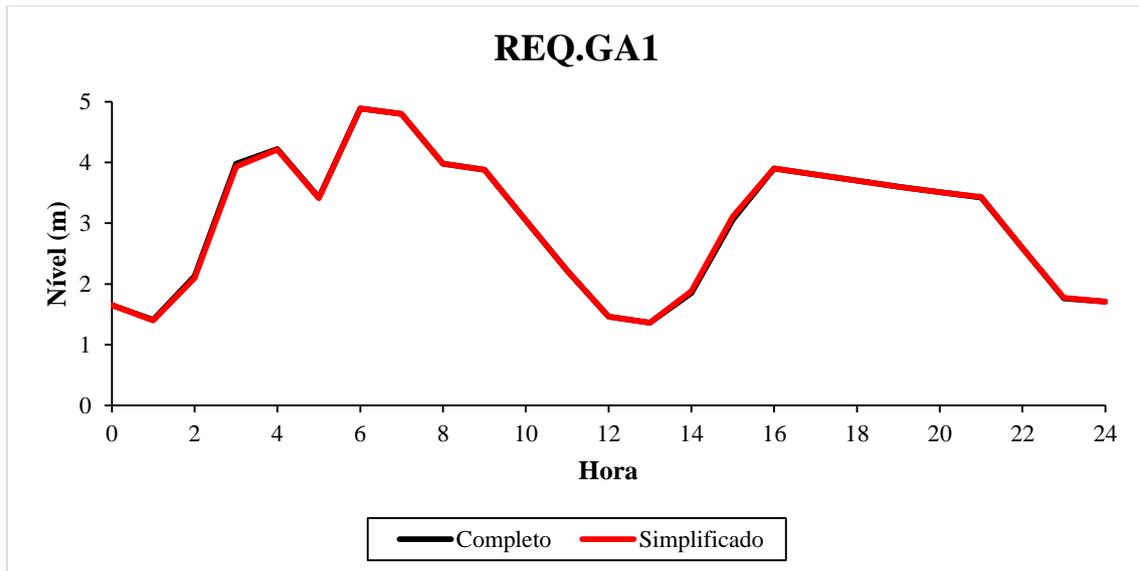


Figura 5.4 – Comparação entre os níveis do REQ.GA1 do modelo hidráulico completo e simplificado

É perceptível pelo gráfico que a trajetória dos níveis do modelo completo e simplificado praticamente coincidem em todas as horas do dia, sendo 0,05 m a maior diferença horária entre eles. Já no caso do RAP.SA2, a maior diferença foi de 0,2 m.

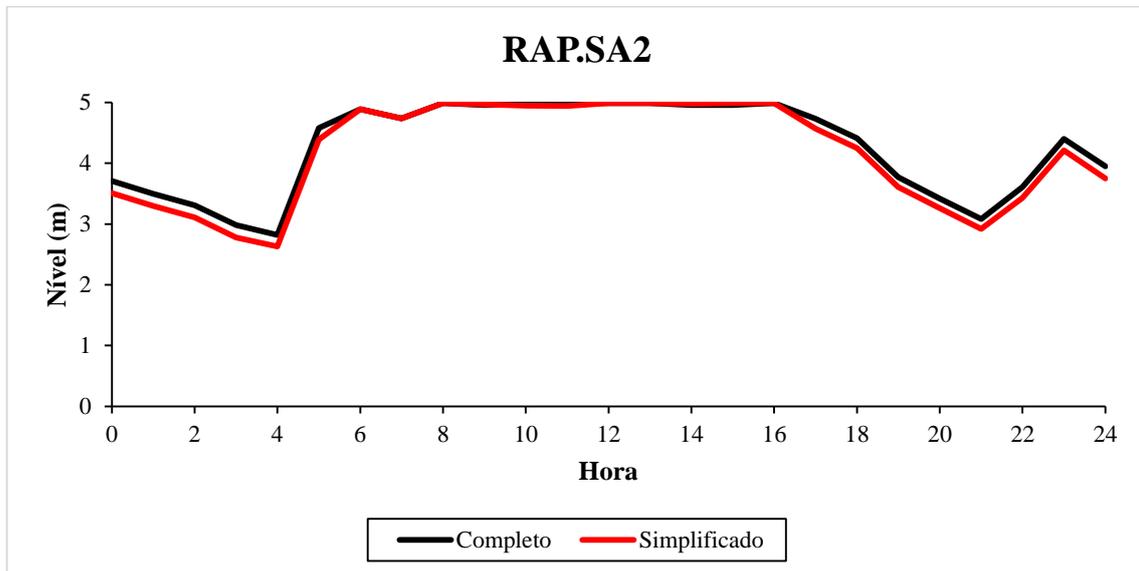


Figura 5.5 – Comparação entre os níveis do RAP.SA2 do modelo hidráulico completo e simplificado

Apesar das diferenças entre os níveis do modelo completo e simplificado foi possível constatar pelo erro médio absoluto (MAE), que o modelo simplificado reproduziu os resultados do modelo completo com alta fidelidade, desse modo, realizou-se a otimização de ambos os modelos, a partir da mesma solução inicial. A Tabela 5.6 apresenta a

comparação entre o modelo completo e simplificado, com os indicadores de desempenho penalidade, custo (função objetivo), aptidão (função objetivo + penalidades) e tempo de processamento.

Tabela 5.6 – Comparação entre o modelo hidráulico completo e simplificado

Indicadores de desempenho	COMPLETO	SIMPLIFICADO
Número de avaliações	10.000	10.000
Aptidão inicial	1.980.000	1.957.000
$P_{\text{pressão}} (\lambda_1 = 100)$	0	0
$P_{\text{NAmin}} (\lambda_2 = 10.000)$	0	0
$P_{\text{NAmax}} (\lambda_3 = 10.000)$	60000	110000
$P_{\text{acionamentos}} (\lambda_{4_g1} = 10.000; \lambda_{4_g2} = 10.000;$ $\lambda_{4_g3} = 2.000; \lambda_{4_g4} = 1.000)$	106433	99328
$P_{\text{diferençaentreNA}} (\lambda_5 = 10.000)$	118371	91712,7
Pressão abaixo do limite ¹ (un.)	0	0
Nível abaixo do limite mínimo ² (un.)	0	0
Nível acima do limite máximo ³ (un.)	6	11
Acionamentos acima do limite ⁴ (un.)	57	55
Diferença entre o nível inicial e final acima do limite ⁵ (un.)	9	7
Custo (R\$)	78348,6	83204,6
Aptidão (Custo + Penalidades)	363153	384245
Tempo máximo EPANET2 (s)	20,5	16,7
Tempo EPANET2 (h)	4,08	3,23
Tempo Total (h)	4,1	3,27
% do tempo gasto com o EPANET2	99,1	98,8

*NA = nível de água no reservatório

¹ pressões entre 0 e 10 m.c.a;

² níveis abaixo de 10% do nível máximo;

³ níveis acima de 98% do nível máximo;

⁴ maior que 1 para bombas de 11000cv; maior que 2 para bombas de 5500 cv; maior que 3 para bombas entre 5500 e 200 cv; maior que 4 para bombas com potência igual ou inferior a 200 cv e todas as válvulas;

⁵ diferenças entre nível inicial e final maior que 5%.

O efeito dos erros entre o modelo completo e simplificado comentados anteriormente é comprovado com a observação da Tabela 5.6, onde a maior diferença foi verificada na P3, penalidade referente ao nível de reservatório acima do limite máximo. As aptidões (custo + penalidades) obtidas não foram iguais, o que significa que as regras encontradas também são diferentes. Isso ocorreu porque o modelo completo iniciou a simulação com aptidão de

1.980.000 e o simplificado com 1.957.000, o que levou a otimização para pontos do espaço de busca diferentes e, conseqüentemente, soluções distintas geradas após a simulação, embora a solução inicial seja a mesma para ambos os modelos.

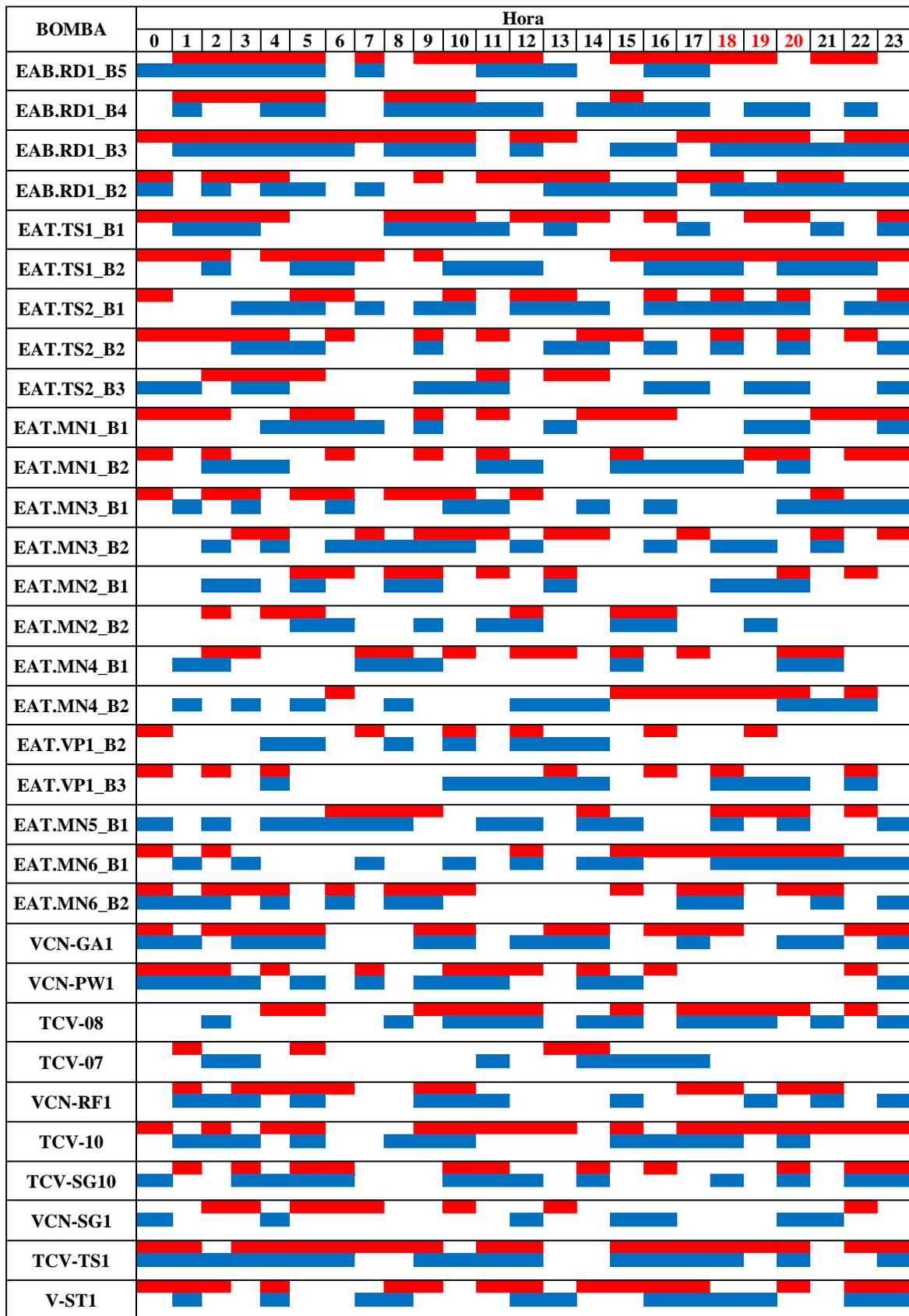
Embora, a diferença entre as aptidões seja considerada pequena, o modelo completo apresentou melhor desempenho para esse indicador (363.153), apresentando nenhuma pressão abaixo de 10 m.c.a e nenhum nível do reservatório resultou em níveis menores que 10% do nível máximo, no entanto, a regra gerada pelo modelo completo resultou em 9 níveis maiores que 98% do nível máximo durante as 24 horas de operação, 57 acionamentos acima dos limites e 9 casos em que a *diferença entre o nível inicial e final* foram maiores que 5%.

O modelo completo gerou menor valor da função objetivo do que o simplificado (78.348,6 reais), com a solução encontrada apresentando menor número de bombas ligadas (Figura 5.6). No entanto, a regra obtida (Figura 5.6) com o modelo simplificado, levou a menor número de penalizações por acionamentos (P4) e por *diferenças entre o nível inicial e final nos reservatórios* (P5), mostrando que o menor custo não representa necessariamente melhor solução do ponto de vista operacional, embora a P3 tenha sido maior.

Assim, como mostrado nos trabalhos de Shamir e Salomons (2008) e Broad *et al.* (2010), o modelo simplificado reduziu o tempo de processamento. A redução foi de 20,2% (50 min). Como a simulação hidráulica utiliza praticamente todo o tempo de processamento do modelo otimizador, a redução no tempo de simulação do modelo hidráulico reflete no tempo de processamento total.

O decréscimo do tempo só não foi maior devido à complexidade do sistema Descoberto, visto que possui muitas unidades operacionais, que não podem ser simplificadas e também porque, o sistema completo já estava relativamente simplificado, com poucos trechos passíveis de maior simplificação.

O modelo simplificado foi utilizado para as próximas simulações, em função do menor tempo de processamento, e por representar bem os resultados produzidos pelo modelo hidráulico, gerando solução muito próxima à do modelo completo.



█ Equipamento ligado (Regra do modelo completo)
█ Equipamento ligado (Regra do modelo simplificado)

Figura 5.6 - Regras operacionais obtidas pela simulação do modelo completo e simplificado

5.3 - TRATAMENTO DAS PENALIDADES

Todos os cenários foram simulados com o AG, o modelo hidráulico simplificado e a solução inicial utilizada nos testes anteriores. Foram realizadas 10.000 avaliações da função aptidão em todos os cenários, com população igual a 10 e 1.000 gerações.

Os cenários do tratamento 1 foram analisados pelos indicadores custo (função objetivo) e tempo de processamento, além dos indicadores *nível dos reservatórios abaixo ou acima dos limites*, e da *diferença entre o nível inicial e final nos reservatórios acima do limite*, pois se trata das penalidades referentes aos níveis dos reservatórios (P6 e P7).

Os resultados foram analisados qualitativamente e não em termos absolutos, pois em todos os cenários com tratamento houve a adição das penalizações graduais, logo uma mesma solução apresenta diferentes aptidões e valores das penalidades que dependem do cenário. Os resultados das simulações do cenário sem tratamento (cenário 1) e com o tratamento 1 (cenário 2 a 9) são apresentados na Tabela 5.7.

Tabela 5.7 – Cenário sem tratamento e cenários simulados com o tratamento 1

Indicadores de desempenho	Cenário 1	Cenário 2	Cenário 3	Cenário 4	Cenário 5	Cenário 6	Cenário 7	Cenário 8	Cenário 9
Número de avaliações	10.000	10.000	10.000	10.000	10.000	10.000	10.000	10.000	10.000
Aptidão inicial (Custo + Penalidades)	1957000	2339000	2290000	2360000	2400000	2339000	2292210	2364640	2403350
P _{pressão} ($\lambda_1 = 100$)	0	0	0	0	0	0	0	0	0
P _{NAmin} ($\lambda_2 = 10.000$)	0	-	-	-	-	-	-	-	-
P _{NAmax} ($\lambda_3 = 10.000$)	110000	-	-	-	-	-	-	-	-
P _{acionamentos} ($\lambda_{4.g1} = 10.000$; $\lambda_{4.g2} = 10.000$; $\lambda_{4.g3} = 2.000$; $\lambda_{4.g4} = 1.000$)	99328	86464,1	104673	87987,4	61095	98778,9	101773	86123,9	62463,8
P _{diferençaentreNA} ($\lambda_5 = 10.000$)	91712,7	-	-	-	-	-	-	-	-
P _{NAminmax} ($\lambda_6 = 10.000$)	-	360885	381470	390476	359471	355926	376889	431263	358173
P _{diferençaentreNA} ($\lambda_7 = 10.000$)	-	93195,4	109732	63762,4	72356,2	88345,5	89346,8	66051,1	80307,5

Tabela 5.7 – Continuação

Indicadores de desempenho	Cenário 1	Cenário 2	Cenário 3	Cenário 4	Cenário 5	Cenário 6	Cenário 7	Cenário 8	Cenário 9
Custo (R\$)	83204,6	77793,9	77783,0	80076,3	84799,8	73029,3	77617,3	90209,6	81994,4
Aptidão (Custo + Penalidades)	384245	618338	673658	622302	577722	616080	645626	673647	582938
Tempo (h)	3,3	2,0	4,0	3,4	1,9	2,6	2,6	4,5	2,1
Pressões abaixo do limite ¹ (un.)	0	0	0	0	0	0	0	0	0
Nível abaixo ou acima dos limites ² (un.)	11	3	7	5	2	4	7	9	1
Acionamentos acima do limite ³ (un.)	55	52	57	63	39	58	59	62	46
Diferença entre o nível inicial e final acima do limite ⁴ (un.)	7	9	12	10	10	8	11	11	11

¹ pressões entre 0 e 10 m.c.a de acordo com a ABNT NBR 12218/1994;

² níveis acima de 98% do nível máximo e abaixo de 10% do nível máximo;

³ maior que 1 para bombas de 11000cv; maior que 2 para bombas de 5500 cv; maior que 3 para bombas entre 5500 e 200 cv; maior que 4 para bombas com potência igual ou inferior a 200 cv e todas as válvulas;

⁴ diferenças entre níveis maior que 5% do nível inicial.

Conforme Tabela 5.7 é possível observar que o cenário que levou ao menor custo (função objetivo) foi o cenário 6, no valor de 73.029,3 reais, ao contrário do cenário 8, para qual foi verificado o maior valor da função objetivo (90.209,6 reais).

Todos os custos dos cenários criados foram reduzidos se comparado ao cenário 1, sem tratamento, exceto, os cenários 5 e 8. Isso se deve ao impacto causado pela junção das penalidades 2 e 3, que se transformaram na penalidade 6, na função aptidão.

Sabe-se que a otimização foi iniciada com níveis dos reservatórios em 98% de sua capacidade total, então, certamente o algoritmo priorizou soluções em que ocorre menos acionamentos (valor total de acionamentos), pois, eles estão ligados ao aumento dos níveis, que são imediatamente penalizados se ultrapassarem o limite de 98%.

Pelo grande impacto da P6 na aptidão, o algoritmo tende a reduzir as ações de ligar as bombas nas soluções e, conseqüentemente, os custos. Vale ressaltar que isso não é uma

regra, já que, o custo também depende da potência das bombas ligadas e do horário (ponta e fora de ponta) em que as bombas estão ligadas.

O menor tempo de processamento foi verificado no cenário 5 (1,9 horas) e o maior no cenário 8 (4,5 horas), isso se deve aos diferentes pontos do espaço de busca que os cenários foram direcionados durante a otimização pois, ainda que todos os cenários tenham o mesmo tratamento das penalidades, a penalização gradual foi diferente entre eles, ou seja, dependendo do cenário a penalidade 6 teve maior ou menor impacto na função aptidão, levando à otimização por diferentes soluções.

Como já comentado, o EPANET2 utiliza praticamente todo o tempo de processamento do código. Assim, soluções (regras operacionais) mais estáveis hidraulicamente necessitam de menor número de iterações para resolver as equações de equilíbrio hidráulico, logo utilizam menor tempo nas simulações do que regras menos estáveis, o que significa que o tempo de processamento depende diretamente das soluções geradas em cada avaliação durante a otimização.

Assim, é provável que os pontos do espaço de busca para os quais o cenário 5 foi direcionado durante a otimização, levaram a soluções hidraulicamente/operacionalmente mais estáveis do que os pontos em que o cenário 8 realizou a busca. Isso pode ser comprovado pelos valores dos indicadores *nível abaixo ou acima dos limites* e *diferença entre o nível inicial e final acima do limite*, em que o primeiro indicador, no cenário 5, apresentou o segundo menor valor, se comparado aos outros cenários. Por outro lado, o cenário 8 apresentou o maior valor desse indicador, entre os cenários com tratamento.

Quase todos os tempos de processamento dos cenários com tratamento foram menores do que o cenário sem tratamento, com exceção do cenário 3, 4 e 8, o que significa que o tratamento 1 tem grande probabilidade de levar a otimização para pontos do espaço de busca com soluções operacionalmente melhores, do que o cenário sem tratamento.

É importante mencionar que todas as simulações com tratamento se mostraram melhores do que o cenário sem tratamento (cenário 1) no quesito *nível abaixo ou acima dos limites*, o que pode ser justificado pela união da P2 a P3 que se transformou na P6, fazendo que a penalidade adquirisse maior dimensão do que as outras penalidades. Assim, a sua redução

causou maior impacto na função aptidão e, portanto, menores violações dessa penalidade. Isso foi mostrado pelo fato do melhor resultado desse indicador ser obtido no cenário 9, em que os intervalos entre os limites são os mais amplos e, com isso, as penalizações graduais tendem a ser maiores e, portanto, o impacto na aptidão também.

Entretanto, o indicador *diferença entre o nível inicial e final acima do limite* não foi reduzido em nenhum cenário com tratamento se comparado ao cenário 1, justamente devido ao impacto da P6 na aptidão, assim, o decréscimo da P6, gera maior redução da aptidão do que o decréscimo da P7. Logo, o algoritmo prioriza soluções em que a P6 é reduzida em detrimento da P7, talvez devido ao valor relativo da penalização final.

Embora, nenhum cenário tenha reduzido o indicador *diferença entre o nível inicial e final*, escolheu-se para as simulações com o tratamento 2 e 3, o cenário que alcançou o menor valor para esse indicador, que foi o cenário 6. Além de levar ao menor custo de operação, o cenário levou também a um bom valor de *nível abaixo ou acima dos limites* (4) e do tempo de processamento (2,6 horas) se comparado aos outros cenários.

A Tabela 5.8 apresenta os resultados dos indicadores de desempenho para 4 cenários, em que o primeiro trata-se do cenário 1 (sem tratamento das penalidades), ou seja, com as penalidades utilizadas por Gebrim (2013), que são a P1 referente ao atendimento às demandas, a P2 aos níveis dos reservatórios abaixo de um limite mínimo, a P3 aos níveis dos reservatórios acima de um limite máximo, a P4 ao número de acionamentos das bombas e válvulas acima do limite estabelecido e a P5 a *diferença entre o nível inicial e final dos reservatórios* maior que o limite definido.

Os outros 3 cenários são resultantes dos tratamentos das penalidades, de modo que cenário 2 possui apenas o tratamento da P2 e P3 que se transformaram na P6, e da P5 que foi alterada para a P7 (tratamento 1); o cenário 3 além do tratamento das penalidades citadas anteriormente, também incorporou o tratamento da P4 (tratamento 2); por fim o cenário 4 com os mesmos tratamentos do cenário 3, porém com a adição do tratamento da P1 (tratamento 3).

Tabela 5.8 – Cenários sem tratamento das penalidades e com tratamento 1, 2 e 3

Indicadores de desempenho	Cenário 1	Cenário 2	Cenário 3	Cenário 4
Número de avaliações	10.000	10.000	10.000	10.000
Aptidão inicial (Custo + Penalidades)	1957000	2339000	2414600	2417460
$P_{\text{pressão}} (\lambda_1 = 100)$	0	0	0	0
$P_{\text{NAmin}} (\lambda_2 = 10.000)$	0	-	-	-
$P_{\text{NAmax}} (\lambda_3 = 10.000)$	110000	-	-	-
$P_{\text{acionamentos}} (\lambda_{4_g1} = 10.000; \lambda_{4_g2} = 10.000; \lambda_{4_g3} = 2.000; \lambda_{4_g4} = 1.000)$	99328	98778,9	145882	166798
$P_{\text{diferençaentreNA}} (\lambda_5 = 10.000)$	91712,7	-	-	-
$P_{\text{NAminmax}} (\lambda_6 = 10.000)$	-	355926	351503	323869
$P_{\text{diferençaentreNA}} (\lambda_7 = 10.000)$	-	88345,5	93726,7	90081,9
Custo (R\$)	83204,6	73029,3	84199	75361,9
Aptidão (Custo + Penalidades)	384245	616080	675311	656111
Tempo total (h)	3,3	2,6	2,7	1,7
Pressões abaixo do limite ¹ (un.)	0	0	0	0
Nível abaixo ou acima dos limites ² (un.)	11	4	3	1
Acionamentos acima do limite ³ (un.)	55	58	46	52
Diferença entre o nível inicial e final acima do limite ⁴ (un.)	7	8	10	10
Acionamentos totais (un.)	389	395	387	387

¹ pressões entre 0 e 10 m.c.a de acordo com a ABNT NBR 12218/1994;

² níveis acima de 98% do nível máximo e abaixo de 10% do nível máximo;

³ maior que 1 para bombas de 11000cv; maior que 2 para bombas de 5500 cv; maior que 3 para bombas entre 5500 e 200 cv; maior que 4 para bombas com potência igual ou inferior a 200 cv e todas as válvulas;

⁴ diferenças entre níveis maior que 5% do nível inicial.

Para as simulações da Tabela 5.8, considerou-se além dos indicadores de desempenho utilizados anteriormente, custo (função objetivo), tempo, *nível abaixo ou acima dos limites*, *diferença entre o nível inicial e final acima do limite*, os indicadores acionamentos acima do limite e pressões entre 0 e 10 m.c.a, já que os tratamentos 2 e 3 se referem, respectivamente, a P4 e P1.

Analisando a Tabela 5.8, percebe-se que o cenário 2 apresentou o menor custo de operação (73.029,3) em 24 horas e o cenário 3 o maior no valor de 84.199 reais. Como na análise anterior, o indicador de desempenho aptidão não foi analisado quantitativamente, já que os

tratamentos das penalidades aumentam o valor da aptidão de forma diferenciada dependendo do cenário, dessa forma, não teria sentido a análise numérica do indicador.

No que tange ao tempo de processamento, o cenário 4 se mostrou melhor (1,7 h). É provável que essas soluções tenham sido encontradas em função do tratamento realizado e, conseqüentemente, do impacto na aptidão, o que justifica o fato de que todos os cenários com tratamento reduziram consideravelmente o tempo de processamento, ao contrário do cenário 1 (sem tratamento).

Pela Tabela 5.8, observa-se que o cenário 4 apresentou menor valor do indicador *nível abaixo ou acima dos limites* (1), devido ao impacto da P1 (tratamento 3) na função aptidão. Assim, a otimização foi conduzida a uma solução com menor valor do indicador referente a P6 (*nível abaixo ou acima dos limites*), o que gerou aumento do indicador referente a P7 (*diferença entre nível inicial e final acima do limite*) no valor de 10. Por outro lado, esse indicador referente a P5 no cenário 1 (sem tratamento) foi o menor (7) e o indicador *nível abaixo ou acima dos limites* o maior (11), mostrando que essas penalidades são inversamente proporcionais.

A simulação inicia com 98% do nível máximo, e ao final do tempo de simulação o nível deve estar próximo ao nível inicial, especificamente, a *diferença entre nível inicial e final* deve ser, no máximo, 5% para o indicador ser zero. Logo, para alcançar o nível inicial é preciso ocorrer o bombeamento pelo menos ao final das 24 horas de simulação. Assim, é possível que aconteça o aumento do número de acionamentos (valor total de acionamentos).

O cenário 4 levou ao maior valor do indicador *diferença entre níveis* (10) e menor número de acionamentos (387), igualmente ao cenário 3, enquanto que o cenário 2, levou ao maior número de acionamentos (395) e menor valor do indicador *diferença entre níveis* (8), dentre os cenários com tratamento.

O cenário 3 apresentou o menor número de acionamentos acima dos limites no valor de 46, seguido pelo cenário 4. Isso ocorreu devido ao impacto da penalidade 4 na função aptidão, causado pelo tratamento 2, já que dobrou-se o valor da parcela da penalidade referente a penalização completa.

O maior número desse indicador foi observado no cenário 2, pois o tratamento 1 causou impacto maior nas penalidades 6 e 7 do que na penalidade 4, o que é justificado pelos valores baixos dos 2 indicadores referentes aos níveis, respectivamente, 4 e 8. Além disso, não foi realizado o tratamento 2 nesse cenário. A penalidade 1 de não ocorrência de pressão negativa ou inferior à 10 m.c.a nos nós de consumo analisados foi satisfeita em todos os cenários, garantindo a continuidade do abastecimento ao longo de todo o período de operação e a geração de soluções viáveis do ponto de vista da aplicação da regra à operação real. A Figura 5.7 apresenta as pressões durante o período de operação em alguns nós de consumo do sistema Descoberto, utilizando a regra operacional obtida no cenário 4.

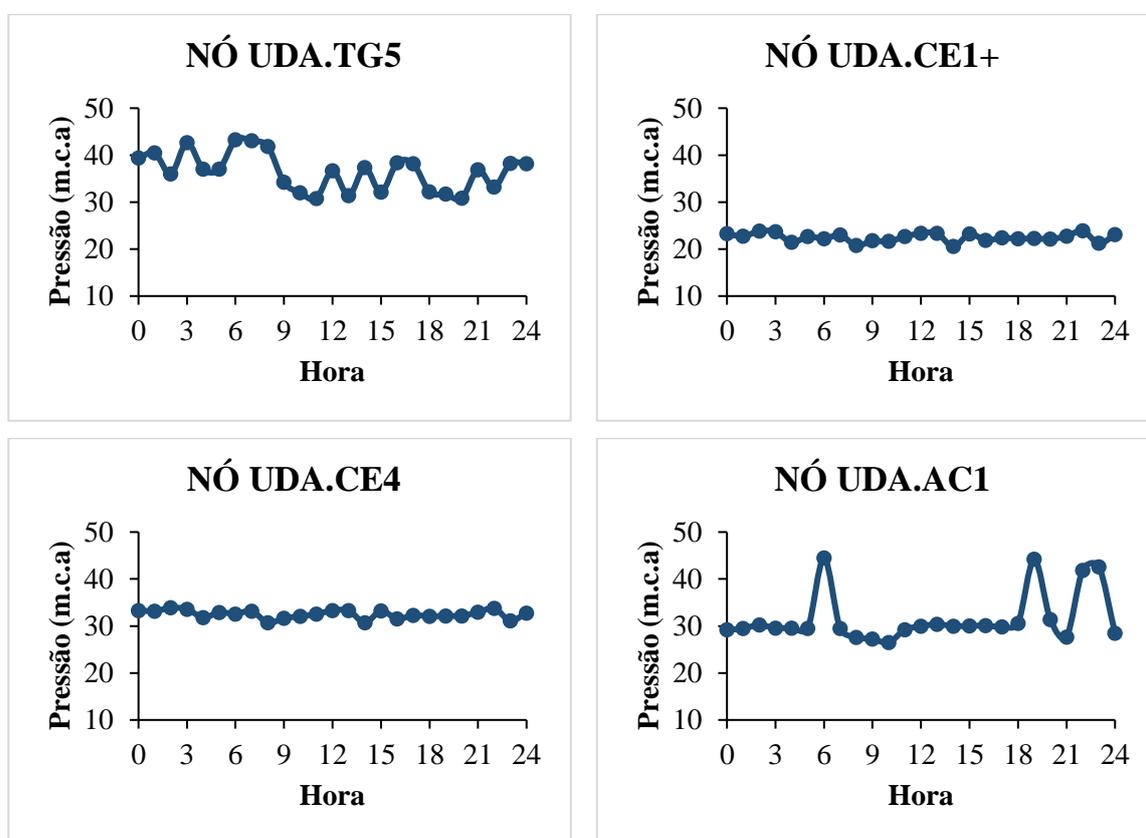


Figura 5.7 – Pressões nos nós de consumo do sistema Descoberto

Segundo Cunha (2009) a otimização energética da operação de um sistema de abastecimento de água provavelmente causará uma redução das pressões e, conseqüentemente, das perdas de água no sistema, pois a otimização irá buscar soluções que reduzam o tempo de funcionamento dos conjuntos motobombas para reduzir os custos operacionais.

A metodologia proposta para tratamentos das penalidades foi satisfatória na obtenção de soluções que reduziram as penalidades, como no caso do tratamento 1, que reduziu os níveis

abaixo e acima do limite se comparado ao cenário 1, o tratamento 2, que diminuiu o número de acionamentos acima do limite se comparado ao cenário 2, exceto no caso do indicador *diferença entre o nível inicial e final* (cenário 2 comparado ao 1).

O tratamento 3, empregado no cenário 4, não resultou em melhoria na penalidade 1, pois todos os cenários apresentaram pressões acima de 10 m.c.a. No entanto, ele levou ao melhor desempenho geral, porque obteve maiores reduções dos indicadores *níveis acima ou abaixo do limite* e tempo de processamento, além de apresentar valores próximos dos menores para os indicadores custo e número de acionamentos acima do limite se comparados aos outros cenários. Dessa forma, esse cenário foi utilizado nas próximas simulações.

5.4 - TÉCNICA SEEDING

A Tabela 5.9 apresenta os resultados da otimização utilizando diferentes soluções iniciais. As simulações com esta técnica foram realizadas a partir do cenário 4, com o tratamento 3, pois foi o que apresentou o melhor desempenho nas simulações anteriores.

A solução inicial 1 foi obtida aleatoriamente com a semente aleatória igual a 100, a solução 2 é a solução inicial utilizada no cenário 4, a solução 3 é proveniente da simulação do cenário 4 com 100.000 avaliações da função aptidão e a solução 4 é a solução obtida a partir da simulação de 100.000 avaliações da função aptidão da regra operacional empregada na operação real do sistema Descoberto mostrada no trabalho de Gebrim (2013).

Tabela 5.9 – Simulações empregando diferentes soluções iniciais e a regra de referência

Indicadores de desempenho	Solução	Solução	Solução	Solução	Referência
	1	2	3	4	
Número de avaliações	10.000	10.000	10.000	10.000	1
Aptidão inicial (Custo + Penalidades)	2695250	2417460	549824	546739	-
$P_{\text{pressão}} (\lambda_1 = 100)$	0	0	0	0	0
$P_{\text{NAmin}} (\lambda_2 = 10.000)$	-	-	-	-	-
$P_{\text{NAmax}} (\lambda_3 = 10.000)$	-	-	-	-	-
$P_{\text{acionamentos}} (\lambda_{4_g1} = 10.000; \lambda_{4_g2} = 10.000; \lambda_{4_g3} = 2.000; \lambda_{4_g4} = 1.000)$	129397	166798	96710,2	67550,6	11980,2
$P_{\text{diferençaentreNA}} (\lambda_5 = 10.000)$	-	-	-	-	-
$P_{\text{NAminmax}} (\lambda_6 = 10.000)$	434190	323869	320022	325399	1342740

Tabela 5.9 - Continuação

Indicadores de desempenho	Solução 1	Solução 2	Solução 3	Solução 4	Referência
$P_{\text{diferençaentreNA}} (\lambda_7 = 10.000)$	78106,6	90081,9	43656,8	38620,1	89199,5
Custo (R\$)	81721,7	75361,9	81618,4	81327,5	85934,5
Aptidão (Custo + Penalidades)	723415	656111	542007	512897	1529850
Tempo máximo EPANET2 (s)	14,4	12,3	4,9	1,2	-
Tempo EPANET2 (h)	5,12	1,66	1,21	1,06	-
Tempo total (h)	5,15	1,7	1,27	1,09	-
Pressões abaixo do limite ¹ (un.)	0	0	0	0	0
Nível abaixo ou acima dos limites ² (un.)	10	1	0	0	31
Acionamentos acima do limite ³ (un.)	51	52	38	26	0
Diferença entre o nível inicial e final acima do limite ⁴ (un.)	12	10	6	5	9

¹ pressões entre 0 e 10 m.c.a de acordo com a ABNT NBR 12218/1994;

² níveis acima de 98% do nível máximo e abaixo de 10% do nível máximo;

³ maior que 1 para bombas de 11000cv; maior que 2 para bombas de 5500 cv; maior que 3 para bombas entre 5500 e 200 cv; maior que 4 para bombas com potência igual ou inferior a 200 cv e todas as válvulas;

⁴ diferenças entre níveis maior que 5% do nível inicial.

Pela Tabela 5.9 é possível perceber que a solução em que a otimização é iniciada influencia diretamente o resultado da otimização, já que se trata da otimização com algoritmos evolutivos. Como obtido nos trabalhos de Savic *et al.* (1997) e Machado *et al.* (2008) a utilização da técnica *seeding* gerou melhor aptidão do que os resultantes do emprego de solução aleatória, além da utilização de menor tempo computacional, já que para encontrar uma boa solução é necessário um menor número de avaliações da função aptidão. No entanto, se a informação genética presente nos indivíduos da população inicial não for suficiente para a evolução da aptidão, o algoritmo genético pode sofrer de convergência prematura e ficar preso em soluções ótimas locais (Affenzeller e Wagner, 2004).

Pela Tabela 5.9, observa-se que a evolução da aptidão da solução 3 e 4 foi inferior se comparada a evolução da solução 1 e 2, pois a partir de determinada quantidade de avaliações da função ocorre a convergência do algoritmo, então a aptidão não evolui de forma significativa.

Dentre as soluções iniciais utilizadas, a que gerou melhores resultados foi a solução de melhor aptidão inicial, que é a solução 4 obtida a partir de uma regra aplicado na operação do sistema, já que os indivíduos presentes na solução estão mais aptos que as outras soluções. Além da melhor aptidão, a solução 4 encontrou os menores valores para todos os indicadores, exceto o custo (solução 2), já que este indicador possui relação inversamente proporcional às penalidades. Embora os indicadores obtidos com a simulação da solução 4 tenham apresentado melhores resultados se comparados às simulações com outras soluções iniciais, essa solução apresentou elevado número de acionamentos (26).

A questão já comentada sobre a relação do tempo com a qualidade das soluções (soluções mais equilibradas hidráulicamente) fica evidente com essas simulações e observação dos indicadores referentes aos tempos da Tabela 5.9, pois a simulação com a melhor aptidão inicial foi a que utilizou menor tempo computacional (solução 4). Isso pode ser comprovado pelo tempo máximo utilizado em cada avaliação do EPANET2, sendo o menor (1,2 s) o utilizado na solução 4 e pelo menor tempo total (1,09 h). O maior tempo total foi de 5,15 h utilizado na solução 1 (aleatória), com tempo máximo para cada simulação no EPANET2, no valor de 14,4 s.

A última coluna da Tabela 5.9 apresenta a simulação realizada com a regra de referência e os valores encontrados para os indicadores, exceto os indicadores referentes aos tempos. O valor da função objetivo obtido para a regra de referência foi de 85.934,5 reais e para a regra otimizada (solução 4) de 81.327,5 reais, revelando uma economia do custo total de bombeamento de 5,3%. A redução nos custos poderia ser maior, no entanto as captações do Descoberto apresentam elevada eficiência operacional, pois a companhia possui um grupo de trabalho específico (Grupo de Trabalho de Eficiência Energética) que tem como objetivo principal o desenvolvimento de um sistema para gerenciamento do consumo de energia (CAESB, 2014).

Embora, a regra de referência não tenha acionamentos acima dos limites, o indicador de níveis acima do limite de 98% do nível máximo do reservatório foi elevado. Acredita-se que isso ocorreu devido a imprecisão do modelo de otimização em decorrência das diferenças nos resultados gerados pelo modelo hidráulico simplificado e/ou das alterações realizadas na discretização horária dos estados das bombas e válvulas para a simulação da regra de

referência, tanto que a utilização da regra de referência na operação real, gerou níveis suficientemente próximos, mas não superiores, aos limites máximos impostos.

Vale ressaltar que os resultados da otimização encontrados na simulação da regra de referência foram diferentes dos encontrados no estudo de Gebrim (2013), pois o modelo do presente trabalho sofreu diversas alterações como a simplificação do modelo hidráulico e o tratamento das penalidades.

A melhor simulação realizada por Gebrim (2013) resultou em uma aptidão de 92.998, tendo apenas penalidades referentes aos acionamentos acima do limite e custo (função objetivo) de 78.283 reais, logo a regra obtida pelo autor é considerada melhor que a regra gerada pela solução 4. A regra de Gebrim (2013) foi obtida pela simulação de 60.000 avaliações da função aptidão, no entanto, o tempo de processamento computacional foi excessivamente elevado (15,8h), enquanto a regra otimizada foi obtida pela simulação utilizando 10.000 avaliações da aptidão com a solução 4.

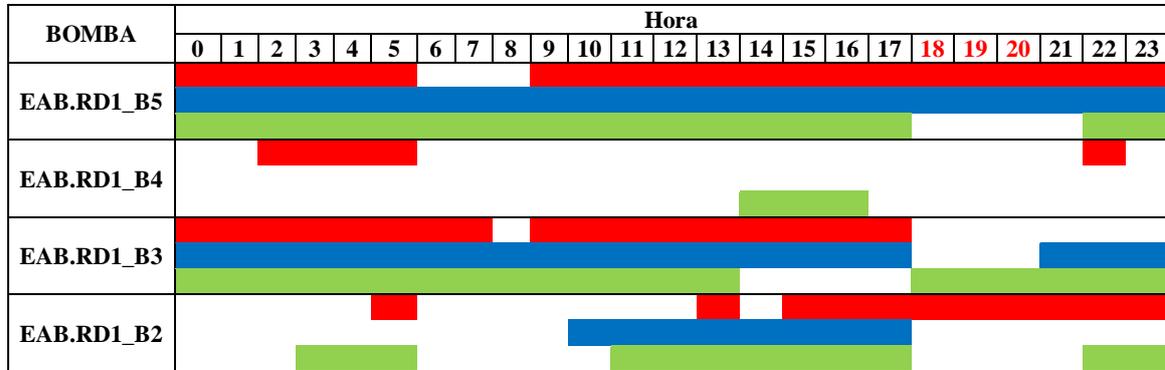
Menciona-se que as alterações sofridas pelo modelo atual como a simplificação, tratamento das penalidades, uso da *seeding* e as diferenças nos valores das variáveis utilizadas nas simulações como o número de avaliações, níveis dos reservatórios e soluções iniciais, dificultam a comparação direta entre os resultados gerados pelas regras operacionais obtidas pelo modelo de Gebrim (2013) e pelo modelo atual.

5.4.1 - Regras operacionais

As Figuras 5.8 a 5.14 ilustram a comparação da regra operacional obtida na simulação com a solução inicial 4, uma regra de referência praticada na companhia para a operação do sistema Descoberto e a regra gerada por Gebrim (2013), onde as bombas ligadas ou as válvulas abertas estão representados pelas cores vermelho para a regra otimizada (solução 4), azul para a regra de referência empregada pela Caesb e verde para a regra de Gebrim (2013).

Apesar do maior número de acionamentos, todas as bombas da classe de potência 1 e 2 (entre 5500 e 11000 cv) apresentadas na Figura 5.8 estão dentro do limite de acionamentos, pois os pesos atribuídos a essas classes de penalidades foram maiores que para as classes de

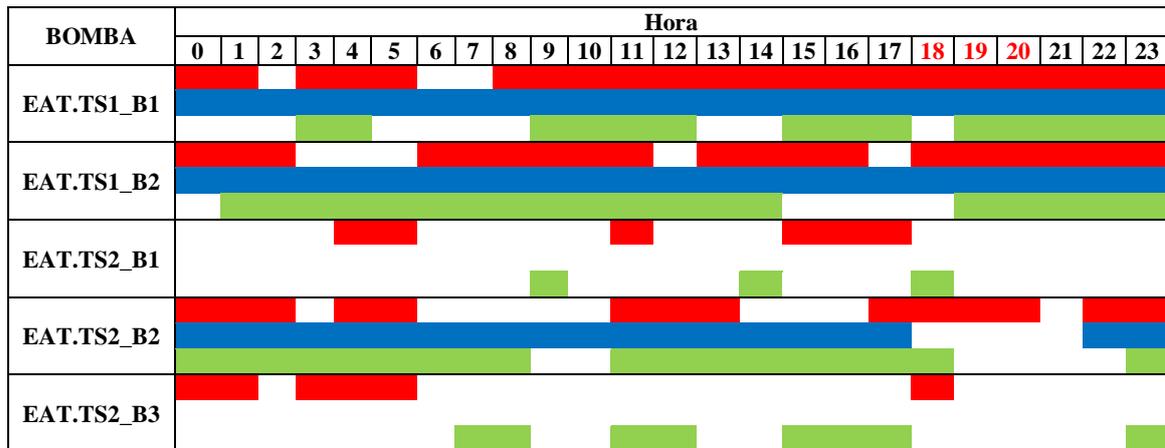
menor potência. As bombas de potência entre 550 e 800 cv (Figura 5.9) também estão dentro do limite de acionamentos na regra otimizada, exceto a bomba EAT.TS2_B2 que ultrapassou o limite, com um acionamento adicional.



Legenda:

- Equipamento ligado (Regra otimizada)
- Equipamento ligado (Regra referência)
- Equipamento ligado (Regra Gebrim, 2013)

Figura 5.8 – Regras operacionais para bombas de potência igual a 5.500 e 11.000 cv



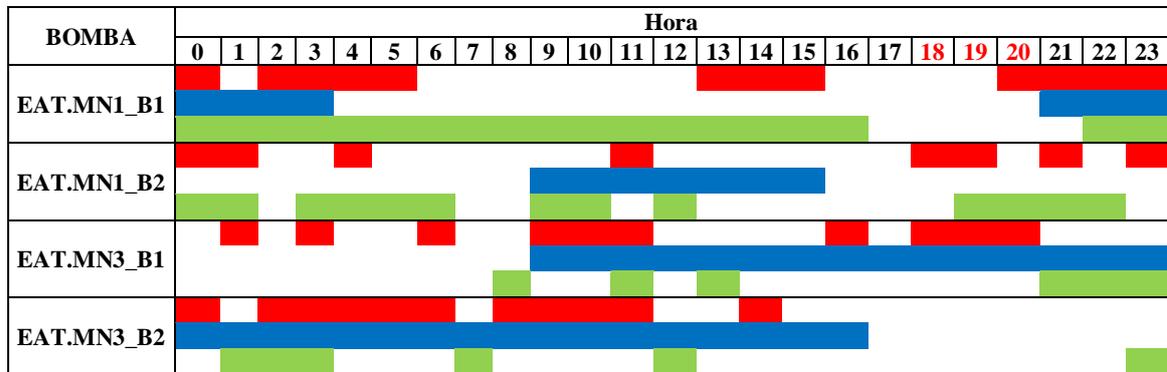
Legenda:

- Equipamento ligado (Regra otimizada)
- Equipamento ligado (Regra referência)
- Equipamento ligado (Regra Gebrim, 2013)

Figura 5.9 – Regras operacionais para bombas de potência igual a 550 e 800 cv

A regra de referência possui casos em que bombas nem são ligadas durante a operação, como o a bomba EAT.TS2_B1, EAT.TS2_B3 (Figura 5.9) e EAT.MN4_B2 (Figura 5.11), o que representa uma situação favorável do ponto de vista operacional, em relação a solução 4 em que bombas como a EAT.MN1_B2 e EAT.MN3_B1 (Figura 5.10) da regra otimizada e a regra de Gebrim (2013), apresentaram muitos acionamentos. Algumas bombas ficam ligadas

apenas por uma hora e desligadas em seguida, situação não usual e nem recomendada na operação de SAAs.

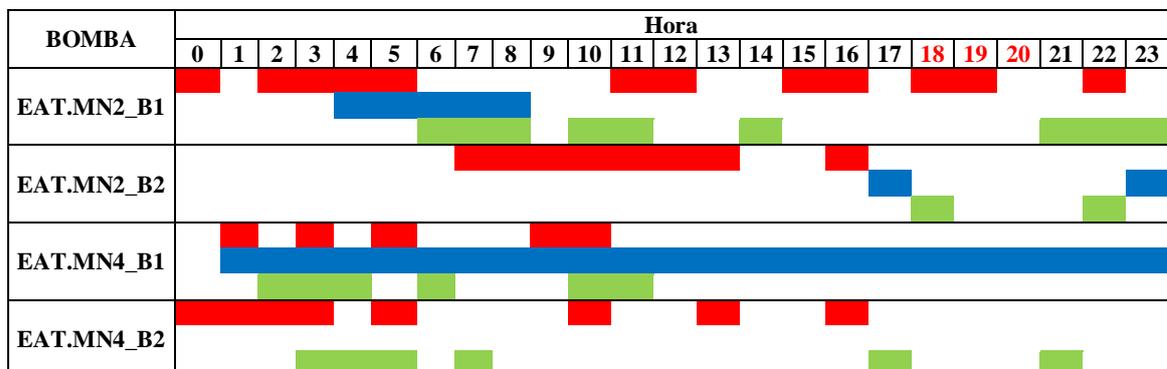


Legenda:

- Equipamento ligado (Regra otimizada)
- Equipamento ligado (Regra referência)
- Equipamento ligado (Regra Gebrim, 2013)

Figura 5.10 – Regras operacionais para bombas de potência igual a 375 cv

É perceptível pela Figura 5.11, a tendência de evitar acionamentos no horário de ponta na regra de referência. No caso da solução otimizada, como o objetivo é minimizar os custos (função objetivo), o modelo também evita esses horários, no entanto, pela Figura 5.11, percebe-se que em alguns casos, bombas da regra otimizada e da regra de Gebrim (2013) estão ligadas no horário de ponta, devido aos conflitos entre o custo com as penalidades referentes aos níveis pois, em algumas situações, é necessário o acionamento de bombas no horário de ponta para manter os níveis nos reservatórios (penalidade 6) e para os *níveis iniciais e finais* serem praticamente os mesmos.

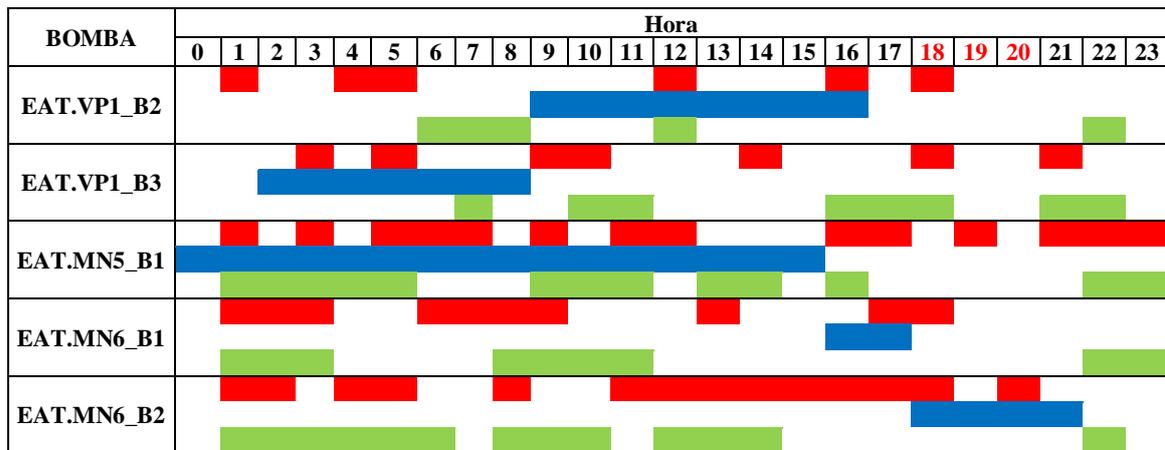


Legenda:

- Equipamento ligado (Regra otimizada)
- Equipamento ligado (Regra referência)
- Equipamento ligado (Regra Gebrim, 2013)

Figura 5.11 – Regras operacionais para bombas de potência igual a 150 e 200 cv

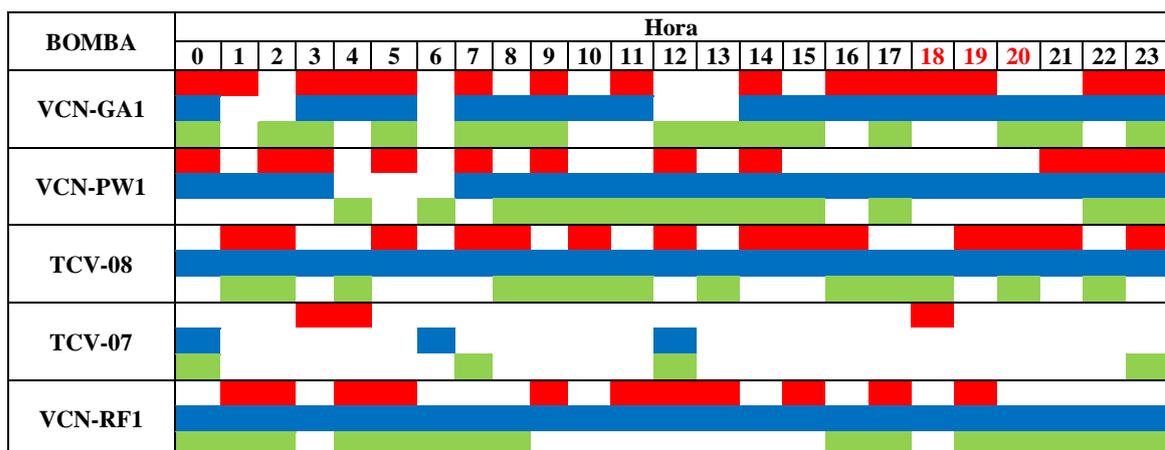
Há muitos acionamentos, acima do limite recomendado, das bombas de menor potência e válvulas (Figura 5.12 a 5.14), já que os pesos atribuídos à essas penalidades para essas classes foram os menores se comparados as classes de maiores potências. Além disso, os equipamentos dessa classe são ligados apenas por uma hora e depois desligados, sendo que em algumas ocasiões como a da bomba EAT.VP1_B2, o tempo total de funcionamento diário da bomba é muito baixo, o que pode não compensar os danos operacionais causados pelos acionamentos mesmo em casos onde há redução dos custos energéticos e/ou redução das penalidades.



Legenda:

- Equipamento ligado (Regra otimizada)
- Equipamento ligado (Regra referência)
- Equipamento ligado (Regra Gebrim, 2013)

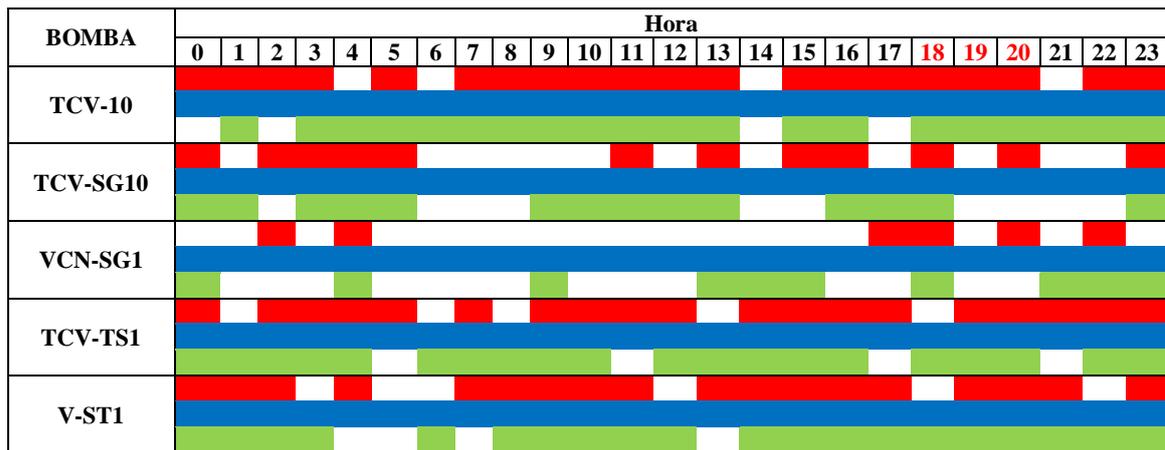
Figura 5.12 – Regras operacionais para bombas de potência igual a 100 e 150 cv



Legenda:

- Equipamento ligado (Regra otimizada)
- Equipamento ligado (Regra referência)
- Equipamento ligado (Regra Gebrim, 2013)

Figura 5.13 – Regras operacionais para válvulas 1



Legenda:

- Equipamento ligado (Regra otimizada)
- Equipamento ligado (Regra referência)
- Equipamento ligado (Regra Gebrim, 2013)

Figura 5.14 – Regras operacionais para válvulas 2

As regras operacionais obtidas pelo modelo de otimização e a regra de Gebrim (2013) possuem mais acionamentos que as regras de referência, pois os operadores do sistema visam realizar o menor número de manobras nos equipamentos, para evitar desgastes. Regras operacionais com grande quantidade de acionamentos das bombas podem até reduzir os custos com energia elétrica, mas podem aumentar consideravelmente os custos de manutenção dos equipamentos (Lansey e Awumah, 1994). Alterações nos valores das penalidades reduzem os acionamentos, mas não de forma significativa, além de aumentar as penalidades de maior relevância para a operação como a penalidade referente aos *níveis dos reservatórios acima ou abaixo do limite*.

Com o objetivo de reduzir o número de acionamentos, testou-se uma simulação onde a penalidade 7 foi substituída pelo aumento da duração total da simulação no EPANET2, que passou de 24h para 96h, visando o equilíbrio do modelo hidráulico e, conseqüentemente, níveis iniciais e finais iguais ou suficientemente próximos ao final da simulação. A Tabela 5.10 apresenta os resultados da simulação da solução 4 sem a penalidade 7 que correspondem as últimas 24h de simulação.

Tabela 5.10 – Simulação da solução 4 sem a penalidade 7

Indicadores de desempenho	Simulação sem P7
Número de avaliações	10.000
Aptidão inicial (Custo + Penalidades)	557839
Custo (R\$)	81727,4

Tabela 5.10 – Continuação

Indicadores de desempenho	Simulação sem P7
Aptidão (Custo + Penalidades)	385989
Tempo máximo EPANET2 (s)	3,5
Tempo EPANET2 (h)	4,17
Tempo total (h)	4,3
Pressões abaixo do limite ¹ (un.)	0
Nível abaixo ou acima dos limites ² (un.)	0
Acionamentos acima do limite ³ (un.)	21
Diferença entre o nível inicial e final acima do limite ⁴ (un.)	5

¹ pressões entre 0 e 10 m.c.a de acordo com a ABNT NBR 12218/1994;

² níveis acima de 98% do nível máximo e abaixo de 10% do nível máximo;

³ maior que 1 para bombas de 11000cv; maior que 2 para bombas de 5500 cv; maior que 3 para bombas entre 5500 e 200 cv; maior que 4 para bombas com potência igual ou inferior a 200 cv e todas as válvulas;

⁴ diferenças entre níveis maior que 5% do nível inicial.

O número de acionamentos foi reduzido de 26 para 21, porém, o indicador *diferença entre o nível inicial e final acima do limite* não sofreu decréscimo, embora, todos os níveis iniciais e finais dos reservatórios apresentaram valores próximos. Como a duração da simulação foi aumentada, o tempo utilizado pelo EPANET2 também aumentou se comparado a simulação aplicando a solução 4 (Tabela 5.9).

5.4.2 - Níveis dos reservatórios

A regra de referência demonstrou ao final do período simulado níveis inferiores aos do início da simulação em alguns casos, como para o reservatório do Riacho Fundo, apresentado na Figura 5.15. Segundo Gebrim (2013) esse fato pode indicar que, na operação real, o controle da operação não é baseado somente na condição de que o nível final seja maior ou igual ao nível inicial, mas sim que o nível inicial seja suficiente para suportar o ciclo diário de consumo, assim como na regra de Gebrim (2013), onde o nível inicia a operação com 30% do nível máximo e termina a operação com 78,6%.

Em situações de otimização em tempo real, quando o sistema alcança o ciclo regular de funcionamento, espera-se que a cada ciclo de otimização a condição inicial de reservação seja compatível com a condição ótima para a operação do próximo período, sem a necessidade de ajustes nos volumes iniciais reservados (Gebrim, 2013).

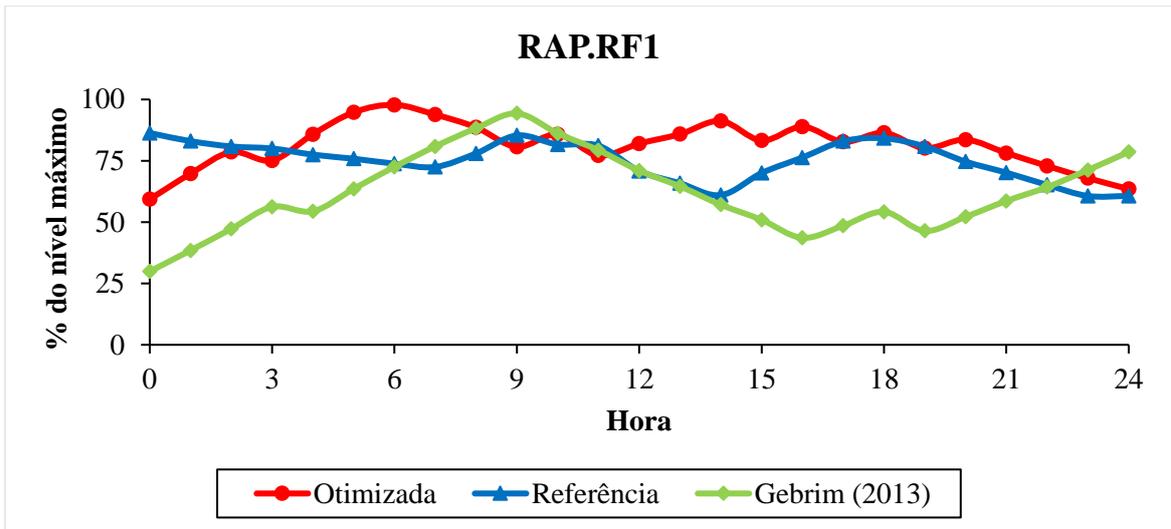


Figura 5.15 – Trajetória dos níveis do reservatório Riacho Fundo

Observando-se a Figura 5.16 relativa ao reservatório RAP.MN1 da regra otimizada é possível perceber a oscilação do nível de água do mesmo, devido à elevada quantidade de manobras no sistema que a otimização determina para as bombas que constituem as elevatórias EAT.MN3 e EAT.MN4, que por sua vez, possuem influência no comportamento desse reservatório. O reservatório de M Norte 1 para a regra otimizada apresentou os menores níveis dentre todos os reservatórios, sendo que o menor nível foi obtido na hora 12 e apresentou valor superior ao limite mínimo imposto de 10%.

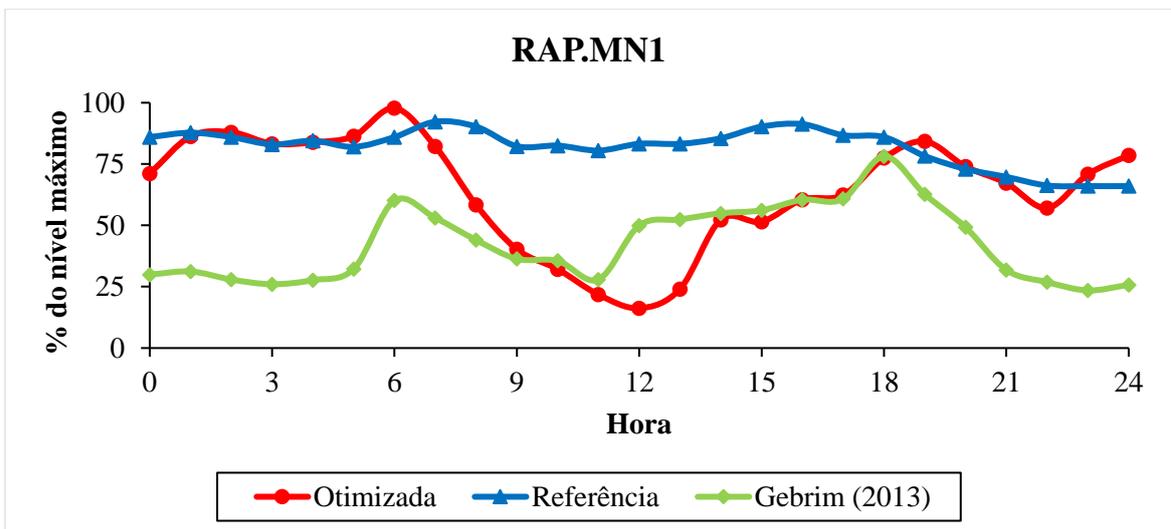


Figura 5.16 – Trajetória dos níveis do reservatório M Norte 1

Reservatórios como o RAP.GA2 do Gama apresentaram grande oscilação para todas as regras. Entretanto, a regra otimizada demonstrou elevado aproveitamento da capacidade do reservatório, pois a válvula (TCV-08) localizada à montante do reservatório é acionada

diversas vezes ao dia (Figura 5.13), situação semelhante ocorre no caso da regra de Gebrim (2013). A utilização da capacidade do reservatório (Figura 5.16 e Figura 5.17) é vantajosa no aspecto de qualidade da água a ser distribuída por proporcionar uma maior recirculação e, conseqüentemente, concentração de cloro adequada no reservatório (Carrijo, 2004).

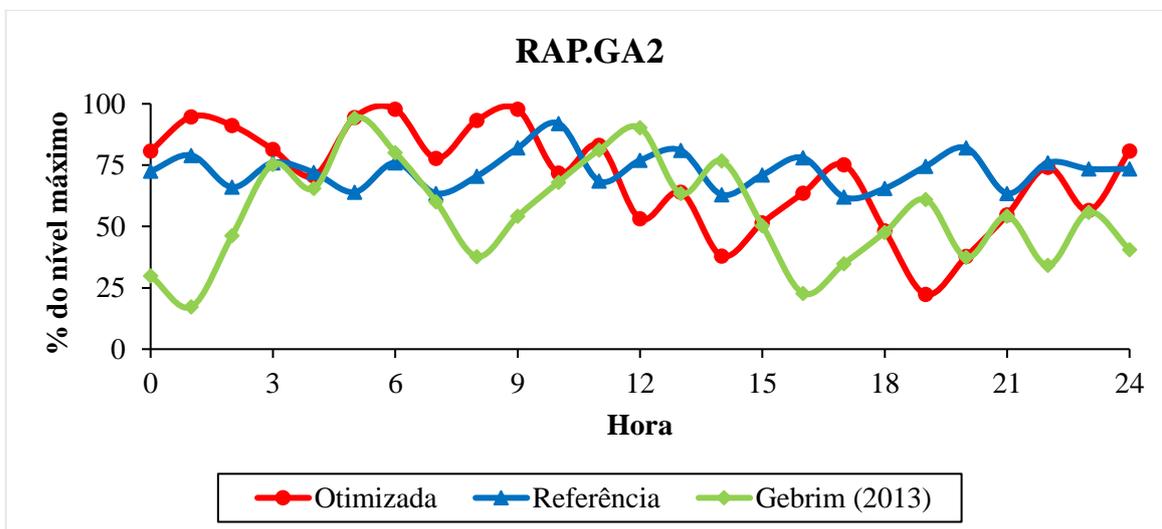


Figura 5.17 – Trajetória dos níveis do reservatório Gama 2

Pelas Figura 5.18 e Figura 5.19, reservatórios de Samambaia e de M Norte 2, respectivamente, é possível observar que a solução 4 (regra otimizada) apresenta maiores níveis dos reservatórios que a regra de referência e a regra de Gebrim (2013). A capacidade dos reservatórios é melhor aproveitada, mas evitando o esvaziamento.

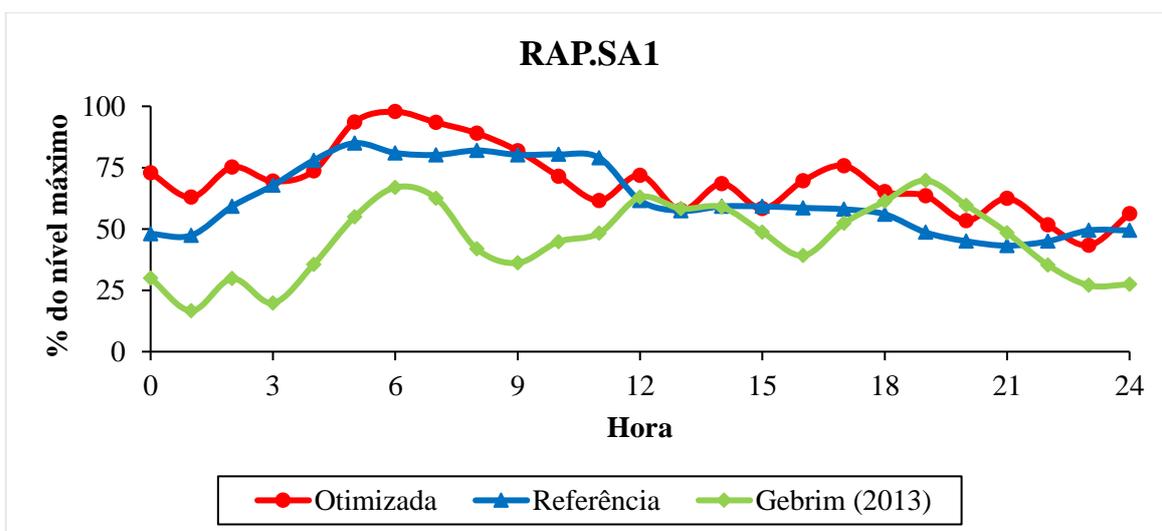


Figura 5.18 – Trajetória dos níveis do reservatório Samambaia

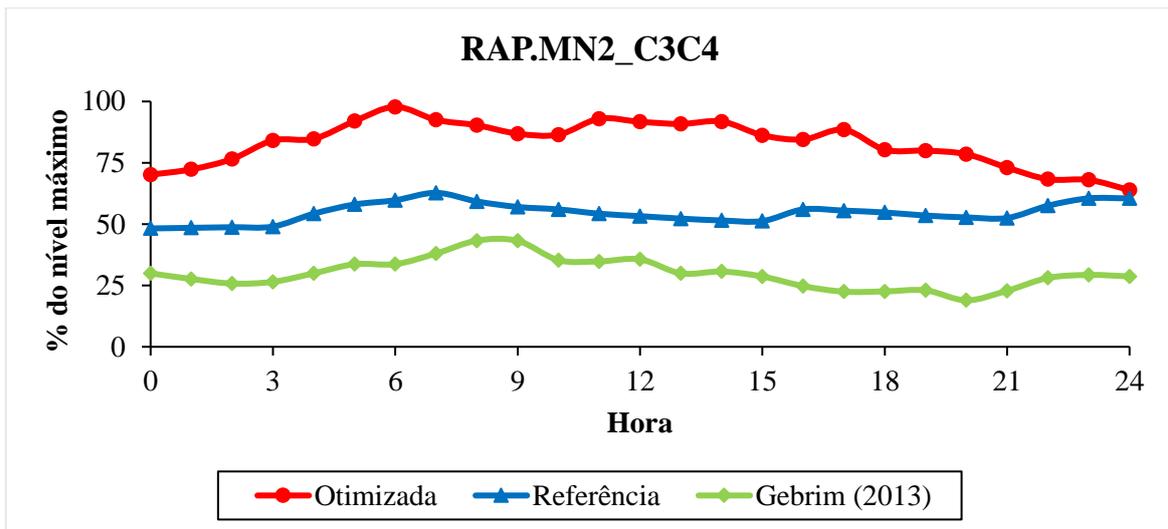


Figura 5.19 – Trajetória dos níveis do reservatório M Norte 2

A importância da segurança operacional para os operadores é clara com a manutenção dos elevados níveis nos reservatórios na regra de referência. Elevados valores de níveis na operação otimizada também ocorreram na maioria dos reservatórios em decorrência da simulação iniciar com o reservatório praticamente cheio, por imposição da penalidade.

A simulação que resultou na regra otimizada foi iniciada às 6 horas e, com isso, os níveis iniciais na hora 0 são diferentes para a regra otimizada, assim como na regra de referência, demonstrando que a condição operacional ideal de níveis iniciais não ocorre com todos os reservatórios na mesma proporção de armazenamento, e que o arranjo de níveis iniciais que geram uma operação otimizada deve ser analisado mais detalhadamente, em conjunto com critérios de confiabilidade operacional (Gebrim, 2013).

Os níveis dos reservatórios gerados pela regra otimizada estão dentro dos limites operacionais, no entanto é importante mencionar que a utilização de diferentes abordagens para as demandas como demandas históricas, demandas previstas, demandas observadas ou demandas médias, geram diferentes resultados provenientes da otimização como mostrado no estudo de Cunha (2009).

Apesar dos problemas encontrados no modelo, como excesso de acionamentos e da otimização ser de objetivo único, é possível a utilização do modelo para o planejamento operacional do sistema e posteriormente a utilização para a otimização em tempo real, já que o tempo computacional foi reduzido consideravelmente se comparado ao tempo

demonstrado no trabalho de Gebrim (2013) que variou entre 3 a 73 horas, enquanto o modelo do presente trabalho utilizou entre 1,09 a 5,15 horas, para o modelo de otimização que emprega o AG na linguagem C++.

O modelo foi capaz de gerar regras com elevada segurança na operação dos reservatórios, o que pode encorajar sua utilização pelos operadores do sistema, além disso, poderá ser utilizado futuramente na geração de regras após a inserção do sistema Corumbá IV, que está em fase de construção, e será interligado ao sistema Descoberto, para suprir sua demanda, considerando o cenário hídrico atual no Distrito Federal. Com a nova configuração, as regras operacionais hoje utilizadas terão que ser adaptadas e soluções otimizadas por simulação podem servir de base para a operação real sem experiência prévia.

6 - CONCLUSÕES E RECOMENDAÇÕES

O presente trabalho teve como principal objetivo a melhoria no desempenho, em termos de qualidade da solução otimizada e tempo de processamento do modelo de otimização de operação de bombas em redes de distribuição de água proposto por Gebrim (2013), utilizando o sistema de abastecimento do Rio Descoberto, localizado no Distrito Federal, como caso de estudo.

Para a otimização da operação do SAA do Rio Descoberto foram testados os algoritmos de otimização AG, DDS e DE, além de simplificação do modelo hidráulico, utilização de uma abordagem de penalização da função aptidão baseada na lógica *fuzzy* e emprego da técnica *seeding*.

A implementação do código na linguagem MATLAB não resultou em ganhos no tempo de processamento do modelo, já que o MATLAB é uma linguagem interpretada, logo ela necessita de um tempo adicional de processamento se comparado as linguagens compiladas como o C++. No entanto permitiu a utilização direta de algoritmos já disponíveis na plataforma.

Na comparação dos algoritmos de otimização, verificou-se que as aptidões não apresentaram diferenças significativas, porém o DDS levou à melhor solução, seguido do AG. O tempo de processamento do AG foi inferior, em decorrência da linguagem de programação utilizada. Apesar do maior tempo computacional utilizado pelo DDS, esse algoritmo possui elevado potencial para a otimização da operação de SAAs, pois necessita de apenas um parâmetro o que leva à facilidade na utilização. A solução encontrada pelo DE foi a menos favorável. No entanto, o tempo de processamento com esse algoritmo foi menor do que com o DDS, o que demonstra que ele pode ser utilizado para problemas onde deseja-se reduzir o tempo de processamento do modelo de otimização.

A simplificação do modelo hidráulico reduziu o tempo de processamento em 20%, se comparado ao modelo completo. Isso pode ser importante visto que parte considerável do tempo de processamento é gasto nas simulações hidráulicas. Entretanto, a simplificação gerou imprecisões nas simulações subsequentes, como na simulação da regra de referência.

O tratamento das penalidades utilizando a abordagem *fuzzy* se mostrou adequado e satisfatório para o problema, se comparado com a simulação sem tratamento, para a obtenção de regras operacionais que reduziram os indicadores de penalidades, os custos e os tempos de processamento, com exceção do indicador *diferença entre o nível inicial e final acima do limite*.

A redução nos tempos de processamento nas simulações com tratamentos das penalidades provavelmente ocorreu porque as soluções geradas necessitaram de menor número de iterações para resolver as equações hidráulicas de conservação de massa e energia do simulador hidráulico, sendo esta uma questão essa que precisa ser melhor explorada.

A utilização da técnica *seeding* de inicialização melhorou a qualidade da solução otimizada e ainda reduziu consideravelmente o tempo de processamento da otimização, mostrando a influência direta da solução inicial no processo de evolução da otimização. No entanto, se a solução inicial não contiver a informação “genética” necessária para o avanço da otimização, o algoritmo pode sofrer convergência prematura e ficar preso em mínimos locais.

A regra operacional obtida com a otimização reduziu os custos em 5,3% em comparação com a simulação da regra de referência e ainda levou a níveis elevados nos reservatórios, garantindo, assim, segurança operacional dos reservatórios, o que pode encorajar a utilização do modelo na operação real. No entanto, se comparado aos resultados de Gebrim (2013), os custos foram maiores, isso ocorreu devido às alterações realizadas no modelo atual e as diferenças nas simulações.

O tempo computacional do modelo de otimização utilizando o AG em C++, variou entre 1,09 e 5,15 horas, enquanto o modelo anterior utilizou entre 3 e 73 horas, pois o modelo atual necessita de um número de avaliações inferior para encontrar uma solução considerada boa. No entanto, o número de acionamentos de bombas e válvulas foi elevado, situação não desejável na operação real, sendo um problema que também ocorria no modelo de Gebrim (2013) e precisa ser melhor explorado.

Apesar da otimização realizada não ser multiobjetivo percebe-se que a utilização do modelo para operação real pode ser considerada, já que ele se aproxima da regra de referência em

aspectos como os níveis dos reservatórios, pressões que garantem o abastecimento e necessita de um tempo de processamento relativamente pequeno, considerando a complexidade do sistema e o número de avaliações necessário para encontrar uma boa solução. No entanto, antes da utilização real seria necessário o emprego de técnicas para a redução dos acionamentos e desenvolvimento de um previsor de demanda, aqui considerada conhecida.

Com as simulações do modelo de otimização observou-se que o simulador hidráulico tradicional EPANET2 é responsável por praticamente todo o tempo de simulação, logo sugere-se a utilização de metamodelos em substituição ao simulador hidráulico tradicional como mostrado por Broad *et al.* (2010); Rao e Salomons (2007) e Martinez *et al.* (2007).

Sugere-se a utilização de métodos de busca local integrados aos algoritmos de busca global como o AG e DE, a fim de melhorar a qualidade das soluções, além da utilização da abordagem *fuzzy* nas penalidades que poderá ser mais vantajosa na busca das soluções.

Pelas simulações dos cenários utilizando a abordagem *fuzzy*, notou-se que os resultados dos indicadores são sensíveis a pequenas alterações dos limites. Assim, é necessário o estudo de outras situações operacionais para verificar a eficácia da técnica, além da utilização de diferentes funções de pertinência e da comparação com outras abordagens de funções de penalidades para averiguar se essa metodologia é superior a técnicas mais sofisticadas como as penalidades auto-adaptativas (Farmani e Wright, 2003).

O modelo apresentou excesso de acionamentos acima do limite, então recomenda-se a utilização de técnicas para minimizar o número de acionamentos, como a mostrada por Odan (2013) que altera os estados das bombas, a partir da solução na otimização.

Embora na otimização tenham sido encontrados níveis elevados nos reservatórios, garantindo segurança na operação, verifica-se ainda necessidade da realização de um estudo de confiabilidade hidráulica do sistema. Também recomenda-se a obtenção de regras em situações de emergências operacionais.

Para a utilização do modelo de otimização para operação em tempo real é necessário o desenvolvimento de um modelo para previsão de demanda, pois a operação precisa se adaptar às alterações de demanda no caso de operação real.

Recomenda-se ainda, a inserção do sistema Corumbá IV ao modelo hidráulico, que será integrado ao sistema Descoberto, para a geração de regras operacionais otimizadas, antes da entrada em operação do novo sistema. Isso certamente auxiliará à CAESB no estabelecimento das regras operacionais do sistema como um todo.

REFERÊNCIAS BIBLIOGRÁFICAS

- Affenzeller, M. e Wagner, S. (2004). “The influence of population genetics for the redesign of genetic algorithm.” In: *Systems Science*, 30(4), 41–49.
- Alonso, J., Alvarruiz, F., Guerrero, D., Hernandez, V., Ruiz, P., Vidal, A., Martinez, F., Vercher, J. e Ulanicki, B. (2000). “Parallel computing in water network analysis and leakage minimization.” In: *Journal of Water Resources Planning and Management*, 126(4), 251-260
- ANA - Agência Nacional de Águas. (2010). *Região Metropolitana da RIDE DF*. Disponível em: <<http://atlas.ana.gov.br/Atlas/forms/analise/RegiaoMetropolitana.aspx?rme=1>> Acesso em 05 de março de 2016.
- Associação Brasileira de Normas Técnicas. NBR 12218: *Projeto de rede de distribuição de água para abastecimento público*. Rio de Janeiro, 1994.
- Balla, M. C. e Lingireddy, S. (2000). “Distributed genetic algorithm model on network of personal computers.” In: *Journal of computing in civil engineering*, 14(3), 199-205.
- Berhe, H. W. (2012). “Penalty function methods using matrix laboratory (MATLAB).” In: *African Journal of Mathematics and Computer Science Research*, 5(13), 209-246.
- Bezerra, S. T. M. (2009). *Sistema Fuzzy para Controle Piezométrico de Sistemas de Distribuição de Água Visando à Economia de Água e Energia*. Tese de Doutorado, Universidade Federal da Paraíba, Centro de Tecnologia, João Pessoa, PB, 142p.
- Bhave, P. R. e Gupta, R. (2004). “Optimal design of water distribution networks for fuzzy demands.” In: *Civil Engineering and Environmental Systems*, 21(4), 229–245.
- Boulos, P. F., Jacobsen, L. B., Heath, J.B. e Kamojjala, S. (2014). “Real-time modeling of water distribution systems: A case study.” In: *Journal American Water Works Association*, 391-401.
- Brion, L. M. e Mays, L. W. (1991). “Methodology for optimal operation of pumping stations in water distribution systems.” In: *Journal of Hydraulic Engineering*. 117(11), 1551–1569.
- Broad, D. R., Dandy G. C. e Maier H. R. (2005). “Water distribution system optimization using metamodels.” In: *Journal of Water Resources Planning and Management*, 131(3), 172-180.
- Broad, D. R., Maier, H. R. e Dandy, G. C. (2010). “Optimal operation of complex water distribution systems using metamodels.” In: *Journal of Water Resources Planning and Management*, 131(3), 172-180.
- Buehren, M. *Differential Evolution*. MATLAB Central. Disponível em: <<https://www.mathworks.com/matlabcentral/fileexchange/18593-differential-evolution>> Acesso em: 20 de nov. 2016.
- CAESB - Companhia de Saneamento Ambiental do Distrito Federal. (2012). *Relatório da Qualidade da Água Distribuída pela Caesb em 2011*. 1-7, Brasília, DF.

- CAESB - Companhia de Saneamento Ambiental do Distrito Federal. (2014). *SIÁGUA – Sinopse do Sistema de Abastecimento de Água do Distrito Federal*. CAESB/DP/PPC, 22ed, Brasília, 138p.
- Camboim, W. L. L. (2012). *Aplicações de Técnicas de Controle Fuzzy para Sistemas de Abastecimento de Água visando melhoria no rendimento energético e hidráulico*. Tese de Doutorado, Universidade Federal da Paraíba, Centro de Tecnologia, João Pessoa, PB, 124p.
- Camboim, W. L. L., Silva, S.A. e Gomes, H. P. (2014). “Aplicação de técnicas Fuzzy no controle de pressão em sistemas de abastecimento de água.” In: *Revista Engenharia Sanitária e Ambiental*. 19(1), 67-77.
- Carrijo, I. B. (2004). *Extração de regras operacionais ótimas de sistemas de distribuição de água através de algoritmos genéticos multiobjetivo e aprendizado de máquina*. Tese de Doutorado, Universidade de São Paulo, Escola de Engenharia de São Carlos, São Carlos, SP, 217p.
- Carrijo, I. B. e Reis, L.F.R. (2006). “Extração de regras operacionais ótimas de sistemas de distribuição de água através de algoritmos genéticos multiobjetivo e aprendizado de máquina – Aplicação ao Sistema de Macro-distribuição de Água de Goiânia.” In: *Revista Brasileira de Recursos Hídricos*. 11(2), 161-172.
- Carvalho, E. C. R. (2014). *Solução de problemas de otimização com restrições usando estratégias de penalização adaptativa e um algoritmo do tipo PSO*. Dissertação de Mestrado, Universidade Federal de Juiz de Fora, Juiz de Fora, MG, 148p.
- Cheung, P. B. (2004). *Análise de reabilitação de redes de distribuição de água para abastecimento via algoritmos genéticos multiobjetivo*. Tese de Doutorado, Universidade de São Paulo, Escola de Engenharia de São Carlos, São Carlos, SP, 268p.
- Coello, C. A. C. (1999). “A Comprehensive survey of evolutionary-based multiobjective optimization techniques.” In: *Knowledge and Information Systems*. 1(3), 269–308.
- Coello, C. A. C. (2002). “Theoretical and numerical constraint handling techniques used with evolutionary algorithms: a survey of the state of the art.” In: *Computer Methods in Applied Mechanics and Engineering*, 191(11-12), 1245-1287.
- Coello, C. A. C. (2006). “Evolutionary Multi-objective optimization: A historical view of the field.” In: *IEEE Computational Intelligence Magazine*, 1(1), 28-36.
- Cunha, A. A. R. (2009). *Otimização Energética em Tempo Real da Operação de Sistemas de Abastecimento de Água*. Dissertação de Mestrado, Universidade de São Paulo, Escola de Engenharia de São Carlos, São Carlos, SP, 179p.
- Dandy, G. C., Simpson, A. R., e Murphy, L. J. (1996). “An improved genetic algorithm for pipe network optimization.” In: *Water Resources Research*, 32(2), 449-457.
- Dandy, G. C., Wilkins, A. e Rohrlach, H. (2010). “A methodology for comparing evolutionary algorithms for optimizing water distribution systems.” In: *12th Water Distribution System Analysis Symposium*, 786-798, Tucson, Arizona, USA.

- Deb, K. (2000). "An efficient constraint handling method for genetic algorithms." In: *Computer Methods in Applied Mechanics and Engineering*, 186(2-4), 311-338.
- Deb, K. (2011). "Multi-objective optimization using evolutionary algorithms: An introduction." *KanGAL Report*, Number 2011003. Disponível em: <<http://www.iitk.ac.in/kangal/papers/k2011003.pdf>> Acesso em 20 de mar. 2016.
- Eliades, D. G. e Kyriakou, M. (2016). *EPANET-MATLAB-Toolkit*. Disponível em: <<https://github.com/OpenWaterAnalytics/EPANET-Matlab-Toolkit>> Acesso em: 20 ago. 2016.
- Eusuff, M. M. e Lansey, K. E. (2003). "Optimization of water distribution network design using the shuffled frog leaping algorithm." In: *Journal of Water Resources Planning and Management*, 129(3), 210–225.
- Farmani, R. e Wright, J. A. (2003). "Self-adaptive fitness formulation for constrained optimization," In: *IEEE Transactions on Evolutionary Computation*, 7(5), 445–455.
- Formiga, K.T.M. e Chaudhry, F.H. (2002). "Utilização de algoritmos evolucionários multiobjetivos no dimensionamento de redes de abastecimento de água." In: *Seminário de Planejamento, Projeto e Operação de Redes de Abastecimento de Água: O Estado da Arte e Questões Avançadas*, João Pessoa, Brasil.
- Gandibleux, X., Morita, H. e Katoh, N. (2001). "The Supported Solutions Used as a Genetic Information in a Population Heuristic." In: Zitzler, E., Deb, K., Thiele, L., Coello Coello, C.A., Corne, D.W. (eds.) *Computer Science*, Heidelberg, Berlin.
- Gargano, R., e Pianese, D. (2000). "Reliability as tool for hydraulic network planning." In: *Journal of Hydraulic Engineering*, 126(5), 354-364.
- Gebrim, D.V.B. (2013). *Otimização operacional de sistemas de abastecimento de água com objetivo de redução de custo de energia elétrica*. Dissertação de Mestrado, Universidade de Brasília, Departamento de Engenharia Civil e Ambiental, Brasília, DF, 120p.
- Geem, Z. W. (2006). "Comparison harmony search with other meta-heuristics in water distribution network design." In: *8th Annual Water Distribution Systems Analysis Symposium*, 1-9, Cincinnati, Ohio, E.U.A.
- Geem, Z. W., Kim, J. H. e Loganathan, G. V. (2001). "A New Heuristic Optimization Algorithm: Harmony Search." In: *Simulation*, 76(2), 60-68.
- Geem, Z. W., Kim, J. H. e Loganathan, G. V. (2002). "Harmony search optimization: application to pipe network design." In: *Int. J. Model. Simulat.*, 22(2), 125-133.
- Geem, Z. W. (2015). "Multiobjective Optimization of Water Distribution Networks Using Fuzzy Theory and Harmony Search." In: *Water.*, 7, 3613-3625.
- Glover, F. e Kochenberger, G. *Handbook of Metaheuristics*. Kluwer Academic Publishers, USA, 556p.

- Gomes, H. P. e Carvalho, P. S. O. (2012). *Manual de sistemas de bombeamento – eficiência energética*. Editora Universitária, João Pessoa, PB, Brasil, 189p.
- Gupta, I. (1969). “Linear programming analysis of water supply system.” In: *AIIE Trans.*, 1(1), 56-61.
- Homaifar, A., Lai, S. H. Y. e Qi, Z. (1994). “Constrained optimization via genetic algorithms.” In: *Simulation*, 62, 242-254.
- Jameel, A. F. e Radhi, A. Z. (2014). “Penalty Function Method For Solving Fuzzy Nonlinear Programming Problem.” In: *International Journal of Recent Research in Mathematics Computer Science and Information Technology*, 1(1), 1-13.
- James, K., Campbell, S. L. e Godlove, C. E. (2002). *Água e energia: aproveitando as oportunidades de eficientização de água e energia não exploradas nos sistemas municipais*. Alliance – Aliança para Conservação de Energia, Washington, E.U.A., 160p.
- Jamieson, D.G., Shamir, U., Martinez, F. e Franchini, M. (2007). “Conceptual design of a generic, real-time, near-optimal control system for water-distribution networks.” In: *Journal of Hydroinformatics*, 9(1), 3-14.
- Jowitt, P.W. e Germanopoulos, G. (1992). “Optimal pump scheduling in water-supply networks.” In: *Journal of Water Resources Planning and Management*, 118(4), 406-422.
- Juliff, K. (1993). “A multi-chromosome genetic algorithm for pallet loading.” In: *Fifth International Conference on Genetic Algorithms*, San Mateo, CA, USA.
- Jung, B. e Karney, B. (2006). “Hydraulic Optimization of Transient Protection Devices Using GA and PSO Approaches.” In: *Journal of Water Resources Planning and Management*, 132(1), 44-52.
- Karmeli, D., Gadish. Y. e Meyers. S. (1968). “Design of optimal water distribution networks.” In: *J. of Pipeline Div.*, 94(1), 1-9.
- Kusakci, A. O. e Can, M. (2012). “Constrained Optimization with Evolutionary Algorithms: A Comprehensive Review.” In: *Southeast Europe Journal of Soft Computing*, 1(2), 16-24.
- Lacerda, I. S., Lucena, K. F. M., Galvão, C. O. (2015). “Regras difusas para sistemas adutores sujeitos a incertezas nas demandas.” In: *XXI Simpósio Brasileiro de Recursos Hídricos*, 1-8, Brasília, DF, Brasil.
- Lacerda, I. S., Lucena, K. F. M., Galvão, C. O., Machado, E. C. M. N. (2008). “Construção de regras para operação multiobjetivo de sistemas de abastecimento de água.” In: *IX Simpósio de Recursos Hídricos do Nordeste*, 1-18, Salvador, BA, Brasil.
- Lansley, K. E. (2006). “The evolution of optimizing water distribution system applications.” In: *8th Annual Water Distribution Systems Analysis Symposium*, 1-20, Cincinnati, Ohio, USA.

- Lansey, K.E. e Awumah, K. (1994). “Optimal pump operations considering pump switches.” In: *Journal of Water Resources Planning and Management*, 120(1), 17-34.
- Le Riche, R. G., Knopf-Lenoir, C. e Haftka, R. T. (1995). “A segregated genetic algorithm for constrained structural optimization.” In: *Sixth International Conference on Genetic Algorithms*, Pittsburgh, PA, USA.
- Lin, M. D., Liu, Y. H., Liu, G. F. e Chu, C. W. (2007). “Scatter search heuristic for least-cost design of water distribution networks.” In: *Journal Engineering Optimization*, 39(7), 857-876.
- López-Ibáñez, M. (2009). *Operational Optimisation of Water Distribution Networks*. Tese de Doutorado, Edinburgh Napier University, School of Engineering and the Built Environment, UK, 215p.
- Luke, S. (2013). *Essentials of Metaheuristics*. Lulu, 2 ed, California, USA, 261p.
- Machado, E. C. N., Cruz, C. A., Rêgo, M. G., Carvalho, M. W. A., Lacerda, I. S. e Galvão, C. O. (2008). “Operação multi-objetivo de sistemas de abastecimento de água: algoritmo genético.” In: *VIII Seminário Hispano-Brasileiro sobre Sistemas de Abastecimento Urbano de Água*, Lisboa, Portugal.
- Mackle, G., Savic, D. A. e Walters, G. A. (1995). “Application of Genetic Algorithms to Pump Scheduling for Water Supply”. In: *Genetic Algorithms in Engineering Systems: Innovations and Applications*, 400-405, Sheffield, UK.
- Maier, H.R., Kapelan, Z., Kasprzyk, J., Kollat, J., Matott, L.S., Cunha, M.C., Dandy, G.C., Gibbs, M. S., Keedwell, E., Marchi, A., Ostfeld, A., Savic, D., Solomatine, D.P., Vrugt, J.A., Zecchin, A.C., Minsker, B.S., Barbour, E.J., Kuczera, G., Pasha, F., Castelletti, A., Giuliani, M. e Reed, P.M. (2014). “Evolutionary algorithms and other metaheuristics in water resources: Current status, research challenges and future directions.” In: *Environmental Modelling & Software*, 62, 271-299.
- Maier, H.R., Simpson, A.R., Zecchin, A.C., Foong, W.K., Phang, K., Seah, H. e Tan, C. (2003). “Ant colony optimization for design of water distribution systems.” *Journal Water Resources Planning and Management*, 129(3), 200–209.
- Marchi, A., Dandy, G., Wilkins, A., Rohrlach, H. (2014). “Methodology for comparing evolutionary algorithms for optimization of water distribution systems.” In: *Journal of Water Resources Planning and Management*, 140(1), 22-31.
- Martinez, F., Hernandez, V., Alonso, J. M., Rao, Z. e Alvisi, S. (2007). “Optimizing the operation of the Valencia water-distribution network.” In: *Journal of Hydroinformatics*, 9(1), 65–78.
- Michalewicz, Z. (1995). “Genetic Algorithms, Numerical Optimization, and Constraints.” In: Eshelman, L. J. (ed.), *Sixth International Conference on Genetic Algorithms*, Pittsburgh, PA, USA.
- Michalewicz, Z. e Schoenauer, M. (1996). “Evolutionary algorithms for constrained parameter optimization problems.” In: *Evolutionary Computation*, 4(1), 1–32.

- Michalewicz, Z., Dasgupta, D., Le Riche, R. G. e Schoenauerx, M. (1996). “Evolutionary Algorithms for Constrained Engineering Problems.” In: *Computers & Industrial Engineering*, 30(4), 851-870.
- Mohan, S. e Babu, K.S.J. (2010). “Optimal water distribution network design with honey-bee mating optimization.” In: *Journal of Computing in Civil Engineering*, 24(1), 117-126.
- Montalvo, I., Izquierdo, J., Schwarze, S. e Pérez-García, R. (2010). “Multi-objective particle swarm optimization applied to water distribution systems design: An approach with human interaction.” *Mathematical and Computer Modelling*, 52, 1219-1227.
- Morgan, D. e Goulter, I. (1985). “Optimal urban water distribution design.” In: *Water Resources Research*, 21(5), 642-652.
- Nicklow, J.W., Reed, P.M., Savic, D., Dessalegne, T., Harrell, L., Chan-Hilton, A., Karamouz, M., Minsker, B., Ostfeld, A., Singh, A., Zechman, E. (2010). “State of the art for genetic algorithms and beyond in water resources planning and management.” In: *Journal of Water Resources Planning and Management*, 136(4), 412-432.
- Nitivattananon, V., Sadowski, E.C. e Quimpo, R.G. (1996). “Optimization of water supply system operation.” In: *Journal of Water Resources Planning and Management*, 122(5), 374-384.
- Odan, F. K. (2010). *Previsão de Demanda para Sistemas de Abastecimento de Água*. Dissertação de Mestrado, Universidade de São Paulo, Escola de Engenharia de São Carlos, São Carlos, SP, 77p.
- Odan, F. K. (2013). *Estudo de Confiabilidade aplicado à Otimização da Operação em Tempo Real de Redes de Abastecimento de Água*. Tese de Doutorado, Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, SP, 210 p.
- Odan, F.K., Reis, L.F.R. e Kapelan, Z. (2014). “Use of metamodels in real -time operation of water distribution systems.” In: *Procedia Engineering*, 89, 449 – 456.
- Odan, F.K., Reis, L.F.R. e Kapelan, Z. (2015). “Real-time multiobjective optimization of operation of water supply systems.” In: *Journal of Water Resources Planning and Management*, 141(9).
- Oliveira, A. C. M. (2001). *Algoritmos evolutivos para problemas de otimização numérica com variáveis reais*. Trabalho de conclusão de curso, Instituto Nacional de Pesquisas Espaciais, São José dos Campos, SP, 52p.
- Ormsbee, L.E. e Lansey, K.E. (1994). “Optimal control of water supply pumping systems.” In: *Journal of Water Resources Planning and Management*, 120(2), 237-252.
- Ormsbee, L.E. e Reddy, S.L. (1995). “Nonlinear heuristic for pump operations.” In: *Journal of Water Resources Planning and Management*, 121(4), 302-309.
- Ormsbee, L.E., Walski, T.M., Chase, D.V. e Sharp, W.W. (1989). “Methodology for improving pump operation efficiency.” In: *Journal of Water Resources Planning and Management*, 115(2), 148-164.

- Ostfeld, A. e Tubaltzev, A. (2008). “Ant colony optimization for least-cost design and operation of pumping water distribution systems.” In: *Journal of Water Resources Planning and Management*, 134(2), 107-118.
- Patton, A. L., Punch III, W. F. e Goodman, E. D. (1995). “A standard GA approach to native protein conformation prediction.” In: *Sixth International Conference on Genetic Algorithms*, Pittsburgh, PA, USA.
- Pedrosa, L. A. F. (2006). *Um Modelo de Operação de Sistemas Adutores de Abastecimento de Água com vistas a Minimização dos Custos Energéticos*. Tese de Doutorado, Universidade Federal de Campina Grande, Centro de ciências e tecnologia, Campina Grande, PB, 174p.
- Perelman, L., e Ostfeld, A. (2007). “An adaptive heuristic cross-entropy algorithm for optimal design of water distribution systems.” In: *Engineering Optimization*, 39(4), 413 - 428.
- Pezeshk, S., Helweg, O.J., Oliver, K.E. (1994). “Optimal operation of ground-water supply distribution systems.” In: *Journal of Water Resources Planning and Management*, 120(5), 573-586.
- Poles, S., Fu, Y. e Rigoni, E. (2009). “The Effect of Initial Population Sampling on the Convergence of Multi-Objective Genetic Algorithms.” In: Barichard V., Ehr Gott M., Gandibleux X., T'Kindt V. (eds.) *Economics and Mathematical Systems*, Heidelberg, Berlin.
- Porto, R.M. (2006). *Hidráulica Básica*. 4. ed. EESC/USP, São Carlos.
- Rao, Z. e Salomons, E. (2007). “Development of a real-time, near optimal control process for water-distribution networks.” In: *Journal of Hydroinformatics*, 9(1), 25–38.
- Ribeiro, G.P. (2005). *Operação Otimizada do Sistema Adutor Metropolitano Utilizando Algoritmos Genéticos*. Estudo de caso: SAM Leste da Região Metropolitana de São Paulo (R.M.S.P.). Tese de Doutorado, Universidade de São Paulo, Escola Politécnica, São Paulo, SP, 239p.
- Righetto, A.M. (2002). “Operação Ótima de Sistema Urbano de Distribuição de Água.” *Anais do Seminário-Planejamento, Projeto e Operação de Redes de Abastecimento de Água: O Estado da Arte e Questões Avançadas*, 1-16, João Pessoa, PB.
- Revelli, R. e Ridolfi, L. (2002). “Fuzzy approach for analysis of pipe networks.” In: *Journal of Hydraulic Engineering*, 128(1), 93-10.
- Saha, C., Das, S., Pal, K. e Mukherjee, S. (2016). “A Fuzzy Rule-Based Penalty Function Approach for Constrained Evolutionary Optimization.” In: *IEEE Transactions on Cybernetics*, 46(12), 2953 – 2965.
- Sakarya, A.B.A. e Mays, L.W. (2000). “Optimal operation of water distribution pumps considering water quality.” In: *Journal of Water Resources Planning and Management*, 126(4), 210–220.
- Savic, D. A., Walters, G. A. e Schwab, M. (1997). “Multiobjective genetic algorithms for pump scheduling in water supply.” In: *Computer Science*, 1305, 227–236.

- Savic, D.A. e Walters, G.A. (1997). “Genetic algorithms for least-cost design of water distribution networks.” In: *Journal of Water Resources Planning and Management*, 123(2), 67-77.
- Semeria, L. (2007). *Accelerating MATLAB using MEX-files*. EE Times. Disponível em: <http://www.eetimes.com/document.asp?doc_id=1275425> Acesso em: 09 de mar. 2017.
- Shamir, U. (1974). “Optimal design and operation of water distribution systems.” In: *Water Resources Research*, 10(1), 27-36.
- Shamir, U. e Salomons, E. (2008). “Optimal real-time operation of urban water distribution systems using reduced models.” In: *Journal of Water Resources and Management*, 134(2), 181-185.
- Simões, M. G. e Shaw, I. S. (2007). *Controle e Modelagem Fuzzy*. Editora Blucher, 2ed. São Paulo, SP, Brasil, 186p.
- Simpson, A. R., Dandy, G. C. e Murphy L. J. (1994). “Genetic Algorithms compared to other techniques for pipe optimization.” In: *Journal Water Resources Planning and Management*, 120(4), 423-443.
- Sivakumar, P., Prasad, R.K., Chandramouli, S., Majumder. S. (2014). “Uncertainty analysis of water distribution networks using linked EPANET-vertex method.” In: *International Journal of Innovative Research in Science, Engineering and Technology*, 3(12), 17900-17911.
- Smith, A. E. e Coit, D. W. (1997). Constraint Handling Techniques - Penalty Functions. In: Back, T., Fogel, D. B., Michalewicz, Z. (eds.). *Handbook of Evolutionary Computation*. Oxford University Press and Institute of Physics Publishing, Bristol, U.K.
- Sousa, A. C. e Soares, A. K. (2014). “Modelo para otimização da operação de sistemas de distribuição de água utilizando o algoritmo genético multiobjetivo SPEA.” In: *Exacta – EP*, 12(3), 313-324.
- Storn, R., S. e Price, K. (1997). “Differential Evolution: a simple and efficient heuristic for global optimization over continuous spaces.” In: *Journal of Global Optimization*, 11(4), 341–359.
- Suribabu, C. R. (2010). “Differential evolution algorithm for optimal design of water distribution networks.” In: *Journal of Hydroinformatics*, 12(1), 66-82.
- Suribabu, C.R. e Neelakantan, T.R. (2006). “Particle swarm optimization compared to other heuristic search techniques for pipe sizing.” In: *Journal of Environmental Informatics*, 8(1), 1-9.
- Tolson, B. A. (2005). *Automatic calibration, management and uncertainty analysis: Phosphorus transport in the Cannonsville watershed*. Tese de Doutorado, Cornell University, Ithaca, USA.
- Tolson, B. A. (2015). *Dynamically Dimensioned Search (DDS) algorithm*. Distribution package. 12p.

- Tolson, B. A., Asadzadeh, M., Zecchin, A. C. e Maier H. R. (2008). "A new algorithm for water distribution system optimization: Discrete dynamically dimensioned search." In: *World Environmental and Water Resources Congress*, 1–10, Honolulu, Hawaii, USA.
- Tolson, B.A. e Shoemaker, C.A. (2007). "Dynamically dimensioned search algorithm for computationally efficient watershed model calibration." In: *Water Resources Research*, 43(1), 1-16.
- Tsutiya, M.T. (2006). *Abastecimento de Água*. EPUSP, 3.ed, São Paulo, 643p.
- Van Le, T. (1995). "A Fuzzy Evolutionary Approach to Constrained Optimization Problems." In: *Second IEEE Conference on Evolutionary Computation*, Perth, Australia.
- Vasan, A. e Simonovic, S.P. (2010). "Optimization of water distribution network design using differential evolution." In: *Journal of Water Resources Planning and Management*, 136(2), 279-287.
- Vicente, R. V. (2005). *Modelo de operação para centros de controle de sistemas de abastecimento de água - estudo de caso: sistema adutor metropolitano de São Paulo*. Dissertação de Mestrado, Universidade de São Paulo, Escola Politécnica, São Paulo, SP, 163p.
- Walski, T.M., Chase, D.V. e Savic, D.A. (2003). *Water distribution modeling*. Haestad Methods, 1ed, Connecticut, USA, 441p.
- Walters, G.A. e Lohbeck, T.K. (1993). "Optimal layout of tree networks using genetic algorithms," In: *Engineering Optimization*, 22, 27-48.
- Wardlaw, R. e Sharif, M. (1999). "Evaluation of Genetic Algorithms for Optimal Reservoir System Operation." In: *Journal of Water Resources Planning and Management*, 125(1), 25-33.
- Wu, Y. Z. e Simpson, A. R. (2002). "A self-adaptive boundary search genetic algorithm and its application to water distribution systems." In: *Journal of Hydraulic Research*, 40(2), 191-203.
- Yeniay, O. (2005). "Penalty function methods for constrained optimization with genetic algorithms." In: *Mathematical and Computational Applications*, 10(1), 45-56.
- Yu, G., Powell, R.S., Sterling, M.J.H. (1994). "Optimized pump scheduling in water distribution systems." In: *Journal of Optimization Theory and Applications*, 83(3), 463-488.
- Zadeh, L.A. (1965). "Fuzzy Sets." *Information and Control*, 8, 338-353.
- Zessler, U. e Shamir, U. (1989). "Optimal operation of water distribution systems." In: *Journal of Water Resources Planning and Management*, 115(6), 735-752.
- Zheng, F., Simpson, A. R. e Zecchin, A. (2011) "Performance Study of Differential Evolution with Various Mutation Strategies Applied to Water Distribution System

Optimization.” In: *World Environmental and Water Resources Congress*, ASCE, Palm Springs, California, USA.

Zheng, F., Simpson, A.R. e Zecchin, A. (2012). “A performance comparison of differential evolution and genetic algorithm variants applied to water distribution system optimization.” In: *Anais do World Environmental and Water Resources Congress*, 2954-2963.