

Estratégias relax-and-fix na solução de um problema de roteamento periódico em arcos capacitados (PCARP)

Jailson Domingos de Oliveira (UFPR) jailson-de-oliveira@hotmail.com
Cassius Tadeu Scarpins (UFPR) cassiusts@gmail.com

Resumo:

Nesse trabalho, aplicou-se uma estratégia baseada na heurística relax-and-fix como método de solução para o Problema de Roteamento Periódico em Arcos Capacitados (PCARP- *Periodic Capacitated Arc Routing Problem*). Considerou-se o caso especial em que os veículos não têm a necessidade de voltar ao depósito no final de um período e, ainda, têm a possibilidade de folgar em qualquer dia do horizonte de tempo. O PCARP é um problema pouco explorado na literatura. Configura-se como um problema *NP-hard*, sendo comumente aplicado em coleta de resíduos urbano, inspeção de linhas de força, despejo de sal em vias com neves, entrega de correspondência entre outros. Desenvolveu-se 4 estratégias diferentes para heurística relax-and-fix e avaliando-se seus desempenhos. Os testes computacionais realizados indicaram que as estratégias propostas para heurística são rápidas na determinação de soluções iniciais para o problema. Destaca-se que das 23 instâncias testadas em nenhum caso se esgotou a memória do computador, fato que ocorre com frequência na tentativa de resolver o problema por métodos exatos.

Palavras chave: Relax-and-fix, Problema de Roteamento Periódico em Arcos Capacitados, Heurística.

Relax-and-fix strategies in solving the periodic capacitated arc routing problem

Abstract

In this paper, applies a strategy based on relax-and-fix heuristic as solving method for the Periodic and Capacitated Arc Routing Problem (PCARP). Consider the special case where the vehicles do not have the need to return to the depot at the end of the period and still have the possibility to have a day off at any day in time horizon. PCARP is a NP-hard problem underexplored in the literature, and is commonly applied to the collection of waste collection, power lines inspection, pouring salt on snowy roads, mail delivery, and so on. Developed for different strategies for the relax-and-fix heuristic and evaluating their performances. Computational tests indicated that the proposed strategies for the heuristic are quick to determine the initial solutions to the problem. Regarding this paper it can be highlighted that of the twenty-three instances tested in any case the computer memory has elapsed, which occurs frequently when trying to solve the problem by means of exact methods.

Key-words: Relax-and-fix, Periodic Capacitated Arc Routing Problems, heuristic.

1. Introdução

A literatura traz estudos em três grandes problemas logísticos: gerenciamento de armazéns, localização de facilidades e roteamento de veículos. Nesse último, em especial, a literatura

sugere dois objetivos explorados: a redução do número de veículos utilizados e/ou a redução da distância total percorrida (DOULABI; SEIFE, 2013). O Problema de Roteamento de Veículos (PRV) ou *Vehicle Routing Problems* (VRP) é considerado um problema clássico da literatura (DANTZIG; RAMSER, 1959). Para Franco e Oliveira (2012) entende-se por roteamento de veículos, um conjunto de problemas que tem por objetivo determinar as melhores rotas para uma frota de veículos atenderem um conjunto de usuários finais (arcos ou nós).

Problemas de Roteamento de Veículos tem recebido a atenção de muitos pesquisadores no decorrer dos anos, seja com o surgimento de novos modelos e abordagens, ou pelo número elevado de aplicações práticas comparados a outros problemas clássicos. Destaca-se: distribuições de jornais, alimentos, bebidas, produtos diversos, transporte escolar, recolhimento de lixo, entrega de correspondência, serviço de emergência entre outros (GOLDBARG, 2005).

Para Goldbarg (2005) os problemas de roteamento de veículos estão entre os mais complexos da área de otimização combinatória. Isto deve-se pelo grande número de variáveis, diversidade de restrição e objetivos, pertencendo a classe de problemas *NP-Hard* em que, não há algoritmos em tempo polinomial para encontrar soluções ótimas.

Segundo Goldbarg (2005) problemas de roteamento de veículos podem ser classificados em duas classes: Problema de Roteamento em Nós (associado aos ciclos Hamiltonianos) e Problemas de Roteamento em Arcos (associados aos ciclos Eulerianos). Problemas de roteamento em nós consistem em determinar uma ou mais rotas que passe por todos os nós ou vértices de um grafo enquanto que o roteamento em arcos tem por objetivo determinar uma ou mais rotas que atenda todos os arcos ou arestas de um grafo (MONROY; AMAYA; LANGEVIN, 2013).

Nesse trabalho, considerou-se o problema de roteamento em arcos. Mais especificamente, o Problema de Roteamento Periódico em Arcos Capacitados (PCARP), baseando-se na formulação matemática proposta por Batista (2013). Como abordagem de solução, foi proposto uma heurística do tipo *relax-and-fix* descrita em Wolsey (1998) que decompõem o modelo original em submodelos que podem ser resolvidos de modo exato, sem, entretanto, garantir o ótimo do problema original. Além de propor diferentes estratégias para heurística, esse trabalho procurou contribuir com a literatura verificando a viabilidade e desempenho da heurística quando aplicada no PCARP.

2. Problemas de Roteamento em Arcos

O Problema de Roteamento em Arcos (*Arc Routing Problems* - ARP) é um dos mais antigos problemas relacionados a grafos, sendo a primeira referência advinda do problema das pontes de Königsberg (1736) resolvido pelo famoso matemático suíço Euler. O termo “percurso em arco” foi estabelecido mais claramente em 1995 por Eiselt *et al.* (1995).

O problema de roteamento em arcos, refere-se a problemas de roteamento onde a atividade principal é cobrir os arcos e suas demandas. Enquanto que problemas de roteamento em nós, a atividade é servi-los e os arcos são de interesse apenas como caminhos que conectam os nós (EISEL *et al.*, 1995i). Segundo Eisel *et al.* (1995i) o principal objetivo dos ARPs é determinar o menor custo para atravessar um conjunto de arcos de um grafo, com ou sem restrições.

De acordo com Corberán e Prins (2010) os ARPs tem sido menos estudados do que os problemas de roteamento em nós, mas, tem ganhado destaque nas últimas décadas e desenvolvimentos impressionantes foram alcançados. Existem um grande número de situações reais que podem ser modelados por esse tipo de problema tais como: a coleta de lixo urbano, a inspeção de linhas de forças, o despejo de sal em vias com neves, entrega de correspondências, entre outros (CORBERÁN; PRINS, 2010).

Diversos são os problemas que se encaixam na classificação ARPs, entre os quais, destaca-se: o problema das pontes de Königsberg, Problema do Carteiro Chinês, o Problema do Carteiro Rural, Problema de Roteamento em Arcos Capacitados e Problema de Roteamento Periódico em Arcos Capacitados.

2.1 Problemas de Roteamento Periódico em Arcos Capacitados

O Problema de Roteamento em Arcos Capacitado (*Capacitated Arc Routing Problem* - CARP), foi proposto por Golden e Wong (1981) e envolve a “capacidade” dos veículos. Essa capacidade, muitas vezes pode estar associada à carga máxima ou a um tempo máximo de trabalho ou, também, a uma distância percorrida. Os ARPs capacitados são mais difíceis de serem resolvidos e seu estudo é mais recente (CORBERÁN; PRINS, 2010).

Como o CARP envolve restrições operacionais que limitam a capacidade de atendimento, sua resolução de maneira exata é considerada difícil. Golden e Wong (1981) apresentam a primeira formulação matemática para o CARP propondo um algoritmo para solucionar o modelo e demonstrar que o problema é *NP-hard*. Devido sua complexidade, diversos métodos heurísticos têm sido desenvolvidos para a resolução do CARP. Eiselt *et al.* (1995ii) classificam esses métodos em três categorias: 1) os construtivos simples; (2) os construtivos de duas fases; e 3) os de melhoria.

O CARP é frequentemente utilizado para resolver problemas de coleta de resíduos urbanos que necessitam de tomadas de decisões de níveis operacionais, onde rotas são definidas num prazo curto, diariamente, em geral. Todavia, em alguns problemas é necessário tomadas de decisões de nível tático, com horizontes de tempo maior e que ainda envolvam restrições de periodicidades. Nesse caso, destaca-se o Problema de Roteamento Periódico em Arcos Capacitados (BATISTA; SCARPIN, 2015).

O Problema de Roteamento Periódico em Arcos Capacitados (*Periodic Capacitated Arc Routing Problem* - PCARP) é uma extensão do CARP. Ao invés do problema ser resolvido para tomada de decisões de nível operacional são utilizados para decisões de nível tático, para horizontes de tempo maiores do que um período e com restrições de frequência (LACOMME; PRINS; RAMDANE-CHÉRIF, 2002i).

Em muitos problemas, alguns arcos possuem uma necessidade de serem atendidos com maior frequência. Por exemplo, no caso da coleta de resíduos urbanos que em grandes centros a produção é maior do que em bairros. Logo, essas regiões demandam uma frequência maior de coleta, em que se faz necessário um planejamento baseado em múltiplos períodos. Portanto, resolver PCARP implica a determinação simultânea de decisões táticas: determinando para cada rua um número de tratamento de acordo com sua frequência; e operacionais: determinando uma rota, de forma que, para cada período, as ruas sejam atendidas. O custo total depende da combinação das decisões de designação e roteamento (LACOMME; PRINS; RAMDANE-CHÉRIF, 2005).

Um dos primeiros modelos matemáticos foram propostos por Chu, Labadi e Prins (2004) que utilizaram *softwares* comerciais para resolução do problema e conseguiram soluções apenas para instâncias pequenas com 5 períodos e 10 arestas. Chu, Labadi e Prins (2006) descrevem o PCARP como extensão do CARP e aplicam dois algoritmos para resolução. O primeiro é uma heurística gulosa e o segundo é o algoritmo *Scatter Search* (SS) baseado em busca local, utilizados para avaliar dois conjuntos de casos.

Lacome, Prins e Ramdame-Chérif (2005) afirmam que pesquisas sobre o PCARP são ainda embrionários e muito recentes. Utilizaram um Algoritmo Genético híbrido cujos objetivos eram

minimizar tanto a frota de veículos quanto os custos das viagens (LACOMME, PRINS e RAMDAME-CHERÉF, 2002ii).

Kansou e Yassine (2009) trabalharam com o PCARP de forma que cada serviço ocorresse de acordo com uma combinação de dias que satisfizessem as frequências, nomeado Problema de Roteamento Periódico Misto em Arcos Capacitados. Para resolução combinaram uma heurística de inserção e a meta-heurística de Colônia de Formigas.

Batista (2014) apresentou um modelo com características diferenciadas das já propostas na literatura, como: não necessitar voltar a um depósito ao final de um dia e a possibilidade de um veículo folgar em um determinado dia. Seu modelo foi aplicado na manutenção preditiva e preventiva de linhas férreas. Utilizou um *software* comercial para resolver, conseguindo chegar na solução ótima em apenas uma das 23 instâncias proposta pelo autor.

O PCARP é um problema *NP-hard* uma vez que inclui CARP como caso particular e, quanto mais restrições, mais difícil é encontrar uma solução (LACOMME; PRINS; CHÉRIF, 2002). Logo, métodos exatos para sua solução são, até o momento na literatura, ineficientes quando trabalhado com problemas reais devido sua complexidade. O que leva ao desenvolvimento e proposta de novos métodos para determinar soluções viáveis.

3. Formulação Matemática

O modelo proposto em Batista (2014), parte de um grafo não direcionado $G = (X, E)$ com n nós, $X = \{x_1, x_2, \dots, x_n\}$, e m arestas com $E = \{e_1, e_2, e_3, \dots, e_m\}$ que devem ser percorridos por nk carros definidos pelo conjunto $K = \{1, 2, \dots, nk\}$. Cada aresta e é formada por um par de nós chamado de $x_{ij} = (x_i, x_j)$, os quais são associado a um custo c_{ij} .

Batista (2014) considerou a capacidade dos veículos usados, como sendo a máxima distância que estes podem se deslocar, ou seja, 1 aresta por dia.

O horizonte de tempo H é formado por np períodos $H = \{1, 2, \dots, np\}$, onde cada período é representado por p . A demanda nesse modelo é considerado a periodicidade de cada aresta que é expressa na quantidade máxima de períodos em que uma aresta deve ser atendido ao menos uma vez $MP(x_{ij})$.

O modelo é baseado em Programação Linear Binária, com três variáveis descritas abaixo:

$$\begin{aligned}
 x_{ijkp} & \begin{cases} 1, & \text{se o carro } k \text{ se desloca do ponto } i \text{ para o } j \text{ no período } p \\ 0, & \text{caso contrário} \end{cases} \\
 p_{ijp} & \begin{cases} 1, & \text{se o arco } x_{ij} \text{ não respeita a periodicidade no período } p \\ 0, & \text{caso contrário} \end{cases} \\
 f_{ikp} & \begin{cases} 1, & \text{se o carro } k \text{ fica parado no ponto } i \text{ no dia } p \\ 0, & \text{caso contrário} \end{cases}
 \end{aligned}$$

As variáveis p_{ijp} e f_{ikp} são propostas para que o modelo tenha viabilidade. Cada variável p_{ijp} está associada a uma punição PU_{ij} , caso uma aresta não tenha sua periodicidade atendida o que permite que haja atrasos no atendimento de algumas aresta.

Já a variável f_{ikp} permite que um carro folgue em um determinado dia, permitindo em algumas vezes obter uma solução com menor quantidade de deslocamentos.

Modelo:

$$\min Z = \sum_{[i,j] \in E} \sum_{k=1}^{nk} \sum_{p=1}^{np} c_{ij} * x_{ijkp} + \sum_{[i,j] \in E} \sum_{p=1}^{np} PU_{ij} * p_{ijp} \quad (1)$$

Sujeito à:

$$\sum_{[i,j] \in E} x_{ijkp} + f_{jkp} - \sum_{[i,j] \in E} x_{jik,p+1} - f_{jk,p+1} = 0 \quad \forall j \in X, \forall k \in K, \forall p \in H \quad (2)$$

$$\sum_{[i,j] \in E} x_{ijkp} + \sum_{[i,j] \in E} x_{jikp} + \sum_{i=1}^n f_{ikp} = 1 \quad \forall p \in H, \forall k \in K \quad (3)$$

$$\sum_{k=1}^{nk} \left(x_{ijkp} + x_{jikp} + x_{ijk,p+1} + x_{jik,p+1} + \dots \right) + p_{ijp} \geq 1 \quad \forall [i,j] \in E, \forall p \in H \quad (4)$$

$$x_{ijkp}, p_{ijp}, f_{ikp} \in \{0,1\} \quad \forall [i,j] \in E, \forall k \in K, \forall p \in H \quad (5)$$

O objetivo do modelo (1) é a minimização dos custos de deslocamentos e atrasos. A restrição (2) garante o fluxo diário dos carros permitindo folgar. Pode ser observado nessa restrição que se um carro chega em um ponto em um determinado dia somando a possibilidade do veículo já estar no ponto, deve ser igual ao movimento no dia seguinte ou permanência no mesmo. A restrição (3) garante que todos os carros terão uma designação para cada dia p , isto é, que cada veículo no dia p pode se deslocar de i para j , ou de j para i , ou ainda permanecer parado em um ponto i . A restrição (4) se refere à periodicidade que deve ser atendida em cada aresta. Observa-se que se as variáveis p_{ijp} assumam valor 1, indica que a aresta e_{ij} teve sua periodicidade atrasada em um dia implicando em uma punição PU_{ij} na função objetivo. Por fim, a restrição (5) determina que todas as variáveis do modelo sejam binárias.

4. Heurística Relax-and-fix

A heurística *relax-and-fix* tem sido aplicada nas últimas décadas para solucionar problemas de otimização combinatória, em especial, problemas de dimensionamento de lotes como mostram os trabalhos desenvolvidos por Dastidar e Nagi (2004), Ferreira *et al* (2009) e Ferreira *et al* (2010). A heurística foi desenvolvida por Wolsey (1998) e consiste em um método de decomposição de um modelo de programação inteira mista em submodelos menores disjuntos, que podem ser resolvidos rapidamente, porém sem a garantia de resolução do problema original de forma ótima.

A heurística baseia-se na partição das variáveis inteiras ou binárias do modelo criando n conjuntos distintos, $P_i, i = 1, \dots, n$. Observa-se que o conjunto de variáveis originais são divididos em três grupos: um contendo variáveis inteiras e binárias, um segundo contendo as variáveis relaxadas, e o último com variáveis fixadas. O número n indica o número de iterações da heurística.

Em uma iteração i , as variáveis do conjunto P_i são definidas como inteiras e binárias, as variáveis $(1, \dots, i-1)$ são fixadas e as demais variáveis $(i+1, \dots, n)$ são relaxadas. O submodelo resultante é resolvido. Caso o submodelo seja infactível, então se para a metodologia, pois não pode ser encontrado uma solução viável para o modelo original com as variáveis $(1, \dots, i-1)$ fixadas. Caso contrário, as variáveis do conjunto P_i são fixadas, repete-se o processo para os demais conjuntos até que todas as variáveis $(i, \dots, n-1)$ sejam todas

fixadas e as variáveis da partição P_n sejam inteiras ou binárias. Assim, obtém-se uma solução inicial para o modelo original.

Segundo a literatura a heurística *relax-and-fix* tem demonstrado bom desempenho para determinar soluções iniciais para modelos de dimensionamento de lotes, esses resultados são posteriormente melhorados com a aplicação de uma heurística de melhoramento, como a heurística *fix-and-optimize*, estratégias utilizadas pelos autores Dastidar e Nagi (2005) e Ferreira *et al.* (2009).

Os processos de escolha das partições das variáveis do problema afetam o desempenho da heurística, uma vez que seu objetivo é resolver submodelos inteiros mistos menores (MORAES; SANTOS, 2012).

Nesse trabalho foram implementadas quatro estratégias diferentes para a heurística *relax-and-fix* e aplicadas ao PCARP proposto por Batista (2014). Nas três primeiras estratégias foram realizadas decomposição por período, na quarta foi proposto uma decomposição por adjacência de nós. Descreve-se cada estratégia a seguir:

Relax-and-Fix Forward (RFF) - Nessa estratégia o problema original foi dividido em P subproblemas a serem resolvidos de modo que cada subproblema corresponda a um dia p do horizonte de tempo. A ordem da resolução é a -cronológica, isto é, iniciando a resolução no subproblema correspondente ao primeiro período do horizonte de tempo e finalizando no subproblema correspondente ao último período.

A figura 1 (a) ilustra o funcionamento dessa estratégia. Para o primeiro subproblema, primeira iteração da heurística, as variáveis binárias referentes a $p = 1$ foram mantidas e as variáveis para os demais períodos foram relaxadas (valores no intervalo $[0,1]$). Após a resolução do primeiro subproblema, as variáveis binárias do primeiro período são fixadas, tornam-se parâmetros, as variáveis do segundo período ($p = 2$) são mantidas binárias, e as demais dos período ($3, \dots, p$) são relaxadas. O processo se repete até que o último subproblema, seja resolvido.

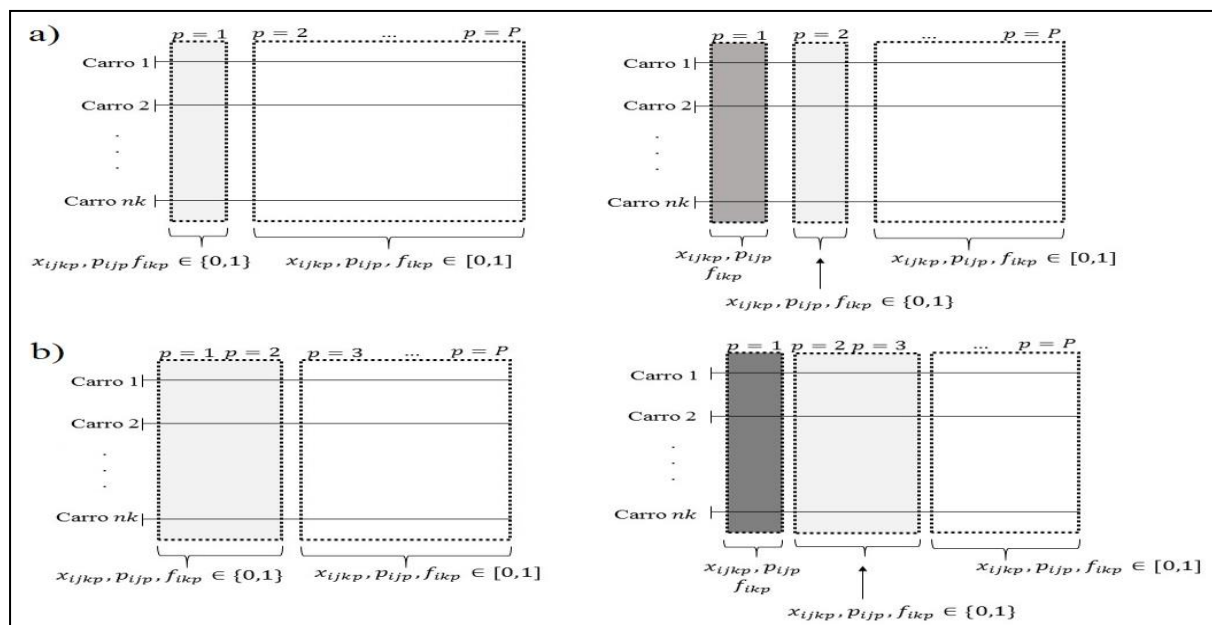


Figura 1 – Representação da heurística *Relax-and-fix*

Relax-and-fix com Overlapping 1 (RFO1) - Essa estratégia, se baseia na partição do problema original em P períodos com sobreposição de partições. Seja uma partição p as variáveis

correspondente aos subproblemas p e $p - 1$ são mantidas binárias e as demais variáveis subsequentes a esses períodos são relaxadas. Na próxima iteração, as variáveis do período $p - 1$ são fixadas, as variáveis dos períodos p e $p + 1$ mantidas binárias e as demais variáveis são relaxadas. O processo ocorre até que o último subproblema seja resolvido.

A figura 1 (b) ilustra essa estratégia. Para o primeiro subproblema temos as variáveis correspondente ao primeiro e segundo período ($p = 1$ e $p = 2$) do horizonte de tempo, mantidas binárias e para os demais períodos foram relaxadas. Após a resolução desse primeiro subproblema, as variáveis binárias do primeiro período são fixadas (tornando-se parâmetro), as variáveis referentes ao segundo e terceiro período ($p = 2$ e $p = 3$) são mantidas binárias, e as demais variáveis são relaxadas. O processo ocorre da mesma forma até a resolução do último subproblema correspondente ao último período.

Relax-and-fix com Overlapping 2 (RFO2) – Essa estratégia, baseia-se na sobreposição de partições, semelhante a estratégia anterior RFO1. Nessa proposta cada subproblema é resolvido mantendo as variáveis binárias correspondentes aos períodos $p - 3$, $p - 2$, $p - 1$ e p , sendo as demais variáveis dos períodos subsequentes relaxadas. No próximo subproblema, as variáveis correspondentes aos períodos $p - 3$ e $p - 2$ são fixadas, as variáveis dos períodos $p - 1$, p , $p + 1$ e $p + 2$ são mantidas binárias e as demais variáveis subsequentes relaxadas, assim sucessivamente até o último subproblema ser resolvido.

Relax-and-fix nós Adjacentes (RFNA) – Nessa estratégia as partições foram baseadas em adjacências dos nós. Escolhe-se um nó para ser o nó raiz. A relaxação ocorre a partir desse nó, isto é, criamos um subconjunto que é formado pelo nó raiz e seus respectivos nós adjacentes, todas as variáveis referentes a esse conjunto de nós são mantidas binárias e as demais variáveis são relaxadas. Na segunda iteração as variáveis correspondentes ao nó raiz e seus adjacentes são fixadas as variáveis correspondentes aos nós adjacentes dos adjacentes do nó raiz são mantidas binárias e as demais relaxadas. Finalizando o processo quando todas as variáveis referentes aos nós do grafo G sejam fixas ou binárias.

A figura 2 descreve o funcionamento dessa estratégia, consideremos um grafo $G = (X, A)$ formado por 8 nós e 10 arestas. Inicia-se, neste exemplo, pelo nó raiz 6, para esse primeiro subproblema todas as variáveis correspondentes aos nós adjacentes a 6 são mantidas binárias e as demais relaxadas. Na segunda iteração, as variáveis referentes aos nós adjacentes de 6 são fixadas, as adjacentes ao nó 1 são mantidas binárias e as demais relaxadas. Na terceira e última iteração as variáveis referentes aos nós adjacentes a 1 e 6 são fixadas e as adjacentes aos nós 2, 4 e 5 são binárias.

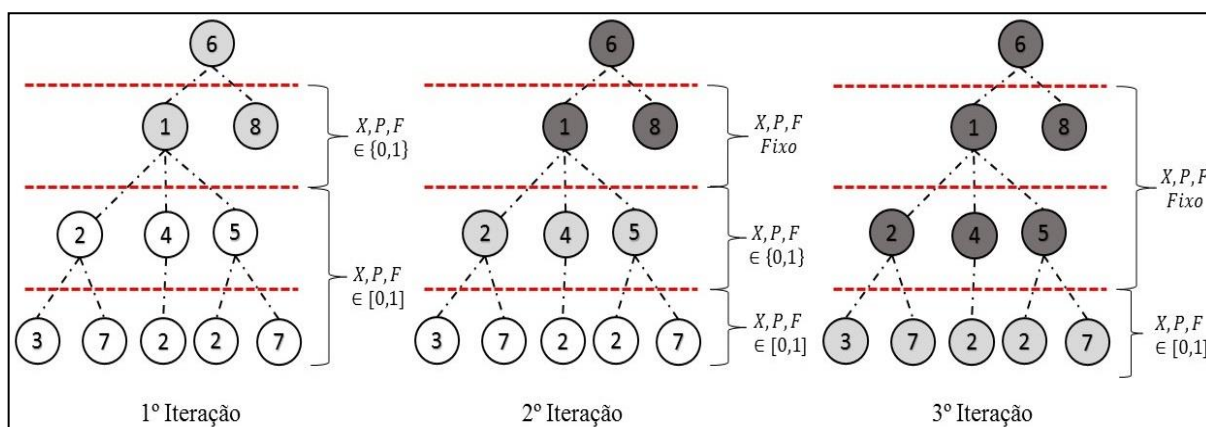


Figura 2 – Representação da heurística Relax-and-fix RFNA

5. Experimentos Computacionais

As estratégias propostas para a heurística *relax-and-fix* para a resolução do PCARP, foram implementadas no Microsoft Visual Studio 2015 utilizando pacote de otimização IBM ILOG CPLEX 12.4. Os testes foram realizados utilizando um computador com processador Intel Core i7-5500U com velocidade de 2.4GHz, memória de RAM de 8GB sob o sistema operacional Windows 10.

Foram utilizadas 23 instâncias proposta no trabalho de Batista (2014), para verificar o desempenho das estratégias sugeridas nesse trabalho. As características para cada uma das instâncias, como número de nós, número arestas, quantidade de carros e o horizonte de tempo (Dias) estão dispostos na Tabela 1. Percebe-se que as instâncias propostas dependendo do número de dias, carros e arestas tornam os problemas complexos de serem resolvidos, fato esperado devido a natureza do problema *NP-hard*.

Problema	Nós	Arestas	Período	Carros	Neces.	Dias
gdb1	12	22	2	1	32	38
gdb2	12	26	3	1	44	53
gdb3	12	22	2	2	31	19
gdb4	11	19	3	2	34	21
gdb5	13	26	4	2	53	32
gdb6	12	22	2	3	32	14
gdb7	12	22	3	3	37	45
gdb8	27	46	4	3	67	27
gdb9	27	51	3	4	78	24
gdb10	12	25	4	4	70	21
gdb11	22	45	2	1	50	60
gdb12	13	23	3	1	33	40
gdb13	10	28	2	2	33	20
gdb14	7	21	3	2	32	20
gdb15	7	21	4	2	48	29
gdb16	8	28	2	3	40	24
gdb17	8	28	3	3	52	21
gdb18	9	36	4	3	81	33
gdb19	8	11	3	4	29	12
gdb20	11	22	4	4	53	16
gdb21	11	33	2	5	46	12
gdb22	11	44	4	5	88	22
gdb23	11	55	2	6	82	17

Fonte: Batista (2014)

Tabela 1- Características dos 23 problemas

A Tabela 2 mostra o valor da função objetivo e o tempo gasto em segundos na execução de cada heurística proposta, para cada uma das 23 instâncias. Foi estipulado um tempo limite de 1 hora ou 3600 segundos para as três primeiras heurísticas propostas (RFF, RFO1, RFO2) e de 300 segundos por iteração para a quarta heurística proposta (RFNA). Os símbolos “--” indicam que a estratégia gerou subproblemas infactíveis, enquanto que os símbolos “*” que a heurística utilizou o tempo limite e não chegou em uma solução.

A última coluna da tabela refere-se aos valores da função objetivo de cada instância do trabalho de Batista (2014), os quais foram resolvidas pelo *software* CPLEX 12.4 utilizando a configuração default do solver, com um tempo limite para obtenção de uma solução de 24 horas ou 86400 segundos. Os símbolos “***” indicam que os problemas esgotaram a memória do

computador antes de chegar a uma solução ótima ou atingir o tempo limite.

A estratégia RFF apresentou o pior desempenho em relação as demais, das 23 instâncias 19 dessas foram infactíveis. A infactibilidade ocorreu com grande frequência nas três primeiras estratégias propostas. Esse fato é devido ao funcionamento da heurística *relax-and-fix*, pois o modelo exige que os veículos retornem ao nó onde iniciaram a rota, em muitos casos um determinado veículo encontra-se em um nó muito longe do nó onde iniciou sua rota, não havendo dias suficientes no horizonte de tempo para seu retorno ao nó inicial, o que torna infactível o problema.

	RFF		RFO1		RFO2		RFNA		EXATO (Batista, 2014)	
<i>Problema</i>	<i>Solu.</i>	<i>Tempo</i>	<i>Solu.</i>	<i>Tempo</i>	<i>Solu.</i>	<i>Tempo</i>	<i>Solu</i>	<i>Temp</i>	<i>Solu.</i>	<i>Tempo</i>
gdb1	--	--	--	--	--	--	53426	358	4450	39000**
gdb2	--	--	--	--	--	--	7620	360	11622	70423**
gdb3	--	--	--	--	--	--	344	349	340	36906**
gdb4	--	--	--	--	14503	105	10479	387	3555	49595**
gdb5	--	--	--	--	--	--	22699	657	1750	86400*
gdb6	--	--	4478	260	459	180	456	109	456	86400*
gdb7	--	--	--	--	15520	270	3550	649	518	55767**
gdb8	--	--	--	--	--	--	--	--	12395	86400*
gdb9		*		*		*	--	--	13454	86400*
gdb10		*		*		*	--	--	734	86400*
gdb11	--	--	--	--	--	--	--	--	64437	31510**
gdb12	--	--	--	--	--	--	12558	129	536	86400*
gdb13	--	--	620	111	620	111	620	51	620	86400*
gdb14	--	--	--	--	145	105	145	31	145	261
gdb15	--	--	--	--	3144	1681	142	156	140	63135**
gdb16	--	--	185	182	184	145	182	175	182	35301**
gdb17	--	--	3180	305	--	--	168	174	168	77693**
gdb18	--	--	--	--		*	404	737	415	86400*
gdb19	2135	122	129	145	135	125	125	53	125	47566**
gdb20	--	--	--	--	6258	137	--	--	2280	67903**
gdb21	--	--	--	--	212	122	212	123	212	38181**
gdb22		*		*		*	470	746	433	86400*
gdb23	--	--	--	--	336	1432	345	423	336	86400*

Fonte: Autor

Tabela 2 - Resultados das estratégias propostas para heurísticas *relax-and-fix*

A estratégia RFO1 apresentou desempenho superior a primeira estratégia. Conseguiu resolver 5 das 23 instâncias, chegando ao mesmo valor para a função objetivo que Batista (2014) para o problema gdb13, com um tempo inferior, enquanto que o método exato levou 24 horas a heurística levou 111 segundos (1 minuto e 51 segundos).

A estratégia RFO2, obteve melhor desempenho que RFO1 e RFF, chegou em uma solução viável para 11 dos 23 problemas, sendo que para as instâncias gdb13, gdb14, gdb21 e gdb23 obteve o mesmo valor que o método exato executado por 24 horas para a função objetivo, em um tempo de execução muito inferior. É possível observar ainda que para as instâncias gdb6, gdb16 e gdb19 a heurística teve bom desempenho chegando em um valor para função objetivo próxima do método exato e, novamente, com um tempo muito inferior a esse. Esses valores podem posteriormente ser melhorados com a aplicação de uma heurística de melhoria como,

por exemplo, a heurística *fix-and-optimize*.

A última estratégia proposta nesse trabalho RFNA, obteve os melhores desempenho. Alcançou uma solução viável para 18 das 23 instâncias propostas em Batista (2014). A estratégia obteve os mesmos valores que o método exato para os problemas gdb6, gdb13, gdb14, gdb16, gdb17, gdb19 e gdb21, com um tempo de execução inferior. Para os problemas gdb2 e gdb18 teve desempenho superior, encontrando soluções com valores menores que o exato executado por 24 horas, e com tempos de execução menores.

Para as instâncias gdb3, gdb15, gdb22 e gdb23 a estratégia RFNA, obteve soluções próximas do método exato e com tempos muito inferior a esse, destacamos a possibilidade da utilização de uma heurística de melhoria para esses casos.

Pode-se destacar que, para as quatro estratégias propostas, em nenhum caso esgotou-se a memória do computador, algo que ocorreu com frequência na execução do método exato.

6. Considerações Finais

Nesse trabalho estudou-se o Problema de Roteamento Periódico em Arcos Capacitados (PCARP). Utilizou-se a heurística *relax-and-fix* como método de solução para o modelo proposto por Batista (2014). Foram desenvolvidas quatro estratégias de decomposição das variáveis binárias do problema e realizado testes com 23 instâncias propostas na literatura.

A heurística *relax-and-fix* mostrou-se uma boa estratégia para determinar uma solução inicial para o modelo PCARP estudado nesse trabalho. Houve ganho em tempo computacional e em nenhum caso houve estouro de memória do computador. Em alguns casos, os valores da função objetivo podem ficar distantes da solução ótima, o que pode ser futuramente melhorada com uma heurística de melhoria.

De acordo com os testes realizados, verificou-se que a melhor heurística proposta foi a *Relax-and-Fix Nós Adjacentes*, que apresentou maior número de soluções quando comparada com as demais. Esta obteve solução viável para 18 instâncias das 23 propostas, sendo que, para duas dessas obteve soluções melhores que o método exato executado por 24 horas e interrompido por este tempo.

Futuramente, pode-se implementar novas estratégias para heurística *relax-and-fix*, baseando-se em outros particionamento do modelo e implementar uma heurística de melhoria do tipo *fix-and-optimize*, com o objetivo de melhorar as soluções encontradas pelo heurística *relax-and-fix*.

Referências

BATISTA, G. V. *Proposta de Um Modelo Matemático para o Problema de Roteamento em Arcos Capacitado e Periódico*. Dissertação de Mestrado. UFPR. 2014. Disponível em: <<http://dspace.c3sl.ufpr.br/dspace/bitstream/handle/1884/35767/R%20-%20D%20%20GUILHERME%20VINICYUS%20BATISTA.pdf?sequence=1>>.

BATISTA, G. V.; SCARPIN, C. T. *Problema de Roteamento em Arcos Capacitado e Periódico Aplicado a um Contexto Real*. Revista Produção Online. Vol. 15, n. 3, p. 1080-1098, 2015.

CHU, F.; LABADI, N.; PRINS, C. *The Periodic Capacitated Arc Routing Problem Linear Programming Model, Metaheuristic and Lower Bounds*. Journal of Systems Science and Systems Engineering, v. 13, n. 4, p. 423–435, 2004.

CHU, F.; LABADI, N.; PRINS, C. *A Scatter Search for the Periodic Capacitated Arc Routing Problem*. European Journal of Operational Research. Vol. 169, p. 586–605, 2006.

CORBERÁN, A.; PRINS, C. *Recent Results on Arc Routing Problems: An Annotated Bibliography*. Networks, p. 50–69, 2010.

- DASTIDAR, S. G.; NAGI, R.** *Scheduling injection molding operations with multiple resource constraints and sequence dependent setup times and costs.* Computers & Operations Research. Vol. 32, p. 2987-3005, 2005.
- DANTZIG, G. B.; RAMSER, J. H.** *The Truck Dispatching Problem.* Management Science. Vol. 6, n. 1, p.80-91, 1959.
- DOULABI, S. H. H.; SEIFE, A.** *Lower and upper bounds for location-arc routing problems with vehicle capacity constraints.* European Journal of Operational Research. Vol. 224, n. 1, p.189–208, 2013.
- EISELT, H. A.; GENDREAU, M.; LAPORTE, G.** *Arc Routing Problems, Part I: The Chinese Postman Problem.* Operations Research. Vol. 43, n. 2, p. 231–242, 1995i. Disponível em: <<http://or.journal.informs.org/cgi/doi/10.1287/opre.43.2.231>>.
- EISELT, H. A.; GENDREAU, M.; LAPORTE, G.** *Arc Routing Problems , Part II : The Rural Postman Problem.* Operations Research. Vol. 43, n. 3, p. 399–414, 1995ii. Disponível em: <<http://or.journal.informs.org/content/43/3/399.full.pdf>>.
- FERREIRA, D.; MORABITO, R.; RANGEL, S.** *Solution Approaches for the Soft Drink Integrated Production Lot Sizing and Scheduling Problem.* European Journal of Operational Research. Vol. 196, p. 697-706, 2009.
- FERREIRA, D.; MORABITO, R.; RANGEL, S.** *Relax and Fix Heuristics to Solve One-Stage Onemachine Lot-Scheduling Models for Small-Scale soft Drink Plants.* Computers & Operations Research. Vol.37, p. 684–691, 2010.
- FRANCO, E. F. J.; OLIVEIRA, H. C. B.** *Adaptação da Meta-Heurística GRASP na resolução do problema de Roteamento de Veículos com Janela de Tempo.* Rio de Janeiro. Vol. 4, n. 3, p. 271-287, 2012.
- GOLDEN, B. L.; WONG, R. T.** *Capacitated Arc Routing Problem.* Networks. Vol. 11, p. 305–315, 1981.
- GOLDBARG, M. C.** *Otimização Combinatória e Programação Linear: Modelos e Algoritmos.* 2. ed. Rio de Janeiro: Elsevier, 2005.
- KANSOU, A.; YASSINE, A.** *Ant Colony System for the Periodic Capacitated Arc Routing Problem.* Proc. 2009 International Network Optimization Conference. Anais... p.1–7, 2009.
- LACOMME, P.; PRINS, C.; RAMDANE-CHÉRIEF, W.** *Evolutionary Algorithms for Multiperiod Arc Routing Problems.* In: IPMU 2002 (Ed.); 9th Int. Conf. on Information Processing and Management of Uncertainty in Knowledge-Based systems. Anais. p.1–8, 2002i.
- LACOMME, P.; PRINS, C.; RAMDANE-CHÉRIEF, W.** *Planning Problems in Arc Routing.* In: F. U.-E. de Valencia (Ed.); Actes de PMS 2002 (EURO Working Group on Project Management and Scheduling). Anais... p.232–235, 2002ii.
- LACOMME, P.; PRINS, C.; RAMDANE-CHÉRIEF, W.** *Evolutionary Algorithms for Periodic Arc Routing Problems.* European Journal of Operational Research. Vol. 165, p. 535–553, 2005.
- MONROY, I. M.; AMAYA, C. A.; LANGEVIN, A.** *The periodic capacitated arc routing problem with irregular services.* Discrete Applied Mathematics. Vol. 161, p. 691–701, 2013.
- MORAES, L. C. C.; SANTOS, M. O.** *Heurística Relax-and-Fix para o Problema de Dimensionamento de Lotes com Janela de Tempo de Produção.* 2012. Disponível em: <<http://www.din.uem.br/sbpo/sbpo2012/pdf/arq0479.pdf>>.
- NEGREIROS, M.; COELHO JUNIOR, W. R.; PALHANO, A. W. D. C.; et al.** *O Problema do Carteiro Chinês, Algoritmos Exatos e um Ambiente MVI para Análise de suas Instâncias: Sistema XNÊS.* Pesquisa Operacional. Vol. 29, n. 2, p. 323–363, 2009.
- WOLSEY, L. A.** *Wiley-Interscience Series in Discrete Mathematics and Optimization: Integer Programming.* Wiley. 1998.