

MARCOS RIBEIRO QUINET DE ANDRADE

**HEURÍSTICAS PARA A SOLUÇÃO DO PROBLEMA DE
ROTEAMENTO PERIÓDICO DE UNIDADES MÓVEIS DE
PISTONEIO**

Tese de doutorado apresentada ao Programa de Pós-Graduação em Computação da Universidade Federal Fluminense, como requisito para obtenção do Grau de Doutor. Área de Concentração: Algoritmos e Otimização.

Orientadores: Prof. Dr. Luiz Satoru Ochi e Profa. Dr. Simone de Lima Martins

Niterói

2016

MARCOS RIBEIRO QUINET DE ANDRADE

**HEURÍSTICAS PARA A SOLUÇÃO DO PROBLEMA DE
ROTEAMENTO PERIÓDICO DE UNIDADES MÓVEIS DE PISTONEIO**

Tese de doutorado apresentada ao Programa de Pós-Graduação em Computação da Universidade Federal Fluminense, como requisito parcial para obtenção do Grau de Doutor. Área de Concentração: Algoritmos e Otimização.

Aprovada em Fevereiro de 2016.

BANCA EXAMINADORA

Prof. Luiz Satoru Ochi – Orientador - UFF

Profa. Simone de Lima Martins – Orientadora - UFF

Prof. Carlos Alberto de Jesus Martinhon- UFF

Profa. Isabel Cristina Mello Rosseti - UFF

Profa. Marcia Helena Costa Fampa - UFRJ

Prof. Virgílio José Martins Ferreira Filho - UFRJ

Niterói
2016

RESUMO

O problema de roteamento periódico de unidades móveis de pistoneio (PRP-UMP) consiste em, dado um conjunto de poços não-surgentes de óleo espalhados ao longo de um campo produtor terrestre, determinar rotas para cada dia de trabalho, com duração limitada ao turno de trabalho, que maximize a coleta de óleo realizada por uma ou mais unidades móveis de pistoneio. O objetivo deste trabalho é realizar o estudo da aplicação de diferentes métodos heurísticos neste problema NP-difícil, de aplicação prática na região nordeste do Brasil.

Diferentes métodos construtivos e de busca local foram desenvolvidos e utilizados na composição das heurísticas *Greedy Randomized Adaptive Search Procedure* (GRASP), *Iterated Local Search* (ILS) e *Simulated Annealing* (SA) para solução do PRP-UMP para uma e múltiplas UMP's.

Palavras-chave: Metaheurísticas, Problema de Roteamento de Veículos, Unidade Móvel de Pistoneio, Pistoneio de poços surgentes, GRASP, *Iterated Local Search*, *Simulated Annealing*.

ABSTRACT

The Problem of periodic mobile piston pump unit routing problem (PMPPURP) consists of, given a set of surface oil wells scattered over an onshore producing field, determining routes for each day of a planning horizon, with limited duration work shift, that maximize the collection of oil carried by one or more mobile piston pump units. The aim of this research is the study of the application of different heuristic methods in a NP-hard problem of practical application in northeastern Brazil.

Different construction and local search methods were developed and used in the composition of GRASP, Simulated Annealing and ILS heuristics for the problem using one or multiple Mobile Piston Pump Units. The results obtained demonstrate the effectiveness of the proposed methods.

Keywords: Metaheuristics, Vehicle Routing Problem, Surface Oil Wells Pumping, Mobile Piston Pump Unit, GRASP, Iterated Local Search, Simulated Annealing.

LISTA DE FIGURAS

Figura 2.1: Unidade Móvel de Pistoneio.....	15
Figura 2.2: Exemplo de rotas geradas por múltiplas Unidades Móveis de Pistoneio.	16
Figura 3.1: Metaheurística GRASP para o PRP-UMP.....	24
Figura 3.2: Algoritmo Construtivo C1	26
Figura 3.3: Exemplo de funcionamento do algoritmo Construtivo C1.	27
Figura 3.4: Algoritmo Construtivo C2.	28
Figura 3.5: Exemplo de funcionamento do algoritmo Construtivo C2.	29
Figura 3.6: Algoritmo Construtivo C3.	30
Figura 3.7: Exemplo de funcionamento do algoritmo Construtivo C3.	32
Figura 3.8: Algoritmo de Busca Local BL1.	33
Figura 3.9: Exemplo de funcionamento da busca local BL1.....	34
Figura 3.10: Exemplo de funcionamento da busca local BL2.....	35
Figura 3.11: Exemplo de funcionamento da busca local BL3.....	36
Figura 3.12: Algoritmo de Busca Local BL4.	37
Figura 3.13: Algoritmo de Busca Local BL5.	39
Figura 4.1: Metaheurística ILS.....	42
Figura 4.2: Metaheurística ILS para o PRP-UMP com uma UMP.	43
Figura 4.3: Metaheurística ILS para o PRP-UMP para múltiplas UMP's.	46
Figura 4.4: Perturbação P1.	48
Figura 4.5: Exemplo da aplicação da perturbação P1.	49
Figura 4.6: Exemplo da aplicação da perturbação P2.	50
Figura 4.7: Perturbação P3.	51
Figura 4.8: Exemplo da aplicação da perturbação P3.	52
Figura 4.9: Perturbação P4.	53
Figura 4.10: Perturbação P5.	54
Figura 4.11: Perturbação P5.	55
Figura 4.12: Perturbação P7.	56
Figura 4.13: Exemplo da operação 2-optimal da rota de uma UMP.	57
Figura 4.14: Perturbação P8.	57
Figura 4.15: Exemplo do operador de troca 1-0 entre as rotas de duas UMP's.	58
Figura 4.17: Exemplo do operador de troca 1-1 entre as rotas de duas UMP's.	59
Figura 4.16: Perturbação P9.	59

Figura 4.18: Perturbação P10.	60
Figura 4.19: Exemplo do operador de troca 2-optimal entre as rotas de duas UMP's.....	61
Figura 4.20: Metaheurística <i>simulated annealing</i> para o PRP-UMP.	64
Figura 4.21: Metaheurística GRASP com <i>simulated annealing</i> para o PRP-UMP com a utilização de uma UMP.	66
Figura 4.22: Metaheurística GRASP com <i>simulated annealing</i> para o PRP-UMP com a utilização de múltiplas UMP's.	69
Figura 5.1: Resultados do GRASP e das heurísticas híbridas empregando uma UMP para a obtenção de um alvo de dificuldade média	88
Figura 5.2: Resultados do GRASP e das heurísticas híbridas empregando uma UMP para a obtenção de um alvo de dificuldade alta.	88
Figura 5.3: Resultados das heurísticas híbridas empregando 5 UMP's para a obtenção de um alvo de dificuldade média.....	100
Figura 5.4: Resultados das heurísticas híbridas empregando 5 UMP's para a obtenção de um alvo de dificuldade alta.....	100

LISTA DE TABELAS

Tabela 5.1: Resultados computacionais do GRASP versões C3BL4 e C3BL5.	74
Tabela 5.2: Resultados computacionais do GRASP versões C3BL4 e C3BL5 para 200 iterações empregando $\alpha=0.1$ e $\alpha=0.05$	76
Tabela 5.3: Resultados computacionais do GRASP versões C3BL4 e C3BL5 limitadas por tempo de execução, empregando $\alpha=0.1$ e $\alpha=0.05$	78
Tabela 5.4: Resultados computacionais do GRASP com ILS, versões C3BL4 e C3BL5 limitadas a 200 execuções, empregando $\alpha=0.1$ e $\alpha=0.05$	81
Tabela 5.5: Resultados computacionais do GRASP com ILS, versões C3BL4 e C3LS5 com limite no tempo de execução, empregando $\alpha=0.1$ e $\alpha=0.05$	82
Tabela 5.6: Resultados computacionais do GRASP com <i>simulated annealing</i> (SA), versões C3BL4 e C3BL5, limitadas a 200 iterações, empregando $\alpha=0.1$ e $\alpha=0.05$	84
Tabela 5.7: Resultados computacionais do GRASP com <i>simulated annealing</i> (SA), versões C3BL4 e C3BL5, limitadas por tempo de execução, empregando $\alpha=0.1$ e $\alpha=0.05$	85
Tabela 5.8: Resultados computacionais do CPLEX e das heurísticas ILS e SA propostas (empregando C3BL5, $\alpha=0.05$), limitadas por tempo de execução.....	90
Tabela 5.9: Resultados computacionais do CPLEX e das heurísticas ILS e SA propostas (empregando C3BL5, $\alpha=0.05$), com tempo de execução de 7200 segundos.	92
Tabela 5.10: Resultados computacionais do CPLEX e das heurísticas ILS e SA propostas (empregando C3BL5, $\alpha=0.05$), com tempo de execução de 10800 segundos..	93
Tabela 5.11: Resultados computacionais do GRASP C3BL5 com ILS e SA, para múltiplas UMP's, limitado a 200 iterações.	96
Tabela 5.12: Tempos computacionais do GRASP C3BL5 com ILS e SA, para múltiplas UMP's, limitado a 200 iterações.	97
Tabela 5.13: Resultados computacionais do GRASP C3BL5 com ILS e SA, para múltiplas UMP's, com tempo de execução limitado.....	99
Tabela 5.14: Resultados computacionais do CPLEX e das heurísticas ILS e SA propostas empregando 5 UMP's, versão C3BL5, $\alpha=0.05$, limitadas por tempo de execução.....	102
Tabela 5.15: Resultados computacionais do CPLEX e das heurísticas ILS e SA propostas empregando 5 UMP's, versão C3BL5, $\alpha=0.05$, com tempo de execução limitado a 7200 segundos..	104

Tabela 5.16: Resultados computacionais do CPLEX e das heurísticas ILS e SA propostas empregando 5 UMP's, versão C3BL5, alfa=0.05, com tempo de execução limitado a 10800 segundos..	106
---	-----

LISTA DE EQUAÇÕES

Equação 2.1	18
Equação 2.2	18
Equação 2.3	18
Equação 2.4	18
Equação 2.5	18
Equação 2.6	18
Equação 2.7	18
Equação 2.8	18
Equação 2.9	18
Equação 2.10	18
Equação 2.11	20
Equação 2.12	20
Equação 2.13	20
Equação 2.14	20
Equação 2.15	20
Equação 2.16	20
Equação 2.17	20
Equação 2.18	20
Equação 2.19	20
Equação 2.20	21
Equação 2.21	21
Equação 2.22	21
Equação 2.23	21
Equação 2.24	21
Equação 2.25	21
Equação 2.26	21

Equação 2.27	21
Equação 3.1	25

SUMÁRIO

Capítulo 1 – Introdução	13
Capítulo 2 – O Problema do Roteamento Periódico de Unidades Móveis de Pistoneio.....	15
2.1 – Formulação matemática para uma UMP	17
2.2 – Formulação matemática para múltiplas UMP's	19
2.3 –Trabalhos relacionados	22
Capítulo 3 – Heurística GRASP para o PRP-UMP.....	24
3.1 – Métodos Construtivos Propostos para o PRP-UMP.....	25
3.1.1 – Construtivo C1.....	26
3.1.2 – Construtivo C2.....	28
3.1.3 – Construtivo C3.....	30
3.2 – Estratégias de busca local	32
3.2.1 – Busca Local BL1	33
3.2.2 – Busca Local BL2	34
3.2.3 – Busca Local BL3	35
3.2.4 – Busca Local BL4	37
3.2.5 – Busca Local BL5	39
Capítulo 4 – Heurísticas híbridas GRASP+ILS e GRASP+SA	41
4.1 – Iterated Local Search.....	41
4.1.1 – <i>Iterated Local Search</i> para o PRP-UMP considerando a utilização de uma UMP	43
4.1.2 – <i>Iterated Local Search</i> para o PRP-UMP considerando a utilização de múltiplas UMP's.....	45
4.2 – Estratégias de perturbação	47
4.2.1 – Perturbação P1	48
4.2.2 – Perturbação P2.....	49
4.2.3 – Perturbação P3.....	51
4.2.4 – Perturbação P4.....	53

4.2.5 – Perturbação P5	54
4.2.6 – Perturbação P6	55
4.2.7 – Perturbação P7	56
4.2.8 – Perturbação P8	57
4.2.9 – Perturbação P9	58
4.2.10 – Perturbação P10	60
4.3 – Estratégia reativa de seleção das perturbações	61
4.4 – Simulated Annealing	62
4.4.1 – Simulated annealing para o PRP-UMP considerando a utilização de uma UMP	65
4.4.2 – Simulated Annealing para o PRP-UMP considerando a utilização de múltiplas UMP's	68
Capítulo 5 – Resultados Computacionais	72
5.1 – Resultados obtidos com a utilização de uma UMP	72
5.1.1 – Resultados obtidos por meio do uso da heurística GRASP	73
5.1.2 – Comparação entre os resultados obtidos por meio dos métodos heurísticos híbridos	80
5.1.3 – Avaliação probabilística dos resultados empregando uma UMP	86
5.1.4 – Comparação entre os resultados obtidos por meio dos métodos heurísticos híbridos e os obtidos por meio de um método exato	89
5.2 – Resultados obtidos por meio da utilização de múltiplas UMP's	94
5.2.1 – comparação entre os resultados obtidos por meio dos métodos híbridos heurísticos	95
5.2.2 – Avaliação probabilística dos resultados empregando múltiplas UMP's	100
5.2.3 – Comparação entre os resultados obtidos por meio dos métodos heurísticos híbridos e os obtidos por meio de um método exato	101
Capítulo 6 – Conclusões	107
Referências	110

CAPÍTULO 1 – INTRODUÇÃO

Esta tese aborda uma aplicação prática do problema de roteamento de veículos (PRV), onde a partir de um depósito inicial, um ou mais veículos, pertencentes a uma frota homogênea, atendem um conjunto de clientes, sendo cada um destes visitado por um único veículo e somente uma vez, e, ao final das atividades, retornam ao depósito de origem, estabelecendo uma rota de custo mínimo para cada veículo para o atendimento ao conjunto de clientes.

Uma variante do PRV conhecida na literatura como o problema de roteamento periódico de veículos (PRPV) consiste em planejar roteiros otimizados para uma frota de veículos num dado período (horizonte de planejamento).

O problema tratado neste trabalho é uma aplicação prática para a versão periódica do PRV que consiste na resolução do problema de construção de rotas otimizadas para unidades móveis de pistoneio (UMP's) em campos terrestres contendo poços não-surgentes, dentro de um horizonte de planejamento de atividades previamente estabelecido denominado problema de roteamento periódico de unidades móveis de pistoneio (PRP-UMP) [Aloise *et al*, 2001; Goldberg *et al*, 2004, 2010; Gonçalves *et al*, 2005; Nascimento, 2010].

Dentre as metaheurísticas existentes na literatura, GRASP (*Greedy Randomized Adaptive Search Procedure*) [Feo e Resende, 1989], [Resende e Ribeiro, 2003] tem se mostrado como um dos métodos mais eficientes para a resolução de diversos problemas de otimização combinatória [Layeb *et al*, 2013]. As técnicas de *Iterated Local Search* (ILS) [Lourenço *et al*, 2003] e *Simulated Annealing* (SA) [Kirkpatrick *et al*, 1983; Cerny, 1985; Kokubugata, H. & Kawashima, H., 2008] são empregadas com sucesso como métodos de otimização global às soluções geradas pelo GRASP [Junior *et al*, 2010; Zeferino *et al*, 2011].

Neste contexto, propomos novos algoritmos heurísticos usando conceitos de GRASP, ILS e SA para a resolução do PRP-UMP empregando uma ou múltiplas UMP's, cujo objetivo é coletar o maior volume de óleo possível para um horizonte de planejamento de atividades. Para isso, são apresentadas diferentes versões de algoritmos construtivos e de busca local utilizados na implementação do *framework* GRASP e métodos de perturbação empregados nos *frameworks* do ILS e do SA, além de utilizarmos métodos exatos baseados em formulações matemáticas da literatura [Gonçalves *et al*, 2005], [Duarte, 2006] usando o programa de otimização CPLEX [IBM, 2016], para a resolução do PRP-UMP empregando uma e múltiplas UMP's.

O presente trabalho está organizado da seguinte forma. O Capítulo 2 descreve brevemente o problema abordado e uma revisão dos principais trabalhos da literatura relacionados ao problema. O Capítulo 3 apresenta a heurística GRASP proposta para o PRP-UMP. No Capítulo 4 são descritas as versões das heurísticas híbridas GRASP+ILS, e GRASP+SA e os operadores de perturbação propostos, empregando uma ou múltiplas UMP's. No Capítulo 5 apresentam-se os resultados computacionais, e no Capítulo 6, as conclusões obtidas e as etapas futuras de pesquisa.

CAPÍTULO 2 – O PROBLEMA DO ROTEAMENTO PERIÓDICO DE UNIDADES MÓVEIS DE PISTONEIO

O problema de roteamento de veículos pode ser aplicado em áreas petrolíferas terrestres, um cenário encontrado em diversas regiões do mundo, e que no Brasil pode ser observado na região nordeste, entre os estados do Rio Grande do Norte e Ceará, em uma localidade denominada Bacia Potiguar. Trata-se de uma área conhecida na exploração de petróleo em campos terrestres, classificados como *surgentes*, quando a pressão interna dos gases é suficiente para expelir o óleo até a superfície, e *não-surgentes*, nas situações que se torna necessária a utilização de um equipamento mecânico para a coleta do óleo.

O fator determinante na decisão do método de exploração de poços não-surgentes é a vazão apresentada por esse. Quando essa vazão é elevada, justifica-se a instalação de um equipamento de coleta fixo na localização de cada poço. Em poços de menor produtividade, para os quais a instalação deste tipo de estrutura não se justifica financeiramente, mas existe um potencial produtivo dos poços que não deve ser desprezado, adota-se um sistema de coleta utilizando unidades móveis de pistoneio (UMP's).

Uma UMP consiste em um caminhão adaptado, com um tanque de armazenamento e todo o maquinário necessário para a operação de pistoneio por elevação, ilustrado pela Figura 2.1, que pode ser colocado em funcionamento em um curto espaço de tempo. Ao completar a coleta de óleo de um poço, a unidade encontra-se livre para seguir em direção a outro poço do campo e repetir a operação, enquanto o poço recém-visitado começa seu processo de reenchimento natural, até que seja visitado novamente depois de certo período de tempo. O processo de reenchimento cessa quando a coluna de fluidos do poço atinge seu nível estático.



Figura 2.1: Unidade Móvel de Pistoneio.

A área de uma região de poços não-surgentes, como a observada na Bacia Potiguar, pode abranger milhares de quilômetros quadrados e conter uma grande quantidade de poços com potencial de produção que os tornem candidatos economicamente viáveis à operação de pistoneio. Portanto, torna-se necessário um método que estabeleça, para cada dia de um horizonte de eventos, os conjuntos de poços que serão visitados a cada dia de trabalho e sua ordem de visita, de forma que a relação entre a distância percorrida entre os poços e o volume de óleo coletado em cada um deles seja a mais produtiva possível.

Dessa maneira, o problema de roteamento periódico de unidades móveis de pistoneio (PRP-UMP) consiste na definição, para cada dia que compõe um horizonte de planejamento estabelecido, de um roteiro otimizado de visitas realizadas pelas UMP's a poços não-surgentes, de forma que o volume de óleo coletado seja o maior possível, tendo como ponto de partida e término uma estação de tratamento de óleo (ETO), onde as UMP's descarregam o óleo coletado. A Figura 2.2. ilustra um modelo onde os poços são representados por círculos e os trajetos feitos pelas UMP's são representados por arestas direcionadas, indicando os pontos de origem e destino do trajeto.

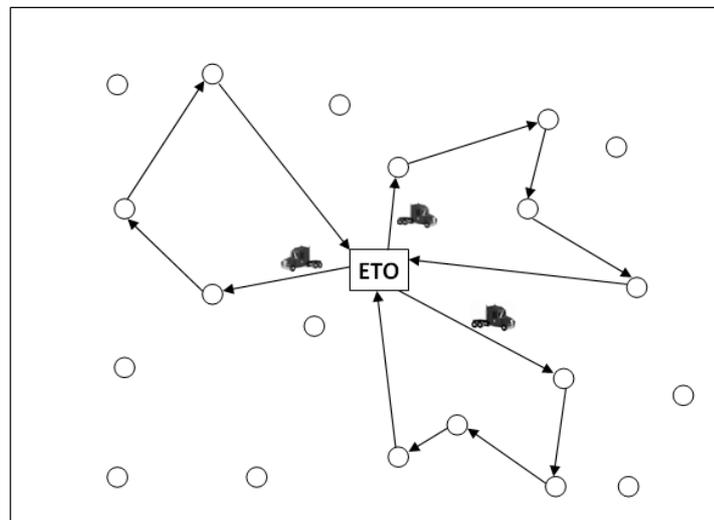


Figura 2.2: Exemplo de rotas geradas por múltiplas Unidades Móveis de Pistoneio.

A definição da rota a partir da ETO e o retorno a essa deve levar em consideração o turno de trabalho dos operadores das UMP's. Para efeitos práticos, caminhões-tanque auxiliares acompanham as UMP's para evitar que essas tenham que retornar à ETO antes do término do turno de trabalho, por já ter coletado um volume de óleo que alcance sua capacidade de armazenamento, de tal forma que essa é considerada ilimitada [Goldbarg *et al*, 2004; Aloise *et al*, 2001].

2.1 – FORMULAÇÃO MATEMÁTICA PARA UMA UMP

Esta formulação, desenvolvida por [Duarte, 2006], considera o campo terrestre a ser atendido por uma UMP como o grafo não direcionado $G = (V, E)$, onde $V = \{0, 1, 2, \dots, v\}$, que representa o conjunto de poços de superfície, sendo a estação de tratamento de óleo (ETO) representada por $v = 0$. O conjunto de arestas do grafo é representado por $E = \{(i, j) \mid i, j \in V \wedge i \neq j\}$, e o seguinte conjunto de valores representa as informações relacionadas aos poços a serem visitados:

- t_{ij} , que representa o tempo de percurso entre os poços i e j ;
- o_i^d , que representa o volume de óleo disponível para pistoneio no poço i , que foi visitado por uma UMP há d dias (o_i^0 representa o volume de óleo disponível no poço quando a coluna de óleo atinge seu nível estático, que representa sua capacidade máxima; assume-se que no primeiro dia de operações todos os poços apresentam inicialmente sua capacidade máxima);
- m_i^d , que representa o tempo necessário para a montagem e desmontagem do equipamento de coleta temporário, acrescido do tempo da operação de pistoneio devidamente informada, considerando que o poço foi pistoneado pela última vez há d dias e o volume de óleo disponível, representado por o_i^d ;

O PRP-UMP consiste em encontrar um conjunto de rotas para uma UMP, uma para cada um dos P dias de um horizonte de planejamento estabelecido, com duração máxima de M unidades de tempo, correspondentes ao turno de trabalho dos operadores da UMP, de forma a maximizar o volume de óleo coletado em cada poço, assim como o tempo de operação em cada um deles. Não são feitas restrições a respeito do limite de capacidade de armazenamento da UMP devido aos caminhões-tanque adicionais que acompanham a UMP em seu percurso. As constantes e variáveis utilizadas nesse modelo são definidas formalmente a seguir.

As variáveis de decisão consideradas na formulação do problema são:

$$x_i^{ud} = \begin{cases} 1, & \text{se o poço } i \text{ foi visitado no dia } u, \text{ cuja última visita foi realizada há } d \text{ dias,} \\ & \text{sendo } 1 \leq i \leq v, 1 \leq u \leq P, 1 \leq d \leq u. \text{ Quando } u = d, \text{ trata-se da primeira} \\ & \text{visita ao poço no período de atividades planejado.} \\ 0, & \text{caso contrário} \end{cases}$$

$$y_{ij}^u = \begin{cases} 1, & \text{caso o poço } j \text{ seja visitado pela UMP logo após o poço } i \text{ no dia } u, \\ & \text{sendo } 0 \leq i, j \leq v, i \neq j, 1 \leq u \leq P \\ 0, & \text{caso contrário} \end{cases}$$

$$\text{Maximizar } F = \sum_{u=1}^P \sum_{d=1}^u \sum_{i=1}^v x_i^{ud} o_i^{u-d} \quad (2.1)$$

sujeito a:

$$\sum_{i \in V} y_{ij}^u \leq 1, \forall j \in \{0, 1, \dots, v\}, \forall u \in \{1, 2, \dots, P\} \quad (2.2)$$

$$\sum_{j \in V} y_{ij}^u \leq 1, \forall i \in \{0, 1, \dots, v\}, \forall u \in \{1, 2, \dots, P\} \quad (2.3)$$

$$\sum_{j \in V - \{0\}} y_{0j}^u = \sum_{i \in V - \{0\}} y_{i0}^u = 1, \forall u \in \{1, 2, \dots, P\} \quad (2.4)$$

$$\sum_{j \in V} y_{ij}^u - \sum_{j \in v} y_{ji}^u = 0, \forall i \in \{0, 1, \dots, v\}, \forall u \in \{1, 2, \dots, P\} \quad (2.5)$$

$$\sum_{i \in V} \sum_{j \in V} y_{ij}^u \leq |N| - 1, \forall N \subseteq V - \{0\}, N \neq \emptyset, \forall u \in \{1, 2, \dots, P\} \quad (2.6)$$

$$\sum_{i \in V} \sum_{j \in V} t_{ij} y_{ij}^u + \sum_{i \in V - \{0\}} \sum_{d=1}^u m_i^{u-d} x_i^{ud} \leq M, \forall u \in \{1, 2, \dots, P\} \quad (2.7)$$

$$\sum_{d=1}^u x_i^{ud} - \sum_{j \in V} y_{ij}^u = 0, \forall i \in \{1, 2, \dots, v\}, \forall u \in \{1, 2, \dots, P\} \quad (2.8)$$

$$x_i^{ud} \in \{0, 1\}, \forall i \in \{1, \dots, v\}, \forall u \in \{1, 2, \dots, P\}, \forall d \in \{1, 2, \dots, u\} \quad (2.9)$$

$$y_{ij}^u \in \{0, 1\}, \forall i \in \{0, 1, \dots, v\}, \forall j \in \{0, 1, \dots, v\}, \forall u \in \{1, 2, \dots, P\} \quad (2.10)$$

A função objetivo F (2.1), do PRP-UMP procura maximizar o volume total de óleo coletado, considerando os poços visitados para cada rota construída para cada dia do horizonte de planejamento, e o volume de óleo disponível na ocasião da visita, que varia em relação ao tempo decorrido desde a última operação de pistoneio.

Em seguida estabelece-se a restrição que somente pode existir uma rota passando por cada poço, a cada dia do horizonte de planeamento, com uma aresta do grafo chegando em cada nó (2.2) e uma aresta partindo (2.3).

Em (2.4), o conjunto de restrições apresentadas garante que o nó zero do grafo, correspondente à ETO, estará incluído em todas as rotas geradas.

O conjunto de restrições (2.5) garante que para qualquer nó que faça parte da rota de um dia u qualquer do horizonte de planeamento, haverá uma aresta de entrada e uma aresta de saída, de tal forma que a rota só estará concluída ao retornar ao nó de origem, que corresponde à ETO. O conjunto de restrições (2.6) é estabelecido para assegurar que qualquer subconjunto de nós N que não inclua o nó zero, não possua mais do que $N-1$ arestas, o que garante que não existirão ciclos entre os nós de N . O único ciclo permitido é a rota completa, que inclui à ETO.

O conjunto de restrições (2.7) existe para assegurar que o tempo total gasto de deslocamento entre os poços de uma rota e o tempo total utilizado na operação de pistoneio não ultrapasse o limite estabelecido pela duração do turno de trabalho M .

O conjunto de restrições (2.8) garante que os valores das variáveis de decisão do problema concordam com as ações observadas: se um poço i é visitado em um dia u do horizonte de planeamento, a aresta que conecta o poço visitado ao próximo poço da rota foi percorrida; caso não haja visita, nenhuma aresta conectada a i deve ter ser marcada como utilizada. Finalmente, os conjuntos de restrições (2.9) e (2.10) denotam que as variáveis de decisão x e y , respectivamente, assumam valores estritamente binários.

2.2 – FORMULAÇÃO MATEMÁTICA PARA MÚLTIPLAS UMP'S

Uma solução do PRP-UMP empregando múltiplas UMP's foi desenvolvida por [Gonçalves, 2005], onde foi apresentada uma formulação matemática que é utilizada neste trabalho para o desenvolvimento de uma solução exata. Enquanto na formulação original o intervalo entre as visitas a um poço qualquer deve respeitar o tempo mínimo necessário para o reenchimento do mesmo, a versão proposta permite que um poço seja visitado sem que o reenchimento até o volume máximo do poço tenha sido concluído, restringe-se apenas que não ocorram duas visitas pela mesma ou por diferentes UMP's no mesmo dia.

A representação dos elementos considerados para o problema é a mesma empregada na solução proposta para uma UMP, com a adição dos seguintes elementos ao conjunto de valores:

- Z representa o conjunto de UMP's empregadas, $|Z| = z$;
- w_{ij}^{uk} , que representa uma variável de fluxo associada a cada aresta (i, j) pertencente à rota da UMP k , no dia u do horizonte de planejamento estabelecido.
- A variável de decisão y_{ij}^u passa a ser representada por y_{ij}^{uk} , pois k representa uma das UMP's empregadas nas operações de pistoneio.
- O conjunto \hat{V} representa o conjunto dos poços, excluindo a ETO ($\hat{V} = V - \{0\}$).

$$\text{Maximizar } F = \sum_{u=1}^P \sum_{d=1}^u \sum_{i=1}^v x_i^{ud} o_i^{u-d} \quad (2.11)$$

Considerando o conjunto de restrições:

$$\sum_{i=0}^n \sum_{j=0, j \neq i}^n \sum_{d=1}^u (t_{ij} + m_i^d) \times y_{ij}^{uk} \leq M, \forall u \in P, \forall d \in P, \forall k \in Z \quad (2.12)$$

$$\sum_{k=1}^z \sum_{i=0; i \neq q}^n \sum_{d=1}^u y_{iq}^{uk} = x_q^{ud}, \forall q \in \hat{V} \text{ e } \forall u \in P, \forall d \in P \quad (2.13)$$

$$\sum_{k=1}^z \sum_{i=0; i \neq q}^n \sum_{d=1}^u y_{qi}^{uk} = x_q^{ud}, \forall q \in \hat{V} \text{ e } \forall u \in P, \forall d \in P \quad (2.14)$$

$$\sum_{i=0; i \neq q}^n y_{iq}^{uk} - \sum_{i=0; i \neq q}^n y_{qi}^{uk} = 0, \forall q \in \hat{V} \text{ e } \forall u \in P \text{ e } \forall k \in Z \quad (2.15)$$

$$\sum_{k=1}^z \sum_{d=1}^u y_{ij}^{uk} \leq \frac{x_i^{ud} + x_j^{ud}}{2}, \forall i \in \hat{V}, \forall j \in \hat{V}, i \neq j \text{ e } \forall u \in P, \forall d \in P \quad (2.16)$$

$$\sum_{d=1}^u x_0^{ud} = 1, \forall u \in P, \forall d \in P \quad (2.17)$$

$$\sum_{j=1}^n y_{0j}^{uk} = 1, \forall u \in P \text{ e } \forall k \in Z \quad (2.18)$$

$$\sum_{i=1}^n y_{i0}^{uk} = 1, \forall u \in P \text{ e } \forall k \in Z \quad (2.19)$$

$$\sum_{j=0; j \neq q}^n w_{qj}^{uk} = \sum_{i=0; i \neq q}^n \sum_{d=1}^u w_{iq}^{uk} + o_q^d \times y_{iq}^{uk}, \forall k \in Z, \forall u \in P, \forall d \in P, \forall q \in \hat{V} \quad (2.20)$$

$$w_{0j}^{uk} = y_{0j}^{uk}, \forall j \in \hat{V}, \forall k \in Z, \forall u \in P \quad (2.21)$$

$$w_{ij}^{uk} \geq y_{ij}^{uk}, \forall i \in V, \forall j \in V, i \neq j, \forall k \in Z, \forall u \in P \quad (2.22)$$

$$y_{ij}^{uk} \geq \frac{w_{ij}^{uk}}{\sum_{j=1}^n \sum_{d=1}^u o_j^d}, \forall i \in V, \forall j \in V, i \neq j, \forall k \in Z, \forall u \in P, \forall d \in P \quad (2.23)$$

$$\sum_{d=1}^u x_i^{ud} \leq 1, \forall i \in \hat{V}, \forall u \in P, \forall d \in P \quad (2.24)$$

$$x_i^{ud} \in \{0, 1\}, \forall i \in V, \forall u \in P, \forall d \in P \quad (2.25)$$

$$y_{ij}^{uk} \in \{0, 1\}, \forall i \in V, \forall j \in V, \forall k \in Z, \forall u \in P \quad (2.26)$$

$$w_{ij}^{uk} \geq 0, \forall i \in V, \forall j \in V, \forall k \in Z, \forall u \in P \quad (2.27)$$

As desigualdades (2.12) estabelecem que o somatório dos tempos de deslocamento e pistoneio das visitas realizadas em uma rota diária não ultrapasse o limite imposto pelo turno de trabalho diário dos operadores de uma UMP.

O conjunto de restrições (2.13), (2.14) e (2.15) estabelece que a cada dia do horizonte de eventos estabelecido, apenas uma UMP pode fazer a operação de pistoneio do poço.

Em (2.16), é estabelecido que dois poços, i e j , a serem visitados em um dia u do horizonte de eventos, caso seja pistoneado o poço i , e logo em seguida o poço j , apenas uma UMP poderá fazê-lo.

A restrição (2.17) garante que o ponto de partida e chegada das UMP's, definido como a ETO estará presente em todas as rotas geradas. As restrições (2.18) e (2.19) são definidas para que as rotas geradas contenham ao menos um poço a ser visitado.

As restrições (2.20) a (2.23) foram estabelecidas para garantir que as rotas construídas não sejam desconexas da origem, empregando a variável de fluxo w_{ij}^{uk} . Em (2.20), é regulado que o fluxo de saída de um poço é igual ao fluxo de entrada, acrescido do volume de óleo pistoneado na visita. A restrição (2.21) estabelece um fluxo unitário aos arcos utilizado pelas UMP's ao partirem da ETO, que não é computado pela função objetivo. A restrição (2.22) estabelece que se uma determinada aresta é utilizada por uma UMP em um dia do horizonte de planejamento, então haverá um fluxo positivo, que identifica este uso. A restrição (2.23) estipula que havendo um fluxo positivo associado a uma aresta, deve haver uma UMP passando por ela no dia atual do período estabelecido.

A restrição (2.24) garante que cada poço pode receber, no máximo, uma visita por dia. Após pistoneado, uma nova visita pode ocorrer no próximo dia, estando o poço plenamente ou parcialmente reenchido, de acordo com o seu coeficiente de reenchimento.

Por fim, as restrições (2.25), (2.26) e (2.27) estabelecem o domínio das variáveis do problema.

2.3 – TRABALHOS RELACIONADOS

Em [Bassi *et al*, 2010] é apresentado o problema de planejamento de sondas de intervenção (PPSI), no qual equipamentos denominados sondas de intervenção realizam operações de manutenção em poços *offshore*, para reestabelecer as condições normais de operação em um poço, minimizando a perda total de óleo causada pela indisponibilidade produtiva decorrente de uma falha. Devido aos altos custos operacionais e do equipamento em si, não é viável ter uma sonda alocada em cada poço do campo petrolífero, portanto, é necessário o planejamento de uma sequência de operações de intervenção aos poços para um horizonte de eventos, similar a problemas de roteamento de veículos. Os autores desenvolveram um algoritmo guloso, de rápida execução, e um GRASP para a geração de conjuntos de rotas para as sondas de intervenção, seguidos por uma reconexão por caminhos, para refinar as soluções obtidas nas etapas anteriores.

O problema de otimização de itinerário de sondas (POIS) é apresentado por [Costa e Ferreira Filho, 2005] para determinar o sequenciamento e o roteamento de um conjunto de sondas de manutenção de um conjunto de poços situados em um campo petrolífero terrestre, na região norte do Espírito Santo, que possui centenas de poços, espalhados geograficamente por dezenas de campos. As sondas de produção terrestre (SPT) são empregadas em operações de manutenção, a fim de minimizar perdas de vazão. Por não ser financeiramente viável a instalação de uma sonda para cada um dos poços, essas devem ser transportadas entre os pontos de demandas conhecidas, tornando-se essencial a obtenção de rotas que minimizem os tempos de transporte e atendam rapidamente os poços que apresentam maior produtividade. Os autores propõem uma metaheurística GRASP para a resolução do POIS.

Uma abordagem utilizando a metaheurística GRASP para o problema do PRP-UMP com múltiplas UMP's foi proposta por [Gonçalves *et al*, 2005; e em Gonçalves, 2005] onde foram apresentadas heurísticas construtivas baseadas no método das pétalas e pela heurística de inserção mais próxima. Como métodos de busca local, empregaram-se a busca local diária, onde podem ser feitas trocas entre as rotas de um mesmo dia, geradas para diferentes UMP's, e a busca local periódica, na qual são alterados os dias de visitação dos poços para que os que

apresentem melhor produtividade possam ser melhor explorados. Como as iterações do GRASP são realizadas de maneira independente, os autores buscaram uma forma de relacionar as soluções de boa qualidade obtidas nas iterações utilizando um módulo de reconexão de caminhos para a exploração de novas soluções que combinem boas características de soluções obtidas em iterações anteriores.

Uma abordagem para uma versão não-periódica do PRP_UMP com múltiplas UMP's empregando técnicas de otimização e inteligência artificial é proposta em [Nascimento e Aloise, 2012]. Por meio de um algoritmo genético padrão, são construídas novas rotas a partir de soluções iniciais, geradas aleatoriamente, para diferentes UMP's. Com base nesse método, foi proposto um algoritmo memético, que incorpora uma etapa de busca local aplicada às novas rotas geradas.

A transgenética computacional, técnica que utiliza conceitos da evolução dos seres vivos por meio da transmissão de alterações genéticas pelas vias intracelular e epigenética foi utilizada como técnica para a resolução do PRP-UMP empregando uma UMP em [Goldberg *et al*, 2010]. Foi proposto um algoritmo transgenético, denominado ProtoG, que a partir de uma população inicial formada por rotas válidas para o problema, constrói um banco de informações contendo sequências de poços, que apresentam uma boa qualidade. Durante a execução do algoritmo, o banco vai sendo atualizado sempre que são encontradas soluções melhores. Nas iterações que se seguem é criado um vetor transgenético, composto por um trecho do banco de informações ou gerado dinamicamente para ser transcrito em soluções sensíveis à manipulação. A rota resultante é mantida se for viável e apresentar qualidade superior à rota original.

Enquanto os estudos apresentados por [Gonçalves, 2005; Nascimento e Aloise, 2012] empregam múltiplas UMP's em suas resoluções, a proposta do algoritmo ProtoG [Goldberg *et al*, 2010] utiliza uma UMP para a coleta de óleo. A proposta apresentada na presente tese considera o modelo que emprega uma e múltiplas UMP's para as atividades de pistoneio. Uma versão preliminar do estudo do PRP-UMP empregando uma UMP e heurísticas GRASP foi apresentada em [Andrade *et al*, 2013], e um estudo mais avançado, apresentando as heurística híbridas GRASP+ILS e GRASP+SA é apresentado em [Andrade *et al*, 2015].

CAPÍTULO 3 – HEURÍSTICA GRASP PARA O PRP-UMP

A metaheurística GRASP consiste em um algoritmo iterativo multipartida utilizado com muita frequência em problemas de otimização combinatória. A cada iteração do GRASP são realizadas duas etapas: uma de construção, responsável pela geração de uma solução viável, seguida por uma busca local, que tenta aprimorar a solução obtida pela etapa anterior. O objetivo do PRP-UMP é coletar o maior volume de óleo possível dos poços, e, portanto, caracteriza-se por ser um problema de maximização, ao contrário do PRV tradicional, que é essencialmente de minimização.

Algoritmo GRASP para o PRP-UMP

```

1:  $s^* \leftarrow 0$ ;
2: para  $i$  de 1 até  $max\_iterações$  faça
3:      $s_0 \leftarrow$  solução_inicial ( $alfa$ );
4:      $s' \leftarrow$  busca_local( $s_0$ );
5:     se  $f(s') > f(s^*)$ 
6:          $s^* \leftarrow s'$ ;
7:     fim se;
8: fim para;
9: retorne ( $s^*$ );

```

Figura 3.1: Metaheurística GRASP para um problema de maximização.

O algoritmo da figura 3.1 caracteriza a metaheurística GRASP para um problema de maximização. A melhor solução, s^* , é inicializada na linha 1 com o valor zero. Na estrutura de repetição compreendida entre as linhas 2 até 8 são realizadas as $max_iterações$ estabelecidas para o GRASP, onde inicialmente é gerada uma solução inicial s_0 pelo método construtivo, que utiliza uma lista restrita de candidatos - LRC de tamanho definido pelo parâmetro $alfa$. Na linha 4 realiza-se uma busca local a partir da solução s_0 , cujo resultado é uma nova solução, s' , que é comparada, na linha 5, com a melhor solução já conhecida. Se observada uma melhora no resultado, a melhor solução é atualizada com o resultado obtido por meio da busca local.

A função de avaliação utilizada no processo de decisão para a escolha de um poço para ser incluído ou retirado da rota da UMP é chamada de atratividade do poço. Neste trabalho

utilizou-se a formulação para cálculo da atratividade proposta por [Goldberg *et al*, 2010], que mostrou-se capaz de contribuir na obtenção de bons resultados, e representada pela equação 3.1, onde A_i^{jd} é o valor da atratividade do poço i , visitado pela última vez há d dias, em relação a UMP, que encontra-se no poço j . O valor o_i^d representa o volume atual do poço i , que foi visitado d dias atrás, e t_{ji} é o tempo de deslocamento entre os poços j e i .

$$A_i^{jd} = \frac{(o_i^d)^2}{t_{ji}} \quad (3.1)$$

A criação de diferentes versões da metaheurística GRASP por meio da combinação entre os métodos propostos para as etapas de construção e busca local torna possível a exploração de diferentes áreas do espaço de soluções do problema, na tentativa de localizar a solução ótima para o problema.

Os poços selecionados para a composição da rota de um dos dias do horizonte de planejamento estabelecido provavelmente apresentarão um volume reduzido no próximo dia de operação da UMP, que varia de acordo com a sua velocidade de reenchimento, o que significa que o valor de atratividade da maioria dos poços recém-pistoneados será baixo, e com isso, dificilmente serão incluídos nas rotas de dias imediatamente posteriores aos da última visita, contribuindo para que o GRASP construa rotas diversas no decorrer do horizonte de operações estabelecido. Eventualmente, alguns poços de maior vazão apresentarão valores de atratividade maiores do que os observados na maioria dos poços, e poderão aparecer com mais frequência nas rotas geradas.

3.1 – MÉTODOS CONSTRUTIVOS PROPOSTOS PARA O PRP-UMP

A fase inicial do GRASP consiste em um método construtivo que seja capaz de gerar uma solução inicial viável para o problema. Distintas metodologias podem ser empregadas nesta etapa, que diferem-se em relação ao desempenho e a qualidade das soluções encontradas.

Os métodos apresentados a seguir utilizam uma lista de candidatos à inserção, constituída por todos os poços que não fazem parte da solução atual, ordenados pelo maior valor obtido por meio de uma função de avaliação, que no caso do PRP-UMP, é a atratividade

de cada poço. Os melhores candidatos à inserção constituem uma lista restrita de candidatos (LRC), cujo tamanho é determinado por um parâmetro *alfa*.

O valor adotado para *alfa* mostra-se um parâmetro importante para a obtenção de soluções de maior qualidade. Enquanto um valor próximo de zero para *alfa* aumenta a possibilidade de escolha de um poço com maior atratividade, por outro lado, se a lista for muito pequena, o conjunto de candidatos para inserção será muito reduzido, o que restringe a diversificação de soluções.

Nas próximas subseções são apresentados os métodos de construção de uma solução inicial para o GRASP proposto.

3.1.1 – CONSTRUTIVO C1

Algoritmo Construtivo C1 ()

1. candidatos \leftarrow poços;
2. $s \leftarrow \emptyset$;
3. enquanto (candidatos > 0)
 4. calcula_atratividade_último (candidatos);
 5. LC \leftarrow insere_ordenado (LC, candidatos);
 6. LRC \leftarrow cria_LRC (LC, *alfa*);
 7. $p \leftarrow$ escolhe_elemento_LRC (LRC);
 8. se (duração da rota após a inserção de $p <$ turno)
 9. $s \leftarrow$ insere_final_rota (s, p);
 10. fim se;
 11. candidatos \leftarrow remove_poço (candidatos, p);
 12. fim enquanto;
 13. retorne (s);

Figura 3.2: Algoritmo Construtivo C1

O método construtivo representado na Figura 3.2 implementa o método de inserção de novos poços no final de uma rota parcial. Na linha 1, cria-se uma lista de candidatos, inicialmente contendo todos os poços da região considerada. A estrutura de repetição compreendida entre as linhas 3 e 12 será repetida enquanto existirem poços candidatos a serem considerados para inserção na solução. Na linha 4 são calculadas as atratividades dos

poços candidatos em relação ao último poço inserido na solução (no caso inicial, onde a rota encontra-se vazia, o cálculo da atratividade é feito em relação à ETO), e estes são inseridos em uma lista de candidatos - LC (linha 5), ordenados de forma não-crescente em relação aos valores obtidos. Em seguida, é criada uma lista restrita de candidatos - LRC de tamanho igual a uma porcentagem *alfa* da LC (linha 6), e na linha 7, é escolhido um poço *p*, aleatoriamente da LRC para compor a rota *s*. Antes de efetivamente passar a fazer parte da solução, é verificado (linha 8) se sua inserção não violará a restrição de duração da rota, considerando o tempo necessário a UMP retornar à ETO; caso a inserção seja considerada válida (linha 8), ele será inserido na última posição da rota que antecede o retorno à ETO. Em ambos os casos, o poço *p* é retirado da lista de candidatos à inserção na linha 11. O processo é repetido, verificando todos os possíveis poços candidatos, e ao ser concluído, produz uma solução viável. A Figura 3.3 ilustra o processo de construção da rota implementado pelo método construtivo C1 proposto.

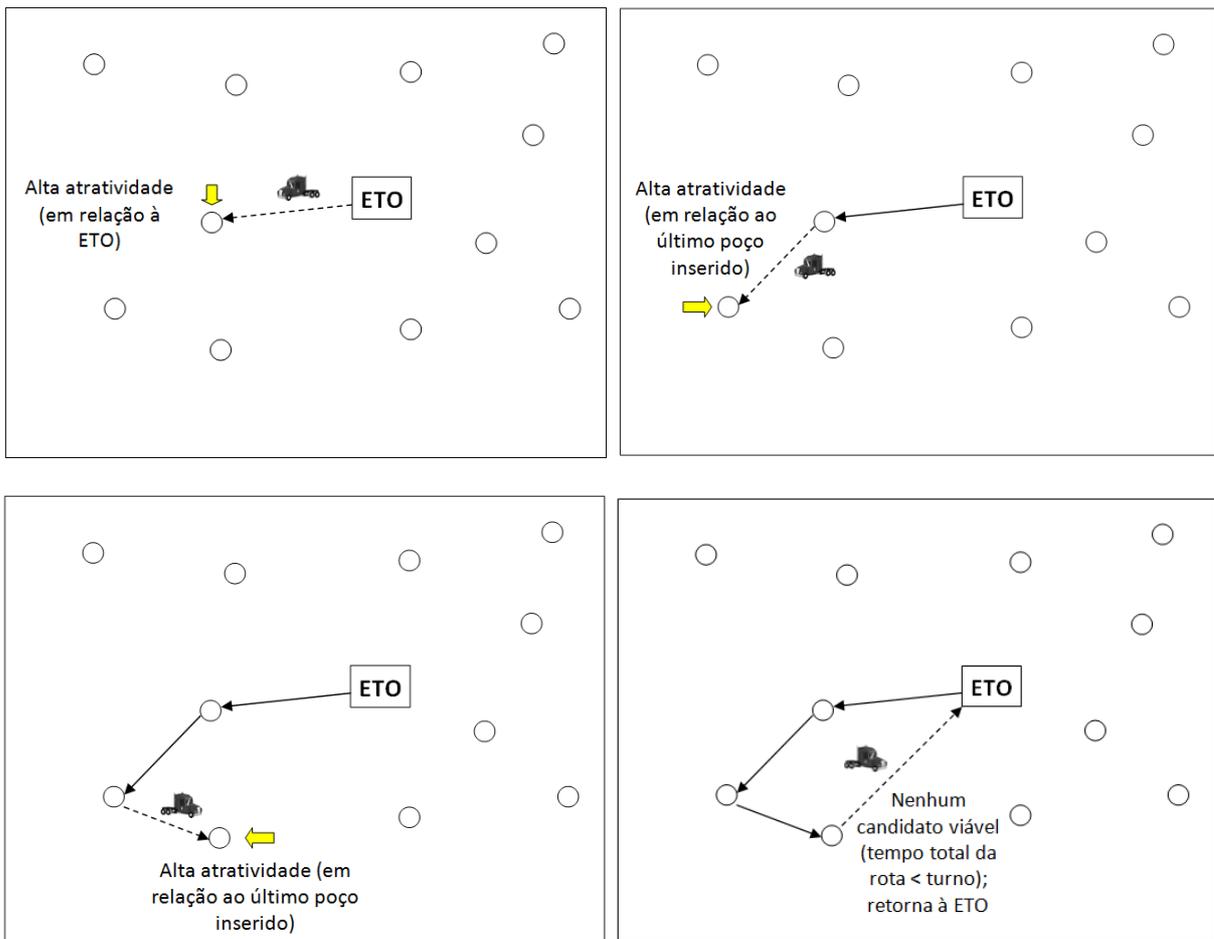


Figura 3.3: Exemplo de funcionamento do algoritmo Construtivo C1.

3.1.2 – CONSTRUTIVO C2

```

Algoritmo Construtivo C2 ( )
1.  candidatos ← poços;
2.  enquanto (candidatos > 0)
3.      calcula_atratividade_ETO (candidatos);
4.      LC ← insere_ordenado (LC, candidatos);
5.      LRC ← cria_LRC (LC, alfa);
6.      p ← escolhe_elemento_LRC (LRC);
7.      determina_melhor_posição_inserção (s, p);
8.      se (duração da rota após a inserção de p < turno)
9.          s ← insere_poço (s, p);
10.     fim se;
11.     candidatos ← remove_poço (candidatos, p);
12. fim enquanto;
13. retorne (s);

```

Figura 3.4: Algoritmo Construtivo C2.

O método construtivo ilustrado pela Figura 3.4 guarda similaridades com o método de inserção no final da rota, apresentado na Figura 3.2, mas com a diferença que o cálculo da atratividade dos poços, feito na linha 3, é realizado considerando a distância entre o poço candidato e a ETO. Nas linhas 4 e 5, são construídas a LC e a LRC, respectivamente. Na linha 6, é selecionado um poço candidato aleatoriamente da LRC para inserção.

Em seguida, na linha 7, é determinada a melhor posição para inserção no poço na rota *s*: são testadas todas as possibilidades de inserção, e seleciona-se aquela que resulte no menor acréscimo à duração da rota. Em seguida, entre as linhas 8 e 10, é verificado se a inserção do poço selecionado não gera uma solução que viole a restrição de duração máxima da rota; caso verdadeiro, o poço efetivamente passa a fazer parte de *s*, ocupando a posição determinada anteriormente. Na linha 11, o poço é retirado da lista de candidatos, tendo sido inserido na solução ou não, e todo o processo é repetido para os demais poços candidatos ainda não testados. O processo de seleção de poços e inserção na rota é ilustrado pela Figura 3.5.

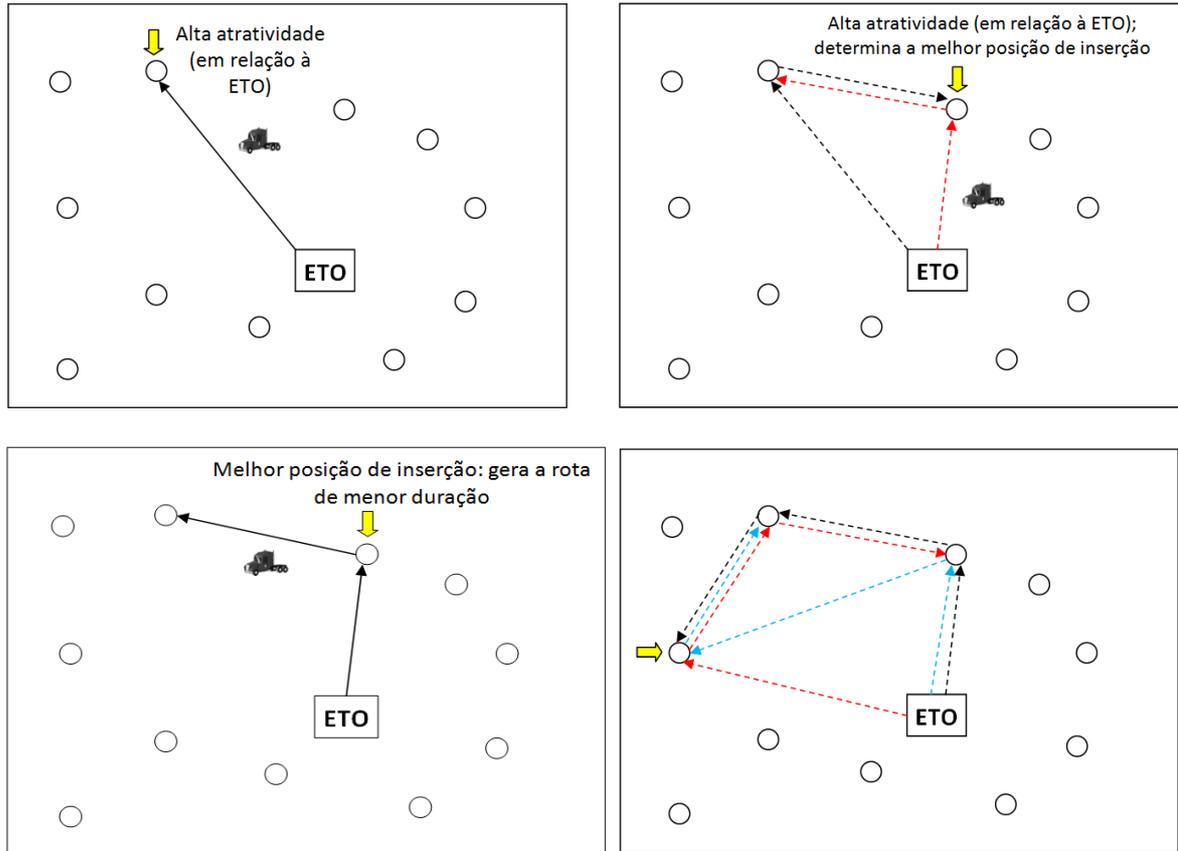


Figura 3.5: Exemplo de funcionamento do algoritmo Construtivo C2.

3.1.3 – CONSTRUTIVO C3

```

Algoritmo Construtivo C3 ( )
1.  candidatos ← poços;
2.  tamanho_solução ← 0;
3.  enquanto (tamanho_solução < 2)
4.      calcula_atratividade_ETO (candidatos);
5.      LC ← insere_ordenado (LC, candidatos);
6.      LRC ← cria_LRC (LC, alfa);
7.      p ← escolhe_elemento_LRC (LRC);
8.      s ← insere_poço (s, p);
9.      tamanho_solução ← tamanho_solução + 1;
10. fim enquanto;
11. atualiza_candidatos ( );
12. atratividade_candidatos ← calcula_atratividade_rota (candidatos);
13. enquanto (candidatos > 0) e (duração da rota s < turno)
14.     LC2 ← insere_ordenado (LC2, atratividade_poços);
15.     LRC2 ← cria_LRC (LC2, beta);
16.     p' ← escolhe_elemento_LRC2 (LRC2);
17.     s ← insere_melhor_posição (s, p');
18.     candidatos ← remove_poço (candidatos, p');
19. fim enquanto;
20. retorne (s);

```

Figura 3.6: Algoritmo Construtivo C3.

O método construtivo ilustrado na Figura 3.6 divide-se em duas etapas: na primeira, compreendida entre as linhas 3 e 10, são seleccionados os dois primeiros poços que farão parte da rota *s*. Na linha 4 é calculada a atratividade de cada poço, levando em consideração sua distância ao ponto de origem (ETO), e baseando-se no resultado desta operação, na linha 5, é construída uma lista de candidatos ordenada não-crescente. Em seguida, é criada uma LRC (linha 6) com tamanho igual a uma porcentagem *alfa* de LC. Finalmente, é seleccionado um poço aleatoriamente da LRC (linha 7), que é inserido na rota *s* atual (linha 8). Na conclusão desta etapa, apesar da rota contar com apenas dois poços, verifica-se a viabilidade da mesma,

e, caso já tenha sido violada a restrição do tempo máximo da rota, o último poço inserido é removido da solução. Na linha 11, os poços que já fazem parte da rota são retirados da lista de candidatos, e em seguida (linha 12) são recalculadas as atratividades dos poços candidatos, mas, ao invés de considerar-se a distância em relação à ETO, são calculadas em relação a cada um dos poços que já foram inseridos na rota na etapa anterior. Portanto, para cada um dos n poços na lista de candidatos, calcula-se sua atratividade para cada um dos m poços que já fazem parte da solução. Na linha 13 é iniciada a segunda etapa da construção, que será repetida enquanto houverem poços candidatos e o tempo máximo da rota, fixado pelo turno de trabalho, não tiver sido ultrapassado. Os valores de atratividades calculados na linha 12 são utilizados para a formação de uma lista de candidatos na linha 14, ordenada de forma não-crescente. Em seguida, na linha 15, é criada uma LRC2 com tamanho igual a uma porcentagem beta da lista de LC2, e um poço, p' , é selecionado aleatoriamente desta LRC2 na linha 16. Já é conhecida a atratividade do poço candidato em relação a um dos poços já inseridos na rota, mas é necessário definir se a melhor posição para sua inserção é imediatamente antes ou imediatamente após o poço da rota ao qual está relacionado, o que é feito na linha 16, caso a restrição de duração da rota com a inserção do poço não tenha sido violada. Finalmente, na linha 18, a lista de candidatos é atualizada, e todas as referências ao poço inserido são eliminadas, e o processo é repetido até que seja gerada uma rota inicial viável. A Figura 3.7 ilustra a construção de uma rota pelo método construtivo C3.

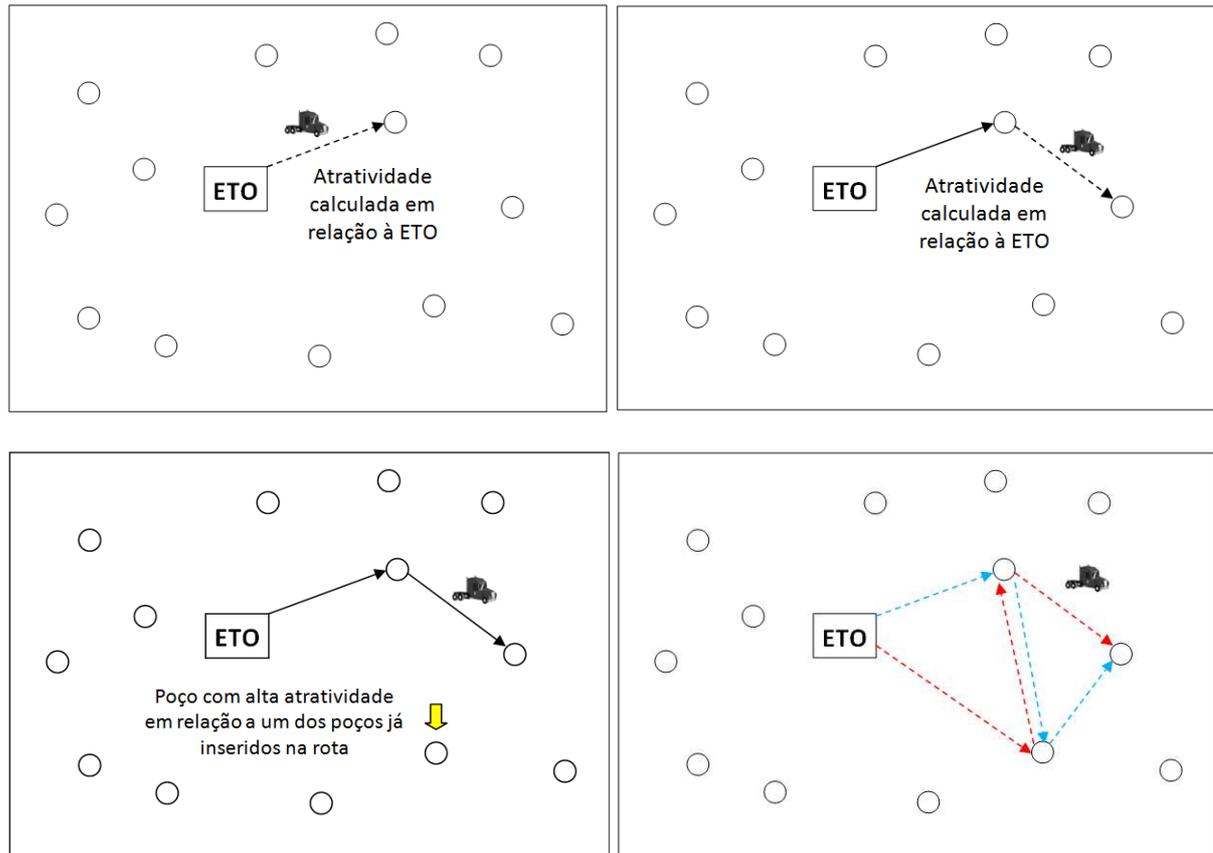


Figura 3.7: Exemplo de funcionamento do algoritmo Construtivo C3.

3.2 – ESTRATÉGIAS DE BUSCA LOCAL

Busca local é uma heurística empregada com o objetivo de intensificar uma solução inicial obtida na etapa de construção por meio da exploração do espaço de soluções. Novas soluções são geradas por meio da aplicação de mudanças locais na solução atual, e se for observada uma melhoria no resultado, ela torna-se a melhor solução conhecida para o problema. O processo é repetido até que não seja encontrada nenhuma outra solução de maior qualidade na vizinhança ou que se atinja o limite de um critério de parada do método.

3.2.1 – BUSCA LOCAL BL1

Algoritmo Busca Local BL1 (s)

```

1.  candidatos  $\leftarrow$  poços;
2.  atratividade_candidatos  $\leftarrow$  calcula_atratividade_rota (candidatos);
3.  fora  $\leftarrow$  cria_lista (candidatos,  $s$ );
4.  maior  $\leftarrow$  encontra_maior (fora, atratividade_candidatos);
5.  menor  $\leftarrow$  encontra_menor ( $s$ , atratividade_candidatos);
6.   $s' \leftarrow s$ ;
7.  enquanto (atratividade (maior) > atratividade (menor))
8.       $s' \leftarrow$  retira_poço ( $s'$ , menor);
9.       $s' \leftarrow$  determina_melhor_posição ( $s'$ , maior);
10.     se (duração da rota  $s'$  após a inserção de (maior) < turno)
11.          $s' \leftarrow$  insere_poço ( $s'$ , maior);
12.     senão
13.         fora  $\leftarrow$  retira_poço (fora, maior);
14.          $s' \leftarrow$  insere_poço ( $s'$ , menor);
15.     fim se;
16.     se ( $f(s') > f(s)$ )
17.          $s \leftarrow s'$ ;
18.         BL1 ( $s$ );
19.         Interrompe_enquanto;
20.     fim se;
21.     maior  $\leftarrow$  encontra_maior (fora, atratividade_candidatos);
22.     menor  $\leftarrow$  encontra_menor ( $s$ , atratividade_candidatos);
23. fim enquanto;
24. retorne ( $s$ );

```

Figura 3.8: Algoritmo de Busca Local BL1.

No primeiro método de busca local, representado pelo algoritmo da Figura 3.8, calcula-se inicialmente (linha 2) a atratividade de todos os poços, considerando a distância de cada um à ETO. Na linha 3 é criada, uma lista contendo os poços que não fazem parte da solução gerada pelo método construtivo, e portanto, candidatos à inserção. Nas linhas 4 e 5

são encontrados, respectivamente, o poço que apresenta a maior atratividade dentre os candidatos à inserção, e o poço de menor atratividade dentre os que fazem parte da rota s atual. As operações contidas entre as linhas 7 e 23 são repetidas enquanto houver um poço na lista de candidatos a inserção cuja atratividade for maior do que o respectivo valor mínimo de um poço já inserido na rota.

Dessa maneira, retira-se da rota atual o poço de menor atratividade (linha 8), e determina-se, na linha 9, a posição de inserção do poço candidato que resultará na rota de menor duração. Caso a rota resultante da inserção na melhor posição tenha duração inferior ao limite estabelecido pelo turno de trabalho, a inserção é efetuada na linha 11; caso contrário, o poço candidato é desconsiderado para as futuras tentativas de inserção e a rota é restaurada à sua composição antes da tentativa de modificação.

Em seguida, verifica-se se o volume da rota atual é superior ao volume da rota original. Se houve uma melhora na solução, é realizada uma nova busca local na rota corrente (linha 18), e a estrutura de repetição será interrompida, não havendo novas tentativas de remoção e inserção de poços para a rota atual, somente para as que serão geradas pela chamada recursiva da busca local em relação à solução s . A Figura 3.9 exemplifica a operação de busca local BL1.

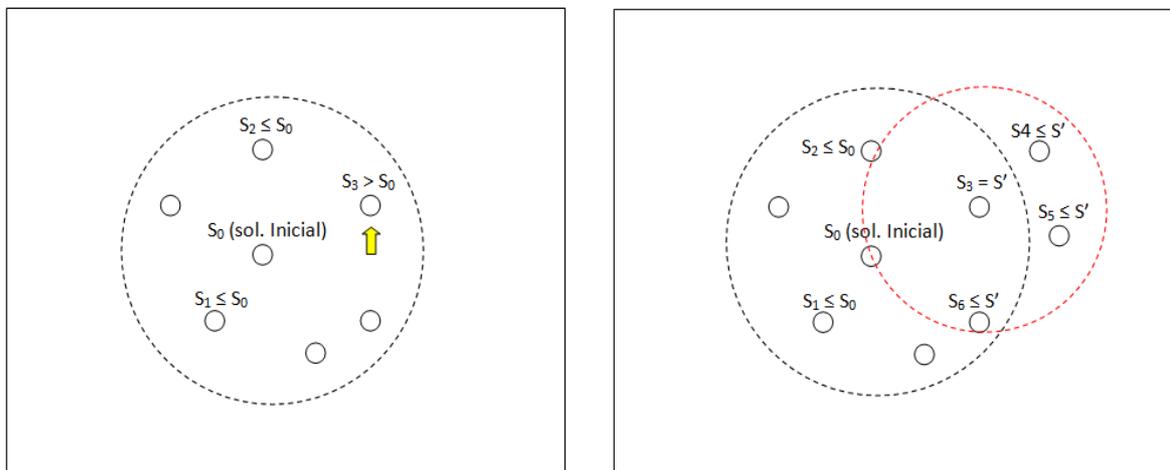


Figura 3.9: Exemplo de funcionamento da busca local BL1.

3.2.2 – BUSCA LOCAL BL2

Por tratar-se de uma variação da busca local BL1, o pseudo-código é praticamente o mesmo apresentado no algoritmo apresentado na Figura 3.8, com a exceção da linha 19, que

foi suprimida nesta versão, pois caso seja obtida pela busca local uma solução s' com volume superior ao da rota s original, após terem sido concluídas todas as chamadas recursivas decorrentes da execução de uma nova busca local usando s' como solução inicial, serão feitas as demais tentativas de remoção e inserção dos atuais poços candidatos em relação a melhor solução conhecida até o momento. A Figura 3.10 ilustra a operação da busca local BL2.

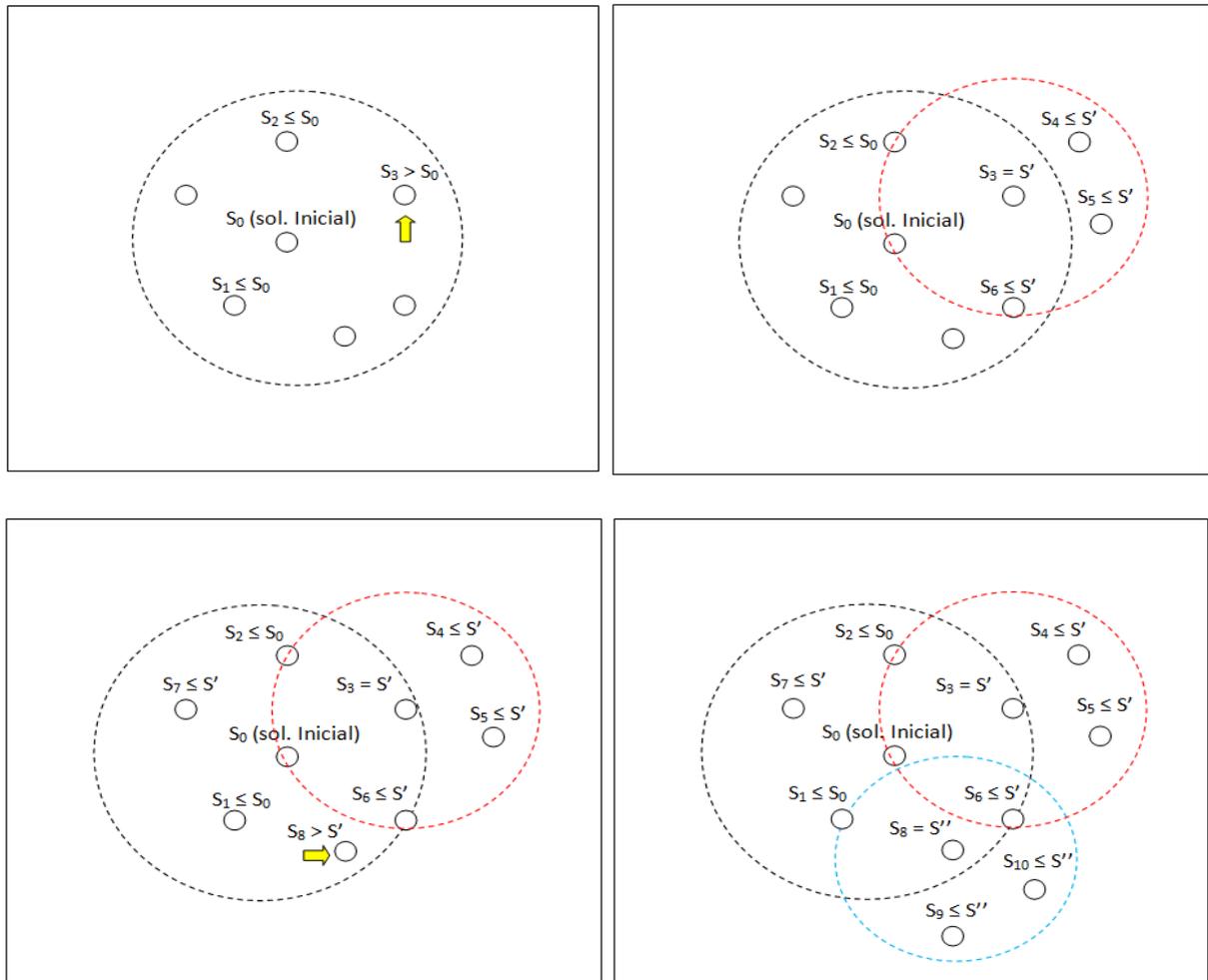


Figura 3.10: Exemplo de funcionamento da busca local BL2.

3.2.3 – BUSCA LOCAL BL3

A busca local BL3 também se baseia na busca local descrita pelo algoritmo da Figura 3.8, com uma diferença fundamental: ao término da estrutura de repetição compreendida entre as linhas 7 e 23, será feita uma nova chamada à função de busca local, utilizando como parâmetro a melhor rota já encontrada até o momento. A operação de seleção de uma solução pela busca local BL3 é ilustrada pela Figura 3.11.

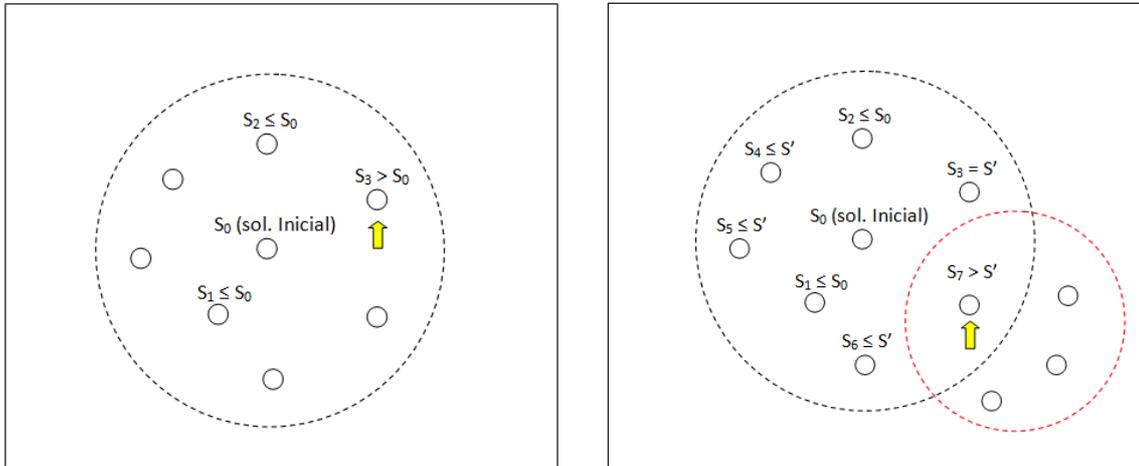


Figura 3.11: Exemplo de funcionamento da busca local BL3.

3.2.4 – BUSCA LOCAL BL4

Algoritmo Busca Local BL4 (s, x)

```

1.   $s' \leftarrow s$ ;
2.  candidatos  $\leftarrow$  poços_fora ( poços,  $s'$ );
3.  atratividade_candidatos  $\leftarrow$  calcula_atratividade (candidatos);
4.  LC  $\leftarrow$  cria_lista (candidatos,  $s'$ );
5.  LRC  $\leftarrow$  cria_LRC (LC,  $\alpha$ );
6.   $p \leftarrow$  seleciona_candidato (LRC);
7.  determina_melhor_posição ( $p, s'$ );
8.   $s' \leftarrow$  insere_poço ( $s', p$ );
9.  enquanto (duração da rota  $s' >$  turno)
10.     menos_atrativo  $\leftarrow$  calcula_menor_atratividade ( $s'$ );
11.      $s' \leftarrow$  retira_poço ( $s',$  menos_atrativo);
12. fim enquanto;
13. para  $i$  de 1 até  $x$  faça
14.      $p' \leftarrow$  seleciona_candidato (LC);
15.     determina_melhor_posição ( $p', s'$ );
16.     se (duração da rota após a inserção de ( $p'$ )  $<$  turno)
17.          $s' \leftarrow$  insere_poço ( $s', p'$ );
18.     senão
19.         candidatos  $\leftarrow$  remove_poço (candidatos,  $p'$ );
20.     fim se;
21. fim para;
22. se ( $f(s') > f(s)$ )
23.      $s \leftarrow s'$ ;
24. fim se;
25. retorne ( $s$ );

```

Figura 3.12: Algoritmo de Busca Local BL4.

A quarta estratégia de busca local proposta, ilustrada pelo algoritmo da figura 3.12, inicia com a formação de uma lista de poços candidatos, que é composta por todos os poços considerados para a operação de pistoneio, exceto os que já fazem parte da solução atual. Em seguida, é feito o cálculo da atratividade em relação à ETO para todos os poços candidatos

(linha 3). Nas linhas 4 e 5 são criadas a LC e a LRC, respectivamente, sendo a LC formada pelos poços candidatos em ordem não-crescente de atratividade, e a LRC formada pelos n primeiros poços da LC, sendo n a porcentagem *alfa* da quantidade de poços candidatos. Na sequência (linha 6), seleciona-se aleatoriamente um poço candidato da LRC, e antes de torná-lo parte da solução, determina-se a posição de inserção que resultará no menor acréscimo ao tempo de duração da rota (linhas 7 e 8). Se a inserção realizada torna a solução inviável por extrapolar o tempo máximo de duração da rota, estabelecido pelo turno de trabalho, serão repetidas as instruções entre as linhas 9 e 12, que retiram o poço de menor atratividade da rota, até que a duração da rota seja igual ou inferior ao turno.

Considerando os poços candidatos restantes, sendo que os poços retirados da solução durante a busca local não fazem parte deste conjunto, são realizadas x tentativas (onde x é um valor pré-estabelecido, menor do que o número de candidatos) de inserção de um novo poço p' na rota, sendo determinada a posição de inserção que representa o menor acréscimo à duração da rota (linha 15). O candidato p' passa a ser parte efetiva da rota somente se não for violada a restrição de duração da mesma (linha 17), caso contrário, é retirado da lista de candidatos à inserção (linha 19).

3.2.5 – BUSCA LOCAL BL5

Algoritmo Busca Local BL5 (s)

```

1.   $s' \leftarrow s$ ;
2.  melhora  $\leftarrow 1$ ;
3.  enquanto (melhora = 1)
4.      candidatos  $\leftarrow$  poços_fora_com_volume_acima_50% (poços,  $s'$ );
5.      atratividade_candidatos  $\leftarrow$  calcula_atratividade_rota (candidatos);
6.      LC  $\leftarrow$  insere_ordenado (LC, candidatos);
7.      LRC  $\leftarrow$  cria_LRC (candidatos,  $\alpha$ );
8.      menor  $\leftarrow$  encontra_menor ( $s'$ , atratividade_candidatos);
9.       $s' \leftarrow$  retira_poço ( $s'$ , menor);
10.      $p \leftarrow$  seleciona_candidato (LRC);
11.     determina_melhor_posição ( $p$ ,  $s'$ );
12.     se (duração de  $s'$  após a inserção de ( $p$ ) < turno)
13.          $s' \leftarrow$  insere_poço ( $s'$ ,  $p$ );
14.     fim se;
15.     se ( $f(s') > f(s)$ )
16.          $s \leftarrow s'$ ;
17.     fim se;
18.      $s'' \leftarrow$  Busca Local BL5 ( $s'$ );
19.     se ( $f(s'') > f(s')$ )
20.          $s' \leftarrow s''$ ;
21.         se ( $f(s'') > f(s)$ )
22.              $s \leftarrow s'$ ;
23.         fim se;
24.     senão
25.         melhora  $\leftarrow 0$ ;
26.     fim se;
27. fim enquanto;
28. retorne ( $s$ );

```

Figura 3.13: Algoritmo de Busca Local BL5.

A quinta estratégia de busca local desenvolvida para o PRP-UMP, apresentada na Figura 3.13, inicia com a criação de uma lista de candidatos composta pelo conjunto de poços disponíveis para pistoneio que apresentem no momento ao menos metade do seu volume máximo e o respectivo cálculo de seus valores de atratividade em relação à ETO (linhas 4 e 5). Uma vez calculadas a atratividade dos poços, é criada uma LC em ordem não-crescente destes valores (linha 6), e em seguida, na linha 7, uma LRC cujo tamanho é igual a uma porcentagem *alfa* do número de elementos da LC. Entre as linhas 3 e 27, são realizadas as operações de geração de uma nova solução a partir da rota atual, que serão realizadas continuamente enquanto a solução obtida na iteração corrente apresentar uma melhoria em relação à solução anterior. No início de cada iteração é identificado o poço que faz parte da solução com menor atratividade para em seguida, ser removido da rota (linhas 8 e 9, respectivamente). Um poço candidato é selecionado aleatoriamente da LRC (linha 10) e é determinada a melhor posição para a inserção desse poço na rota (linha 11), mas a inclusão desse poço candidato ocorrerá somente se a duração da rota resultante não ultrapassar o tempo estabelecido como turno de trabalho dos operadores da UMP (linhas 12 a 14). Na linha 15, verifica-se se a solução atual é superior a melhor solução conhecida, que em caso verdadeiro, será atualizada. Na linha 18 é realizada uma nova busca local a partir da solução gerada na iteração corrente. Se a solução gerada por esta chamada recursiva da busca local for superior à solução gerada no início da iteração atual, ela será utilizada na próxima iteração (linha 20), e ainda, compara-se esta solução com a melhor já encontrada, que será atualizada, caso mostre-se superior (linhas 21 a 23).

CAPÍTULO 4 – HEURÍSTICAS HÍBRIDAS GRASP+ILS E GRASP+SA

4.1 – ITERATED LOCAL SEARCH

A metaheurística *Iterated Local Search* (ILS) [Lourenço *et al*, 2003] foi adotada como uma técnica de otimização global aplicada às soluções geradas pelo GRASP. Em algumas ocasiões, um método heurístico pode encontrar uma solução que é um ótimo local, onde nenhum de seus vizinhos representa uma solução melhor do que esse ótimo. Para evitar essa situação, o ILS realiza uma perturbação na solução atual s , o que o leva a uma nova solução s' , que deve ser suficientemente diferente de s para permitir que a aplicação de uma operação de busca local em s' gere soluções que representem a exploração de uma nova área do espaço de soluções, que pode, eventualmente, resultar na obtenção da solução exata do problema.

A Figura 4.1 ilustra as operações de um algoritmo ILS básico para um problema de maximização. Inicialmente é utilizado um método, como uma heurística construtiva, para a obtenção de uma solução inicial, s_0 (linha 1), ao problema em questão, que é submetida a uma busca local na linha 2, que gera uma nova solução, s . Em seguida, entre as linhas 4 e 11, são realizadas várias operações de perturbação da solução, até um limite previamente estabelecido de *max_iterações*, onde s é submetida a uma operação de perturbação (linha 6), responsável por modificar a solução atual, obtendo uma solução s' . Na linha 7, uma busca local é realizada na solução s' , gerando uma nova solução, s'' . Essa é comparada, na linha 8, com a solução inicial s , e caso seja observada uma melhoria segundo o critério estabelecido (como a maximização da coleta de óleo), a nova solução encontrada se torna a melhor rota conhecida. Ao término das iterações, o melhor resultado encontrado é retornado (linha 12).

Algoritmo ILS (s)

1. $s_0 \leftarrow \text{Algoritmo_Construtivo} ();$
2. $s \leftarrow \text{Busca_Local} (s_0);$
3. iterações $\leftarrow 0;$
4. enquanto iterações $<$ max_iterações
5. iteraões \leftarrow iterações $+1;$
6. $s' \leftarrow \text{Perturba_Solução} (s);$
7. $s'' \leftarrow \text{Busca_Local} (s');$
8. se ($f(s'') > f(s)$)
9. $s \leftarrow s'';$
10. fim se;
11. fim enquanto;
12. retorne (s);

Figura 4.1: Metaheurística ILS para um problema de maximização.

4.1.1 – ITERATED LOCAL SEARCH PARA O PRP-UMP CONSIDERANDO A UTILIZAÇÃO DE UMA UMP

Algoritmo PRP-UMP GRASP + ILS para 1 UMP ()

```

1.  $s \leftarrow 0$ ;
2. poços  $\leftarrow$  lê entrada de dados (conjunto_poços, volume, distância);
3. para  $x$  de 1 até (limite de iterações ou limite de tempo) faça
4.     para  $y$  de 1 até (máximo de dias) faça
5.          $s \leftarrow$  Construtivo ( );
6.          $s' \leftarrow$  Busca_Local ( $s$ );
7.         se  $f(s') > f(s)$ 
8.              $s \leftarrow s'$ ;
9.         fim se;
10.    para  $j$  de 1 até (limite de iterações do ILS) faça
11.         $p \leftarrow$  Selecciona_Perturbação_Intra_Rota ( );
12.         $s'' \leftarrow$  Perturba ( $s, p$ );
13.        se  $f(s'') > f(s)$ 
14.             $s \leftarrow s''$ ;
15.        fim se;
16.         $s''' \leftarrow$  Busca_Local ( $s''$ );
17.        se  $f(s''') > f(s)$ 
18.             $s \leftarrow s'''$ ;
19.        fim se;
20.    fim para;
21. fim para;
22. fim para;
23. retorne ( $s$ ).

```

Figura 4.2: Metaheurística ILS para o PRP-UMP com uma UMP.

A Figura 4.2 ilustra o GRASP com ILS proposto para a operação de pistoneio considerando a utilização de somente uma UMP. Na linha 2, são acessados os dados sobre o conjunto de poços disponíveis para pistoneio, contendo as informações sobre o volume inicial de cada poço e distância entre eles. A partir da linha 3 são feitas n iterações do GRASP,

limitadas por um critério de parada, sendo possível optar por um número pré-estabelecido de iterações ou quando for atingido um limite de tempo estabelecido para a execução do algoritmo. Para cada dia do horizonte de eventos definido, é feita uma iteração do GRASP seguida do ILS, a partir da linha 4, quando um método construtivo é aplicado no conjunto de dados de entrada (linha 5), resultando em uma solução inicial s , a partir da qual é realizada uma operação de busca local, usando um dos algoritmos disponíveis, descritos no capítulo anterior (linha 6). Em seguida, compara-se a solução s' obtida com a solução s , e caso seja observada uma melhoria no resultado (linha 7), s' passa a ocupar a posição de solução atual. No próximo passo são realizadas, entre as linhas 10 e 20, perturbações à rota atual, limitadas por um número de iterações previamente estabelecido. Para cada uma dessas iterações, seleciona-se uma das perturbações intra-rota desenvolvidas a partir de um repositório de perturbações (linha 11). Inicialmente, todas as operações de perturbações estão disponíveis no repositório, e quando uma delas é selecionada aleatoriamente para ser aplicada à solução, é retirada temporariamente deste repositório. A perturbação escolhida não estará disponível nas próximas iterações, até que todas as demais disponíveis tenham sido selecionadas e aplicadas à solução da iteração corrente. Quando este repositório tornar-se vazio, o que significa que todas as estratégias de perturbação foram utilizadas em diferentes iterações, ele é preenchido novamente com todas as perturbações disponíveis, que serão selecionadas uma a uma, repetindo este processo até o término das iterações.

A aplicação da perturbação p selecionada em s (linha 12) resulta em uma nova solução s' , que é comparada com a solução atual s para verificar se a nova solução gerada representa um aprimoramento, o que se for observado, substituirá s como a solução atual. A partir da solução resultante da perturbação será aplicada uma nova operação de busca local (linha 17), gerando uma solução identificada como s'' , que substituirá a solução corrente caso o volume de óleo coletado pela UMP seja superior ao obtido pela solução s .

4.1.2 – ITERATED LOCAL SEARCH PARA O PRP-UMP CONSIDERANDO A UTILIZAÇÃO DE MÚLTIPLAS UMP'S

Algoritmo PRP-UMP GRASP + ILS para Múltiplas UMP'S ()

1. poços \leftarrow lê entrada de dados (conjunto_poços, volume, distância);
2. para x de 1 até (limite de iterações ou limite de tempo) faça
3. para y de 1 até (máximo de dias) faça
4. para i de 1 até (número de UMP's) faça
5. $s(i) \leftarrow$ construtivo (poços);
6. $s'(i) \leftarrow$ Busca_Local ($s(i)$);
7. se $f(s'(i)) > f(s(i))$
8. $s(i) \leftarrow s'(i)$;
9. fim se;
10. Atualiza_Poços_Candidatos ();
11. fim para;
12. para i de 1 até (número de UMP's) faça
13. para j de 1 até (limite de iterações) faça
14. $j \leftarrow$ (aleatório (UMP's) $\neq i$)
15. $p' \leftarrow$ seleciona perturbação inter_rota ();
16. $s''(i), s'''(i) \leftarrow$ perturba ($s(i), s(j), p'$);
17. se tempo($s''(i)$) < turno
18. se $f(s''(i)) > f(s(i))$
19. $s(i) \leftarrow s''(i)$;
20. $s'(i) \leftarrow$ Busca_Local ($s(i)$);
21. se $f(s'(i)) > f(s(i))$
22. $s(i) \leftarrow s'(i)$;
23. fim se;
24. fim se;
25. fim se;
26. se tempo($s'''(i)$) < turno
27. se $f(s'''(i)) > f(s(i))$
28. $s(i) \leftarrow s'''(i)$;

```

Algoritmo PRP-UMP GRASP + ILS para Múltiplas UMP'S ( )
29.            $s'(i) \leftarrow \text{Busca\_Local}(s(i));$ 
30.           se  $f(s'(i)) > f(s(i))$ 
31.                $s(i) \leftarrow s'(i);$ 
32.           fim se;
33.       fim se;
34.   fim se;
35.   fim para;
36.   fim para;
37.   Atualiza_Informações dos poços ( );
38.   fim para;
39. fim para;
40.  retorne (rotas  $s$ );

```

Figura 4.3: Metaheurística ILS para o PRP-UMP para múltiplas UMP's.

A versão do ILS considerando a utilização de múltiplas UMP's, representada pela Figura 4.3. Após a leitura do conjunto de dados referentes ao campo de pistoneio na linha 1, iniciam-se as iterações limitadas por uma quantidade pré-estabelecida ou por um limite de tempo fixo (linha 2), e para cada uma dessas iterações, é realizada uma para cada dia de um horizonte de eventos estabelecido (linha 3), para cada uma das UMP's que realizarão o pistoneio (linha 4). Para cada UMP i é realizada a construção de uma solução viável $s(i)$ para o problema (linha 5), e em seguida é realizada uma busca local, gerando a solução $s'(i)$. Se a solução $s'(i)$ apresentar um melhor resultado em relação a função objetivo f do que a solução $s(i)$, ela a substituirá como a melhor rota atual para a UMP i . Antes da execução do GRASP para a próxima UMP, a lista de poços disponíveis para pistoneio é atualizada. Entre as linhas 12 e 41 o ILS é executado para as soluções geradas pelo GRASP para cada uma das UMP's. A partir da linha 13, para um número previamente definido de iterações, seleciona-se aleatoriamente uma UMP diferente da que está sendo considerada na iteração corrente. Na linha 15, seleciona-se uma das perturbações inter-rotas desenvolvidas a partir de um repositório, cujo gerenciamento das perturbações disponíveis é apresentada em 4.3.2, e a operação escolhida é aplicada nas soluções $s(i)$ e $s(j)$, gerando as novas soluções $s''(i)$ e $s'''(i)$.

As operações compreendidas entre as linhas 17 e 25 verificam se a duração da solução $s''(i)$ gerada é viável, baseada em sua duração (linha 17), e em caso verdadeiro, compara-se seu resultado com a rota $s(i)$ original. Se observada a obtenção de uma solução de maior qualidade, ela passa a ser considerada como a rota atual, e uma operação de busca local será realizada a partir dela na linha 20, gerando uma solução identificada por $s'(i)$. Esta nova solução é comparada com a solução a partir da qual foi feita a busca local, verificando-se se uma nova melhor rota para a UMP i foi encontrada. As operações entre as linhas 26 e 34 são exatamente as mesmas, porém, aplicadas à solução $s'''(i)$, que é a segunda das rotas geradas pela aplicação de uma operação de perturbação inter-rota aplicada às rotas $s(i)$ e $s(j)$ originais. Na linha 37, as informações dos poços visitados pelas i UMP's são atualizadas para a construção das rotas do próximo dia do horizonte de eventos. Ao término das iterações estabelecidas, as melhores rotas obtidas para cada uma das UMP's são retornadas à função principal.

4.2 – ESTRATÉGIAS DE PERTURBAÇÃO

As heurísticas de otimização empregadas baseiam-se no fato que os algoritmos de busca local não exploram todo o espaço de soluções de um problema, e ao atingirem um ótimo local, podem não explorar outras vizinhanças que podem conter soluções melhores do que às já conhecidas. Para tal, o ILS aplica um método de perturbação à melhor solução já conhecida, fazendo modificações nessa que sejam significativas o suficiente para permitir que a busca local explore diferentes regiões do espaço de soluções, mas que não modifique drasticamente, o que descaracterizaria a solução original.

São apresentadas as seguir dez estratégias de perturbação propostas para o ILS e o SA aplicado ao PRP-UMP. Enquanto as sete primeiras baseiam-se em modificações intra-rota, retirando elementos que serão substituídos por outros de um conjunto de candidatos, cuja composição pode variar, dependendo da estratégia escolhida, as demais estratégias são métodos clássicos de operações inter-rota. Uma solução para o PRP-UMP representa uma rota que se inicia na ETO, passa por uma sequência de poços e no final retorna a mesma ETO do qual partiu. Para algumas das estratégias descritas a seguir, considera-se como sendo a primeira metade de uma rota composta por n poços o conjunto $(p_1, p_2, \dots, p_{\lceil n/2 \rceil})$, cuja ordem dos elementos representa a sequência de visitação de poços pela UMP após esta partir da ETO, e a segunda metade da rota é representada pelo conjunto, $(p_{\lceil n/2 \rceil + 1}, \dots, p_n)$, que representa

a sequência de poços visitados a partir do término da primeira metade da rota até atingir o último poço visitado pela UMP antes que ela retorne à ETO.

As escolhas das estratégias de perturbação intra-rota relacionadas às posições de retirada e inserção de poços na rota da UMP têm o objetivo de garantir que as perturbações aplicadas à rota não se concentrem em trechos específicos da mesma. Na construção da solução realizada pelo GRASP os poços com maior atratividade são selecionados primeiro, e caso não violem a restrição de duração, são inseridos nas primeiras posições da rota para visita. Com a definição de regiões específicas de uma rota que sofrerão modificações, espera-se aplicar perturbações que não seriam realizadas em estratégias cujos critérios de alterações sejam baseados somente nos valores de contribuição de cada candidato.

As operações de perturbação inter-rota são realizadas entre a rota da UMP da iteração corrente e a rota de uma das outras UMP's em operação, selecionada aleatoriamente, a não ser no caso de existirem apenas duas UMP's em operação, pois só haverá mais uma rota para ser selecionada. A cada perturbação inter-rota, duas novas rotas são criadas e comparadas com a solução atual para verificar se alguma delas representa uma nova solução de maior qualidade.

4.2.1 – PERTURBAÇÃO P1

Algoritmo P1(s_0)

1. $s \leftarrow s_0$;
2. $LC \leftarrow \text{cria_lista}(\text{candidatos}, s)$;
3. $p_1 \leftarrow \text{aleatório1}(LC)$;
4. $p_2 \leftarrow \text{aleatório2}(s)$;
5. $s \leftarrow \text{remove_poço}(s, p_2)$;
6. $s \leftarrow \text{insere_poço}(s, p_1)$;
7. se $(f(s) > f(s_0))$
8. $s_0 \leftarrow s$;
9. fim se;
10. retornar (s_0) ;

Figura 4.4: Perturbação P1.

Na estratégia de perturbação P1, apresentada pela Figura 4.4, cria-se uma cópia da rota original s_0 , identificada como s , que será manipulada. Uma lista de candidatos, formada por todos os poços que não fazem parte da rota s , é criada na linha 2, e um poço p_1 desta lista é selecionado aleatoriamente na linha 3 para ser inserido na rota, (linha 6), após a seleção e subsequente retirada de um poço p_2 , escolhido aleatoriamente dentre os poços que compõem a metade inicial da rota s . O poço p_1 é inserido na rota, na mesma posição ocupada anteriormente por p_2 . Em seguida, verifica-se se a rota gerada apresenta qualidade superior à da rota original, e em caso positivo, ela substitui a rota original como a melhor rota já encontrada (linhas 7 a 9). A Figura 4.5 ilustra as operações de perturbação do método P1.

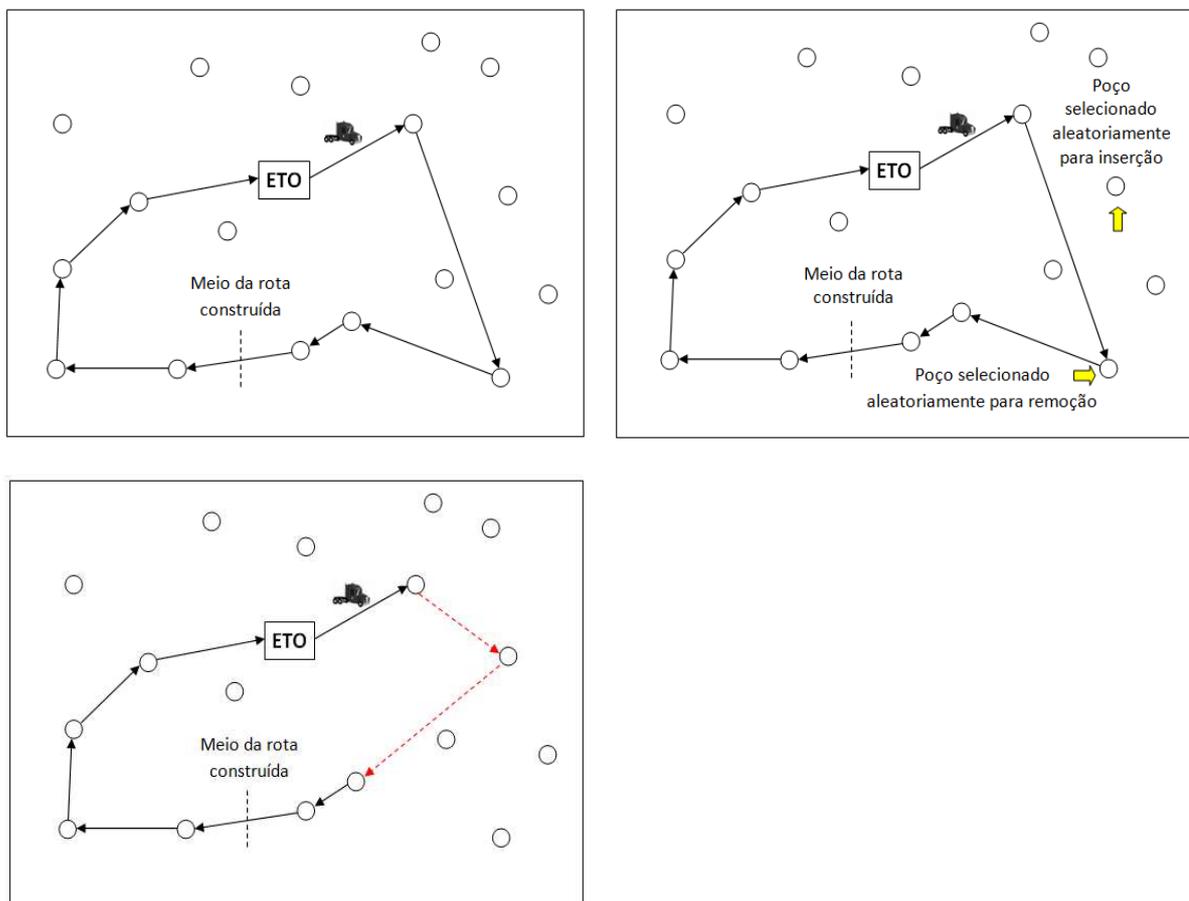


Figura 4.5: Exemplo da aplicação da perturbação P1.

4.2.2 – PERTURBAÇÃO P2

Similar a estratégia proposta para a perturbação P1, diferindo-se pela escolha do poço que será retirado ser feita aleatoriamente dentre os que fazem parte da segunda metade da rota, e o poço selecionado aleatoriamente da lista de candidatos ser inserido em uma posição

escolhida aleatoriamente na primeira metade da rota. As demais operações são feitas da mesma maneira que a estratégia P1. A Figura 4.6 ilustra as mudanças feitas na rota pela perturbação P2.

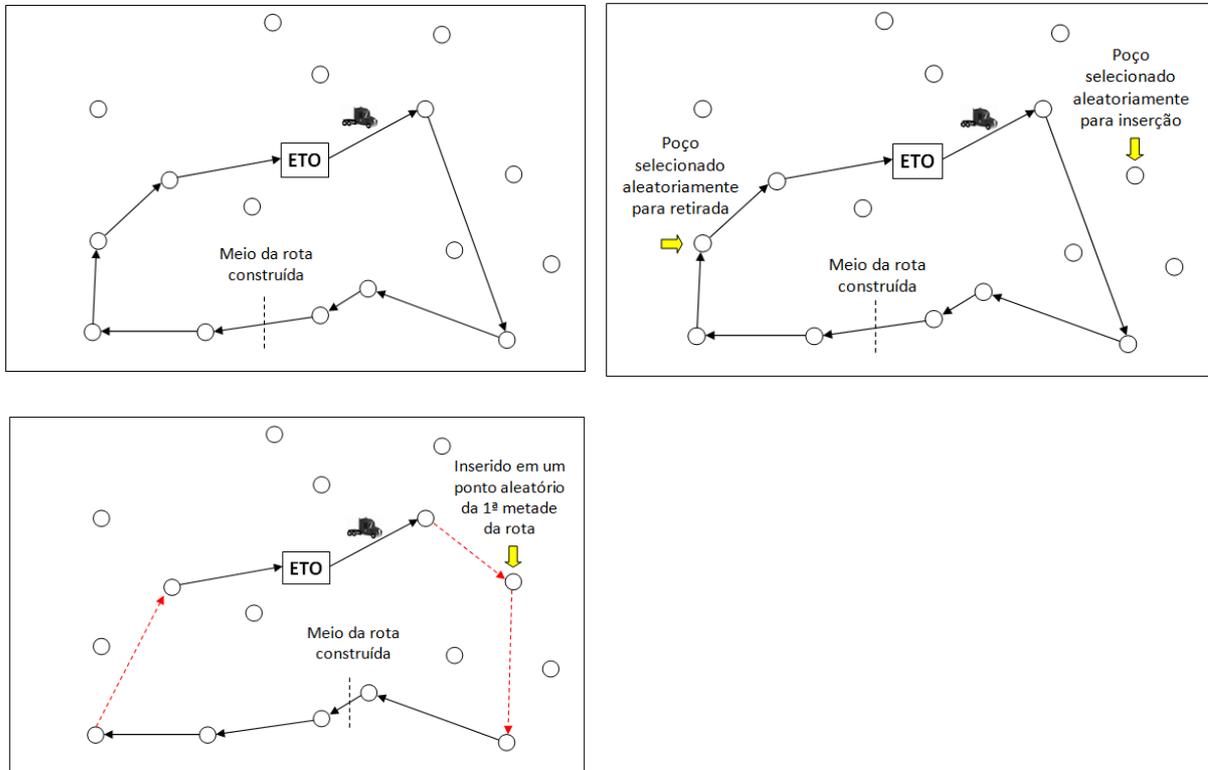


Figura 4.6: Exemplo da aplicação da perturbação P2.

4.2.3 – PERTURBAÇÃO P3

Algoritmo P3 (s_0)

1. $s \leftarrow s_0$;
2. $LC \leftarrow \text{cria_lista}(\text{poços}, s)$;
3. $LRC \leftarrow \text{cria_lista}(LC, s, \text{alfa})$;
4. $p_1 \leftarrow \text{aleatório1}(s)$;
5. $s \leftarrow \text{remove_poço}(s, p_1)$;
6. $\text{candidato} \leftarrow \text{seleciona_poço}(LRC)$;
7. $s \leftarrow \text{insere_poço}(s, \text{candidato})$;
8. se $(f(s) > f(s_0))$
9. $s_0 \leftarrow s$;
10. fim se;
11. retornar (s_0);

Figura 4.7: Perturbação P3.

A estratégia de perturbação P3, apresentada na Figura 4.7, faz na linha 1 uma cópia da rota original, s_0 , para uma rota s , que sofrerá as operações de perturbação. Na linha 2, é criada uma lista de candidatos, contendo os poços que não fazem parte da solução s , e a partir dela, é criada uma LRC (linha 3) a partir dos poços candidatos, cujo tamanho é regulado por um parâmetro *alfa*. Na linha 4 é selecionado aleatoriamente um poço localizado na primeira metade da rota s , para que em seguida (linha 5) seja retirado da mesma. Nas linhas 6 e 7, respectivamente, um poço é selecionado aleatoriamente da LRC e inserido no lugar do poço retirado anteriormente, gerando uma nova rota, s . Na linha 8, a qualidade da rota s é comparada com a da rota s_0 original, e caso seja superior, passa a ocupar o lugar de s_0 como a melhor solução conhecida (linha 9). A Figura 4.8 ilustra as operações de geração de uma nova rota por meio da perturbação P3.

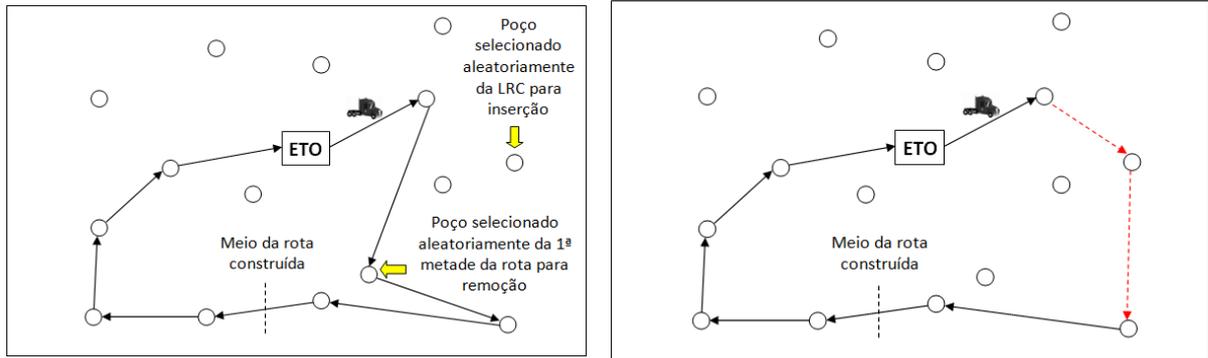


Figura 4.8: Exemplo da aplicação da perturbação P3.

4.2.4 – PERTURBAÇÃO P4

Algoritmo P4 (s)

1. $s \leftarrow s_0$;
2. $p_1 \leftarrow \text{aleatório}(s)$;
3. $pos_1 \leftarrow \text{posição}(p_1)$;
4. $pos_2 \leftarrow pos_1$;
5. enquanto ($pos_2 = pos_1$)
6. $pos_2 \leftarrow \text{posição_aleatória}(s)$;
7. fim enquanto;
8. $s \leftarrow \text{remove_poço}(s, p_1)$;
9. $s \leftarrow \text{insere_poço}(s, p_1, pos_2)$;
10. se ($f(s) > f(s^*)$)
11. $s^* \leftarrow s$;
12. fim se;
13. retorne (s^*);

Figura 4.9: Perturbação P4.

A troca da posição de um poço dentro da rota estabelecida é implementada por meio da perturbação P4, apresentada na Figura 4.9. Um poço p_1 é selecionado aleatoriamente (linha 2), para ter sua posição na rota atual modificada. Em seguida, entre as linhas 5 a 7, uma posição na rota, necessariamente distinta da posição original de p_1 , é selecionada. Em seguida, p_1 é retirado de sua posição original e reinserto na rota na nova posição escolhida (linhas 8 e 9, respectivamente). A nova solução gerada é comparada com a melhor já conhecida, e se apresentar melhora, ela passa a ser considerada a melhor solução para o problema.

4.2.5 – PERTURBAÇÃO P5

Algoritmo P5 (s)

1. $s \leftarrow s_0$;
2. $p_1 \leftarrow \text{aleatório}(s)$;
3. $p_2 \leftarrow p_1$;
4. enquanto ($p_2 = p_1$)
5. $p_2 \leftarrow \text{aleatório}(s)$;
6. fim enquanto;
7. $pos_1 \leftarrow \text{posição}(p_1)$;
8. $pos_2 \leftarrow \text{posição}(p_2)$;
9. $s \leftarrow \text{remove_poço}(s, p_1)$;
10. $s \leftarrow \text{remove_poço}(s, p_2)$;
11. $s \leftarrow \text{insere_poço}(s, p_1, pos_2)$;
12. $s \leftarrow \text{insere_poço}(s, p_2, pos_1)$;
13. se ($f(s) > f(s^*)$)
14. $s^* \leftarrow s$;
15. fim se;
16. retorne (s^*);

Figura 4.10: Perturbação P5.

A perturbação P5, apresentada na Figura 4.10, realiza a troca de posição de dois poços, selecionados aleatoriamente, da rota atual. Inicialmente, um poço p_1 é selecionado de maneira aleatória (linha 2), e logo em seguida é escolhido o poço p_2 , que será trocado de lugar com p_1 . Uma vez definidas as posições na rota que serão modificadas, os poços são retirados (linhas 9 e 10) e reinseridos na rota (linhas 11 e 12), cada um ocupando a posição deixada pelo outro poço retirado. Por fim, é verificado se a rota modificada é superior em relação à melhor solução conhecida, e em caso verdadeiro, passa a ser referenciada como a melhor solução já encontrada para o PRP-UMP.

4.2.6 – PERTURBAÇÃO P6

Algoritmo P6 (s)

1. $s \leftarrow s_0$;
2. $p_1 \leftarrow \text{aleatório}(s)$;
3. $p_2 \leftarrow p_1$;
4. enquanto ($p_2 = p_1$)
5. $p_2 \leftarrow \text{aleatório}(s)$;
6. fim enquanto;
7. enquanto ($p_1 \neq p_2$)
8. $aux \leftarrow p_1$;
9. $p_1 \leftarrow \text{próximo}(p_1)$;
10. $s \leftarrow \text{insere}(s, aux, \text{próximo}(p_2))$;
11. $s \leftarrow \text{remove}(s, aux)$;
12. fim enquanto;
13. se ($f(s) > f(s^*)$)
14. $s^* \leftarrow s$;
15. fim se;
16. retorne (s^*);

Figura 4.11: Perturbação P5.

A inversão de uma sequência de poços da rota é implementada por meio da perturbação P6, apresentada na Figura 4.11. Dois poços, p_1 e p_2 , são selecionados aleatoriamente da rota, com uma restrição que garanta que são poços distintos (linhas 2 a 6). Uma vez definida a sub-rota que será invertida, os poços serão retirados sequencialmente do início e reinsertos na posição adequada para que a ordem de visita seja modificada para o intervalo compreendido entre as posições originais de p_1 e p_2 (linhas 7 a 12). Uma vez concluída a inversão da sub-rota, é verificado se rota modificada apresenta alguma melhoria em relação à melhor rota já conhecida, e caso esta suposição se confirma, ela passa a ocupar a posição da melhor rota conhecida para o PRP-UMP.

4.2.7 – PERTURBAÇÃO P7

Algoritmo P7 (s)

1. $a_1 \leftarrow \text{seleciona_aresta}(s)$;
2. $a_2 \leftarrow \text{seleciona_aresta}(s)$;
3. $p_1 \leftarrow \text{inicio_aresta}(a_1)$;
4. $p_2 \leftarrow \text{inicio_aresta}(a_2)$;
5. $p_3 \leftarrow \text{final_aresta}(a_1)$;
6. $p_4 \leftarrow \text{final_aresta}(a_2)$;
7. $s \leftarrow \text{remove_aresta}(s, a_1)$;
8. $s \leftarrow \text{remove_aresta}(s, a_2)$;
9. $s \leftarrow \text{cria_aresta}(s, p_1, p_2)$;
10. $s \leftarrow \text{cria_aresta}(s, p_3, p_4)$;
11. se $(f(s) > f(s^*))$
12. $s^* \leftarrow s$;
13. fim se;
14. retorne (s^*) ;

Figura 4.12: Perturbação P7.

A operação de perturbação P7, representada pela Figura 4.12, implementa o operador intra-rota *2-optimal* [Layeb *et al*, 2013]. Nas linhas 1 e 2 são selecionadas as arestas da rota atual que serão removidas, e são marcados os poços dos quais se originam (linhas 3 e 4) e que terminam (linhas 5 e 6) estas arestas. Nas linhas 7 e 8, as arestas selecionadas da rota original são removidas, e em seguida (linhas 9 e 10) são criadas novas arestas de forma a manter a estrutura da rota. O último passo consiste na comparação entre a rota modificada pela perturbação e a melhor rota já encontrada para o problema. Caso seja observada uma melhoria na nova solução, ela passa a ocupar a posição de melhor solução encontrada para o problema até aquele momento. A Figura 4.13 exemplifica as operações feitas em um trecho de uma rota criada para uma UMP.

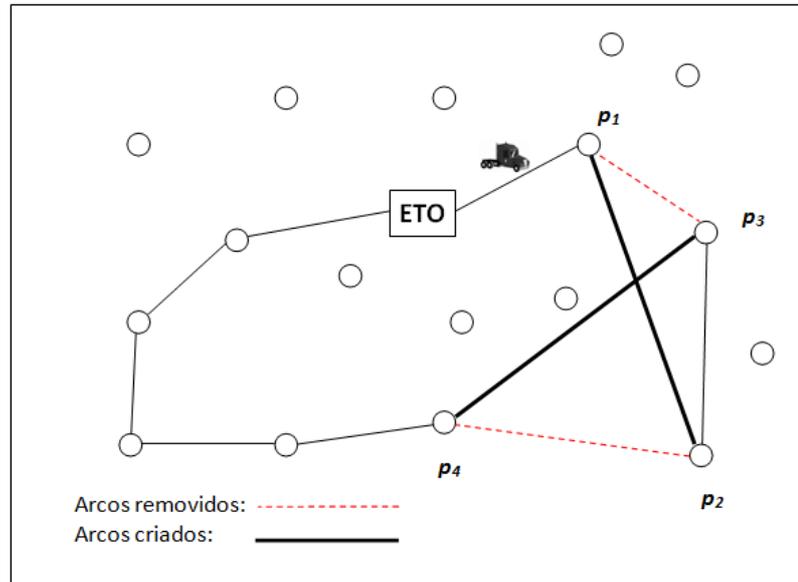


Figura 4.13: Exemplo da operação 2-óptima da rota de uma UMP.

4.2.8 – PERTURBAÇÃO P8

A estratégia de perturbação P8 é o primeiro método de perturbação inter-rotas, empregada nas execuções onde mais de uma UMP faz as operações de pistoneio aos poços. Consiste na retirada de um poço de uma rota s_1 , para que seja inserido na melhor posição possível de uma rota s_2 . A Figura 4.14 ilustra o pseudo-código para a implementação da perturbação P5.

Algoritmo P8 (s_1, s_2)

1. $p_1 \leftarrow \text{seleciona_poço}(s_1)$;
2. $s_1 \leftarrow \text{remove_poço}(p_1)$;
3. $\text{determina_melhor_posição}(p_1, s_2)$;
4. $s_2 \leftarrow \text{insere_poço}(s_2, p_1)$;
5. retorne (s_1, s_2);

Figura 4.14: Perturbação P8.

Na linha 1, é selecionado aleatoriamente um poço pertencente a rota s_1 para que seja trocado pelo operador 1-0 [Layeb *et al*, 2013]. Após sua retirada (linha 2), é determinada a melhor posição de inserção na rota s_2 (linha 3), isto é, a posição que representará o menor

acrécimo no tempo total da rota, e em seguida, a operação de inserção é efetuada (linha 4). A Figura 4.15 exemplifica o processo da troca de poço entre as rotas.

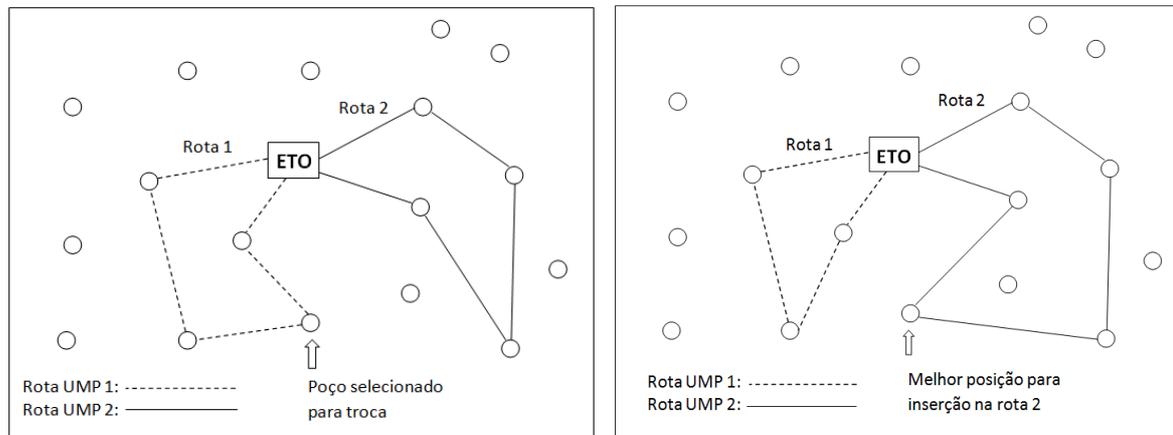


Figura 4.15: Exemplo do operador de troca 1-0 entre as rotas de duas UMP's.

4.2.9 – PERTURBAÇÃO P9

A estratégia de perturbação P9 implementa o operador de troca 1-1 [Layeb *et al*, 2013], no qual são selecionadas as rotas de duas UMP's distintas e em cada uma é selecionado um poço de forma aleatória. Os poços escolhidos são retirados de suas rotas originais e inseridos na outra rota selecionada, na posição em que sua inclusão resulte na rota de menor duração possível. A Figura 4.16 ilustra o pseudo-código da perturbação P9.

Algoritmo P9 (s_1, s_2)

1. $p_1 \leftarrow \text{seleciona_poço}(s_1)$;
2. $p_2 \leftarrow \text{seleciona_poço}(s_2)$;
3. $s_1 \leftarrow \text{remove_poço}(p_1)$;
4. $s_2 \leftarrow \text{remove_poço}(p_2)$;
5. $\text{determina_melhor_posição}(p_2, s_1)$;
6. $\text{determina_melhor_posição}(p_1, s_2)$;
7. $s_1 \leftarrow \text{insere_poço}(s_1, p_2)$;
8. $s_2 \leftarrow \text{insere_poço}(s_2, p_1)$;
9. **retorne** (s_1, s_2);

Figura 4.16: Perturbação P9.

Nas linhas 1 e 2 são selecionados os poços p_1 e p_2 , respectivamente, que serão manipulados pelo operador de troca 1-1. Em seguida, os poços são removidos de suas rotas originais (linhas 3 e 4), a melhor posição de inserção na rota em que serão inseridos é determinada nas linhas 5 e 6, e finalmente, nas linhas 7 e 8, novas rotas são constituídas com a inserção dos poços. A Figura 4.17 exemplifica uma operação de troca baseada na perturbação P6 entre as rotas de duas UMP's.

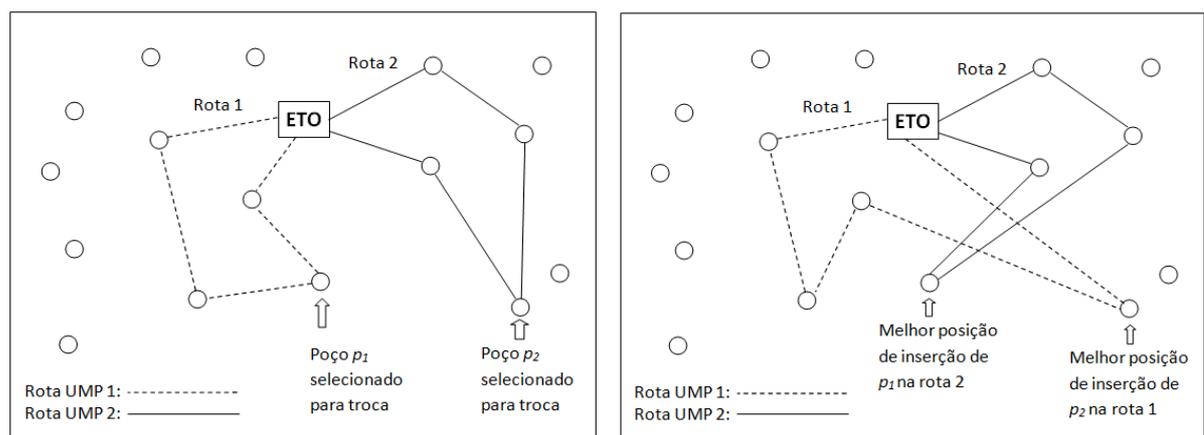


Figura 4.17:

Exemplo do operador de troca 1-1 entre as rotas de duas UMP's.

4.2.10 – PERTURBAÇÃO P10

A perturbação P10 implementa o operador de troca *2-optimal* [Layeb *et al*, 2013], que, considerando as rotas construídas para duas UMP's distintas, seleciona duas arestas, a_1 e a_2 , das rotas s_1 e s_2 , respectivamente. Em seguida, cria uma nova aresta conectando o nó inicial da aresta a_1 com o nó final da aresta a_2 , e outra aresta conectando o nó inicial da aresta a_2 com o nó final da aresta a_1 . As arestas a_1 e a_2 , selecionadas originalmente, são removidas, e duas novas rotas são geradas. A Figura 4.18 ilustra o pseudo-código da perturbação P10.

Algoritmo P10 (s_1, s_2)

1. $a_1 \leftarrow \text{seleciona_aresta}(s_1)$;
2. $a_2 \leftarrow \text{seleciona_aresta}(s_2)$;
3. $s_1 \leftarrow \text{cria_aresta}(s_1, \text{inicio}(a_1), \text{final}(a_2))$;
4. $s_2 \leftarrow \text{cria_aresta}(s_2, \text{inicio}(a_2), \text{final}(a_1))$;
5. $s_1 \leftarrow \text{remove_aresta}(s_1, a_1)$;
6. $s_2 \leftarrow \text{remove_aresta}(s_2, a_2)$;
7. retorne (s_1, s_2);

Figura 4.18: Perturbação P10.

Nas linhas 1 e 2 são selecionadas aleatoriamente as arestas a_1 e a_2 , pertencentes às rotas s_1 e s_2 , respectivamente. Duas novas arestas são criadas, uma originando-se no vértice inicial da aresta a_1 e terminando no vértice final da aresta a_2 , e a outra começando em no vértice inicial de a_2 e terminando no vértice final de a_1 . As arestas selecionadas originalmente são excluídas (linhas 5 e 6), resultando em duas novas rotas, formadas a partir das rotas originais. A Figura 4.19 ilustra a operação de troca *2-optimal*, implementada pela perturbação P10.

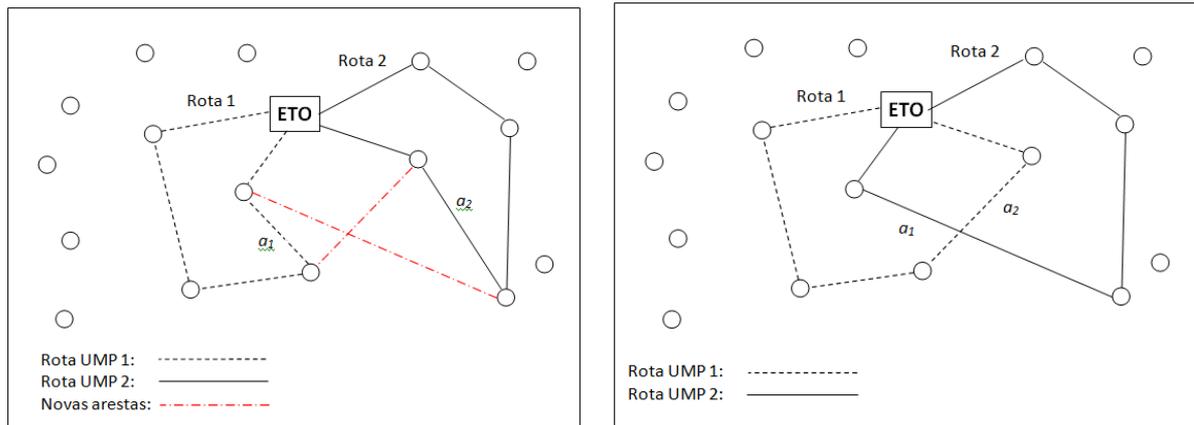


Figura 4.19: Exemplo do operador de troca 2-optimal entre as rotas de duas UMP's.

4.3 – ESTRATÉGIA REATIVA DE SELEÇÃO DAS PERTURBAÇÕES

Foi empregada uma estratégia reativa para a seleção de um dos métodos de perturbação propostos a ser aplicado na solução atual, gerada por uma das heurísticas. O conjunto de perturbações consideradas para seleção é formado de acordo com a quantidade de UMP's empregadas nas operações de pistoneio, pois as operações intra-rota são utilizadas quando há somente uma rota construída por uma UMP a cada dia, e as operações inter-rota fazem modificações entre as rotas de diferentes UMP's que operam simultaneamente.

Para ambos os casos, a cada iteração da heurística, um dos métodos de perturbação é selecionado de uma lista P contendo todas as perturbações disponíveis para uso. Inicialmente, a lista P contém todos os métodos de perturbação propostos para a quantidade de UMP's empregada. Quando uma perturbação é selecionada para ser aplicada, ela é removida da lista P , tornando a seleção de perturbação da iteração seguinte mais restrita. A perturbação selecionada para ser utilizada na iteração atual é selecionada aleatoriamente dentre as disponíveis na lista. Quando a lista encontra-se vazia, todos os métodos de perturbação são reinsertidos na lista P .

Durante as N iterações iniciais da heurística, todos os métodos de perturbação são empregados para a composição da lista P de operações disponíveis para seleção. O valor atribuído para N é igual a 25% do limite empregado como critério de parada, que pode ser um número máximo de iterações previamente definido ou o tempo limite de execução do algoritmo. A cada $|P|$ iterações realizadas, cada uma das operações de perturbação empregadas terá sido utilizada uma vez, e a melhor solução obtida dentre as geradas nestas

iterações é identificada, e a perturbação responsável por gerá-la é pontuada em uma classificação que identifica o total de vezes que cada perturbação foi capaz de obter o melhor resultado no sub-grupo de N iterações do algoritmo.

Quando terminadas estas iterações preliminares, um subconjunto P' , formado pelas perturbações que obtiveram a melhor classificação, representando as operações que foram capazes de gerar as soluções de melhor qualidade, é selecionado para ser utilizado nas demais iterações da heurística. Foi estabelecido que para o conjunto das sete operações de perturbação intra-rota o subconjunto P' é formado por três elementos, e para as operações inter-rota, que possuem três operadores de perturbação, o subconjunto P' é formado por dois elementos.

4.4 – SIMULATED ANNEALING

O *simulated annealing* é uma metaheurística de otimização global proposta por [Kirkpatrick, Gellat & Vechhi, 1983], baseada no processo utilizado na metalurgia para temperar ou enrijecer metais e vidros por meio da elevação da temperatura da substância até seu ponto de fusão, para em seguida resfriá-lo gradualmente, em um processo controlado, o que permite que o material seja misturado em um estado cristalino mais estável, estruturalmente mais forte e de menor energia. Durante o processo de resfriamento, o material passa por vários estados possíveis, até atingir seu estado ótimo.

Em problemas de otimização combinatória, podemos relacionar o processo de resfriamento da têmpera com a geração de soluções de um PRV. Em um problema de minimização, a diminuição de temperatura representa a diminuição da probabilidade de aceitar a exploração de soluções de pior qualidade, à medida que o espaço de soluções do problema é explorado. A possibilidade de explorar soluções de pior qualidade é uma parte fundamental do processo, pois minimiza as chances do algoritmo ficar preso a um ótimo local, explorando áreas do espaço de soluções que não seriam analisadas de outra forma.

Assim como na heurística proposta baseada em ILS, os testes do GRASP+SA empregaram as versões do GRASP resultantes da combinação do método construtivo C3 com as estratégias de busca local BL4 e BL5, pois estas versões mostraram-se capazes de obter os melhores resultados nos testes realizados com as diversas versões do GRASP.

A Figura 4.20 ilustra o algoritmo de *simulated annealing*. Inicialmente, a solução atual s é registrada como a melhor solução conhecida, s^* (linha 1), e as variáveis de controle de

temperatura T e número de iterações são inicializadas (linhas 2 e 3). O *simulated annealing* é executada entre as linhas 4 e 22 enquanto o valor de temperatura mínimo não for atingido. Para o valor da temperatura atual, serão realizadas *max_iterações* entre as linhas 5 e 19; uma das estratégias de perturbação da solução disponíveis para a temperatura será selecionada e aplicada a solução atual, gerando uma nova rota, s' (linhas 7 e 8, respectivamente). Em seguida, obtém-se o valor de *delta*, que representa a diferença entre o valor obtido pela função objetivo f para a solução gerada pela perturbação e a solução inicial (linha 9), e caso seja observado uma melhoria na solução, de acordo com o critério adotado (linha 10), a nova solução será utilizada nas próximas iterações, além de ser registrada como a melhor solução conhecida (linha 13). Se a nova solução apresentar qualidade inferior à solução original, é calculada a probabilidade para determinar se a solução gerada será aceita para a próxima iteração, como uma estratégia de escapar de possíveis ótimos locais (linha 16). No início do processo, quando a temperatura está mais elevada, as chances de se aceitar soluções piores é maior, porém, conforme a temperatura se resfria, a probabilidade do algoritmo aceitar soluções de qualidade inferior diminui, e a heurística passa a agir como um método de subida de encosta, aceitando apenas movimentos de melhora de solução, o que evidencia a obtenção de um ótimo local, que eventualmente pode ser o ótimo global.

Ao atingir o número máximo de iterações para o valor de temperatura T definido no início da estrutura interna de repetição, a temperatura é diminuída e o contador de iterações é reiniciado (linhas 20 e 21, respectivamente), e todo o processo de perturbação é executado novamente para a solução s atual. Repete-se o processo até que a temperatura mínima tenha sido atingida, e termina com o retorno da melhor solução encontrada (linha 24).

Algoritmo *Simulated Annealing* (s)

1. $s^* \leftarrow s$;
2. iterações $\leftarrow 0$;
3. $T \leftarrow$ temperatura_inicial;
4. enquanto ($T >$ temperatura_mínima) faça;
5. enquanto (iteraões \leq max_iteracoes) faça
6. iteraões \leftarrow iteraões +1;
7. perturbação \leftarrow Selecciona_Perturbação ();
8. $s' \leftarrow$ Perturba_Solução (s , perturbação);
9. $delta \leftarrow f(s') - f(s)$;
10. se ($delta < 0$)
11. $s \leftarrow s'$;
12. se ($f(s') > f(s^*)$)
13. $s^* \leftarrow s'$;
14. senão
15. $x \leftarrow$ Gera_Aleatório (0, 1);
16. se ($x < \exp^{(-delta/T)}$)
17. $s \leftarrow s'$;
18. fim se;
19. fim enquanto;
20. $T \leftarrow T * alfa$;
21. iteraões $\leftarrow 0$;
22. fim enquanto;
23. $s \leftarrow s^*$;
24. retorne (s);

Figura 4.20: Metaheurística *simulated annealing* para um problema de maximização.

4.4.1 – SIMULATED ANNEALING PARA O PRP-UMP CONSIDERANDO A UTILIZAÇÃO DE UMA UMP

Algoritmo PRP-UMP GRASP+SA para 1 UMP ()

1. poços \leftarrow lê entrada de dados (conjunto_poços, volume, distância);
2. para x de 1 até (limite de iterações ou limite de tempo) faça
3. para y de 1 até (máximo de dias) faça
4. $s \leftarrow$ Construtivo (poços);
5. $s' \leftarrow$ Busca_Local (s);
6. se $f(s') > f(s)$
7. $s \leftarrow s'$;
8. fim se;
9. $s^* \leftarrow s$;
10. iteraões $\leftarrow 0$;
11. $T \leftarrow$ temperatura_inicial;
12. enquanto ($T >$ temperatura_mínima) faça
13. enquanto (iteraões \leq max_iteraões) faça
14. iteraões \leftarrow iteraões +1;
15. $p \leftarrow$ Selecciona_Perturbação_Intra_Rota ();
16. $s'' \leftarrow$ Perturba (s, p);
17. $delta \leftarrow f(s'') - f(s)$;
18. se ($delta > 0$)
19. $s \leftarrow s''$;
20. $s^* \leftarrow s$;
21. senão
22. $x \leftarrow$ Gera_Aleatório (0, 1);
23. se ($x < \exp^{(-delta/T)}$)
24. $s \leftarrow s''$;
25. fim se;
26. fim se;

Algoritmo PRP-UMP GRASP+SA para 1 UMP ()

```

27.          fim enquanto;
28.           $T \leftarrow T * \text{alfa}$ ;
29.          iterações  $\leftarrow 0$ ;
30.          fim enquanto;
31.      fim para;
32.  fim para;
33.  retorne ( $s^*$ ).

```

Figura 4.21: Metaheurística híbrida GRASP+SA para o PRP-UMP com a utilização de uma UMP.

A versão proposta do algoritmo de *simulated annealing* para o PRP-UMP utilizando uma unidade móvel de pistoneio é apresentada na Figura 4.21. Após a leitura inicial das informações sobre o conjunto de poços considerados para visitaç o, iniciam-se as iteraç es do GRASP, limitadas por um valor pr -estabelecido ou um limite de tempo definido, e para cada dia de um horizonte de eventos proposto, realiza-se a constru o de uma solu o inicial s para o problema na linha 4, a partir qual ser  realizada uma busca local em seguida, na linha 5, resultando em uma solu o s' , que   comparada com a solu o s original (linha 6), e, caso seja observada uma melhoria da solu o em rela o ao volume de  leo pistoneado pela UMP, ela ser  considerada a solu o atual. Na linha 9, a melhor solu o obtida por meio do GRASP   atribu da a vari vel s^* , que representa a melhor solu o conhecida para o conjunto de poços que est  sendo analisado.

Nas linhas 10 e 11 s o inicializadas as vari veis de controle do n mero de itera es que ser o realizadas pela t mpera simulada, compreendida entre as linhas 12 a 31. Uma das opera es de perturba o intra-rota dispon veis para sele o a partir do reposit rio de perturba es   selecionada na linha 15, e aplicada   solu o s na linha 16, resultando em uma nova solu o, s'' . Caso a nova solu o encontrada apresente menor dura o que a anterior, ela passa a substituir-la como a solu o atual (linha 18).

A diferen a entre os valores obtidos pela fun o objetivo f aplicada  s solu es s'' e s , respectivamente,   atribu da, na linha 17,   vari vel de decis o *delta*. Caso o valor de *delta* seja positivo (linha 18), o que ocorre quando a solu o gerada pela busca local a partir da solu o perturbada mostra-se superior   solu o inicial, a solu o s , que ser  utilizada na

próxima iteração é atualizada com a solução s'' encontrada, assim como s^* , que representa a melhor solução conhecida até o momento (linhas 19 e 20, respectivamente).

Caso o valor calculado para delta seja menor ou igual a zero, é realizado um cálculo de probabilidade para determinar se a solução s'' , uma solução de pior qualidade do que s , será aceita como a solução atual, implementando uma estratégia para tentar escapar de ótimos locais nas próximas iterações (linhas 22 a 25).

Antes de começar a próxima iteração do SA, o valor da temperatura é atualizado (linha 28) e o número de iterações é reinicializado com o valor zero, para que sejam feitas *max_iterações* a partir da solução s atual. Ao terminarem todas iterações do GRASP para todos os dias do horizonte de eventos, a melhor solução obtida, s^* , é retornada.

A estratégia reativa para a seleção de um dos métodos de perturbação propostos, apresentada no item 4.3 é utilizada no algoritmo de SA proposto da mesma forma que foi empregada na solução que empregou a heurística ILS.

4.4.2 –SIMULATED ANNEALING PARA O PRP-UMP CONSIDERANDO A UTILIZAÇÃO DE MÚTIPLAS UMP'S

Algoritmo PRP-UMP GRASP+SA para Múltiplas UMP'S ()

1. poços \leftarrow lê entrada de dados (conjunto_poços, volume, distância);
2. para x de 1 até (limite de iterações ou limite de tempo) faça
3. para y de 1 até (máximo de dias) faça
4. para i de 1 até (número de UMP's) faça
5. $s(i) \leftarrow$ Construtivo (poços);
6. $s'(i) \leftarrow$ Busca_Local ($s(i)$);
7. se $f(s'(i)) > f(s(i))$
8. $s(i) \leftarrow s'(i)$;
9. fim se;
10. Atualiza_Poços_Candidatos ();
11. fim para;
12. para i de 1 até (número de UMP's) faça
13. $s^*(i) \leftarrow s(i)$;
14. $iterações \leftarrow 0$;
15. $T \leftarrow$ Temperatura_Inicial;
16. enquanto ($T > temperatura_mínima$) faça;
17. enquanto ($iterações \leq max_iterações$) faça
18. $iterações \leftarrow iterações + 1$;
19. $j \leftarrow$ (aleatório (UMP's) $\neq i$)
20. $p' \leftarrow$ Seleciona Perturbação Inter_Rota ();
21. $s''(i), s'''(i) \leftarrow$ Perturba ($s(i), s(j), p'$);
22. $delta_1 \leftarrow f(s''(i)) - f(s(i))$;
23. $delta_2 \leftarrow f(s'''(i)) - f(s(i))$;
24. se $delta_1 > 0$
25. $s(i) \leftarrow s''(i)$;
26. $s^*(i) \leftarrow s(i)$;
27. senão

```

28.          x ← gera_aleatório (0, 1);
29.          se (x < exp(-delta_1/T))
30.              s(i) ← s''(i);
31.          fim se;
32.      fim se;
33.      se delta_2 > 0
34.          s(i) ← s'''(i);
35.          s*(i) ← s(i);
36.      senão
37.          x ← gera_aleatório (0, 1);
38.          se (x < exp(-delta_2/T))
39.              s(i) ← s'''(i);
40.          fim se;
41.      fim se;
42.      fim enquanto;
43.      T ← T * alfa;
44.      iterações ← 0;
45.      fim enquanto;
46.      s(i) ← s*(i);
47.      Atualiza_Informações_Dos_Poços ( );
48.      fim para;
49.      fim para;
50.      fim para;
51.      retorne (rotas s);

```

Figura 4.22: Metaheurística GRASP com *simulated annealing* para o PRP-UWP com a utilização de múltiplas UWP's.

O algoritmo para a resolução do PRP-UMP empregando GRASP e *simulated annealing* considerando a utilização simultânea de múltiplas UMP's, apresentado pela Figura 4.22, inicia com a leitura dos dados de entrada dos poços a serem considerados para visitaç o e em seguida, o in cio das itera es do GRASP, que ocorrem entre as linhas 2 at  50. Entre as linhas 4 e 11 inicia-se a constru o de uma solu o inicial para cada uma das i UMP's, por meio de um m todo construtivo (linha 5), seguido de uma busca local (linha 6). Caso a busca local encontre uma solu o $s'(i)$ com um maior valor para a fun o objetivo f do que a da solu o $s(i)$ encontrada pela busca local, a solu o atual   atualizada (linha 8). Na linha 10 a lista de po os dispon veis para pistoneio   atualizada antes de iniciar a etapa do *simulated annealing* para as i UMP's.

Antes do in cio das opera es do *simulated annealing*, a melhor solu o $s(i)$ de cada uma das UMP's, s o copiadas para $s^*(i)$, que representa a melhor solu o j  encontrada para a UMP i . Nas linhas 14 e 15 s o inicializados o n mero de itera es e a temperatura inicial da t mpera, respectivamente, para o in cio das itera es da UMP i . Enquanto a temperatura m nima n o tiver sido atingida (linha 16) e o n mero m ximo de itera es n o tiver sido alcan ado (linha 17),   selecionada na linha 19, uma UMP j , necessariamente diferente da UMP i atual, para que sua rota, $s(j)$, seja utilizada pela perturba o inter-rota p' , selecionada na linha 20, que ao ser aplicada nas solu es $s(i)$ e $s(j)$, resulta nas novas solu es $s''(i)$ e $s'''(i)$.

As diferen as entre os resultados da fun o objetivo aplicada a estas solu es e o valor obtido utilizando a solu o $s(i)$ atual s o armazenadas nas vari veis $delta_1$ e $delta_2$. Na linha 24, verifica-se se o valor de $delta_1$   positivo, o que significa que a nova solu o $s''(i)$   melhor do que a solu o $s(i)$ atual. Neste caso, $s''(i)$ passa a ocupar o lugar da solu o $s(i)$ atual (linha 25) e   armazenada em $s^*(i)$, a melhor solu o encontrada para a UMP i (linha 26). Caso contr rio, ser  calculada a probabilidade do algoritmo utilizar a solu o $s''(i)$ na pr xima itera o, mesmo que essa apresente um resultado inferior ao da solu o $s(i)$ quando aplicada a fun o objetivo f , mas como uma forma de tentar explorar diferentes  reas do espa o de solu es do problema, por meio da diversifica o das solu es analisadas. Entre as linhas 33 at  41 a mesma an lise   realizada para a outra solu o, $s'''(i)$, gerada pela opera o de perturba o da linha 21.

Quando o n mero m ximo de itera es da t mpera utilizando a temperatura atual for atingido, o valor da temperatura   diminuido (linha 43), representando o resfriamento da

têmpera, o que para o algoritmo representa a diminuição da probabilidade de aceitação de soluções de menor qualidade para a próxima iteração. Ao terminar as iterações da têmpera para a UMP i atual, o valor de $s(i)$ é atualizado com a melhor solução conhecida para i . Na linha 47, as informações dos poços disponíveis para pistoneio são atualizadas para que o algoritmo seja executado para outra UMP. Depois do GRASP e o *simulated annealing* terem sido executadas para todas as UMP's, o melhor resultado encontrado para cada UMP é retornada na linha 51.

CAPÍTULO 5 – RESULTADOS COMPUTACIONAIS

Nesta seção são apresentados os resultados computacionais obtidos pelas diferentes versões do GRASP, resultantes das combinações entre os métodos construtivos associados a uma das estratégias de busca local apresentadas, e as heurísticas híbridas GRASP+ILS e GRASP+SA empregando as operações de perturbação propostas, apresentadas no Capítulo 4, além dos resultados obtidos por meio da implementação de um método exato no IBM ILOG CPLEX. Os testes foram realizados em um computador Pentium Dual Core 1.87 Ghz e 3 GB de memória principal. Os programas foram implementados na linguagem C segundo o padrão ANSI e compilados com GCC versão 4.9.2, e foi empregado o gerador de números aleatórios Mersenne Twister versão 1.23 [Matsumoto and Nishimura, 1998].

5.1 – RESULTADOS OBTIDOS COM A UTILIZAÇÃO DE UMA UMP

Os testes foram realizados a partir do conjunto de 27 instâncias propostas em [Goldbarg *et al*, 2010]. As instâncias foram geradas a partir do conjunto de instâncias empregadas para o problema do caixeiro viajante assimétrico (PCVA) da TSPLIB [Reinelt, 1995], e a parte numérica presente na identificação de cada instância identifica a quantidade de poços disponíveis para visitação. Os tempos de reenchimento dos poços para que atinjam 99,9% de sua capacidade máxima, assim como o volume máximo dos poços foram gerados aleatoriamente dentro de faixas de valores fornecidas pela Petrobrás. A taxa de reenchimento dos poços é representada por uma tabela, cuja primeira coluna representa o volume máximo de cada poço e a cada coluna n subsequente, o volume acumulado do reenchimento caso o poço tenha sido visitado a $n-1$ dias atrás.

Para cada instância, executou-se uma versão do GRASP identificada pelo método construtivo empregado em conjunto com uma das estratégias de busca local, por exemplo, exemplo: C1BL1 representa o GRASP composto pelo método construtivo C1 com a busca local BL1. Nestes primeiros testes realizados, tentou-se determinar as versões do GRASP capazes de alcançar os melhores resultados, com o propósito de concentrar os esforços e utilização de outras técnicas de otimização para a obtenção de melhores resultados a partir das versões que inicialmente demonstraram o maior potencial em encontrar as melhores soluções para o PRP-UMP.

Para cada uma das 15 combinações possíveis, executaram-se preliminarmente 10 execuções, cada uma com um valor distinto de semente, e com o número de iterações do GRASP

fixado em 200 e o parâmetro *alfa* em 0.1. Os resultados destes testes são apresentados no apêndice 1, na forma de *gaps* (isto é, a diferença percentual entre os valores encontrados por cada estratégia gerada pelos métodos construtivo e de busca local [Goldbarg *et al*, 2010]). As colunas identificadas como 'Vol. Max.' representam os *gaps* entre as melhores soluções obtidas dentre todas as execuções realizadas para o GRASP aqui proposto e os resultados da literatura. Os valores de *gaps positivos* identificam uma melhora na solução do algoritmo GRASP proposto neste trabalho em relação aos resultados da literatura, enquanto os valores de *gaps negativos* representam resultados inferiores aos melhores conhecidos. As colunas identificadas como 'Vol. Med.' representam os *gaps* entre os valores médios calculados a partir dos volumes obtidos nas execuções e o valor máximo encontrado. As colunas 'Vol Min.' apresentam os *gaps* calculados entre a melhor e a pior solução obtida entre as execuções realizadas. Uma vez que a média de um conjunto de valores não pode ser superior ao maior valor contido no conjunto, o valor destes *gaps* sempre será nulo ou negativo, e indica se a média dos resultados está próxima ou distante do melhor resultado observado. Por exemplo, um *gap* igual a 2 significa que o resultado obtido com aquela estratégia para uma certa instância é dois por cento melhor em relação ao resultado com o qual está sendo comparado. A coluna '*T Med.(s)*' indica a quantidade média de tempo, medida em segundos, que foi necessária ao algoritmo para completar as 200 iterações.

5.1.1 – RESULTADOS OBTIDOS POR MEIO DO USO DA HEURÍSTICA GRASP

A partir dos resultados alcançados nestes testes preliminares empregando somente o GRASP foi possível observar que as versões C3BL4 e C3BL5 foram capazes de obter resultados melhores às demais, que apresentaram diferenças pouco significativas entre si, e, portanto, não foram utilizadas nos demais testes realizados. Os resultados obtidos pelos testes realizados a partir das estratégias C3BL4 e C3BL5, analisados a seguir, são apresentados na Tabela 5.1. Os melhores resultados obtidos pelas heurísticas, independente se são melhores ou piores do que os apresentados na literatura estão destacados em vermelho. Comparando os resultados de volume máximo obtido com os apresentados em [Goldbarg *et al*, 2010], pôde-se observar que obtiveram-se melhores soluções para 21 das 27 instâncias testadas, para ambas estratégias. Comparando os melhores resultados obtidos por cada uma das versões do GRASP, independente se estes foram melhores aos resultados apresentados pela literatura, pode-se observar que a estratégia C3BL4 foi capaz de encontrar os melhores resultados para três das instâncias testadas, enquanto C3BL5 mostrou-se melhor em 22 instâncias, e ocorreram dois empates.

Instância	C3BL4				C3BL5			
	Vol. Max.	Vol. Med.	Vol. Min.	T. Med. (s)	Vol. Max.	Vol. Med.	Vol. Min.	T. Med (s)
br17	1,78	-0,09	-0,24	0,85	1,82	-0,12	-0,24	0,84
ftv33	0,07	-0,08	-0,11	2,37	0,11	-0,11	-0,31	2,35
ftv35	0,27	-0,08	-0,18	2,81	0,33	-0,11	-0,15	2,81
ftv38	-0,58	-0,06	-0,15	4,31	-0,60	-0,04	-0,15	4,31
p43	0,43	-0,04	-0,09	3,97	0,50	-0,07	-0,15	4,09
ftv44	0,68	-0,14	-0,4	5,22	0,72	-0,13	-0,38	5,38
ftv47	0,69	-0,21	-0,55	5,40	0,75	-0,15	-0,54	5,29
ry48p	0,55	-0,23	-0,45	6,00	0,54	-0,17	-0,35	5,89
ft53	1,03	-0,26	-0,5	9,30	1,08	-0,26	-0,45	9,12
ftv55	0,36	-0,11	-0,28	9,45	0,36	-0,14	-0,26	9,54
ftv64	0,53	-0,16	-0,33	16,48	0,63	-0,16	-0,21	16,81
ftv70	0,38	-0,15	-0,27	18,63	0,48	-0,16	-0,27	18,81
ft70	-1,11	-0,23	-0,46	12,64	-1,05	-0,19	-0,44	12,64
ftv90	0,55	-0,23	-0,48	37,91	0,56	-0,18	-0,42	37,91
ftv100	0,18	-0,26	-0,6	48,68	0,18	-0,23	-0,54	48,21
ftv110	0,74	-0,07	-0,21	61,90	0,78	-0,02	-0,12	61,90
ftv120	1,10	-0,2	-0,3	86,24	1,18	-0,12	-0,23	82,95
kro124	0,76	-0,08	-0,14	37,79	0,77	-0,03	-0,14	37,07
ftv130	0,76	-0,2	-0,3	106,91	0,84	-0,23	-0,39	107,95
ftv140	0,38	-0,08	-0,2	129,81	0,41	-0,05	-0,14	127,29
ftv150	1,17	-0,21	-0,35	163,84	1,26	-0,19	-0,23	163,84
ftv160	-3,41	-0,19	-0,41	174,27	-3,43	-0,22	-0,34	177,68
ftv170	-3,13	-0,33	-0,64	218,44	-3,05	-0,29	-0,52	218,44
rbg323	-2,49	-0,28	-0,5	791,90	-2,45	-0,31	-0,38	776,67
rbg358	-0,30	-0,55	-1,09	875,67	-0,23	-0,48	-0,98	850,65
rbg403	0,20	-0,06	-0,15	1196,25	0,28	-0,05	-0,09	1173,02
rbg443	0,00	-0,19	-0,53	1779,70	0,07	-0,10	-0,48	1762,42

Tabela 5.1: Resultados computacionais do GRASP versões C3BL4 e C3BL5. Os campos ‘Vol. Max’ representam o *gap* entre a solução obtida pela heurística e o melhor resultado da literatura; os campos ‘Vol. Med.’ representam o *gap* entre o valor médio e o melhor resultado da heurística; os campos ‘Vol. Min.’ representam o *gap* entre o pior e o melhor resultado obtido pela heurística. Valores positivos representam resultados melhores do que os apresentados na literatura, e valores negativos representam resultados piores que os da literatura. Os resultados destacados em vermelho representam a melhor solução obtida por meio de um dos métodos heurísticos propostos, independente se são melhores do que os resultados da literatura.

Foram realizados novos testes com as estratégias C3BL4 e C3BL5 com as mesmas quantidades de execuções e iterações por execução utilizadas anteriormente, mas com a alteração do parâmetro *alfa*, agora fixado em 0.05. A diminuição do valor do parâmetro representa a utilização de uma LRC menor, tornando a escolha aleatória de um candidato a compor a solução mais restrita aos candidatos que apresentam maior atratividade. Para instâncias compostas por um pequeno número de poços, garante-se que haja um número mínimo de poços na LRC, para que a seleção de um elemento da lista não seja gulosa. Os resultados são apresentados na Tabela 5.2. Pode-se observar que a adoção de um valor menor para o parâmetro *alfa* contribuiu para que as versões testadas do GRASP encontrassem soluções de maior qualidade: em 17 instâncias o melhor resultado foi obtido atribuindo a *alfa* o valor 0.05, em 9 instâncias o melhor resultado foi encontrado com *alfa* igual a 0.1 e foi observada a ocorrência de um empate.

Quando comparados os resultados em relação ao método de busca local adotado, a estratégia C3BL5 foi responsável em encontrar os melhores resultados para 22 das instâncias testadas, em quatro instâncias os melhores resultados foram obtidos por meio da execução da estratégia C3BL4, e em um caso foi observado o empate entre as soluções dos métodos testados.

GRASP, 200 iterações, 10 Execuções

Instância	ProtoG	C3BL4 alfa=0.1			C3BL4 alfa=0.05			T. Med. (s)	C3BL5 alfa=0.1			C3BL5 alfa=0.05			T. Med. (s)
		Vol. Max.	Vol. Med.	Vol. Min.	Vol. Max.	Vol. Med.	Vol. Min.		Vol. Max.	Vol. Med.	Vol. Min.	Vol. Max.	Vol. Med.	Vol. Min.	
br17	1047,29	1,78	-0,09	-0,24	1,78	-0,07	-0,23	0,88	1,82	-0,12	-0,24	1,80	-0,08	-0,19	0,83
ftv33	1129,01	0,07	-0,08	-0,11	0,74	-0,17	-0,33	2,05	0,11	-0,10	-0,34	0,82	-0,16	-0,29	2,26
ftv35	1129,95	0,27	-0,08	-0,18	0,29	-0,10	-0,17	2,49	0,33	-0,11	-0,15	0,38	-0,04	-0,16	2,75
ftv38	1140,88	-0,58	-0,06	-0,15	-0,51	-0,06	-0,18	3,97	-0,60	-0,04	-0,15	-0,43	-0,05	-0,11	4,14
p43	1214,27	0,43	-0,04	-0,09	0,62	-0,15	-0,25	4,14	0,50	-0,07	-0,15	0,63	-0,14	-0,21	3,93
ftv44	1141,08	0,68	-0,14	-0,40	1,01	-0,19	-0,49	5,56	0,72	-0,13	-0,38	1,00	-0,20	-0,44	5,12
ftv47	1128,71	0,69	-0,21	-0,55	0,99	-0,17	-0,58	5,02	0,75	-0,15	-0,54	1,04	-0,10	-0,57	5,19
ry48p	1137,67	0,55	-0,23	-0,45	0,55	-0,27	-0,43	6,03	0,54	-0,17	-0,35	0,53	-0,18	-0,45	5,77
ft53	1127,47	1,03	-0,26	-0,50	1,29	-0,30	-0,62	8,27	1,08	-0,26	-0,45	1,28	-0,22	-0,54	8,94
ftv55	1141,98	0,36	-0,11	-0,28	0,41	-0,20	-0,34	8,61	0,36	-0,14	-0,26	0,43	-0,13	-0,33	9,17
ftv64	1167,26	0,53	-0,16	-0,33	0,78	-0,19	-0,33	16,75	0,63	-0,16	-0,21	0,84	-0,19	-0,30	16,32
ftv70	1172,97	0,38	-0,15	-0,27	0,45	-0,20	-0,31	18,48	0,48	-0,16	-0,27	0,44	-0,11	-0,27	17,91
ft70	1099,74	-1,11	-0,23	-0,46	-1,09	-0,07	-0,15	13,07	-1,05	-0,19	-0,44	-1,09	-0,08	-0,15	12,15
ftv90	1166,04	0,55	-0,23	-0,48	0,55	-0,21	-0,48	35,16	0,56	-0,18	-0,42	0,59	-0,21	-0,45	36,81
ftv100	1168,97	0,18	-0,26	-0,60	0,17	-0,23	-0,60	44,94	0,18	-0,23	-0,54	0,15	-0,24	-0,58	46,81
ftv110	1168,58	0,74	-0,07	-0,21	1,45	-0,15	-0,52	62,75	0,78	-0,02	-0,12	1,48	-0,16	-0,43	61,29
ftv120	1176,92	1,10	-0,20	-0,30	1,87	-0,22	-0,68	84,67	1,18	-0,12	-0,23	1,91	-0,22	-0,68	82,13
kro124	1164,64	0,76	-0,08	-0,14	0,78	-0,03	-0,17	34,72	0,77	-0,03	-0,04	0,85	-0,05	-0,19	35,99
ftv130	1174,97	0,76	-0,20	-0,30	1,10	-0,16	-0,26	101,66	0,84	-0,23	-0,19	1,16	-0,13	-0,19	103,8
ftv140	1179,06	0,38	-0,08	-0,20	1,11	-0,17	-0,52	108,94	0,41	-0,05	-0,14	1,15	-0,08	-0,52	116,03
ftv150	1181,29	1,17	-0,21	-0,35	1,14	-0,12	-0,28	129,73	1,26	-0,19	-0,23	1,22	-0,03	-0,27	133,04
ftv160	1178,13	-3,41	-0,19	-0,41	-2,97	-0,19	-0,41	142,79	-3,43	-0,22	-0,34	-3,08	-0,13	-0,41	148,85
ftv170	1183,14	-3,13	-0,33	-0,64	-3,29	-0,19	-0,26	198,50	-3,05	-0,29	-0,52	-3,20	-0,19	-0,17	201,04
rbg323	1105,74	-2,49	-0,28	-0,50	-2,43	-0,21	-0,29	613,58	-2,45	-0,31	-0,38	-2,39	-0,23	-0,29	611,44
rbg358	1099,23	-0,30	-0,55	-1,09	-0,35	-0,43	-1,15	788,13	-0,23	-0,48	-0,98	-0,30	-0,41	-1,10	793,97
rbg403	1133,16	0,20	-0,06	-0,15	0,17	-0,03	-0,13	1155,83	0,28	-0,05	-0,09	0,23	-0,02	-0,09	1161,41
rbg443	1236,88	0,00	-0,19	-0,53	-0,06	-0,23	-0,50	1734,80	0,07	-0,10	-0,48	-0,08	-0,25	-0,49	1727,86

Tabela 5.2: Resultados computacionais do GRASP versões C3BL4 e C3BL5 para 200 iterações empregando alfa=0.1 e alfa=0.05. Os campos ‘Vol. Max’ representam o *gap* entre a solução obtida pela heurística e o melhor resultado da literatura; os campos ‘Vol. Med.’ representam o *gap* entre o valor médio e o melhor resultado da heurística; os campos ‘Vol. Min.’ representam o *gap* entre o pior e o melhor resultado obtido pela heurística. Valores positivos representam resultados melhores do que os apresentados na literatura, e valores negativos representam resultados piores que os da literatura. Os resultados destacados em vermelho representam a melhor solução obtida por meio de um dos métodos heurísticos propostos, independente se são melhores do que os resultados da literatura.

Em ambos os conjuntos de testes, para 14 instâncias observou-se que o tempo de execução do GRASP com 200 iterações para qualquer uma das estratégias adotadas foi superior aos tempos de execução do algoritmo ProtoG [Goldbarg *et al*, 2010], aumentando de acordo com o tamanho da instância testada. Neste contexto, para efetuarmos uma comparação mais justa entre as versões do GRASP propostas e o ProtoG da literatura, foi feito um cálculo de ajuste dos tempos computacionais do algoritmo ProtoG da literatura para determinar os tempos de execução equivalentes no computador empregado neste trabalho. Estes tempos foram utilizados como critério de parada das versões do GRASP propostas em um novo experimento para a qual foram realizadas 20 execuções para cada instância, empregando sementes distintas. Foram utilizados dois valores para *alfa*: 0.1 e 0.05, e os resultados são apresentados na Tabela 5.3.

Na coluna 1 estão os nomes das instâncias utilizadas. Da coluna 2 até a 7 são apresentados os resultados obtidos para o GRASP C3BL4, cuja execução foi limitada por um tempo máximo pré-estabelecido. A coluna 2 representa os *gaps* dos melhores resultados obtidos dentre as 20 execuções em relação aos resultados da literatura, na coluna 3 estão os *gaps* entre as médias dos resultados e os resultados da coluna 2, e na coluna 4 estão os *gaps* entre os menores e os maiores resultados obtidos dentre as execuções. Da mesma forma, são apresentados os resultados para a execução do GRASP C3BL5 com o parâmetro *alfa* igual a 0.05 nas colunas 5, 6 e 7. As colunas 8 até 13 representam os resultados encontrados com a estratégia C3BL5, e, finalmente, na coluna 14 são apresentados os limites de tempo empregados para cada instância testada.

GRASP, Limite de Tempo, 20 Execuções													
Instância	C3BL4 alfa=0.1			C3BL4 alfa=0.05			C3BL5 alfa=0.1			C3BL5 alfa=0.05			T. lim. (s)
	Vol. Max.	Vol. Med.	Vol. Min.	Vol. Max.	Vol. Med.	Vol. Min.	Vol. Max.	Vol. Med.	Vol. Min.	Vol. Max.	Vol. Med.	Vol. Min.	
br17	1,93	-0,41	-0,82	1,93	-0,39	-0,82	1,96	-0,4	-0,77	1,99	-0,35	-0,73	125
ftv33	0,3	-0,37	-0,74	0,74	-0,64	-0,99	0,37	-0,32	-0,76	0,77	-0,61	-0,99	15
ftv35	0,08	-0,35	-0,82	0,29	-0,22	-0,51	0,1	-0,34	-0,85	0,36	-0,22	-0,52	16
ftv38	-0,54	-0,53	-1,19	-0,54	-0,19	-0,42	-0,43	-0,47	-1,12	-0,49	-0,21	-0,35	17
p43	0,4	-0,21	-0,41	0,68	-0,54	-0,63	0,44	-0,24	-0,4	0,42	-0,47	-0,54	18
ftv44	0,65	-0,54	-1,16	1,01	-0,64	-1,12	0,63	-0,52	-1,14	1,08	-0,63	-1,19	19
ftv47	0,6	-0,46	-0,99	0,99	-0,52	-0,91	0,59	-0,49	-0,93	1,02	-0,49	-0,84	20
ry48p	0,15	-0,3	-0,7	0,48	-0,31	-0,65	0,14	-0,31	-0,71	0,56	-0,3	-0,63	20
ft53	0,41	-0,36	-0,81	1,02	-0,45	-0,88	0,45	-0,29	-0,74	1,03	-0,41	-0,82	45
ftv55	0,46	-0,82	-1,7	0,32	-0,33	-0,73	0,48	-0,8	-1,66	0,32	-0,24	-0,65	15
ftv64	0,42	-0,43	-0,93	0,46	-0,31	-0,55	0,43	-0,44	-0,91	0,49	-0,24	-0,48	15
ftv70	-0,22	-0,33	-0,79	0,51	-0,47	-0,99	-0,25	-0,27	-0,8	0,55	-0,42	-0,99	15
ft70	-1,25	-0,42	-0,91	-1,4	-0,21	-0,42	-1,28	-0,38	-0,87	-1,39	-0,16	-0,44	40
ftv90	0,39	-0,58	-1,29	0,41	-0,34	-0,95	0,41	-0,61	-1,25	0,41	-0,36	-0,95	15
ftv100	0,04	-0,64	-1,13	0,1	-0,46	-0,82	0,01	-0,59	-1,11	0,17	-0,45	-0,79	15,5
ftv110	0,69	-0,29	-0,7	0,84	-0,3	-0,55	0,76	-0,23	-0,69	0,93	-0,27	-0,49	15
ftv120	1,00	-0,29	-0,53	1,3	-0,48	-1,00	1,06	-0,29	-0,48	1,29	-0,5	-0,94	15
kro124	0,77	-0,43	-0,83	0,78	-0,36	-0,54	0,87	-0,46	-0,79	0,82	-0,34	-0,48	35
ftv130	0,68	-0,46	-1,07	0,9	-0,41	-0,95	0,75	-0,4	-1,03	0,97	-0,43	-0,91	15
ftv140	0,33	-0,41	-0,83	0,71	-0,51	-0,88	0,31	-0,44	-0,83	0,75	-0,51	-0,9	15
ftv150	0,72	-0,44	-1,3	1,06	-0,5	-0,89	0,75	-0,43	-1,24	1,14	-0,44	-0,82	15
ftv160	-3,44	-0,33	-0,78	-3,06	-0,43	-0,96	-3,47	-0,35	-0,79	-3,08	-0,43	-0,97	15
ftv170	-3,63	-0,4	-1,16	-3,18	-0,33	-0,7	-3,65	-0,38	-1,11	-3,15	-0,33	-0,67	15,5
rbg323	-2,64	-0,51	-1,13	-2,68	-0,74	-1,14	-2,67	-0,44	-1,16	-2,67	-0,75	-1,07	20
rbg358	-1,21	-0,63	-1,55	-0,36	-1,59	-2,31	-1,2	-0,65	-1,48	-0,35	-1,51	-2,24	20
rbg403	0,1	-0,25	-0,49	0,25	-0,37	-0,77	0,16	-0,26	-0,49	0,21	-0,35	-0,68	20
rbg443	-0,03	-0,45	-0,91	0,28	-0,82	-1,02	0,02	-0,46	-0,9	-0,04	-0,75	-1,07	20

Tabela 5.3: Resultados computacionais do GRASP versões C3BL4 e C3BL5 limitadas por tempo de execução, empregando alfa=0.1 e alfa=0.05. Os campos ‘Vol. Max’ representam o *gap* entre a solução obtida pela heurística e o melhor resultado da literatura; os campos ‘Vol. Med.’ representam o *gap* entre o valor médio e o melhor resultado da heurística; os campos ‘Vol. Min.’ representam o *gap* entre o pior e o melhor resultado obtido pela heurística. Valores positivos representam resultados melhores do que os apresentados na literatura, e valores negativos representam resultados piores. Os resultados destacados em vermelho representam a melhor solução obtida por meio de um dos métodos heurísticos propostos, independente se são melhores do que os resultados da literatura.

Analisando os resultados obtidos para as execuções limitadas por tempo, observou-se que a versão C3BL5 obteve o melhor resultado para 19 das instâncias testadas, enquanto o C3BL4 mostrou-se melhor para sete instâncias e um empate foi observado.

Entre as soluções obtidas com a variação do parâmetro *alfa*, em cinco casos o melhor resultado foi encontrado utilizando *alfa*=0.1, e para 21 instâncias as melhores soluções foram obtidas com *alfa*=0.05, e para uma instância ocorreu empate entre os resultados.

Com estes resultados, pode-se observar que a adoção de um valor menor para *alfa*, que conseqüentemente consiste em empregar uma LRC menor na etapa construtiva do GRASP, contribuiu na obtenção de melhores soluções.

Comparando os resultados obtidos por meio das execuções limitadas pelo número de iterações e os resultados encontrados pelas execuções limitadas por um valor máximo de tempo pré-estabelecido, pode-se observar que foram obtidos os melhores resultados com a execução de 200 iterações do GRASP para 18 instâncias, enquanto por meio do critério de parada por tempo de execução foram encontrados resultados melhores para oito instâncias, e um empate foi observado. Nas ocasiões que a estratégia com limite de tempo encontrou melhores resultados, estes foram ligeiramente melhores aos obtidos por meio de um número fixo de iterações (a maior diferença observada foi de 0.34%), que por sua vez, para as mesmas instâncias, foi capaz de gerar resultados iguais ou melhores aos apresentados na literatura. Novamente a versão GRASP C3BL5 mostrou-se capaz de encontrar resultados melhores em relação à estratégia C3BL4; a primeira encontrou os melhores resultados para 23 das instâncias testadas, enquanto a outra mostrou-se melhor para quatro instâncias. Nota-se também que a adoção do valor 0.05 para o parâmetro *alfa* também contribuiu positivamente: para 18 instâncias o melhor resultado foi obtido com este valor, enquanto para oito instâncias, o uso de *alfa*=0.1 resultou em resultados melhores e observou-se a ocorrência de um empate.

Ao observar os valores obtidos para os *gaps* de volume médio, para todas as 27 instâncias constatou-se que nos testes executados com limite de tempo os *gaps* obtidos são menores do que os *gaps* encontrados nos testes realizados com 200 iterações do GRASP. Isto deve-se ao fato de serem realizadas menos iterações do GRASP, devido a imposição de um limite de tempo de execução, o que reduz o espaço de soluções explorado pela estratégia.

Comparando os valores obtidos para os *gaps* de volume mínimo obtidos com a variação de *alfa*, para as 27 instâncias testadas, nas execuções com 200 iterações do GRASP, o menor *gap* foi observado 20 vezes para a adoção de *alfa*=0.1, quatro vezes para *alfa*=0.05 e ocorreram três

empates, e na comparação entre os métodos de busca local empregados, C3BL4 obteve os melhores resultados para duas instâncias, a versão C3BL5 em 22 instâncias e foram observados três empates.

Nas execuções com limite de tempo, em 17 instâncias observou-se que $\alpha=0.05$ obteve os melhores resultados para volume mínimo, enquanto com a adoção de $\alpha=0.1$ foram encontrados os melhores resultados de valor mínimo para dez instâncias. A versão do GRASP C3BL4 encontrou o melhor resultado dentre os mínimos para sete instâncias enquanto a versão C3BL5 mostrou-se capaz de encontrar melhores resultados para 17 instâncias e foi observada a ocorrência de três empates.

Se observadas as diferenças entre os *gaps* de volume mínimo entre as dez execuções realizadas com 200 iterações do GRASP e as 20 execuções com um limite fixo de tempo, em ambas as versões do GRASP com diferentes buscas locais testadas, para todas as instâncias observam-se resultados melhores para o primeiro caso. Estes resultados confirmam a expectativa que por meio de uma maior quantidade de iterações do GRASP os resultados obtidos nas execuções sejam próximos entre si, devido ao maior número de execuções das heurísticas que são realizadas.

5.1.2 – COMPARAÇÃO ENTRE OS RESULTADOS OBTIDOS POR MEIO DOS MÉTODOS HEURÍSTICOS HÍBRIDOS

Em seguida, foram realizados testes, com a heurística híbrida GRASP+ILS para avaliar a contribuição de hibridizar duas heurísticas na obtenção de melhores resultados. Utilizando novamente os critérios de 10 execuções com 200 iterações do GRASP, e 20 execuções do com um limite de tempo pré-definido. Para ambos os casos, o número de iterações do ILS foi fixado em 100, realizadas para cada iteração do GRASP. Os resultados apresentados são os *gaps* entre os valores máximos encontrados pelo GRASP com ILS proposto e os valores máximos apresentados na literatura. Na Tabela 5.4 os resultados obtidos são apresentados da seguinte forma: a coluna 1 apresenta a identificação das instâncias utilizadas; nas colunas 2 e 3 são apresentados, respectivamente, os *gaps* de volumes máximo para os parâmetros $\alpha=0.1$ e $\alpha=0.05$, obtidos por meio da versão do GRASP C3BL4 com ILS, e a coluna 4, o tempo computacional médio das execuções. As colunas 5 até 7 apresentam os resultados correspondentes obtidos utilizando a versão C3BL5 com ILS do GRASP. Note-se que os *gaps* são calculados da mesma forma utilizada na Tabela 5.1: os de

volume máximo são calculados em relação aos melhores resultados apresentados pela literatura.

GRASP+ILS, 200 iterações, 10 Execuções						
Instância	C3BL4 alfa=0.1			C3BL5 alfa=0.05		
	Vol. Max.	Vol. Max.	T. med. (s)	Vol. Max.	Vol. Max.	T. med. (s)
br17	5,49	5,49	1,52	12,31	13,64	1,58
ftv33	0,71	0,89	3,96	6,02	6,28	4,06
ftv35	2,68	3,03	4,33	6,61	6,93	4,82
ftv38	-0,29	-0,03	6,39	5,54	5,79	6,73
p43	0,47	0,58	6,01	1,52	1,51	6,62
ftv44	1,18	1,38	7,45	5,68	5,89	7,82
ftv47	4,27	4,75	7,96	5,83	6,19	8,18
ry48p	3,79	3,79	7,75	6,02	6,20	8,41
ft53	1,37	1,82	11,93	5,31	5,45	11,62
ftv55	3,39	3,58	13,81	2,96	3,01	12,85
ftv64	1,10	1,22	20,39	1,67	2,06	22,04
ftv70	3,06	3,07	24,79	2,85	2,96	26,73
ft70	-1,02	-0,88	17,03	0,50	0,68	21,44
ftv90	2,54	2,74	52,33	2,85	2,78	54,35
ftv100	2,48	2,79	66,47	2,16	2,61	63,61
ftv110	1,73	1,98	86,01	1,47	1,97	83,62
ftv120	1,85	1,95	113,7	1,64	1,98	116,05
kro124	1,86	1,94	51,55	0,93	1,11	62,49
ftv130	1,65	1,54	137,22	1,09	1,49	142,53
ftv140	1,03	1,68	172,67	1,11	1,73	169,93
ftv150	1,44	2,04	204,49	1,40	1,63	208,16
ftv160	-2,92	-2,58	240,72	-2,60	-2,01	237,81
ftv170	-2,84	-2,41	283,00	-2,45	-2,39	277,46
rbg323	-0,28	-0,31	1149,12	-0,33	-0,28	1163,92
rbg358	0,10	-0,09	1523,91	-0,04	0,04	1551,38
rbg403	0,32	0,32	1760,74	0,23	0,36	1749,13
rbg443	0,29	0,28	2856,83	0,19	0,28	2716,83

Tabela 5.4: Resultados computacionais do GRASP+ILS, versões C3BL4 e C3BL5 limitadas a 200 execuções, empregando alfa=0.1 e alfa=0.05. Os campos ‘Vol. Max’ representam o *gap* entre a solução obtida pela heurística e o melhor resultado da literatura. Valores positivos representam resultados melhores do que os apresentados na literatura, e valores negativos representam resultados piores que os da literatura. Os resultados destacados em vermelho representam a melhor solução obtida por meio de um dos métodos heurísticos propostos, independente se são melhores do que os resultados da literatura.

Analisando os resultados obtidos, assim como observado nos resultados obtidos somente por meio do GRASP, a adoção de um valor menor para *alfa*, que na prática representa a utilização de uma LRC menor, foi capaz de obter melhores resultados para a maior parte das instâncias testadas. Nas execuções do GRASP+ILS com 200 iterações para os

dois métodos de busca local analisados, em 22 das 27 instâncias o uso de $\alpha=0.05$ gerou melhores resultados, para quatro instâncias os melhores resultados foram obtidos utilizando $\alpha=0.1$ e ocorreu um empate. Dentre estes resultados a versão GRASP+ILS C3BL5 obteve os melhores resultados para 17 instâncias, enquanto para nove instâncias o GRASP+ILS C3BL4 encontrou resultados melhores, e foi observada a ocorrência de um empate.

GRASP+ILS, Limite de Tempo, 20 Execuções					
Instância	C3BL4 $\alpha=0.1$	C3BL4 $\alpha=0.05$	C3BL5 $\alpha=0.1$	C3BL5 $\alpha=0.05$	T. lim. (s)
	Max. Vol.	Max. Vol.	Max. Vol.	Max. Vol.	
br17	5,30	5,30	12,02	13,06	125
ftv33	0,57	0,76	5,96	6,15	15
ftv35	2,48	2,60	6,62	6,93	16
ftv38	-0,61	-0,31	5,41	5,89	17
p43	0,38	0,68	1,51	1,52	18
ftv44	0,70	1,03	5,57	5,69	19
ftv47	4,03	4,37	6,13	6,19	20
ry48p	3,55	3,62	6,10	6,20	20
ft53	1,05	1,27	5,28	5,45	45
ftv55	2,74	3,18	2,63	3,02	15
ftv64	0,88	0,91	1,63	1,96	15
ftv70	2,39	2,78	2,81	2,88	15
ft70	-1,01	-1,14	0,49	0,62	40
ftv90	2,19	2,40	2,18	2,57	15
ftv100	2,16	2,49	2,08	2,17	15
ftv110	1,92	1,41	1,40	1,44	15
ftv120	1,12	1,59	1,64	1,69	15
kro124	1,47	1,32	0,86	1,07	35
ftv130	1,08	1,08	1,04	1,11	15
ftv140	0,67	1,06	1,07	1,06	15
ftv150	1,01	1,45	1,35	1,43	15
ftv160	-3,26	-2,70	-2,68	-2,13	15
ftv170	-3,14	-2,83	-2,84	-2,78	15
rbg323	-1,19	-1,16	-1,12	-1,07	20
rbg358	-0,61	-0,60	-0,60	-0,52	20
rbg403	0,19	0,26	0,24	0,24	20
rbg443	0,01	0,28	0,11	0,12	20

Tabela 5.5: Resultados computacionais do GRASP+ILS, versões C3BL4 e C3BL5 com limite no tempo de execução, empregando $\alpha=0.1$ e $\alpha=0.05$. Os campos ‘Vol. Max’ representam o *gap* entre a solução obtida pela heurística e o melhor resultado da literatura. Valores positivos representam resultados melhores do que os apresentados na literatura, e valores negativos representam resultados piores que os da literatura. Os resultados destacados em vermelho representam a melhor solução obtida por meio de um dos métodos heurísticos propostos, independente se são melhores do que os resultados da literatura.

Já nas iterações com tempo de execução limitado, cujos resultados são apresentados na Tabela 5.5, para 24 instâncias foi possível encontrar os melhores resultados com $\alpha=0.05$ e nos três casos restantes, a utilização de $\alpha=0.1$ foi capaz de gerar resultados melhores. A versão do GRASP C3BL4 com ILS encontrou melhores resultados para oito instâncias, enquanto a versão C3BL5 obteve melhores resultados para 19 instâncias.

Comparando os melhores resultados apresentados na Tabela 5.4 com os obtidos com o uso somente do GRASP (apresentados na Tabela 5.1), a adoção do ILS mostrou-se um passo fundamental no processo de otimização das rotas, já que em todas as instâncias adotadas ocorreram melhorias nos resultados encontrados.

Em seguida, foram realizados testes com a heurística híbrida GRASP+SA. Os testes realizados seguiram os mesmos números de iterações e limites de tempo dos testes realizados previamente com o GRASP+ILS. Diversos experimentos foram realizados para determinar os valores de temperatura da têmpera e o número máximo de iterações que seriam realizadas com cada valor de temperatura, de maneira que os tempos de execução dos testes limitados por um número máximo de iterações fossem os mais próximos possíveis dos tempos de execução dos testes executados empregando o ILS, a fim de que a comparação dos resultados obtidos por cada um fosse a mais justa possível. Os testes foram realizados com o valor de temperatura inicial igual a 10000, a temperatura mínima igual a 0.0001 e o parâmetro α , que regula o decréscimo da temperatura a cada iteração igual a 0.1. O número máximo de iterações feitas a cada valor de temperatura foi fixado em 10.

GRASP+ILS, 200 iterações, 10 Execuções

Instância	C3BL4 alfa=0.1		C3BL4 alfa=0.05		C3BL5 alfa=0.1		C3BL5 alfa=0.05	
	Vol. Max.	Vol. Max.	T. Med. (s)	Vol. Max.	T. Med. (s)	Vol. Max.	Vol. Max.	T. Med. (s)
br17	5,46	5,50	1,74	12,19	14,34	1,72		
ftv33	0,75	0,86	4,13	5,73	6,82	4,26		
ftv35	2,70	2,94	4,77	6,68	6,93	4,65		
ftv38	-0,19	-0,17	7,29	5,53	5,89	7,48		
p43	0,50	0,41	7,52	1,46	1,48	8,03		
ftv44	1,25	1,36	8,49	5,54	5,69	8,13		
ftv47	4,35	4,76	8,56	5,85	6,26	9,38		
ry48p	3,87	3,79	9,14	6,04	6,20	8,83		
ft53	1,45	1,66	13,93	5,33	5,67	13,56		
ftv55	3,68	4,01	15,37	2,91	3,19	17,04		
ftv64	1,14	1,03	22,49	1,75	2,04	21,95		
ftv70	3,09	3,05	27,58	2,88	3,04	28,52		
ft70	-0,91	-1,05	21,40	0,52	0,66	23,70		
ftv90	2,60	2,74	56,37	2,96	3,09	55,53		
ftv100	2,49	2,77	71,37	2,27	2,85	73,16		
ftv110	1,84	2,01	95,94	1,52	1,94	93,38		
ftv120	1,89	1,91	121,55	1,61	1,93	124,52		
kro124	1,47	1,29	75,49	0,98	1,37	84,63		
ftv130	1,64	1,40	158,13	1,13	1,49	161,15		
ftv140	0,85	1,70	184,27	1,16	1,32	193,47		
ftv150	1,54	1,86	216,48	1,47	1,77	220,53		
ftv160	-2,90	-2,57	262,14	-2,63	-2,04	258,39		
ftv170	-2,82	-2,56	295,41	-2,32	-1,97	299,41		
rbg323	-0,22	-0,41	1158,83	-0,27	-0,29	1157,32		
rbg358	0,07	-0,14	1573,77	-0,03	0,03	1580,38		
rbg403	0,35	0,15	1794,03	0,24	0,26	1789,46		
rbg443	0,14	0,14	2894,47	0,15	0,19	2807,32		

Tabela 5.6: Resultados computacionais do GRASP+SA, versões C3BL4 e C3BL5, limitadas a 200 iterações, empregando $\alpha=0.1$ e $\alpha=0.05$. Os campos 'Vol. Max' representam o *gap* entre a solução obtida pela heurística e o melhor resultado da literatura. Valores positivos representam resultados melhores do que os apresentados na literatura, e valores negativos representam resultados piores que os da literatura. Os resultados destacados em vermelho representam a melhor solução obtida por meio de um dos métodos heurísticos propostos, independente se são melhores do que os resultados da literatura.

A Tabela 5.6 apresenta os resultados obtidos por meio do GRASP+SA utilizando o *gap* entre as melhores soluções encontradas e os melhores resultados apresentados na literatura. A coluna 1 representa a identificação das instâncias testadas, enquanto as colunas 2 e 3 apresentam, respectivamente, os melhores resultados obtidos pela versão C3BL4 com valores de α iguais a 0.1 e 0.05. A coluna 4 apresenta os tempos médios de execução. As colunas 5 à 7 apresentam os respectivos resultados encontrados pela versão C3BL5 do algoritmo

proposto. Comparando os resultados, pode-se observar que o GRASP C3BL4 encontrou os melhores resultados para dez das instâncias testadas e a versão C3BL5 para 17 instâncias restantes. Se comparados os resultados em relação ao valor de *alfa* adotado, os testes realizados com *alfa*=0.1 encontraram os melhores resultados para seis das instâncias, e mudando o valor de *alfa* para 0.05, encontramos resultados melhores para 21 instâncias.

Instância	C3BL4 alfa=0.1	C3BL4 alfa=0.05	C3BL5 alfa=0.1	C3BL5 alfa=0.05	T. lim. (s)
	Vol. Max.	Vol. Max.	Vol. Max	Vol. Max	
br17	5,31	5,28	12,06	13,51	125
ftv33	0,61	0,69	6,02	6,89	15
ftv35	2,42	2,53	6,38	6,93	16
ftv38	-0,63	-0,27	5,52	5,89	17
p43	0,41	0,63	1,37	1,43	18
ftv44	0,61	0,96	5,43	5,64	19
ftv47	4,04	4,35	6,18	6,27	20
ry48p	3,5	3,62	6,13	6,20	20
ft53	0,99	1,25	5,03	5,67	45
ftv55	3,26	3,71	2,94	3,17	15
ftv64	0,81	0,97	1,75	1,94	15
ftv70	2,38	2,82	2,91	3,04	15
ft70	-1,02	-1,18	0,50	0,58	40
ftv90	2,16	2,33	2,13	2,57	15
ftv100	1,15	1,82	2,29	2,72	15
ftv110	1,89	1,32	1,46	1,58	15
ftv120	1,09	1,62	1,57	1,64	15
kro124	1,36	1,77	0,93	1,02	35
ftv130	1,08	1,09	1,07	1,47	15
ftv140	0,65	1,02	1,10	1,39	15
ftv150	0,92	1,46	1,38	1,36	15
ftv160	-3,33	-2,7	-2,54	-2,15	15
ftv170	-3,18	-2,81	-2,63	-2,80	15
rbg323	-1,26	-1,17	-1,09	-0,45	20
rbg358	-0,65	-0,57	-0,57	-0,61	20
rbg403	0,18	0,19	0,24	0,26	20
rbg443	-0,06	0,12	0,12	0,12	20

Tabela 5.7: Resultados computacionais do GRASP com *simulated annealing* (SA), versões C3BL4 e C3BL5, limitadas por tempo de execução, empregando $\alpha=0.1$ e $\alpha=0.05$. Os campos 'Vol. Max' representam o *gap* entre a solução obtida pela heurística e o melhor resultado da literatura. Valores positivos representam resultados melhores do que os apresentados na literatura, e valores negativos representam resultados piores que os da literatura. Os resultados destacados em vermelho representam a melhor solução obtida por meio de um dos métodos heurísticos propostos, independente se são melhores do que os resultados da literatura.

Para os testes realizados com a imposição de um limite de tempo para execução do GRASP com *simulated annealing*, cujos resultados são apresentados na Tabela 5.7, o melhor resultado para as instâncias testadas foi encontrado quatro vezes por meio da versão do GRASP C3BL4, e 21 vezes com o uso da versão C3BL5, e em duas situações ocorreu o empate. A adoção de $\alpha=0.1$ levou ao melhor resultado em dois casos, e com $\alpha=0.05$, foram encontrados resultados melhores para 23 das instâncias testadas, e para duas instâncias foram observados empates nos resultados com valores distintos de α .

Quando comparados os resultados das heurísticas híbridas GRASP+ILS e GRASP+SA nos testes limitados a 200 iterações, o GRASP+ILS encontrou os melhores resultados para 13 instâncias testadas, enquanto o GRASP+SA encontrou resultados melhores para 11 instâncias, e foi observado empate nos resultados para três instâncias. Nos resultados obtidos pelas execuções limitadas por um tempo máximo pré-estabelecido, Para os testes realizados com um limite de tempo de execução, o GRASP+ILS gerou soluções de maior qualidade para nove instâncias, enquanto o GRASP+SA encontrou melhores resultados em 13 instâncias, e foram observados cinco empates. Os resultados apresentados por estes métodos mostraram que a utilização dos métodos de intensificação propostos foi relevante ao objetivo de obter soluções melhores àquelas geradas somente pelo GRASP.

Analisando os resultados obtidos pelas diferentes versões do GRASP combinadas com o *Simulated Annealing* e o *Iterated Local Search* com a utilização de uma UMP para a operação de pistoneio dos poços, pode-se observar que a versão do GRASP C3BL5 foi capaz de encontrar melhores soluções para o PRP-UMP do que as demais versões testadas. Nota-se ainda a predominância de melhores soluções quando empregado o parâmetro $\alpha=0.05$, o que confirma a hipótese de que a seleção aleatória de um elemento de maior atratividade para o problema em questão, pertencente a uma lista restrita de candidatos apresenta melhores resultados quando é empregada uma lista menor.

5.1.3 – AVALIAÇÃO PROBABILÍSTICA DOS RESULTADOS EMPREGANDO UMA UMP

Uma maneira de avaliar os resultados obtidos é por meio do método de avaliação probabilística apresentado por [Aiex, R. M. *et al*, 2002], que realiza um estudo de convergência das soluções encontradas por cada uma das heurísticas propostas. O método baseia-se em obter os tempos de execução necessários a cada heurística para encontrar um

resultado que seja igual ou superior a um valor alvo estabelecido. Em problemas de maximização, quanto maior é o valor do alvo, mais difícil se torna de alcançá-lo.

Para se obter os tempos computacionais necessários para atingir um valor alvo estabelecido, permitiu-se que as heurísticas executassem quantas iterações fossem necessárias para a obtenção de tal valor. Cada um dos métodos propostos foi executado 200 vezes, empregando sementes distintas.

Uma vez concluídos os testes, os tempos computacionais obtidos foram ordenados de forma crescente em uma lista L , e a cada tempo t_i obtido foi associada à probabilidade $p_i = (i - 0,5/200)$, sendo $i = 1, \dots, 200$, a ordem que cada tempo t aparece na lista ordenada L . Com o conjunto de pontos $z_i = (t_i, p_i)$, foi traçado o gráfico de probabilidade de obtenção da solução alvo em um determinado tempo de execução.

Foram estabelecidos alvos considerados de dificuldade média e difícil de serem obtidos. Os testes foram feitos com as heurísticas GRASP, GRASP+ILS e GRASP+SA para uma UMP, empregando como alvo de dificuldade intermediária a média das soluções obtidas por meio do GRASP e nos testes com dificuldade mais elevada a melhor solução obtida pelo GRASP. Para múltiplas UMP's os testes foram feitos empregando as heurísticas híbridas propostas e utilizou-se como alvo médio o menor valor dentre as médias dos resultados encontrados pelas heurísticas, e como alvo difícil foi utilizado o menor valor dentre os melhores resultados das heurísticas, reduzido em 5%.

As Figuras 5.1 e 5.2 mostram as distribuições de probabilidades empíricas empregando uma UMP para a obtenção dos valores alvo médio e difícil, respectivamente, utilizando as heurísticas GRASP, GRASP+ILS e GRASP+SA para a instância *fv55*.

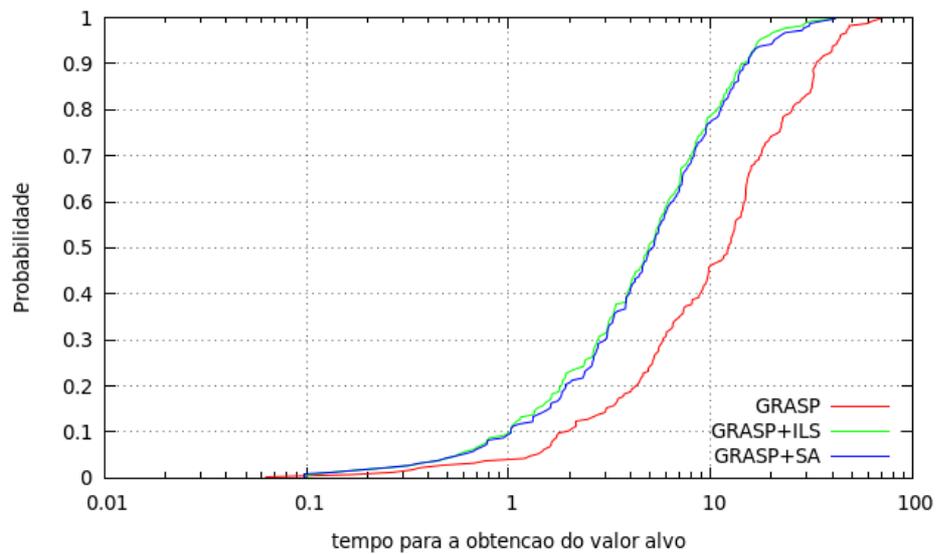


Figura 5.1: Resultados do GRASP e das heurísticas híbridas empregando uma UMP para a obtenção de um alvo de dificuldade média

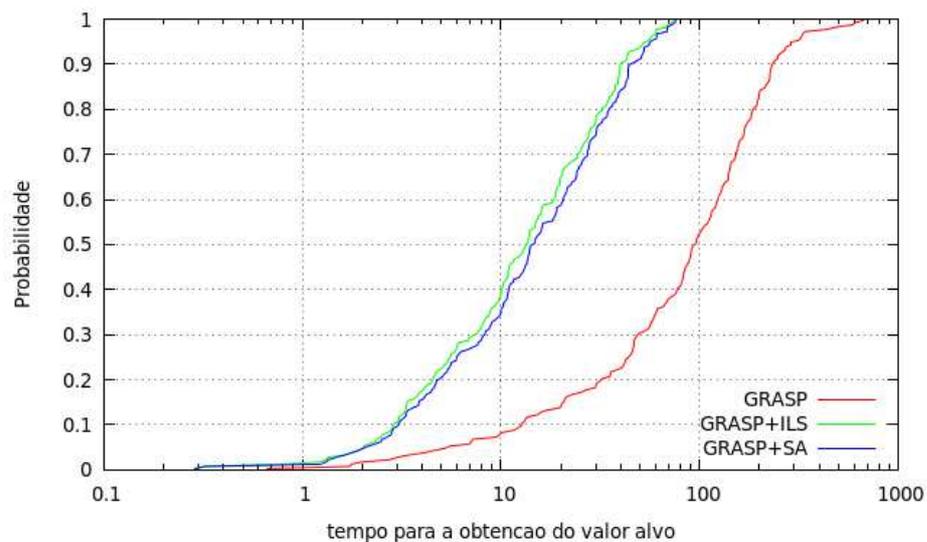


Figura 5.2: Resultados do GRASP e das heurísticas híbridas empregando uma UMP para a obtenção de um alvo de dificuldade alta.

Nos resultados para uma UMP com alvo médio, pode-se observar um comportamento bastante similar entre as heurísticas GRASP+ILS e GRASP+SA, com uma pequena tendência de a primeira possuir uma probabilidade maior de encontrar a solução alvo mais rapidamente. Como esperado, o GRASP apresenta uma convergência para o alvo mais demorada do que as heurísticas híbridas propostas.

Os resultados obtidos com uma UMP empregando um alvo considerado difícil apresentam um comportamento similar ao observado nos resultados com alvo de dificuldade média, novamente com a heurística GRASP+ILS apresentando um desempenho ligeiramente

melhor do que a do GRASP+SA. Já para a heurística GRASP nota-se que o tempo necessário para convergir para o alvo estabelecido é significativamente maior.

5.1.4 – COMPARAÇÃO ENTRE OS RESULTADOS OBTIDOS POR MEIO DOS MÉTODOS HEURÍSTICOS HÍBRIDOS E OS OBTIDOS POR MEIO DE UM MÉTODO EXATO

Um novo conjunto de testes foi realizado para tentar resolver o problema por meio de um método exato utilizando-se o IBM ILOG CPLEX empregando a formulação apresentada no Capítulo 2. Um conjunto inicial de testes foi realizado estabelecendo como critério de parada tempos máximos de execução, estabelecidos para grupos de instâncias, de acordo com o número de poços presentes em cada uma. Foram feitas 10 execuções de cada método heurístico apresentado, cada uma com um valor distinto de semente, utilizando a versão C3BL5, por ter se mostrado capaz de encontrar melhores resultados, de acordo com o observado nos testes anteriores. A Tabela 5.8 apresenta os melhores resultados obtidos dentre as 10 execuções. A coluna 1 representa a instância utilizada, a coluna 2 os resultados obtidos por meio do CPLEX e a coluna 3 os *gaps* duais do CPLEX. As colunas 4 e 5 apresentam os *gaps* calculados entre os resultados obtidos pela heurística GRASP+ILS e os resultados do CPLEX e do ProtoG, respectivamente. Já as colunas 6 e 7 representam os *gaps* entre os resultados obtidos por meio da heurística GRASP+SA e do CPLEX e ProtoG, respectivamente. A coluna 8 representa o limite de tempo estabelecido para cada instância, e que foi empregado por todos os métodos testados. As linhas sombreadas indicam as instâncias que o CPLEX terminou sua execução dentro do limite de tempo estabelecido, e, portanto, representam a solução exata da instância empregando uma UMP nas operações de pistoneio.

Instância	CPLEX	Dual CPLEX	GRASP+ILS		GRASP+SA		T. Lim. (s)
			gap CPLEX	gap ProtoG	gap CPLEX	gap ProtoG	
br17	1201,84	0	0,00	14,76	0,00	14,76	20,00
ftv33	1208,28	0	0,00	7,02	0,00	7,02	20,00
ftv35	1208,28	0	0,00	6,93	0,00	6,93	20,00
ftv38	1208,13	0	0,00	5,89	0,00	5,89	20,00
p43	1228,25	0,53	0,37	1,52	0,35	1,51	20,00
ftv44	1208,26	0	0,00	5,89	0,00	5,89	20,00
ftv47	1208,71	0,33	-0,69	6,35	-0,63	6,41	20,00
ry48p	1208,25	0	0,00	6,20	0,00	6,20	20,00
ft53	1208,66	0,31	-1,59	5,50	-1,26	5,85	45,00
ftv55	1208,70	0,4	-1,77	3,97	-1,68	4,06	45,00
ftv64	1208,89	0,27	-1,12	2,41	-1,07	2,46	45,00
ftv70	1208,91	1,24	0,53	3,61	0,46	3,53	45,00
ft70	1159,14	0,35	-4,28	0,89	-4,17	1,01	45,00
ftv90	1209,06	0,88	0,55	4,26	0,61	4,32	200,00
ftv100	1209,09	0,83	-0,02	3,41	-0,02	3,41	200,00
ftv110	1209,10	1,35	0,46	3,94	0,09	3,56	200,00
ftv120	1209,08	1,18	0,77	3,53	0,71	3,46	200,00
kro124	1209,09	0,47	-1,64	2,11	-1,55	2,21	200,00
ftv130	1213,46	0,86	-0,60	2,66	-0,57	2,69	600,00
ftv140	1211,01	0,96	0,36	3,08	0,05	2,76	600,00
ftv150	1211,94	0,88	0,47	3,08	0,44	3,04	600,00
ftv160	1209,55	0,34	-1,86	0,76	-1,75	0,87	600,00
ftv170	1212,03	0,23	-0,54	1,89	-1,12	1,29	600,00
rbg323	1122,41	1,14	-0,21	1,29	-0,17	1,34	3900,00
rbg358	1130,38	1,16	-0,88	1,93	-1,02	1,78	3900,00
rbg403	1137,13	0,24	0,14	0,49	0,11	0,46	3900,00
rbg443	1240,73	0,13	0,03	0,34	0,03	0,34	3900,00

Tabela 5.8: Resultados computacionais do CPLEX e das heurísticas ILS e SA propostas (empregando C3BL5, $\alpha=0.05$), limitadas por tempo de execução. Os campos 'gap CPLEX' representam o *gap* entre a solução obtida pela heurística e o melhor resultado por meio do método exato. Os campos 'gap ProtoG' representam o *gap* entre a solução obtida pela heurística e o melhor resultado da literatura. Valores positivos representam resultados melhores obtidos pelas heurísticas em relação ao outro valor de comparação, e valores negativos representam resultados piores.

A partir dos resultados apresentados, pode ser observado que o CPLEX foi capaz de obter a solução exata para seis das instâncias testadas, e que as heurísticas de GRASP+ILS e GRASP+SA propostas também alcançaram estes resultados com o limite de tempo imposto. Para as demais instâncias, tanto o GRASP+ILS quanto o GRASP+SA encontraram melhores resultados que o CPLEX para nove instâncias, e ambos foram capazes de encontrar resultados melhores aos apresentados pelo ProtoG com o limite de tempo utilizado.

Foi realizada uma análise estatística utilizando os resultados obtidos a fim de determinar a contribuição das heurísticas propostas. Inicialmente foi utilizado o método de Shapiro-Wilk [Shapiro and Wilk, 1965] como teste de normalidade dos resultados. Os testes estatísticos foram realizando formulações elaboradas para o Microsoft Excel, disponíveis na página da Internet dedicada a métodos estatísticos de análise de dados [Real Statistics, 2016]. Utilizando um nível de significância $\alpha=5\%$, os *p-valores* encontrados a partir dos resultados das heurísticas GRASP+ILS e GRASP+SA são iguais a $p\text{-valor}_{(ILS)}= 6,22E-05$ e $p\text{-valor}_{(SA)}= 4,45E-05$, respectivamente. Como os *p-valores* encontrados são menores do que o valor nível α escolhido, a hipótese nula é rejeitada para ambos os casos, o que indica que os resultados encontrados não seguem a distribuição normal.

Em seguida foi aplicado o teste de Wilcoxon [Wilcoxon, 1945] pareado para determinar se os resultados obtidos pelas heurísticas apresentam uma diferença estatística, empregando formulações elaboradas para o Microsoft Excel [Real Statistics, 2016]. Empregando um nível de significância $\alpha=5\%$, o valor mínimo do somatório das classificações foi $T=103$. Considerando que do conjunto de instâncias testadas 20 resultados das heurísticas apresentaram valores distintos, e, portanto, considerados pelo teste de Wilcoxon, foi adotado o valor $T\text{-crit}=52$. Como para este conjunto de resultados $T > T\text{-crit}$, conclui-se que não há uma diferença estatisticamente significativa entre os resultados encontrados pelas heurísticas.

Com o objetivo de tentar encontrar um maior número de soluções exatas para as instâncias adotadas por meio do CPLEX e comparar o desempenho das heurísticas propostas com o método exato, novos testes foram feitos com o limite de tempo de 7200 segundos para todas as heurísticas. A Tabela 5.9 apresenta estes resultados, organizados com a mesma disposição de colunas empregada na Tabela 5.8, a exceção da omissão da coluna 8, uma vez que o limite de tempo adotado foi de 7200 segundos para todas as instâncias, independente do número de poços que as compõe.

Instância	CPLEX	Dual CPLEX	GRASP+ILS		GRASP+SA	
			<i>gap</i> CPLEX	<i>gap</i> ProtoG	<i>gap</i> CPLEX	<i>gap</i> ProtoG
p43	1228,25	0,53	0,47	1,63	0,52	1,68
ftv47	1208,71	0,33	0,17	7,27	0,06	7,15
ft53	1208,66	0,31	-0,84	6,30	-0,88	6,26
ftv55	1208,70	0,40	-0,50	5,31	-0,65	5,15
ftv64	1208,89	0,27	0,15	3,72	0,10	3,67
ftv70	1208,91	1,24	1,06	4,16	0,90	3,99
ft70	1159,14	0,35	-2,82	2,43	-2,92	2,32
ftv90	1209,06	0,88	0,96	4,69	0,84	4,56
ftv100	1209,09	0,83	0,74	4,19	0,79	4,25
ftv110	1209,10	1,35	1,19	4,70	1,15	4,66
ftv120	1209,08	1,18	1,03	3,79	0,91	3,66
kro124	1209,09	0,47	0,29	4,12	0,25	4,08
ftv130	1213,46	0,86	0,56	3,86	0,63	3,92
ftv140	1211,01	0,96	0,54	3,26	0,46	3,19
ftv150	1211,94	0,88	0,55	3,16	0,45	3,06
ftv160	1209,55	0,34	-0,42	2,24	-0,46	2,20
ftv170	1212,03	0,23	-0,15	2,29	-0,39	2,04
rbg323	1122,41	1,14	0,97	2,49	0,92	2,44
rbg358	1130,38	1,16	0,66	3,52	0,71	3,56
rbg403	1137,13	0,24	0,14	0,49	0,11	0,46
rbg443	1240,73	0,13	0,03	0,34	0,03	0,34

Tabela 5.9: Resultados computacionais do CPLEX e das heurísticas ILS e SA propostas (empregando C3BL5, $\alpha=0.05$), com tempo de execução de 7200 segundos. Os campos '*gap* CPLEX' representam o *gap* entre a solução obtida pela heurística e o melhor resultado por meio do método exato. Os campos '*gap* ProtoG' representam o *gap* entre a solução obtida pela heurística e o melhor resultado da literatura. Valores positivos representam resultados melhores obtidos pelas heurísticas em relação ao outro valor de comparação, e valores negativos representam resultados piores.

Analisando os resultados, o desempenho das heurísticas aumentou satisfatoriamente com adoção de um maior tempo de execução. Para as 21 instâncias testadas, em apenas cinco delas o método exato encontrou resultados melhores aos obtidos através das heurísticas propostas. Comparando os resultados das heurísticas, o GRASP+ILS foi capaz de encontrar melhores resultados para 16 instâncias, enquanto o GRASP+SA obteve resultados melhores para quatro instâncias, e um empate foi observado.

Para os resultados com o limite de tempo igual a 7200 segundos, o teste de Shapiro-Wilk empregando um nível de significância $\alpha=5\%$, encontrou os seguintes p-valores para as heurísticas: $p\text{-valor}_{(ILS)} = 0.000169$ e $p\text{-valor}_{(SA)} = 0.000122$. Como estes valores são inferiores ao nível *alfa* estabelecido, pode-se afirmar que os resultados das heurísticas não seguem a distribuição normal.

Empregando o teste de Wilcoxon pareado para as 21 instâncias cujas soluções exatas não foram encontradas, utiliza-se $T\text{-crit} = 58$, e o valor encontrado para o menor somatório dos rankings é $T=43$. Para esses conjuntos de dados, tem-se que $T < T\text{-crit}$, logo, pode-se afirmar que existe diferença significativa entre os resultados gerados pelas heurísticas propostas.

Mais um conjunto de testes foi executado, aumentando-se o limite de tempo de execução do método exato e das heurísticas para 10800 segundos, e os resultados são apresentados na Tabela 5.10. Com um maior tempo de execução, espera-se que os métodos desenvolvidos sejam capazes de encontrar melhores resultados, e eventualmente seja obtido o resultado exato de cada instância por meio do método desenvolvido em CPLEX.

Instância	CPLEX	Dual CPLEX	GRASP+SA		GRASP+ILS	
			<i>gap</i> CPLEX	<i>gap</i> ProtoG	<i>gap</i> CPLEX	<i>gap</i> ProtoG
p43	1229,73	0,51	0,41	1,69	0,41	1,69
ftv47	1208,71	0,33	0,21	7,31	0,11	7,20
ft53	1210,31	0,22	0,14	7,50	0,12	7,47
ftv55	1211,29	0,27	0,20	6,28	0,13	6,21
ftv64	1210,06	0,21	0,15	3,83	0,18	3,85
ftv70	1210,38	1,13	0,98	4,20	0,92	4,14
ft70	1160,08	0,32	-1,92	3,46	-2,03	3,35
ftv90	1211,51	0,84	0,79	4,72	0,76	4,69
ftv100	1213,41	0,68	0,47	4,28	0,46	4,28
ftv110	1210,95	1,31	1,08	4,75	1,07	4,73
ftv120	1210,61	1,12	0,97	3,87	0,90	3,79
kro124	1209,09	0,47	0,37	4,20	0,26	4,09
ftv130	1215,01	0,74	0,53	3,95	0,58	4,01
ftv140	1213,15	0,70	0,43	3,34	0,32	3,22
ftv150	1214,07	0,72	0,40	3,18	0,32	3,11
ftv160	1210,57	0,30	0,09	2,85	0,17	2,92
ftv170	1212,72	0,22	0,11	2,61	0,03	2,53
rbg323	1126,05	0,96	0,78	2,63	0,71	2,56
rbg358	1134,51	1,03	0,47	3,70	0,53	3,75
rbg403	1137,13	0,24	0,14	0,49	0,11	0,46
rbg443	1240,73	0,13	0,03	0,34	0,03	0,34

Tabela 5.10: Resultados computacionais do CPLEX e das heurísticas ILS e SA propostas (empregando C3BL5, $\alpha=0.05$), com tempo de execução de 10800 segundos. Os campos '*gap* CPLEX' representam o *gap* entre a solução obtida pela heurística e o melhor resultado por meio do método exato. Os campos '*gap* ProtoG' representam o *gap* entre a solução obtida pela heurística e o melhor resultado da literatura. Valores positivos representam resultados melhores obtidos pelas heurísticas em relação ao outro valor de comparação, e valores negativos representam resultados piores.

A análise dos resultados obtidos demonstra a eficácia dos métodos heurísticos, pois para apenas uma das instâncias testadas, identificada como *ft70*, o resultado obtido pelo método exato foi melhor aos resultados encontrados pelos GRASP+ILS e GRASP+SA. Comparando os resultados das heurísticas propostas, o GRASP+ILS mostrou-se capaz de encontrar os melhores resultados para 15 das 21 instâncias testadas, enquanto o GRASP+SA obteve resultados melhores para quatro instâncias e ocorreram dois empates.

Os resultados obtidos através do teste de Shapiro-Wilk utilizando o nível de significância $\alpha=5\%$ para as heurísticas de GRASP+ILS e GRASP+SA propostas foram, respectivamente: $p\text{-valor}_{(ILS)} = 1,93E-05$ e $p\text{-valor}_{(SA)} = 2,19E-05$. Como os $p\text{-valores}$ encontrados são inferiores ao alfa estabelecido, pode-se afirmar que os resultados das heurísticas não apresentam distribuição normal.

Ao aplicar os resultados obtidos ao teste de Wilcoxon pareado com $\alpha=5\%$, e considerando que das 21 instâncias testadas, as heurísticas geraram resultados distintos para 20 destas, o valor crítico empregado é $T\text{-crit}=52$. O menor somatório do ranking de classificação das diferenças entre os resultados das heurísticas é $T=39$, logo, como $T < T\text{-crit}$, pode-se afirmar que existe diferença estatisticamente significativa entre os resultados encontrados.

5.2 – RESULTADOS OBTIDOS POR MEIO DA UTILIZAÇÃO DE MÚLTIPLAS UMP’S

Em seguida foram realizados testes onde múltiplas UMP’s são utilizadas simultaneamente para as operações de pistoneio dos poços, limitadas a um número máximo de cinco UMP’s. Os testes realizados empregaram as heurísticas híbridas GRASP+ILS e GRASP+SA, utilizando a combinação C3BL5 para o GRASP. Baseado nos resultados analisados para as execuções com uma UMP, o valor do parâmetro *alfa* que determina o tamanho percentual da lista restrita de candidatos em relação ao número de poços candidatos foi fixado em 0,05. Não foram encontrados na literatura resultados computacionais empregando múltiplas UMP’s para o conjunto de instâncias empregadas, portanto a análise dos resultados é feita por meio da comparação entre os resultados obtidos pelas heurísticas propostas.

5.2.1 – COMPARAÇÃO ENTRE OS RESULTADOS OBTIDOS POR MEIO DOS MÉTODOS HÍBRIDOS HEURÍSTICOS

Os resultados obtidos pela versão do GRASP C3BL5 utilizando as heurísticas ILS e SA, com a execução do algoritmo dez vezes, sendo cada execução limitada a 200 iterações, são apresentados na Tabela 5.11. A coluna 1 apresenta a identificação das instâncias, as colunas 2 e 3 apresentam os melhores resultados (volume máximo) obtidos pelo ILS e pelo SA, respectivamente, considerando a utilização de duas UMP's. A coluna 4 representa a diferença percentual entre os resultados obtidos entre o ILS e o SA, onde uma diferença percentual representada por um valor negativo indica que o resultado do ILS foi melhor ao obtido pelo SA, e positivo se a relação referente ao maior valor for o contrário. Os resultados apresentados para as colunas 5 a 7, 8 a 10 e 11 a 13 representam os resultados encontrados com a utilização de 3, 4 e 5 UMP's, respectivamente, organizados da mesma maneira utilizada para apresentar os resultados de 2 UMP's nas colunas 2 a 4. Os tempos de execução das heurísticas são apresentados na Tabela 5.12.

Analisando os resultados obtidos por meio da utilização das heurísticas propostas empregando múltiplas UMP's, pode-se observar que as estratégias apresentaram desempenhos parecidos entre si. A diferença percentual entre os resultados do ILS e SA foi, para a ampla maioria das soluções, inferior a 2%; as heurísticas apresentaram números bastante similares em relação à quantidade de instâncias que cada método obteve o melhor resultado e sem a observação de empates, como apresentado a seguir:

- 2 UMP's:
 - GRASP+ILS: 14 instâncias;
 - GRASP+SA: 13 instâncias;
- 3 UMP's:
 - GRASP+ILS: 14 instâncias;
 - GRASP+SA: 13 instâncias;
- 4 UMP's:
 - GRASP+ILS: 15 instâncias;
 - GRASP+SA: 12 instâncias;
- 5 UMP's:
 - GRASP+ILS: 15 instâncias;
 - GRASP+SA: 12 instâncias;

C3BL5 Instância	2 UMP's			3 UMP's			4 UMP's			5 UMP's		
	ILS	SA	Diferença									
br17	1343,42	1339,93	-0,26	1369,14	1356,98	-0,89	1370,04	1357,22	-0,94	1370,89	1362,88	-0,58
ftv33	1866,69	1843,10	-1,26	2863,09	2886,89	0,83	3212,55	3225,69	0,41	4088,32	4067,02	-0,52
ftv35	1822,75	1843,68	1,15	2867,06	2880,63	0,47	3278,48	3253,72	-0,76	4017,75	4031,47	0,34
ftv38	1875,46	1892,09	0,89	2891,44	2884,55	-0,24	3455,20	3415,96	-1,14	4027,61	4039,50	0,30
p43	1912,57	1888,35	-1,27	2950,02	2958,27	0,28	3891,94	3915,02	0,59	4583,31	4614,40	0,68
ftv44	1918,29	1905,40	-0,67	3009,02	3003,12	-0,20	3946,69	3941,82	-0,12	4612,64	4622,60	0,22
ftv47	1971,17	1994,22	1,17	3081,74	3022,81	-1,91	4002,58	4096,51	2,35	4600,28	4590,62	-0,21
ry48p	2021,04	2018,13	-0,14	3075,34	3009,65	-2,14	4075,50	4086,25	0,26	4635,20	4613,48	-0,47
ft53	2019,37	2013,26	-0,30	3054,68	3047,04	-0,25	4063,28	4085,43	0,54	4642,20	4645,20	0,06
ftv55	2057,24	2071,92	0,71	3069,80	3065,93	-0,13	4069,55	4070,53	0,02	4811,62	4819,13	0,16
ftv64	2075,13	2063,90	-0,54	3027,52	3050,08	0,75	4049,45	4053,96	0,11	4814,96	4807,84	-0,15
ftv70	2080,43	2052,03	-1,37	3196,62	3143,66	-1,66	4251,40	4263,52	0,29	4807,64	4825,88	0,38
ft70	2091,22	2023,12	-3,26	3214,38	3265,66	1,60	4178,38	4172,55	-0,14	4817,87	4851,40	0,70
ftv90	2102,76	2124,56	1,04	3072,16	3110,19	1,24	4058,38	4054,42	-0,10	4841,05	4852,61	0,24
ftv100	2164,21	2163,35	-0,04	3166,07	3143,14	-0,72	4086,67	4008,02	-1,92	5006,23	5016,55	0,21
ftv110	2159,62	2120,93	-1,79	3165,23	3193,83	0,90	4041,41	4028,68	-0,31	5028,23	5031,70	0,07
ftv120	2218,15	2235,23	0,77	3191,06	3123,98	-2,10	4073,00	4071,31	-0,04	5150,48	5138,32	-0,24
kro124	2238,79	2257,50	0,84	3090,46	3141,69	1,66	4188,05	4101,60	-2,06	5097,58	5048,80	-0,96
ftv130	2216,48	2228,99	0,56	3182,73	3109,61	-2,30	4145,62	4150,57	0,12	5166,33	5118,53	-0,93
ftv140	2217,79	2204,65	-0,59	3082,27	3119,33	1,20	4148,76	4153,01	0,10	5152,41	5133,49	-0,37
ftv150	2219,02	2235,30	0,73	3110,74	3136,15	0,82	4087,27	4038,92	-1,18	5098,06	5070,13	-0,55
ftv160	2133,20	2230,19	4,55	3035,25	3072,33	1,22	4071,79	4076,86	0,12	5138,46	5111,41	-0,53
ftv170	2124,46	2119,17	-0,25	3084,85	3097,20	0,40	4104,17	4098,66	-0,13	5110,66	5106,93	-0,07
rbg323	2101,07	2116,28	0,72	3107,53	3139,50	1,03	4179,78	4185,38	0,13	5072,20	5039,78	-0,64
rbg358	2053,42	2071,00	0,86	3076,43	3059,13	-0,56	4271,46	4268,65	-0,07	5102,28	5105,37	0,06
rbg403	2100,42	2108,49	0,38	3107,89	3103,35	-0,15	4263,04	4197,00	-1,55	5096,44	5095,62	-0,02
rbg443	2153,90	2138,11	-0,73	3252,32	3204,41	-1,47	4214,55	4206,59	-0,19	5118,05	5112,65	-0,11

Tabela 5.11: Resultados computacionais do GRASP C3BL5 com ILS e SA, para múltiplas UMP's, limitado a 200 iterações. Os campos identificados como 'Diferença' representam a diferença percentual entre os resultados obtidos através das heurísticas ILS e SA propostas.

C3BL5 Instância	2 UMP's		3 UMP's		4 UMP's		5 UMP's	
	ILS	SA	ILS	SA	ILS	SA	ILS	SA
br17	2,48	2,14	2,47	2,04	2,74	2,63	3,78	3,53
ftv33	4,68	4,28	5,81	5,68	6,98	6,84	7,29	7,85
ftv35	4,85	4,81	6,54	5,76	8,89	8,20	13,43	10,33
ftv38	7,98	6,21	10,99	11,06	13,14	12,29	16,55	14,51
p43	7,91	7,74	12,50	11,79	18,03	19,01	30,58	29,03
ftv44	9,60	9,11	14,24	13,06	21,00	19,14	32,40	33,42
ftv47	11,56	10,45	21,62	19,63	28,37	30,71	40,87	39,89
ry48p	13,64	10,53	22,65	21,25	32,80	36,08	53,47	49,05
ft53	17,77	16,31	27,77	24,28	42,72	46,35	81,92	83,57
ftv55	19,14	20,78	29,05	29,40	56,25	50,20	112,59	107,23
ftv64	28,63	27,58	38,59	34,90	77,50	69,64	240,17	226,58
ftv70	40,43	41,02	58,84	56,87	88,35	84,37	287,13	261,50
ft70	41,96	40,35	51,06	55,45	93,51	86,98	285,21	266,35
ftv90	68,13	65,94	80,32	86,26	102,26	98,42	298,14	278,64
ftv100	84,43	87,32	96,60	95,67	113,18	108,37	306,56	296,87
ftv110	88,98	83,62	102,72	99,26	125,30	129,74	333,09	320,28
ftv120	129,04	135,28	166,83	164,27	218,32	212,31	375,82	364,88
kro124	88,77	91,71	120,83	136,12	219,59	254,36	420,42	435,83
ftv130	187,61	192,12	202,24	203,52	317,55	298,45	609,35	547,40
ftv140	218,39	212,69	252,84	259,07	346,93	328,94	662,03	602,75
ftv150	250,72	252,25	300,15	308,88	473,58	452,56	705,28	661,45
ftv160	281,15	290,07	350,80	359,44	504,41	485,39	756,80	734,05
ftv170	381,25	376,07	439,66	438,19	560,42	537,67	847,62	822,94
rbg323	1364,66	1431,95	1644,19	1679,26	2001,04	1893,56	2744,50	2611,96
rbg358	1976,10	1968,48	2159,65	2189,98	2569,70	2405,81	3130,91	3033,89
rbg403	2197,56	2162,68	2431,36	2416,14	2804,79	2775,90	3336,43	3277,55
rbg443	3034,25	3097,70	3305,21	3385,45	3603,62	3596,84	3908,13	3857,97

Tabela 5.12: Tempos computacionais do GRASP C3BL5 com ILS e SA, para múltiplas UMP's, limitado a 200 iterações.

O mesmo conjunto de testes empregando a versão C3BL5 e as heurísticas propostas foi executado utilizando limites no tempo de execução de acordo com o tamanho de cada instância. Os limites impostos são múltiplos dos empregados nos testes realizados empregando somente uma UMP, e foram multiplicados pelo número de UMP's consideradas em cada conjunto de testes, e os resultados são apresentados na Tabela 5.13, cujas colunas seguem o mesmo modelo de apresentação utilizado na Tabela 5.11.

- 2 UMP's:
 - ILS: 18 instâncias;
 - SA: 9 instâncias;

- 3 UMP's:
 - ILS: 16 instâncias;
 - SA: 11 instâncias;
- 4 UMP's:
 - ILS: 14 instâncias;
 - SA: 13 instâncias;
- 5 UMP's:
 - ILS: 12 instâncias;
 - SA: 11 instâncias;
 - Empates: 4 instâncias.

Nos testes realizados com um limite de tempo de execução, pode-se observar que o ILS foi capaz de encontrar melhores resultados para todos os testes realizados com múltiplas UMP's, porém, a diferença na quantidade de instâncias que cada estratégia mostrou-se capaz de encontrar os melhores resultados diminuiu conforme o número de UMP's empregadas foi aumentando.

As diferenças percentuais entre os resultados das heurísticas apresentaram variações pequenas porque foram inferiores a 1% para a maioria das soluções encontradas. Quando comparados estes resultados com os obtidos nas execuções limitadas por um número máximo de iterações, apresentados na Tabela 5.11, observa-se que os resultados das execuções limitadas por tempo de execução foram superiores para as instâncias cujo limite de tempo fixado foi superior ao tempo de execução dos testes limitados por um número pré-estabelecido de iterações. Apenas para a instância '*ftv70*' empregando a heurística ILS com o uso de cinco UMP's foi observado um empate.

C3BL5 Instância	2 UMPs			3 UMPs			4 UMPs			5 UMPs		
	ILS	SA	Diferença	ILS	AS	Diferença	ILS	SA	Diferença	ILS	SA	Diferença
br17	1349,28	1344,59	-0,35	1375,08	1358,89	-1,18	1379,61	1377,19	-0,18	1403,54	1403,54	0,00
ftv33	1903,69	1884,98	-0,98	2917,25	2896,16	-0,72	3220,11	3283,7	1,97	4133,63	4133,63	0,00
ftv35	1892,98	1887,93	-0,27	2975,64	2912,1	-2,14	3296,35	3289,14	-0,22	4084,72	4084,72	0,00
ftv38	1926,33	1915,52	-0,56	2970,82	2925,25	-1,53	3463,13	3472,66	0,28	4086,57	4086,57	0,00
p43	1918,18	1933,79	0,81	2952,11	2999,3	1,60	3939,19	3916,63	-0,57	4717,04	4710,95	-0,13
ftv44	1990,88	1989,84	-0,05	3050,34	3036,53	-0,45	3962,51	3962,65	0,00	4660,19	4653,35	-0,15
ftv47	2013,45	2004,18	-0,46	3085,65	3078,15	-0,24	4017,36	4098,86	2,03	4741,46	4739,77	-0,04
ry48p	2037,57	2055,77	0,89	3082,87	3021,41	-1,99	4096,62	4093,52	-0,08	4760,41	4759,39	-0,02
ft53	2029,23	2021,65	-0,37	3088,5	3051,88	-1,19	4092,73	4091,1	-0,04	4761,67	4780,57	0,40
ftv55	1990,88	2043,08	2,62	2984,48	2914,91	-2,33	3908,15	3875,58	-0,83	4782,02	4797,8	0,33
ftv64	1948,04	1953,37	0,27	2808,72	2800,95	-0,28	3781,98	3773,93	-0,21	4796,47	4775,44	-0,44
ftv70	1908,95	1904,43	-0,24	2946,64	2987,91	1,40	3901,36	3895,44	-0,15	4807,64	4805,82	-0,04
ft70	1999,65	1990,79	-0,44	3084,03	3034,28	-1,61	3896,14	3898	0,05	4803,16	4841,21	0,79
ftv90	2054,53	2052,41	-0,10	2993,11	2968,7	-0,82	3876,12	3880,83	0,12	4821,65	4838,49	0,35
ftv100	2012,45	2037,21	1,23	3026,54	2963,38	-2,09	3816,06	3844,37	0,74	4901,42	4900,18	-0,03
ftv110	2064,75	2087,94	1,12	3014,23	3106,61	3,06	3972,45	3975,86	0,09	4959,17	4995,25	0,73
ftv120	2087,08	2056,45	-1,47	3043,6	3084,31	1,34	3821,12	3805,84	-0,40	5013,82	5009,69	-0,08
kro124	2119,3	2116,54	-0,13	2912,41	2904,57	-0,27	3992,25	3853,41	-3,48	5069,27	5033,67	-0,70
ftv130	2102,97	2147,13	2,10	3040,72	3070,81	0,99	3999,29	3855,17	-3,60	5081,8	5020,16	-1,21
ftv140	2134,69	2130,94	-0,18	2875,91	2900,68	0,86	3988,46	3852,58	-3,41	4998,77	5044,98	0,92
ftv150	2124,54	2115,7	-0,42	2895,59	2892,31	-0,11	3820,38	3865,33	1,18	4942,66	4999,23	1,14
ftv160	2093,99	2044,9	-2,34	2809,99	2863,63	1,91	3809,79	3854,41	1,17	4961,61	5027,19	1,32
ftv170	2088,21	2045,76	-2,03	2910,95	2955,28	1,52	3996,59	3943,45	-1,33	4991,6	5009,72	0,36
rbg323	2041,17	2094,11	2,59	3022,07	3069,11	1,56	4039,13	4058,64	0,48	4876,03	4859,69	-0,34
rbg358	2005,93	2021,7	0,79	2949,15	2903,66	-1,54	3904,53	3915,99	0,29	4923,11	4923,55	0,01
rbg403	2025,38	2025,12	-0,01	3021,08	3045,74	0,82	3957,82	4027,94	1,77	4867,85	4881,09	0,27
rbg443	2123,6	2108,16	-0,73	3103,32	3132,08	0,93	4066,53	4056,73	-0,24	4925,51	4917,05	-0,17

Tabela 5.13: Resultados computacionais do GRASP C3BL5 com ILS e SA, para múltiplas UMP's, com tempo de execução limitado. Os campos identificados como 'Diferença' representam a diferença percentual entre os resultados obtidos através das heurísticas ILS e SA propostas.

5.2.2 – AVALIAÇÃO PROBABILÍSTICA DOS RESULTADOS EMPREGANDO MÚLTIPLAS UMP'S

Os resultados dos testes realizados com as heurísticas híbridas empregando 5 UMP's foram utilizados para a geração dos gráficos 5.3 e 5.4, que representam as probabilidades de convergência para um alvo de dificuldade média e um difícil, respectivamente. Assim como observado nos resultados empregando uma UMP, as heurísticas propostas apresentaram comportamento parecido, apenas a partir do ponto de 40% de probabilidade de convergência que a heurística GRASP+ILS apresentou uma pequena vantagem sobre o GRASP+SA.

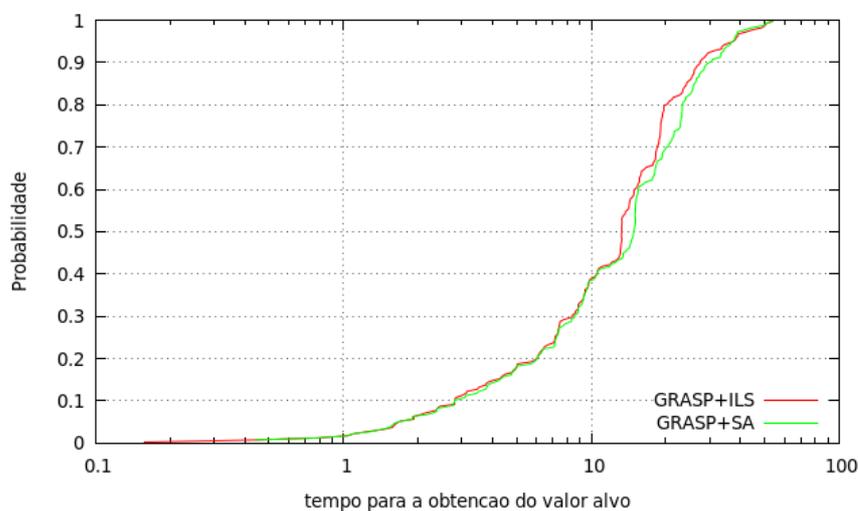


Figura 5.3: Resultados das heurísticas híbridas empregando 5 UMP's para a obtenção de um alvo de dificuldade média.

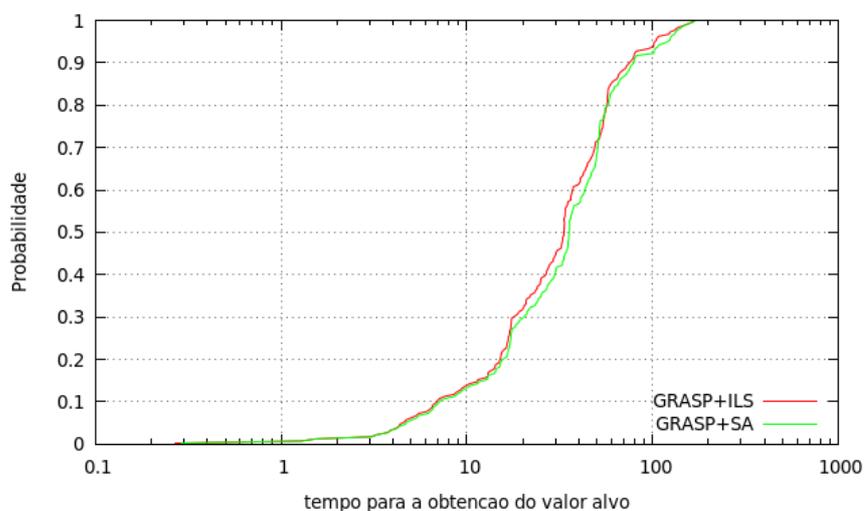


Figura 5.4: Resultados das heurísticas híbridas empregando 5 UMP's para a obtenção de um alvo de dificuldade alta.

5.2.3 – COMPARAÇÃO ENTRE OS RESULTADOS OBTIDOS POR MEIO DOS MÉTODOS HEURÍSTICOS HÍBRIDOS E OS OBTIDOS POR MEIO DE UM MÉTODO EXATO

Com o objetivo de tentar estabelecer os resultados exatos do PRP-UMP quando empregadas múltiplas UMP's foi empregada a formulação matemática apresentada na Seção 4.2, no *solver* IBM ILOG CPLEX, e foram executados testes computacionais com limites de tempo de execução distintos, considerando a utilização de 5 UMP's simultâneas nas atividades de pistoneio. Os resultados obtidos são comparados a seguir com os obtidos por meio das heurísticas, que foram executadas com o mesmo limite de tempo utilizado pela solução exata. Foram realizadas 10 execuções de cada um dos métodos heurísticos propostos, cada uma com um valor distinto de semente, utilizando a versão C3BL5, pois em testes anteriores, esta se mostrou a versão que obteve os melhores resultados para a maior parte das instâncias.

O primeiro conjunto de testes empregou limites de tempos separados em conjuntos, de acordo com o tamanho das instâncias utilizadas. A Tabela 5.14 apresenta os resultados organizados da seguinte forma: a coluna 1 identifica a instância empregada, a coluna 2 os resultados obtidos por meio do CPLEX e a coluna 3 os *gaps* do dual CPLEX. As colunas 4 e 5 apresentam, respectivamente, os resultados obtidos pela heurística GRASP+ILS e os *gaps* calculados entre os resultados da heurística e os resultados do CPLEX. As colunas 6 e 7 representam, respectivamente, os resultados obtidos pela heurística SA e os *gaps* calculados em relação aos resultados do método exato. Na coluna 8 são apresentados os limites de tempo estabelecido para cada instância, utilizado tanto pelas heurísticas quanto pelo método exato. As linhas sombreadas indicam as instâncias que o CPLEX terminou sua execução dentro do limite de tempo estabelecido, e, portanto, representam a solução exata da instância para o PRP-UMP considerando a utilização de 5 UMP's.

Instância	GRASP+ILS				GRASP+SA		T. Lim. (s)
	CPLEX	Dual CPLEX	ILS	gap CPLEX	S.A.	gap CPLEX	
br17	1403,54	0,00	1403,54	0,00	1403,27	-0,02	40,00
ftv33	4133,63	0,00	4131,17	-0,06	4133,63	0,00	40,00
ftv35	4063,72	0,52	4077,51	0,34	4078,03	0,35	40,00
ftv38	4086,57	0,00	4086,57	0,00	4086,57	0,00	40,00
p43	4715,02	0,26	4717,04	0,04	4709,33	-0,12	80,00
ftv44	4702,13	0,15	4660,19	-0,89	4662,51	-0,84	80,00
ftv47	4733,28	0,34	4738,51	0,11	4735,06	0,04	80,00
ry48p	4772,40	0,42	4758,04	-0,30	4761,39	-0,23	80,00
ft53	4881,94	0,35	4883,43	0,03	4885,62	0,08	400,00
ftv55	4936,03	0,17	4932,18	-0,08	4934,83	-0,02	400,00
ftv64	4999,38	0,46	4994,10	-0,11	4992,62	-0,14	400,00
ftv70	5086,33	0,20	5063,74	-0,44	5071,71	-0,29	400,00
ft70	5033,12	0,16	5003,95	-0,58	5000,15	-0,66	400,00
ftv90	5092,03	0,64	5083,26	-0,17	5083,70	-0,16	600,00
ftv100	5064,55	0,35	5064,82	0,01	5063,73	-0,02	600,00
ftv110	5053,16	0,43	5055,75	0,05	5055,08	0,04	600,00
ftv120	5206,45	0,43	5200,46	-0,12	5198,42	-0,15	600,00
kro124	5119,06	0,21	5102,53	-0,32	5108,37	-0,21	600,00
ftv130	5206,41	0,54	5210,17	0,07	5208,63	0,04	1200,00
ftv140	5211,46	0,43	5194,32	-0,33	5196,47	-0,29	1200,00
ftv150	5162,73	0,68	5164,46	0,03	5164,38	0,03	1200,00
ftv160	5173,25	0,30	5173,82	0,01	5174,08	0,02	1200,00
ftv170	5173,29	0,69	5191,13	0,34	5189,69	0,32	1200,00
rbg323	5101,38	0,45	5108,41	0,14	5110,74	0,18	5400,00
rbg358	5148,42	0,67	5150,06	0,03	5149,93	0,03	5400,00
rbg403	5138,47	0,16	5099,35	-0,76	5096,06	-0,83	5400,00
rbg443	5185,24	0,18	5135,62	-0,96	5128,74	-1,09	5400,00

Tabela 5.14: Resultados computacionais do CPLEX e das heurísticas ILS e SA propostas empregando 5 UMP's, versão C3BL5, alfa=0.05, limitadas por tempo de execução. As colunas identificadas como 'gap CPLEX' representam a diferença percentual entre os resultados obtidos pelo método exato e as heurísticas propostas. Valores positivos indicam que o resultado obtido pela heurística foi melhor que o resultado obtido pelo CPLEX.

Os resultados exatos foram encontrados para três instâncias dentre as testadas, cujas linhas na Tabela 5.14 foram destacadas. Para a instância denominada 'br17' o GRASP+ILS obteve êxito em encontrar o ótimo, na denominada 'ftv33' o GRASP+ILS encontrou a melhor solução, e na denominada 'ftv38' ambas as heurísticas propostas alcançaram o melhor resultado possível.

Na comparação entre as heurísticas, o GRASP+ILS encontrou o melhor resultado para 14 instâncias, enquanto o GRASP+SA obteve resultados melhores para 12 instâncias e foi

observada a ocorrência de um empate. Em relação aos resultados obtidos pelo CPLEX, o GRASP+ILS obteve resultados melhores para 12 instâncias, enquanto o GRASP+SA conseguiu resultados melhores que os obtidos pelo método exato para 10 instâncias. Embora para os demais resultados as heurísticas obtiverem resultados inferiores, observa-se que a diferença percentual entre estes e os encontrados pelo CPLEX é inferior a 1% para a quase totalidade das instâncias, o único resultado ligeiramente melhor a este percentual foi observado no resultado do GRASP+SA para a instância identificada como 'rbg443'.

Para a realização da análise estatística dos resultados, foi empregado inicialmente o teste de Shapiro-Wilk com o nível de significância $\alpha=5\%$. As heurísticas obtiveram os valores de probabilidade $p\text{-valor}_{(ILS)}= 3,32E-08$ e $p\text{-valor}_{(SA)}= 3,3E-08$. Como os resultados encontrados para o $p\text{-valor}$ de ambas as heurísticas são superiores ao nível de significância estabelecido, conclui-se que os resultados não apresentam distribuição normal.

Em seguir foi empregado o teste de Wilcoxon pareado com $\alpha=5\%$, e para 26 das 27 instâncias testadas as heurísticas propostas encontraram resultados distintos. Empregando o valor crítico $T\text{-crit}=98$, e considerando que o menor somatório do *ranking* de classificação das diferenças entre os resultados das heurísticas é $T=175$, pode concluir, uma vez que $T > T\text{-crit}$, que não existe uma diferença estatisticamente significativa entre os resultados encontrados.

Uma nova sequência de testes foi executada com o método exato e as heurísticas propostas aumentando o tempo limite de execução para 7200 segundos para todas as instâncias, independente do número de elementos que as compõem com o objetivo de validar a hipótese que as heurísticas apresentam a capacidade de encontrar as soluções exatas das instâncias obtidas por meio de um método exato, e nos demais casos, encontrar soluções superiores. A Tabela 5.15 apresenta estes resultados, utilizando a mesma organização de colunas empregada na Tabela 5.14, com exceção da coluna 7, que indicava os limites de tempo de execução das instâncias e que agora foi omitida, pelo valor ser o mesmo para todas as instâncias.

Instância	CPLEX	GRASP+ILS			GRASP+SA	
		Dual CPLEX	ILS	gap CPLEX	S.A.	gap CPLEX
ftv35	4084,72	0,00	4084,72	0,00	4084,72	0,00
p43	4722,59	0,21	4726,58	0,08	4728,20	0,12
ftv44	4702,93	0,09	4691,25	-0,25	4684,38	-0,39
ftv47	4741,37	0,30	4747,37	0,13	4743,77	0,05
ry48p	4780,06	0,31	4787,44	0,15	4786,49	0,13
ft53	4886,49	0,27	4888,29	0,04	4893,71	0,15
ftv55	4981,51	0,11	4966,37	-0,30	4969,25	-0,25
ftv64	5001,55	0,35	5003,84	0,05	4998,48	-0,06
ftv70	5088,14	0,18	5086,35	-0,04	5082,44	-0,11
ft70	5038,81	0,12	5016,78	-0,44	5016,38	-0,45
ftv90	5093,17	0,62	5107,91	0,29	5101,17	0,16
ftv100	5073,91	0,28	5077,13	0,06	5076,83	0,06
ftv110	5060,42	0,40	5064,39	0,08	5060,62	0,00
ftv120	5212,49	0,29	5218,54	0,12	5221,05	0,16
kro124	5119,06	0,21	5119,06	0,00	5117,38	-0,03
ftv130	5206,41	0,54	5223,18	0,32	5226,41	0,38
ftv140	5211,80	0,41	5219,35	0,14	5209,54	-0,04
ftv150	5168,58	0,43	5189,30	0,40	5183,04	0,28
ftv160	5173,25	0,30	5180,47	0,14	5180,35	0,14
ftv170	5175,09	0,64	5203,66	0,55	5196,53	0,41
rbg323	5116,03	0,39	5125,45	0,18	5124,83	0,17
rbg358	5156,47	0,55	5164,75	0,16	5168,38	0,23
rbg403	5138,47	0,16	5131,58	-0,13	5127,37	-0,22
rbg443	5185,24	0,18	5187,41	0,04	5185,93	0,01

Tabela 5.15: Resultados computacionais do CPLEX e das heurísticas ILS e SA propostas empregando 5 UMP's, versão C3BL5, alfa=0.05, com tempo de execução limitado a 7200 segundos. As colunas identificadas como 'gap CPLEX' representam a diferença percentual entre os resultados obtidos pelo método exato e as heurísticas propostas. Valores positivos indicam que o resultado obtido pela heurística foi melhor que o resultado obtido pelo CPLEX.

Pode-se observar na Tabela 5.15 que a linha correspondente à instância 'ftv35' foi destacada, indicando que tanto o CPLEX quanto as heurísticas propostas foram capazes de encontrar a solução exata. Para as demais instâncias, a heurística GRASP+ILS mostrou-se capaz de encontrar resultados superiores para 17 instâncias, enquanto o GRASP+SA encontrou os melhores resultados para as seis instâncias restantes. Nota-se ainda uma considerável melhoria entre os resultados do método exato e os obtidos por meio das heurísticas: o GRASP+ILS gerou resultados superiores aos obtidos pelo CPLEX em 18 das 23 instâncias cujo resultado exato não é conhecido, e o GRASP+SA obteve resultados melhores do que os encontrados pelo CPLEX em 15 instâncias.

A análise estatística dos resultados feita por meio do teste de Shapiro-Wilk com o nível de significância $\alpha=5\%$ obteve os seguintes resultados: $p\text{-valor}_{(ILS)}= 6,13591\text{E-}05$ e $p\text{-valor}_{(SA)}= 5,84381\text{E-}05$. Como os valores calculados para o $p\text{-valor}$ de cada heurística são inferiores ao valor definido para *alfa*, conclui-se que os resultados encontrados não seguem a distribuição normal.

Os resultados das heurísticas foram, em seguida, aplicados ao teste de Wilcoxon pareado empregando nível de significância $\alpha=5\%$. Das 24 instâncias empregadas, as heurísticas encontraram resultados distintos para 23 delas, que foram utilizados para estabelecer o valor $T=68$. Para o número de instâncias e o valor de alfa estabelecido, é empregado $T\text{-crit}=73$, e como tem-se a relação $T < T\text{-crit}$, pode-se afirmar que os resultados encontrados pelas heurísticas apresentam diferença estatisticamente significativa entre si.

O último conjunto de testes realizados empregou como critério de parada tanto para o método exato quanto para as heurísticas propostas o limite de tempo de execução igual a 10800 segundos. Com os resultados destes testes espera-se avaliar a capacidade das heurísticas em obter melhores resultados que o método exato se os métodos dispuserem de mais tempo de execução para a exploração de mais soluções do espaço de soluções do problema. A Tabela 5.16 apresenta estes resultados, utilizando a mesma organização de colunas empregada na Tabela 5.15.

Comparando os resultados das heurísticas com os obtidos pelo método exato, pode-se notar que, para apenas duas instâncias (identificadas como '*ftv55*' e '*ft70*'), ambas as heurísticas obtiveram resultados inferiores aos obtidos através do CPLEX. Para três instâncias ('*ftv70*', '*ftv100*' e '*rbg403*'), enquanto uma das heurísticas apresentou resultado inferior ao obtido pelo método exato, a outra obteve um resultado superior. Do total de 23 instâncias testadas, os métodos heurísticos propostos superaram o método exato em 18 instâncias.

Analisando os resultados das heurísticas propostas, o GRASP+ILS obteve os melhores resultados para 16 das instâncias testadas, enquanto o GRASP+SA foi capaz de encontrar resultados superiores para seis instâncias. Um empate entre as heurísticas foi observado.

A fim de determinar se existem diferenças significativamente estatísticas entre os resultados dos métodos heurísticos propostos, foi realizado inicialmente o teste de Shapiro-Wilk adotando o nível de significância $\alpha=5\%$. Os $p\text{-valores}$ obtidos a partir dos resultados das heurísticas são os seguintes: $p\text{-valor}_{(ILS)}= 0,002619$ e $p\text{-valor}_{(SA)}= 0,002698$. Como ambos os valores encontrados são inferiores ao valor estabelecido para alfa, pode-se afirmar que os resultados não seguem a distribuição normal.

A análise seguinte foi realizada por meio do teste de Wilcoxon pareado empregando nível de significância $\alpha=5\%$. Considerando que as heurísticas obtiveram resultados distintos para 22 das 23 instâncias empregadas, empregou-se o valor $T = 60$. O valor T-crit obtido a partir da classificação dos resultados é igual a 65, e como o valor de T foi inferior ao valor de T-crit, pode-se concluir que os resultados obtidos por meio das heurísticas ILS e SA propostas apresentam diferença estatisticamente significativa entre si.

Instância	CPLEX	Iterated Local Search			Simulated Annealing	
		Dual CPLEX	ILS	gap CPLEX	S.A.	gap CPLEX
p43	4728,76	0,12	4731,64	0,06	4729,86	0,02
ftv44	4708,03	0,07	4710,43	0,05	4708,75	0,02
ftv47	4744,26	0,22	4752,86	0,18	4749,69	0,11
ry48p	4787,27	0,15	4791,43	0,09	4790,93	0,08
ft53	4895,63	0,13	4896,52	0,02	4898,18	0,05
ftv55	4986,31	0,09	4979,05	-0,15	4973,51	-0,26
ftv64	5005,17	0,29	5007,38	0,04	5011,06	0,12
ftv70	5092,49	0,08	5094,3	0,04	5091,73	-0,01
ft70	5038,81	0,12	5018,73	-0,40	5021,26	-0,35
ftv90	5110,78	0,48	5118,04	0,14	5116,58	0,11
ftv100	5084,94	0,22	5088,93	0,08	5084,63	-0,01
ftv110	5063,17	0,36	5070,11	0,14	5067,83	0,09
ftv120	5219,77	0,17	5224,4	0,09	5222,66	0,06
kro124	5122,63	0,19	5127,83	0,10	5124,90	0,04
ftv130	5213,79	0,43	5228,4	0,28	5228,40	0,28
ftv140	5220,55	0,31	5226,74	0,12	5222,42	0,04
ftv150	5184,06	0,37	5197,68	0,26	5194,03	0,19
ftv160	5179,69	0,16	5185,27	0,11	5183,49	0,07
ftv170	5188,27	0,44	5208,49	0,39	5203,71	0,30
rbg323	5125,43	0,12	5126,77	0,03	5128,48	0,06
rbg358	5167,15	0,20	5174,88	0,15	5171,18	0,08
rbg403	5138,47	0,11	5136,29	-0,04	5141,55	0,06
rbg443	5188,62	0,16	5188,80	0,00	5190,13	0,03

Tabela 5.16: Resultados computacionais do CPLEX e das heurísticas ILS e SA propostas empregando 5 UMP's, versão C3BL5, alfa=0.05, com tempo de execução limitado a 10800 segundos. As colunas identificadas como 'gap CPLEX' representam a diferença percentual entre os resultados obtidos pelo método exato e as heurísticas propostas. Valores positivos indicam que o resultado obtido pela heurística foi melhor que o resultado obtido pelo CPLEX.

CAPÍTULO 6 – CONCLUSÕES

Este trabalho apresenta uma nova proposta de utilizar as metaheurísticas GRASP, *Iterated Local Search* e *Simulated Annealing* aplicadas ao PRP-UMP, considerando a utilização de uma ou múltiplas UMP's. As soluções propostas foram empregadas para a construção de uma heurística GRASP e das heurísticas híbridas GRASP+ILS e GRASP+SA.

Foram apresentados três métodos construtivos de rotas para o PRP-UMP, cinco estratégias de busca local e dez operadores de perturbação, cujas combinações resultaram em diferentes heurísticas GRASP, dentre as quais a denominada C3BL5 foi capaz de encontrar os melhores resultados dentre elas, e para várias das instâncias experimentadas, resultados superiores aos da literatura e dos obtidos por meio de um método exato.

A utilização de um repositório com os operadores de perturbação permitiu que várias operações de modificação fossem aplicadas às rotas construídas pela heurística nas iterações iniciais, e que um subconjunto formado pelos operadores responsáveis por gerar as melhores soluções fosse empregado nas iterações avançadas das heurísticas.

Experimentos computacionais mostraram que os métodos propostos são promissores tanto pelos resultados obtidos como pelo fato de possuírem uma estrutura bastante simples, que facilitará a incorporação de novos módulos que visem melhorar ainda mais seu desempenho.

Os testes computacionais realizados considerando a utilização de uma UMP foram executados inicialmente com todas as versões do GRASP geradas a partir da combinação dos métodos construtivos e de busca local apresentados. As versões identificadas como C3BL4 e C3BL5 apresentaram resultados significativamente superiores aos obtidos pelas outras versões e superiores aos da literatura para até 22 das 27 instâncias empregadas nos testes limitados por um número máximo de iterações. Nos testes com limite de tempo os resultados também foram superiores aos apresentados na literatura para a maior parte das instâncias.

Foram realizados experimentos com a variação do parâmetro *alfa*, que controla o tamanho de uma lista restrita de candidatos de alta atratividade à inserção na rota da UMP. Observou-se que a adoção de um valor menor para *alfa*, que na prática significa trabalhar com uma lista menor formada pelos candidatos de maior atratividade, contribuiu de forma mais significativa para a obtenção das melhores soluções.

O GRASP proposto foi combinado com versões desenvolvidas para o PRP-UMP criando as heurísticas híbridas GRASP+ILS e GRASP+SA, empregando as versões C3BL4 e

C3BL5 com o valor atribuído para *alfa* igual a 0.05. As duas heurísticas híbridas propostas foram capazes de encontrar resultados superiores aos apresentados na literatura para 23 das 27 instâncias, tanto para os testes limitados por um número máximo de iterações quanto nos testes com limite de tempo quando utilizada a versão C3BL5 do GRASP.

Novos testes foram realizados com as heurísticas híbridas com a versão C3BL5 do GRASP e *alfa*=0.05 e o método exato desenvolvido para uma UMP com três limites de tempo distintos. Nos testes iniciais as instâncias foram separadas em blocos de tempo máximo de execução de acordo com seu tamanho. Foram encontrados os resultados exatos para 6 instâncias, que também foram alcançados pelas heurísticas híbridas, que para as demais instâncias foram capazes de encontrar resultados superiores aos obtidos pelo método exato para 9 instâncias. Foi feita a análise estatística entre os resultados das heurísticas híbridas empregando os métodos de Shapiro-Wilk e o teste de Wilcoxon pareado e não foram observadas diferenças significativas entre os resultados.

Em seguida, os testes foram realizados com o limite de tempo igual a 7200 segundos, e os resultados obtidos por meio das heurísticas híbridas apresentaram significativa melhora: em 16 das 21 instâncias testadas, as heurísticas encontraram resultados superiores aos do método exato, e a versão GRASP+ILS foi capaz de encontrar resultados superiores aos gerados pelo GRASP+SA para 16 instâncias. Os testes estatísticos mostraram diferença significativa entre os resultados das heurísticas híbridas.

Um último conjunto de testes, com limite de tempo igual a 10800 segundos foi realizado, e ambas as heurísticas híbridas foram capazes de encontrar resultados superiores aos obtidos por meio do método exato para 20 das 21 instâncias testadas. A heurística GRASP+ILS encontrou os melhores resultados para 15 instâncias e a GRASP+SA para por meio instâncias. A análise estatística dos resultados indicou que há diferença estatisticamente significativa entre os resultados das heurísticas híbridas.

Baseando-se nos resultados obtidos nos testes empregando uma UMP, os experimentos realizados com múltiplas UMP's utilizaram a versão do GRASP C3BL5 com *alfa*=0.05. Para todas os testes realizados com diferentes quantidades de UMP's e execução limitada pelo número de iterações ou tempo de execução, o GRASP+ILS foi capaz de encontrar o melhor resultado com mais frequência do que o GRASP+SA.

Os testes realizados com as heurísticas híbridas propostas e um método computacional exato, seguidos da análise estatística dos resultados foram realizados com a utilização de

cinco UMP's. Nos testes realizados com limites de tempo separados por blocos de acordo com o tamanho das instâncias, foram encontradas as soluções exatas para três das heurísticas empregadas, e ao menos uma das heurísticas híbridas foi capaz de encontrar estes resultados. Para as 24 instâncias restantes, o GRASP+ILS encontrou resultados superiores aos obtidos pelo método exato para 12 instâncias, e o GRASP+SA para dez instâncias. Na comparação entre os resultados obtidos pelas heurísticas propostas, o GRASP+ILS encontrou melhores resultados para 14 instâncias, e o GRASP+SA para 12 instâncias. A análise estatística dos resultados não encontrou diferença significativa entre os resultados das heurísticas propostas.

Nos testes realizados com um limite de tempo de execução de 7200 segundos, o GRASP+ILS encontrou resultados superiores aos obtidos pelo método exato para 17 instâncias, enquanto o GRASP+SA encontrou resultados superiores para 15 instâncias. Na comparação entre as heurísticas, o GRASP+ILS encontrou o melhor resultado para 17 instâncias, enquanto o GRASP+SA mostrou-se melhor para seis instâncias. A análise estatística dos resultados das heurísticas mostrou que há diferença significativa entre eles.

O último conjunto de testes realizados empregando cinco UMP's teve o tempo de execução limitado a 10800 segundos. Do total de instâncias testadas, em apenas dois casos o método exato encontrou resultados superiores aos obtidos pelas heurísticas híbridas propostas. Comparando os resultados das heurísticas, o GRASP+ILS encontrou os melhores resultados para 16 instâncias, enquanto os resultados do GRASP+SA foram superiores para seis instâncias. A análise estatística dos resultados encontrados pelas heurísticas indicou que há diferença significativa entre eles.

Para trabalhos futuros, o estudo do PRP-UMP pode ser aprofundado com a aplicação da hibridização entre métodos exatos e heurísticos, que são métodos promissores para serem aplicados em problemas de otimização [Mestria *et al*, 2010; Silva *et al*, 2013, Tavares *et al*, 1998], e uma versão para o PRP-UMP certamente apresentaria uma contribuição positiva. Finalmente, o desenvolvimento de versões paralelas das heurísticas propostas pode contribuir significativamente na obtenção das soluções exatas das instâncias formadas por uma maior quantidade de poços e que estes resultados possam ser encontrados pelas heurísticas híbridas em um tempo computacional inferior ao necessário por um método exato.

REFERÊNCIAS

Aiex, R. Resende, M. G. C., Ribeiro, C. C. *Probability distribution of solution time in GRASP: An experimental investigations*. Journal of Heuristics, v. 8, p. 343-373, 2002.

Aloise, D. J., Santos, A. C., Barros, C. A., Souza, M. C., Noronha, T. F. *GRASP Reativo aplicado ao Problema do Emprego da Unidade Móvel de Pistoneio (POE-UMP)*. Proceedings of the Congreso Chileno de Investigación Operativa (OPTIMA), Chile, 2001.

Andrade, Marcos R. Q., Ochi, Luiz Satoru; and Martins, Simone L. (2013) "Metaheurística GRASP para a solução do Problema de Roteamento Periódico de Unidades Móveis de Pistoneio.". Proc. of the X Encontro Nacional de Inteligência Artificial e Computacional (ENIAC) Fortaleza - CE (2013).

Andrade, M. R. Q., Ochi, L. S., Martins, S. L. (2015). *Heuristics for the Periodic Mobile Pump Unit Routing Problem*. International Journal of Natural Computing Research, vol 5. No. 1 (pp. 1-25). IGI Publishing, 2015. DOI: 10.4018/ijncr.2015010101.

Bassi, H.V., Ferreira Filho, V. J. M., Bahiense, L. *Solutions for the Workover Rigs Scheduling Problem using Simulation-Optimization and Path-relinking*. Rio Oil & Gas Expo and Conference 2010, 2010, Rio de Janeiro. Proceedings of Rio Oil & Gas Expo and Conference 2010. Rio de Janeiro: IBP - Instituto Brasileiro de Petroleo, 2010. v. 1. p. 1-10.

Cerny, V. (1985). *Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm*. Journal of Optimization Theory and Applications 45: 41–51. doi:10.1007/BF00940812.

Costa, L. R., Ferreira Filho, V. J. M.. *Uma Heurística de Montagem Dinâmica para o Problema de Otimização de Itinerários de Sondas*. SBPO - Simpósio Brasileiro de Pesquisa Operacional, 2005, Gramado - RS. Anais do SBPO. Rio de Janeiro: SOBRAPO, 2005. p. 1-12

Duarte, H. M.. *Um Estudo Algorítmico de Problemas logísticos na Indústria de Petróleo e Gás Natural*. Dissertação de Mestrado em Sistemas e Computação. UFRN, 2006.

Feo and M.G.C. Resende (1989) *A probabilistic heuristic for a computationally difficult set covering problem*. Operations Research Letters, 8:67–71, 1989.

Goldbarg, M. C.; Goldbarg, E. F. G.; Duarte, H. M. (2010) *Transgenetic Algorithm for the Periodic Mobile Piston Pump Unit Routing Problem with Continuous Oil Replenishment*. *International Journal of Innovative Computing and Applications*, v. 2, p. 203-214.

Goldbarg, E. F. G., Goldbarg, M. C., Neto, F. D. M. (2004). *An Evolutionary Approach for the Piston Pump Mobile Unit Problem*. In: Proceedings of the Fifth International Conference on Computer Sciences. Londres: Hermes Science Publishing, 2004. v. 01. p. 281-290.

Gonçalves, L. B., Ochi, L. S., and Martins, S. L. (2005). *A GRASP with adaptive memory for a period vehicle routing problem*. Proceedings of International Conference on Computational Intelligence for Modelling, Control & Automation (CIMCA 2005), vol. 1, pp.721–727, Vienna. IEEE, 2005.

Gonçalves, L. B, *Heurísticas GRASP para o Problema de Roteamento Periodico de Unidades Móveis de Pistoneio*; Dissertação de Mestrado em Computação: IC/UFF.

IBM (2016). IBM ILOG CPLEX Optimizer. <http://www01.ibm.com/software/integration/optimization/cplex-optimizer/>, Acessado em 2016.

Junior, P. L. O, Arroyo, J. E. C., & Souza, V. A. A. (2010). *Heurísticas GRASP e ILS para o Problema No-Wait Flowshop Scheduling Multiobjetivo*. Anais do XLII Simpósio Brasileiro de Pesquisa Operacional (XLII SBPO) (vol. 1, pp: 01-12). Bento Gonçalves, Brasil.

Kirkpatrick, S., Gelatt Jr, C. D.; Vecchi, M. P. (1983). *Optimization by Simulated Annealing*. Science 220: 671–680. doi:10.1126/science.220.4598.671. JSTOR 1690046. PMID 17813860.

Kokubugata, H. and Kawashima, H. (2008). *Application of Simulated Annealing to Routing Problems in City Logistics*, Simulated Annealing, Cher Ming Tan (Ed.), ISBN: 978-953-761907-7, InTech.

Layeb, A., Ammi, M. and Chikhi, S. (2013). *A GRASP Algorithm Based on New Randomized Heuristic for Vehicle Routing Problem*. Journal of Computing and Information Technology - CIT 21, 2013, 1, 35–46. doi:10.2498/cit.1002085

Lourenço, H. R., Martin O. and Stützle T. (2003). *Iterated Local Search*. Handbook of Metaheuristics. Kluwer Academic Publishers, International Series in Operations Research & Management Science 57: 321–353.

Matsumoto, M. and Nishimura, T. (1998). *Mersenne Twister: A 623-Dimensionally Equidistributed Uniform Pseudo-Random Number Generator*. ACM Transactions on Modeling and Computer Simulation, Vol. 8, No. 1, January 1998, pp 3--30.

Mestria, M., Martins, S. L., and Ochi, L. S. (2010) *Heurísticas Híbridas para o Problema do Caixeiro Viajante com Grupamentos*. Anais do XXX Encontro Nacional de Engenharia de Produção (XXX ENEGEP). São Carlos/SP.

Munhoz, P. L., Ochi, L. S., and Souza, M. J. F. (2012). *Um algoritmo baseado em Iterated Local Search para o Problema de Roteamento de Veículos Periódico*. Proc. of XXXII ENEGEP2012 - XXXII Enc. Nac. de Eng. de Prod., 15-18 Outubro, Bento Gonçalves/RS, 2012.

Nascimento, J., P. L. do, (2010). *Metaheurísticas Evolutivas para o Problema de Roteamento de Unidades Móveis de Pistoneio*. Dissertação Mestrado em Engenharia de Produção - UFRN.

Nascimento, J. P. L., Aloise, D. J. (2012). *Algoritmo memético com vocabulary building para o problema de roteamento de unidades móveis de pistoneio*. GEPROS. Gestão da Produção, Operações e Sistemas, v. 1, p. 81-95, 2012.

Penna, P. H. V., Subramanian, A., and Ochi, L. S. *An Iterated Local Search heuristic for the Heterogeneous Fleet Vehicle Routing Problem*. Journal of Heuristics - Elsevier, Volume 19, pp. 201-232, 2013.

Real Statistics Using Excel (2016). <http://www.real-statistics.com/>. Acessado em 2016

Reinelt, G. *A traveling salesman problem library*, ORSA Journal on Computing, v. 3, pp. 376-384, 1991.

Resende, M. G. C. and Ribeiro, C. C. (2003). *Greedy randomized adaptive search procedures*. Handbook of Metaheuristics, (pp. 219–249). Kluwer Academic Publishers.

Shapiro, S. S. and Wilk, M. B. (1965). *An analysis of variance test for normality (complete samples)*. Biometrika 52 (3–4): 591–611. doi:10.1093/biomet/52.3-4.591. JSTOR 2333709. MR 205384. p. 593

Silva, A. R.V., and Ochi, L. S. (2011). *Heurísticas Híbridas para o Problemas de Escalonamento de Projetos com Restrição de Recursos Dinâmicos*. Capítulo 13 do livro: Computação Evolucionária em Problemas da Engenharia, pp: 273-300. ISBN: 978-85-84619-00-5

Subramanian, A. ; Cabral, L. A. F. and Ochi, L. S. . *An Efficient Iterated Local Search Algorithm for the Vehicle Routing Problem with Simultaneous Pickup and Delivery*. XL Simpósio Brasileiro de Pesquisa Operacional, 2008, João Pessoa. Anais do XL SBPO. Rio de Janeiro: SOBRAPO, 2008.

Subramanian, A., Uchoa, E., and Ochi, L. S. *A Hybrid Algorithm for a Class of Vehicle Routing Problems*. Computers & Operations Research (COR), Volume 40, pp. 2519-2531 - Elsevier (2013). DOI information: <http://dx.doi.org/10.1016/j.cor.2013.01.013>.

Tavares, A.L.G., Ochi, L.S. and Boeres, M.C.S. (1998). *Metaheurísticas híbridas baseadas em redes elásticas e algoritmos evolutivos para a solução de problemas de roteamento de uma frota de veículos*. Anais do XXX Simpósio Brasileiro de Pesquisa Operacional (XXX SBPO), SOBRAPO, PUC-PR.

Zeferino, G., Amorim, F. M. S. & Filho, M. F. F. (2011). *Algoritmos, Multi-Start, GRASP e ILS Aplicados ao Problema de P-Medianas*. Anais do XLIII Simpósio Brasileiro de Pesquisa Operacional (XLIII SBPO) (vol. 1, pp: 01-08). Ubatuba, Brasil.

Wilcoxon, F. (1945). *Individual comparisons by ranking methods*. Biometrics Bulletin 1 (6): 80–83.

Apêndice 1:

Instância	C1BL1				C1BL2				C1BL3				C1BL4				C1BL5			
	Vol. Max.	Vol. Med.	Vol.Min.	T.Med.	Vol. Max.	Vol. Med.	Vol.Min.	T.Med.	Vol. Max.	Vol. Med.	Vol.Min.	T.Med.	Vol. Max.	Vol. Med.	Vol.Min.	T.Med.	Vol. Max.	Vol. Med.	Vol.Min.	T.Med.
br17	-4,33	-3,45	-5,66	0,87	-4,35	-3,47	-4,68	0,86	-4,22	-3,37	-4,62	0,85	0,47	-0,28	-1,08	0,86	0,49	-0,26	-1,11	0,85
ftv33	-1,86	-1,66	-3,58	2,31	-1,66	-1,56	-3,47	2,28	-1,69	-1,61	-2,93	2,35	-0,12	-0,31	-1,04	2,33	-0,09	-0,31	-0,96	2,33
ftv35	-1,66	-1,59	-3,47	2,83	-1,59	-1,58	-2,51	2,78	-1,72	-1,69	-2,61	2,86	-0,07	-0,33	-1,36	2,86	-0,02	-0,31	-1,28	2,78
ftv38	-2,07	-1,85	-3,76	4,22	-1,98	-1,76	-3,69	4,35	-1,85	-1,56	-2,40	4,31	-1,01	-0,29	-1,05	4,26	-1,00	-0,25	-1,05	4,22
p43	0,69	-0,71	-2,84	4,01	0,58	-0,63	-2,81	3,97	0,66	-0,69	-2,86	4,13	0,07	-0,28	-0,48	4,05	0,10	-0,26	-0,40	4,05
ftv44	-2,43	-2,07	-3,96	5,22	-2,45	-2,15	-3,03	5,17	-2,25	-1,99	-3,09	5,32	-0,62	-0,23	-1,15	5,17	-0,61	-0,20	-1,13	5,32
ftv47	-2,39	-2,41	-4,43	5,24	-2,64	-2,44	-3,33	5,35	-2,65	-2,46	-3,35	5,35	0,06	-0,34	-1,66	5,45	0,05	-0,26	-1,61	5,29
ry48p	-1,75	-1,42	-3,14	5,89	-1,63	-1,43	-3,20	6,00	-1,67	-1,41	-2,76	5,89	0,22	-0,40	-1,04	6,00	0,21	-0,42	-1,01	6,00
ft53	-4,51	-3,77	-4,32	9,21	-4,61	-3,93	-5,58	9,12	-4,01	-3,61	-4,97	9,21	0,44	-0,68	-1,88	9,39	0,52	-0,62	-1,83	9,30
ftv55	-2,72	-2,54	-3,40	9,26	-2,43	-2,32	-3,22	9,45	-2,56	-2,44	-3,35	9,63	0,08	-0,58	-1,74	9,26	0,16	-0,58	-1,65	9,26
ftv64	-2,42	-2,37	-3,41	16,81	-2,42	-2,43	-3,39	16,65	-2,47	-2,36	-3,26	16,48	-0,02	-0,33	-1,14	17,14	-0,02	-0,33	-1,06	16,48
ftv70	-2,50	-2,30	-4,17	18,63	-2,49	-2,17	-4,95	18,09	-2,52	-2,20	-3,97	18,45	-0,06	-0,41	-1,67	18,27	0,02	-0,33	-1,67	18,09
ft70	-2,80	-2,22	-3,54	12,27	-2,54	-2,00	-3,37	12,76	-2,94	-2,26	-3,55	12,64	-1,89	-0,56	-1,84	12,76	-1,85	-0,49	-1,80	12,76
ftv90	-3,26	-3,02	-4,84	38,28	-3,20	-3,03	-4,93	37,55	-3,23	-3,06	-4,94	37,18	0,22	-0,64	-1,77	38,28	0,26	-0,64	-1,69	37,91
ftv100	-3,19	-3,08	-3,92	49,15	-3,10	-3,03	-3,94	48,21	-3,17	-3,07	-3,83	48,21	0,05	-0,69	-2,04	47,28	0,11	-0,65	-1,97	47,75
ftv110	-3,43	-3,29	-4,22	63,74	-3,30	-3,19	-4,16	63,13	-3,23	-3,10	-4,08	62,52	0,13	-0,69	-2,18	61,90	0,20	-0,61	-2,16	64,35
ftv120	-3,02	-2,96	-4,91	85,42	-2,95	-2,95	-3,90	85,42	-3,09	-2,98	-3,88	86,24	0,05	-0,47	-1,80	82,95	0,08	-0,48	-1,73	84,59
kro124	-2,08	-2,13	-3,19	37,43	-2,22	-2,17	-3,15	36,35	-2,37	-2,22	-3,16	36,35	-0,07	-0,39	-1,36	37,43	0,00	-0,35	-1,33	36,35
ftv130	-3,58	-3,42	-4,37	106,91	-3,41	-3,29	-4,30	108,99	-3,51	-3,34	-4,25	107,95	-0,11	-0,51	-2,12	107,95	-0,03	-0,44	-2,08	108,99
ftv140	-3,83	-3,57	-4,64	131,07	-3,65	-3,42	-4,26	128,55	-3,77	-3,60	-4,57	127,29	-0,68	-0,53	-2,27	131,07	-0,68	-0,53	-2,26	131,07
ftv150	-3,34	-3,24	-4,13	159,16	-3,23	-3,20	-4,18	163,84	-3,22	-3,20	-4,22	162,28	0,04	-0,44	-2,98	160,72	0,01	-0,43	-2,96	162,28
ftv160	-3,81	-3,80	-4,79	174,27	-3,77	-3,78	-4,67	179,39	-3,67	-3,74	-5,09	177,68	-1,56	-0,50	-2,95	174,27	-1,57	-0,42	-2,88	177,68
ftv170	-3,53	-3,41	-5,32	214,24	-3,48	-3,36	-4,29	220,54	-3,50	-3,33	-4,21	218,44	-1,50	-0,76	-1,95	220,54	-1,45	-0,71	-1,91	218,44
rbg323	-0,52	-0,14	-0,90	776,67	-0,49	-1,03	-1,32	776,67	-0,56	-0,08	-1,22	769,05	-1,35	-0,61	-0,96	769,05	-1,31	-0,54	-0,99	784,28
rbg358	-0,51	-0,24	-1,13	850,65	-0,58	-0,35	-1,23	867,33	-0,55	-0,27	-1,25	867,33	-0,28	-0,71	-1,79	875,67	-0,22	-0,73	-1,72	842,31
rbg403	-0,74	-0,65	-1,57	1196,25	-0,71	-0,56	-1,59	1196,25	-0,91	-0,82	-1,72	1207,87	-0,03	-0,42	-1,03	1184,64	-0,04	-0,38	-1,03	1184,64
rbg443	-1,78	-2,47	-5,85	1745,14	-2,37	-2,64	-5,87	1745,14	-2,87	-2,64	-5,16	1762,42	-0,06	-0,71	-1,14	1814,25	0,00	-0,63	-1,07	1745,14

Instância	C2BL1				C2BL2				C2BL3				C2BL4				C2BL5			
	Vol. Max.	Vol. Med.	Vol.Min.	T.Med.	Vol. Max.	Vol. Med.	Vol.Min.	T.Med.	Vol. Max.	Vol. Med.	Vol.Min.	T.Med.	Vol. Max.	Vol. Med.	Vol.Min.	T.Med.	Vol. Max.	Vol. Med.	Vol.Min.	T.Med.
br17	-4,10	-3,27	-4,60	0,86	-4,03	-3,27	-4,60	0,87	-4,00	-3,17	-4,45	0,84	0,53	-0,63	-1,47	0,85	0,58	-0,63	-1,48	0,86
ftv33	-3,27	-3,07	-3,98	2,33	-2,96	-2,87	-3,87	2,37	-3,20	-3,03	-3,91	2,33	-0,34	-0,50	-1,79	2,37	-0,35	-0,41	-1,79	2,37
ftv35	-3,12	-3,08	-4,01	2,86	-3,09	-3,01	-3,87	2,83	-3,31	-3,14	-3,99	2,83	-0,19	-0,44	-1,43	2,86	-0,12	-0,40	-1,45	2,81
ftv38	-3,29	-3,03	-4,88	4,22	-3,14	-2,91	-3,87	4,35	-2,94	-2,79	-3,70	4,35	-1,52	-0,28	-1,77	4,26	-1,46	-0,30	-1,71	4,35
p43	0,09	-1,17	-2,34	4,01	0,11	-1,19	-2,37	4,01	0,55	-1,45	-2,19	3,97	0,32	-0,21	-1,22	4,13	0,39	-0,13	-1,20	3,97
ftv44	-3,28	-3,06	-3,96	5,27	-3,12	-2,95	-4,01	5,32	-3,46	-3,17	-4,07	5,27	-0,35	-0,27	-1,61	5,27	-0,28	-0,26	-1,51	5,22
ftv47	-4,34	-4,17	-5,04	5,40	-4,12	-3,03	-3,95	5,24	-4,00	-3,79	-4,23	5,40	-0,06	-0,37	-1,85	5,45	0,02	-0,36	-1,82	5,24
ry48p	-2,55	-2,28	-3,07	6,00	-2,58	-2,27	-3,01	5,83	-2,35	-2,18	-2,98	5,89	0,08	-0,39	-1,33	5,89	0,13	-0,32	-1,31	5,83
ft53	-1,77	-1,08	-2,74	9,03	-1,74	-1,00	-2,55	9,30	-1,66	-0,95	-2,52	9,39	0,28	-0,53	-0,77	9,39	0,34	-0,45	-0,78	9,03
ftv55	-3,17	-3,14	-4,06	9,45	-3,05	-2,85	-3,72	9,54	-2,88	-2,93	-3,89	9,63	0,01	-0,49	-1,33	9,45	0,03	-0,40	-1,23	9,45
ftv64	-2,90	-2,83	-3,77	16,97	-2,75	-2,64	-3,58	16,48	-2,82	-2,82	-3,82	16,48	-0,11	-0,39	-1,27	16,97	-0,11	-0,34	-1,19	16,81
ftv70	-2,31	-2,10	-3,97	18,45	-2,38	-2,08	-3,88	18,09	-2,40	-2,20	-3,03	18,09	-0,44	-0,49	-1,75	18,09	-0,46	-0,46	-1,75	18,45
ft70	-0,91	-1,27	-3,40	12,76	-1,05	-0,43	-3,29	12,39	-0,95	-0,34	-2,36	12,64	-0,38	-0,43	-1,46	12,76	-0,38	-0,35	-1,47	12,39
ftv90	-3,24	-3,01	-3,82	38,28	-3,12	-2,98	-3,95	37,18	-3,27	-3,02	-3,84	37,55	0,15	-0,55	-2,23	37,91	0,21	-0,56	-2,18	37,91
ftv100	-2,99	-3,03	-4,95	48,68	-3,13	-2,93	-3,74	49,15	-3,14	-3,03	-3,51	49,15	-0,13	-0,52	-2,04	48,68	-0,08	-0,44	-1,98	48,21
ftv110	-3,06	-3,06	-5,22	63,13	-3,11	-2,93	-4,85	63,13	-3,11	-2,99	-4,04	62,52	0,08	-0,42	-2,28	63,13	0,13	-0,34	-2,31	61,90
ftv120	-2,98	-2,94	-4,88	85,42	-2,95	-2,90	-3,84	83,77	-3,05	-2,98	-3,93	84,59	0,10	-0,37	-1,94	85,42	0,13	-0,33	-1,84	82,95
kro124	-2,67	-2,58	-3,61	36,71	-2,48	-2,43	-3,42	36,35	-2,54	-2,47	-3,45	37,79	-0,15	-0,25	-1,83	37,07	-0,11	-0,22	-1,85	36,71
ftv130	-3,46	-3,34	-4,27	106,91	-3,48	-3,37	-4,33	106,91	-3,54	-3,36	-4,30	105,88	-0,03	-0,49	-2,48	106,91	0,03	-0,45	-2,46	107,95
ftv140	-3,63	-3,43	-4,27	132,33	-3,68	-3,42	-4,25	131,07	-3,70	-3,41	-4,22	128,55	-0,20	-0,44	-2,12	128,55	-0,22	-0,40	-2,02	129,81
ftv150	-3,10	-3,09	-4,97	162,28	-3,26	-3,14	-4,99	157,60	-3,29	-3,18	-4,02	160,72	-0,07	-0,39	-2,77	157,60	0,01	-0,33	-2,78	160,72
ftv160	-3,66	-3,66	-5,72	174,27	-3,65	-3,63	-4,55	172,56	-3,72	-3,73	-4,66	175,98	-1,71	-0,41	-2,36	174,27	-1,67	-0,32	-2,26	179,39
ftv170	-3,68	-3,54	-5,43	214,24	-3,26	-3,35	-4,34	220,54	-3,39	-3,32	-4,27	212,14	-1,64	-0,67	-2,14	214,24	-1,62	-0,61	-2,08	216,34
rbg323	-7,09	-6,65	-7,31	769,05	-6,91	-6,50	-7,21	791,90	-6,74	-6,41	-7,12	769,05	-2,18	-0,64	-1,11	791,90	-2,20	-0,59	-1,11	799,51
rbg358	-3,44	-3,20	-4,05	867,33	-3,43	-3,17	-4,02	858,99	-3,14	-2,97	-3,85	875,67	-0,95	-0,54	-1,46	850,65	-0,97	-0,47	-1,43	875,67
rbg403	-3,16	-2,87	-4,65	1219,48	-2,91	-2,72	-3,65	1196,25	-2,70	-2,55	-3,41	1219,48	0,03	-0,37	-1,13	1173,02	0,10	-0,33	-1,09	1219,48
rbg443	0,16	-1,48	-4,60	1779,70	0,07	-1,41	-4,55	1796,97	0,13	-1,46	-4,62	1762,42	-0,02	-0,50	-1,26	1779,70	-0,04	-0,45	-1,21	1796,97

Instância	C3BL1				C3BL2				C3BL3				C3BL4				C3BL5			
	Vol. Max.	Vol. Med.	Vol.Min.	T.Med.	Vol. Max.	Vol. Med.	Vol.Min.	T.Med.	Vol. Max.	Vol. Med.	Vol.Min.	T.Med.	Vol. Max.	Vol. Med.	Vol.Min.	T.Med.	Vol. Max.	Vol. Med.	Vol.Min.	T.Med.
br17	-4,29	-3,42	-4,67	0,87	-4,38	-3,51	-4,75	0,87	-4,23	-3,44	-4,70	0,86	1,78	-0,09	-0,24	0,85	1,82	-0,12	-0,24	0,84
ftv33	-1,11	-0,97	-2,85	2,35	-1,22	-0,98	-2,79	2,35	-0,87	-0,80	-2,73	2,35	0,07	-0,08	-0,11	2,37	0,11	0,01	-0,34	2,35
ftv35	-0,98	-0,90	-1,80	2,78	-1,11	-0,97	-1,87	2,81	-1,09	-0,95	-1,86	2,89	0,27	-0,08	-0,18	2,81	0,33	-0,11	-0,15	2,81
ftv38	-0,86	-0,76	-1,81	4,26	-0,76	-0,72	-1,76	4,22	-0,98	-0,74	-1,61	4,18	-0,58	-0,06	-0,15	4,31	-0,60	-0,04	-0,15	4,31
p43	0,59	-0,66	-1,80	4,13	0,61	-0,67	-1,86	3,97	0,61	-0,61	-1,84	4,13	0,43	-0,04	-0,09	3,97	0,50	-0,07	-0,15	4,09
ftv44	-1,51	-1,15	-1,99	5,27	-1,37	-1,13	-1,83	5,17	-1,46	-1,21	-2,17	5,17	0,68	-0,14	-0,40	5,22	0,72	-0,13	-0,38	5,38
ftv47	-1,96	-1,92	-2,87	5,45	-1,85	-1,79	-2,71	5,24	-1,99	-1,91	-2,83	5,40	0,69	-0,21	-0,55	5,40	0,75	-0,15	-0,54	5,29
ry48p	-1,01	-0,69	-1,41	6,06	-0,86	-0,62	-1,37	5,89	-0,70	-0,51	-1,28	5,89	0,55	-0,23	-0,45	6,00	0,54	-0,17	-0,35	5,89
ft53	-0,20	-0,40	-1,74	9,30	-0,16	-0,48	-1,82	9,21	-0,17	-0,55	-1,91	9,21	1,03	-0,26	-0,50	9,30	1,08	-0,26	-0,45	9,12
ftv55	-1,55	-1,52	-2,45	9,45	-1,43	-1,51	-2,49	9,63	-1,58	-1,47	-1,42	9,45	0,36	-0,11	-0,28	9,45	0,36	-0,14	-0,26	9,54
ftv64	-1,32	-1,22	-2,13	16,97	-1,53	-1,45	-2,38	16,81	-1,38	-1,24	-2,14	16,48	0,53	-0,16	-0,33	16,48	0,63	-0,16	-0,21	16,81
ftv70	-1,53	-1,23	-2,01	18,81	-1,57	-1,24	-2,03	18,81	-1,27	-1,15	-2,07	18,27	0,38	-0,15	-0,27	18,63	0,48	-0,16	-0,27	18,81
ft70	-0,60	-1,12	-2,89	12,76	-0,51	-1,17	-1,91	12,39	-0,57	-1,06	-1,84	12,51	-1,11	-0,23	-0,46	12,64	-1,05	-0,19	-0,44	12,64
ftv90	-1,53	-1,38	-2,28	38,28	-1,64	-1,47	-2,33	37,55	-1,74	-1,50	-2,33	37,91	0,55	-0,23	-0,48	37,91	0,56	-0,18	-0,42	37,91
ftv100	-1,71	-1,55	-2,48	49,15	-1,75	-1,51	-2,30	49,15	-1,78	-1,57	-2,43	47,28	0,18	-0,26	-0,60	48,68	0,18	-0,23	-0,54	48,21
ftv110	-1,42	-1,36	-2,35	63,74	-1,49	-1,39	-2,39	61,90	-1,49	-1,47	-2,58	63,74	0,74	-0,07	-0,21	61,90	0,78	-0,02	-0,12	61,90
ftv120	-1,58	-1,49	-2,41	82,95	-1,64	-1,58	-2,53	84,59	-1,45	-1,45	-2,38	83,77	1,10	-0,20	-0,30	86,24	1,18	-0,12	-0,23	82,95
kro124	-1,56	-1,44	-2,40	37,07	-1,46	-1,49	-2,47	37,07	-1,70	-1,55	-2,48	37,43	0,76	-0,08	-0,14	37,79	0,77	-0,03	-0,04	37,07
ftv130	-1,93	-1,81	-2,79	108,99	-2,01	-1,85	-2,81	105,88	-1,98	-1,83	-2,83	104,84	0,76	-0,20	-0,30	106,91	0,84	-0,23	-0,19	107,95
ftv140	-2,09	-1,81	-2,64	131,07	-2,08	-1,97	-2,86	131,07	-2,23	-1,95	-2,73	132,33	0,38	-0,08	-0,20	129,81	0,41	-0,05	-0,14	127,29
ftv150	-1,85	-1,86	-2,79	162,28	-1,91	-1,87	-2,82	157,60	-1,82	-1,81	-2,75	162,28	1,17	-0,21	-0,35	163,84	1,26	-0,19	-0,23	163,84
ftv160	-2,31	-2,24	-3,12	177,68	-2,23	-2,26	-3,22	175,98	-2,45	-2,41	-3,32	175,98	-3,41	-0,19	-0,41	174,27	-3,43	-0,22	-0,34	177,68
ftv170	-2,28	-2,21	-3,11	216,34	-2,28	-2,18	-3,05	220,54	-2,34	-2,21	-3,13	212,14	-3,13	-0,33	-0,64	218,44	-3,05	-0,29	-0,52	218,44
rbg323	-1,23	-0,96	-1,80	769,05	-1,42	-0,98	-1,68	791,90	-1,58	-1,14	-1,86	769,05	-2,49	-0,28	-0,50	791,90	-2,45	-0,31	-0,38	776,67
rbg358	0,58	-0,94	-2,12	867,33	0,56	-0,95	-2,16	867,33	0,54	-0,91	-2,12	850,65	-0,30	-0,55	-1,09	875,67	-0,23	-0,48	-0,98	850,65
rbg403	0,34	-0,66	-1,94	1196,25	0,36	-0,67	-1,78	1196,25	0,35	-0,83	-1,95	1173,02	0,20	-0,06	-0,15	1196,25	0,28	-0,05	-0,09	1173,02
rbg443	0,27	-1,63	-3,83	1779,70	0,34	-1,67	-3,24	1745,14	0,30	-1,64	-3,79	1796,97	0,00	-0,19	-0,53	1779,70	0,07	-0,10	-0,48	1762,42