5
Algoritmos Duais

5.1
Representação da Solução Dual

A modelagem de uma solução dual para o PMNC baseou-se na formulação dual condensada apresentada na Seção 2.2.3. Esta solução deve informar o valor da função objetivo, o valor da variável dual γ e o valor das variáveis duais λ. Assim, foi definido um conjunto para as variáveis duais λ e o mesmo foi implementado utilizando um *array* de números fracionários com o intuito de podermos recuperar e acessar aleatoriamente cada elemento em tempo constante.

5.2
Dualidade

5.2.1
Complementaridade de Folga

Um relacionamento importante entre soluções primais e duais ótimas é descrito pelas condições de *complementaridade de folga* (detalhes em (Bertsimas e Tsitsiklis, 1997)). Estas condições são enunciadas da seguinte forma para o PMNC: dado x e y vetores que constituem a solução viável para o problema primal e λ o vetor com a solução viável para o problema dual. Os vetores x, y e λ são soluções ótimas para os respectivos problemas se e somente se

$$y_i^*(\gamma - \sum_{i \in U} \max(0, \lambda_i^* - d_{ij})) = 0 \tag{5-1}$$

$$(y_i^* - x_{ij}^*) \max(0, \lambda_i^* - d_{ij}) = 0 \tag{5-2}$$

$$(\sum_{j \in F} y_j - p)\gamma = 0 \tag{5-3}$$

As condições de complementaridade de folga são utilizadas para derivar soluções primais a partir de soluções fornecidas pelo algoritmo *Dual Ascent*;

para efetuar ajustes em soluções duais no algoritmo *Dual Adjustment* e para guiar a escolha da variável para ramificação em cada nó do algoritmo *Branch-and-Ascent*. Estes algoritmos são descritos nas seções 4.2.5, 5.4 e 6.2.1, respectivamente.

5.2.2
Fixação de Variáveis por Custo Reduzido

A fixação de variáveis por custo reduzido, discutida em (Wolsey, 1998), consiste em fixar em zero o valor de determinadas variáveis da formulação primal, levando em consideração o ganho que tal fixação acarretaria no valor da solução desta formulação. A seguir descrevemos esta técnica aplicado ao PMNC.

Seja z_P o valor da melhor solução viável conhecida para as restrições (2-1) a (2-5) e seja z_D o valor de uma solução viável para as restrições (2-14) e (2-15). Considere, também, os custos reduzidos $\overline{c}_\pi(j)$ e $\overline{c}_\pi(ij)$ definidos em (2-13) e (2-17), respectivamente. As condições a seguir são válidas para a fixação das variáveis x_{ij} e y_j em zero:

– Se $\overline{c}_\pi(j) > z_P - z_D$ então a variável y_j pode ser fixada em zero

– Se $\overline{c}_\pi(ij) > z_P - z_D$ então a variável x_{ij} pode ser fixada em zero

A fixação da variável x_{ij} e y_j em zero implica, respectivamente, na eliminação da possibilidade de se atribuir o cliente i a facilidade j e na eliminação da facilidade j como candidato a abertura. Consequentemente, estas fixações possibilitam a redução do tamanho das instâncias, podendo ser utilizada como uma técnica de pré-processamento para, posteriormente, ser aplicado algum método as instâncias reduzidas. Na Seção descrevemos o algoritmo exato *Branch-and-Ascent* que utiliza, além das fixações no pré-processamento, fixações em cada nó da árvore de enumeração com o intuito de reduzir ao máximo o número de nós pesquisados. Abaixo, explicamos como uma variável fixada por custo reduzido permanece fora da solução primal construída pelo método *Primal-Dual*.

A garantia da fixação da variável primal y_j como fechada é feita substituindo-se a parcela $-\gamma$ na restrição dual (2-17) por ∞ para j. Assim, para fixarmos a variável primal y_j como fechada ($y_j = 0$), substituímos a restrição dual (2-17) referente a facilidade j por $\sum_{i \in U} \max(0, \lambda_i - d_{ij}) \leq \infty$. Como consequência, esta restrição não limitará o crescimento do valor de nenhuma variável dual λ_i e o custo reduzido $\overline{c}_\pi(j)$ descrito na equação (2-17) referente a facilidade j assumirá o maior valor possível. Como as facilidades abertas na solução primal possuem $\overline{c}_\pi(j)$ mínimo (preferencialmente $\overline{c}_\pi(j) = 0$), esta facilidade nunca será definida como aberta. Portanto, esta alteração na formulação dual condensada garante a fixação de uma facilidade j como fechada. Para garantirmos a fixação da variável primal x_{ij} em zero (eliminar a possibilidade de se atribuir o cliente i a facilidade j), mantemos nulo o valor desta variável quando executamos qualquer método sobre esta configuração.

O Apêndice A apresenta os resultados obtidos com a fixação por custo reduzido das variáveis x_{ij} e y_j. Para cada instância é descrita as dimensões da mesma, o percentual de redução no número de vértices e arestas, juntamente com o tempo (seg.) gasto para a execução das rotinas responsáveis por estas reduções.

5.3 Dual Ascent

5.3.1 Descrição

Os métodos do tipo *Dual Ascent* (detalhes em (Reeves, 1993) e (Bertsimas e Tsitsiklis, 1997)) partem de uma solução dual viável e, a cada iteração, tentam aumentar o valor das variáveis duais desta solução. O método é executado até que não seja mais possível incrementar nenhuma variável dual mantendo a solução dual viável. Pelas condições de complementaridade de folga descritas na Seção 5.2.1, é possível construir uma solução primal a partir de qualquer solução dual encontrada por este método. Esta construção é descrita na 4.2.5. Assim, o incremento nos valores das variáveis duais são realizados com o intuito de reduzir ao máximo o *Gap* entre as soluções primal e dual.

5.3.2 Implementação

A Figura 5.1 mostra o pseudocódigo do método *Dual Ascent*. Este pseudocódigo é uma adaptação do método apresentado em (Captivo, 1991). Neste pseudocódigo, o primeiro passo (linhas 3 a 5), que consiste na construção de uma solução inicial dual viável π' e no cálculo dos valores iniciais para os conjuntos J^+ e $K(i)$, é concluído em tempo $O(n)$. O segundo passo (linhas 6 a 8) consiste no cálculo dos valores iniciais para os conjuntos θ_j e ξ_j e possui complexidade $O(n^2)$, pois para cada $j \in F$, calculamos um somatório envolvendo cada $i \in U$. No passo seguinte (linha 9) definimos o valor da variável dual γ. Esta definição possui complexidade $O(n)$ em consequência do tempo $O(n)$ gasto no cálculo de ζ. A etapa seguinte (linhas 10 a 22) é composta por um *loop* que é executado enquanto alguma variável dual λ_i sofrer alteração em seu valor. Vamos analisar a complexidade de pior caso de cada iteração deste *loop*. Interno ao *loop*, tentaremos incrementar o valor da variável dual λ_i para cada $i \in J^+$. Para isto, primeiramente definimos o valor de Δ_i em tempo $O(n)$. Δ_i é o maior valor possível para incremento de λ_i sem violar as restrições definidas em (2-15) (Δ_i é o custo reduzido $\overline{c}_\pi(j)$ mínimo obtido analisando cada $j \in F$ - $\overline{c}_\pi(j)$ é calculado como definido em (2-17)). Após a definição de Δ_i, precisamos recuperar o valor da distância d_{ij} da próxima facilidade j retirada do conjunto ordenado c_i^k. Precisamos deste valor, pois o incremento de λ_i será $\min(\Delta_i, d_{ij})$. O pior caso desta etapa acontece quando percorremos todo o conjunto c_i^k tendo, assim, complexidade $O(n)$ (o pior caso somente acontecerá em circunstâncias definidas posteriormente, quando executamos o *Dual Ascent* sobre instâncias reduzidas). Dando continuidade, ajustamos Δ_i em tempo constante, incrementamos os valores de θ_j, para cada $j \in F : \lambda_i - d_{ij} \geq 0$, em tempo $O(n)$ e, incrementamos também, o valor de λ_i, porém em tempo constante. Contudo, cada iteração possui complexidade $O(n^2)$, pois no pior caso $J^+ = F$. Após o *loop*, construímos o conjunto I^+ e a solução primal viável α em tempo $O(n^2)$ cada, e calculamos o valor da função objetivo da solução dual π em tempo $O(n)$. Portanto, a complexidade do método *Dual Ascent* é $O(n^2)$ por iteração.

Os métodos *Dual Ascent* e *Dual Adjustment*, descrito nas Seção 5.4, necessitam, para cada cliente j, da criação de um conjunto contendo as facilidades ordenadas em ordem não-decrescente de distância do respectivo cliente j as facilidades. Para a criação destes relacionamentos foi implementado o algoritmo de ordenação *Heapsort* descrito em (Cormen et al., 2001). Este algoritmo ordena localmente os elementos de um conjunto (apenas um número constante de elementos do conjunto é armazenado fora do conjunto em qualquer etapa do método) e usa uma estrutura de dados chamada *heap*, estrutura esta descrita em (Cormen et al., 2001), para efetuar esta ordenação. O algoritmo de ordenação possui complexidade $O(nlgn)$, onde n é o números de elementos contidos no conjunto a ser ordenado. Durante o carregamento das instâncias são definidos estes relacionamentos gastando um tempo $O(n^2lgn)$, pois necessitamos ordenar n linhas de uma matriz de dimensão nxn.

Procedimento *Dual Ascent*

Entrada: Solução dual vazia π

Saída: Solução dual viável π^* e uma solução primal viável α

para cada $i \in U$ **faça**

  $\lambda_i \leftarrow 0$; $J^+ \leftarrow true$; $K(i) \leftarrow \min(k : \lambda_i \leq d_i^k) + 1$;

fim-para

para cada $j \in F$ **faça**

  $\theta_j \leftarrow \sum_{i \in U} \max(0, \lambda_i - d_{ij})$; $\xi_j \leftarrow \sum_{i \in U} \max(0, \min_{l \in S} d_{il} - d_{ij})$

fim-para

$\zeta \leftarrow \max_{j \in F} \xi_j$; $\gamma \leftarrow -\zeta$; $\delta \leftarrow true$;

enquanto $\delta = true$ **faça**

  $\delta \leftarrow false$;

  para cada $i \in J^+$ **faça**

    $\Delta_i \leftarrow \min_{j \in F}(\overline{c}_\pi(j) = -\gamma - \theta_j : \lambda_i - d_{ij} \geq 0)$;

    se $\Delta_i = 0$ **então** executar a próxima iteração com um novo j **fim-se**

    se $d_i^{K(i)} = \infty$ **então** executar a próxima iteração com um novo j **fim-se**

    enquanto $d_i^{K(i)} - \lambda_i = 0$ **faça** $K(i) \leftarrow K(i) + 1$ **fim-enquanto**

    se $\Delta_i > d_i^{K(i)} - \lambda_i$ **então** $\Delta_i \leftarrow d_i^{K(i)} - \lambda_i$ **fim-se**

    $\delta \leftarrow true$;

    para cada $j \in F : \lambda_i - d_{ij} \geq 0$ **faça** $\theta_j \leftarrow \theta_j + \Delta_i$ **fim-para**

    $\lambda_i \leftarrow \lambda_i + \Delta_i$;

  fim-para

fim-enquanto

Construção do conjunto I^+

Construção da solução primal α

Cálculo do valor da função objetivo de π que é dado por $fO_\pi \leftarrow \sum_{i \in U} \lambda_i + \gamma p$

$\pi^* \leftarrow \pi$

retorne (π^*, α);

Figura 5.1: Algoritmo *Dual Ascent* - Fonte: adaptado de (Captivo, 1991)

5.4 Dual Adjustment

5.4.1 Descrição

Se a solução dual obtida pela heurística *Dual Ascent* e a solução primal derivada desta solução dual satisfazem as condições de complementaridade de folga então a solução é ótima. Caso contrário, é feita uma tentativa de melhorar a solução dual através da heurística *Dual Adjustment*. Nesta heurística , selecionamos alguma facilidade $j^{'}$ para o qual o segundo grupo de restrições de complementaridade de folga (restrições 5-2) é violado. Suponha que seja decrementado $\lambda_{j'}$. Isto cria folga em, pelo menos, duas restrições 2-15 amarradas. A heurística tenta incrementar outros λ_j que são limitados por estas restrições. Se mais de um λ_j pode ser incrementado com o valor $\lambda_{j'}$, então o valor da função objetivo para o dual será incrementada. Porém, se somente um λ_j pode ser incrementado com $\lambda_{j'}$, então o valor da função objetivo para o dual permanece no mesmo nível, mas a folga criada em alguma restrição altera o valor da função objetivo para o primal. O procedimento cicla através deste processo para todas as comunidades $j^{'}$ e repete o procedimento enquanto o valor da função objetivo para o dual é incrementada. Assim, o procedimento decrementa $\lambda_{j'}$ para o próximo menor valor c_j^k e aplica a heurística *Dual Ascent* para aqueles λ_j identificados como prováveis candidatos a incrementar.

5.4.2 Implementação

A Figura 5.2 apresenta o pseudocódigo do método *Dual Adjustment*. Este pseudocódigo é uma adaptação do método apresentado em (Erlenkotter, 1978). Vamos analisar a complexidade deste método. Há um *loop* sobre todos os clientes envolvendo as linhas 3 a 15. Neste *loop*, a linha 4 consiste na construção dos conjuntos I_i^*, I_i^+ e J_j^+, que é realizado em tempo $O(n^2)$. Em seguida, as linhas 5, 6 e 7 são executadas em tempo constante. Na linha 8 é obtido c_j^- e atualizado $K(i)$ gastando um tempo total de $O(n)$. Na linha 9 incrementamos os valores de θ_j, para cada $j \in F : \lambda_i - d_{ij} \geq 0$, em tempo $O(n)$ e, na linha 10, alteramos o valor de λ_i em tempo constante. Prosseguindo, na linha 11, a construção de J^+ é realizado em tempo $O(n)$ e o método *Dual Ascent* possui complexidade $O(n^2)$ por iteração, conforma já descrito. Assim, a linha 11 gasta um tempo $O(\rho^{'} n^2)$ para ser concluída, onde $\rho^{'}$ é o número de iterações do *Dual Ascent*. A linha 12 possui complexidade $O(\rho^{''} n^2)$, onde $\rho^{''}$ é o número de iterações do *Dual Ascent* (a construção de J^+ é realizada em tempo $O(n)$

Procedimento *Dual Adjustment*

1. **Entrada:** Solução dual viável π
2. **Saída:** Solução dual viável π^* e uma solução primal viável α
3. **para** cada $i \in U$ **faça**
4. Construção dos conjuntos I_i^*, I_i^+ e J_j^+
5. **se** $|I_i^+| \leq 1$ **então** executar a próxima iteração com um novo i **fim-se**
6. **se** $J^+_{i^+(j)} = \emptyset \wedge J^+_{i'(j)} = \emptyset$ **então** executar a próxima iteração com um novo i **fim-se**
7. $\omega \leftarrow \lambda_i$;
8. Calcular c_j^- e atualizar $K(i)$;
9. **para** cada $j \in F : \lambda_i - d_{ij} \geq 0$ **faça** $\theta_j \leftarrow \theta_j - (\lambda_i - c_j^-)$ **fim-para**
10. $\lambda_i \leftarrow c_j^-$;
11. Fazer $J^+ \leftarrow J^+_{i^+(j)} \cup J^+_{i'(j)}$ e executar o método *Dual Ascent*
12. Fazer $J^+ \leftarrow J$ e executar o método *Dual Ascent*
13. Executar o método *Dual Ascent*
14. **se** $\lambda_i \neq \omega$ **então** executar a próxima iteração com o mesmo i **fim-se**
15. **fim-para**
16. **retorne** (π^*, α);

Figura 5.2: Algoritmo *Dual Adjustment* - Fonte: adaptado de (Erlenkotter, 1978)

sendo, assim, assintoticamente dominada pelo tempo de execução do método *Dual Ascent*). Por fim, a linha 13, onde é invocado novamente o método *Dual Ascent*, possui complexidade $O(\rho^{'''}n^2)$. Lembrando que, como todas estas etapas compõem cada iteração do método *Dual Adjustment*, o mesmo gasta um tempo $O(\max(\rho^{'}, \rho^{''}, \rho^{'''})n^3)$ por iteração.

5.5 Dual Scaling

5.5.1 Descrição

O método *Dual Scaling* tenta melhorar uma solução π fornecida pelo método *Dual Ascent* através da multiplicação de cada variável dual λ_i que compõe π por um mesmo fator constante β escolhido pseudo-aleatoriamente dentro do intervalo [0.1, 1.0[. Esta etapa de multiplicação é chamada de *scaling*. Após o *scaling* das variáveis duais, obtemos uma nova solução viável $\pi^{'}$ e invocamos o método *Dual Ascent* utilizando como ponto de partida esta solução. Seja $\pi^{''}$ a solução fornecida pelo *Dual Ascent*. Se $\pi^{''}$ for melhor do que π então o *scaling* possibilitou um melhoramento na solução fornecida pelo *Dual Ascent*. Podemos iterar sobre estes passos até que não seja mais possível melhorar a melhor solução encontrada nas iterações anteriores do *Dual Scaling*.

Foram implementadas duas variações do método *Dual Scaling*. A primeira, chamada de *Dual Scaling Não Aleatório*, utiliza o mesmo multiplicador β em todas as iterações do método. A segunda, chamada de *Dual Scaling Aleatório*, altera o multiplicador β sempre que é atualizada a melhor solução encontrada nas iterações anteriores do *Dual Scaling*. A Figura 5.3 apresenta o pseudocódigo do método.

5.5.2 Implementação

Iremos, agora, analisar a complexidade do método *Dual Scaling*, cujo pseudocódigo está descrito na Figura 5.2. Neste pseudocódigo, há um *loop* envolvendo as linhas 3 a 22. Neste *loop*, nas linhas 4 a 8, para cada $j \in F$, multiplicamos a variável dual λ_j por β em tempo constante, atualizamos $K(j)$ em tempo $O(n)$ e calculamos θ_j também em tempo $O(n)$. Assim, a complexidade das linhas 4 a 8 é $O(n^2)$. Nas linhas 9, 10, 11 e 12 invocamos os métodos *Dual Ascent*, *Dual Adjustment* e a heurística de refinamento gastando um tempo total de $O(\rho n^3)$ para ser concluída, onde ρ é o número de iterações do *Dual Adjustment*. As demais linhas são executadas em tempo constante. Portanto, a complexidade do método *Dual Scaling* é $O(\rho n^3)$ por iteração.

A Figura 5.4 apresenta o pseudocódigo utilizado para a seleção do melhor β que será apresentado na seção de resultados posteriormente.

Procedimento *Dual Scaling*

Entrada: Solução dual viável π, fator constante multiplicador β e a *flag* δ informando se o método a ser utilizado é aleatória ou não aleatório

Saída: Solução dual viável π^* e uma solução primal viável α

enquanto Critério de parada não satisfeito **faça**
    para cada $j \in F$ **faça**
        $\lambda_j \leftarrow \lambda_j \times \beta$;
        Atualizar $K(j)$;
        $\theta_j \leftarrow \sum_{i \in U} \max(0, \lambda_i - d_{ij})$;
    fim-para
    $\pi \leftarrow$ *DualAscent*(π);
    $\pi \leftarrow$ *DualAdjustment*(π);
    $\alpha \leftarrow$ *PrimalDual*(π);
    $\alpha \leftarrow$ *BuscaLocal*(α);
    se $fO_\pi > fO_{\pi^*}$ **então**
        $\pi^* \leftarrow \pi$;
        se $\delta = true$ **então**
            Atualizar β com um valor escolhido pseudo-aleatoriamente dentro do intervalo [0.1, 1.0[
        fim-se
    senão
        retorne π^*;
    fim-senão
fim-enquanto
retorne (π^*, α^*);

Figura 5.3: *Algoritmo Dual Scaling*

Procedimento ***Seleção do*** β

1. **Entrada:** Solução dual viável π
2. **Saída:** Solução dual viável π^* e uma solução primal viável α
3. **enquanto** Critério de parada não satisfeito **faça**
4. $\pi^{'} \leftarrow \pi$;
5. Inicializar β com um valor escolhido pseudo-aleatoriamente dentro do intervalo [0.1, 1.0[
6. $(\pi^{'}, \alpha^{'}) \leftarrow$ *DualScaling*$(\pi^{'})$;
7. **se** $fO_\alpha < fO_{\alpha^*}$ **então** $\alpha^* \leftarrow \alpha$ **fim-se**
8. **se** $fO_\pi > fO_{\pi^*}$ **então** $\pi^* \leftarrow \pi$ **fim-se**
9. **fim-enquanto**
10. **retorne** (π^*, α^*);

Figura 5.4: Algoritmo para seleção do β utilizado nos resultados reportados para o *Dual Scaling*

5.6 Fixação Ativa

Nesta seção tratamos do método *Fixação Ativa*, cuja denominação foi recebida por Poggi de Aragão et al. 2001 (Poggi de Aragão et al., 2001).

5.6.1 Fixação Ativa de Arcos

Descrição

A heurística dual *Fixação Ativa de Arcos* têm como objetivo fixar arcos em zero, ou seja, eliminar arcos que constituem a instância de teste, baseado no custo reduzido destes arcos. Como já discutido, esta fixação reduz o tamanho das instâncias e possibilitará uma melhoria na qualidade das soluções duais fornecidas pelo método *Dual Ascent*. Conforme apresentado na Seção 2.2.2, o custo reduzido $\overline{c}_\pi(ij)$ referente ao arco (i, j) é dado por 2-12 e para que este arco seja fixado em zero (a variável x_{ij} ser fixada em zero) ele deve satisfazer a condição $\overline{c}_\pi(ij) > z_P - z_D$ (o fundamento matemático desta fixação é explicada na Seção 5.2.2). Assim, se $\overline{c}_\pi(ij)$ for máximo então aumentaremos a probabilidade do arco (i, j) ser fixado (a probabilidade do arco (i, j) ser eliminado cresce com o aumento de seu custo reduzido $\overline{c}_\pi(ij)$). Considerando um determinado arco (i, j), se fixarmos λ_i em zero então $\mu_{ij} = 0, \forall j \in F$ de acordo com a definição de μ_{ij} em 2-16. Consequentemente, $\overline{c}_\pi(ij) = d_{ij}$. Assim, a imposição desta condição para λ_i e a execução do *Dual Ascent* respeitando esta condição aumentará a probabilidade de serem eliminados arcos na instância de teste. Na escolha do arco (i, j) para a fixação de λ_i em zero, analisamos todos os arcos cujo custo reduzido seja superior ao *Gap* entre a melhor solução primal α^* conhecida e a solução dual corrente π. Dentre estes arcos, selecionamos aquele que possui $\overline{c}_\pi(ij)$ máximo.

Implementação

A Figura 5.5 descreve o pseudocódigo do método *Fixação Ativa de Arcos* cuja complexidade será discutida nesta seção. Neste pseudocódigo, há um *loop* envolvendo as linhas 3 a 19. Dentro deste *loop*, invocamos os métodos *Dual Ascent*, *Dual Adjustment*, o método construtivo Primal-Dual e uma busca local nas linhas 4, 5, 6 e 7, respectivamente. Conforme já analisado, o *Dual Adjustment* é o método mais custoso dentre estes com complexidade $O(\rho n^3)$ por iteração. As linhas 8 e 9 tratam da atualização da melhor solução primal e dual encontradas até o momento. Cada atualização possui complexidade $O(n)$. Na linha 10 é calculado o *Gap* em tempo constante. Na linha 11 é selecionado

Procedimento *Fixação Ativa de Arcos*

1. **Entrada:** Solução dual viável π, as melhores soluções primal α^* e dual π^* encontradas até o momento
2. **Saída:** Solução dual viável π
3. <u>**enquanto**</u> Critério de parada não satisfeito <u>**faça**</u>
4. $\quad \pi \leftarrow$ *DualAscent(π)*;
5. $\quad \pi \leftarrow$ *DualAdjustment(π)*;
6. $\quad \alpha \leftarrow$ *PrimalDual(π)*;
7. $\quad \alpha \leftarrow$ *BuscaLocal(α)*;
8. $\quad$ <u>**se**</u> $fO_\alpha < fO_{\alpha^*}$ <u>**então**</u> $\alpha^* \leftarrow \alpha$ <u>**fim-se**</u>
9. $\quad$ <u>**se**</u> $fO_\pi > fO_{\pi^*}$ <u>**então**</u> $\pi^* \leftarrow \pi$ <u>**fim-se**</u>
10. $\quad Gap = fO_{\alpha^*} - fO_\pi$;
11. $\quad$ Selecionar o arco (i, j) que possui $\overline{c}_\pi(ij)$ máximo dentre todos os arcos que possuem $\overline{c}_\pi(ij) > Gap$
12. $\quad$ Definir o valor máximo permitido para $\lambda_{i'} : i' = i$ como sendo 0.0. As demais variáveis $\lambda_{i''} : i'' \neq i$ não são limitadas
13. $\quad$ <u>**para**</u> cada $i \in U$ <u>**faça**</u>
14. $\quad\quad$ <u>**para**</u> cada $j \in F$ <u>**faça**</u>
15. $\quad\quad\quad$ <u>**se**</u> $\overline{c}_\pi(ij) > Gap$ <u>**então**</u> Fixar o arco (i, j) como fechado <u>**fim-se**</u>
16. $\quad\quad$ <u>**fim-para**</u>
17. $\quad$ <u>**fim-para**</u>
18. <u>**fim-enquanto**</u>
19. <u>**retorne**</u> (π^*, α^*);

Figura 5.5: *Algoritmo Fixação Ativa de Arcos*

o arco que possui o custo reduzido $\overline{c}_\pi(ij)$ mínimo em tempo $O(n^2)$. Na linha 12 é fixada a variável λ_i em tempo constante. Nas linhas 13 a 18 é feita a fixação de arcos por custo reduzido em tempo $O(n^2)$. Contudo, a complexidade do método *Fixação Ativa de Arcos* é $O(\rho n^3)$ por iteração.

5.6.2 Fixação Ativa de Facilidades

Descrição

A heurística dual *Fixação Ativa de Facilidades* têm o mesmo objetivo da heurística dual *Fixação Ativa de Arcos*, porém difere desta no tipo de variável a ser fixada. Enquanto a primeira heurística fixa variáveis y_j, a segunda heurística fixa variáveis x_{ij}. Conforme apresentado na Seção 2.2.2, o custo reduzido $\overline{c}_{\pi}(j)$ referente a facilidade j é dado por 2-13 e para que esta facilidade seja fixada em zero (a variável y_j ser fixada em zero) ele deve satisfazer a condição $\overline{c}_{\pi}(j) > z_P - z_D$ (o fundamento matemático desta fixação é explicada na Seção 5.2.2). Assim, a probabilidade da facilidade j ser eliminado cresce com o aumento de seu custo reduzido $\overline{c}_{\pi}(j)$. Considerando uma determinada facilidade j, se definimos o limite superior de λ_i como sendo d_{ij} então $\mu_{ij} = 0, \forall j \in F$ de acordo com a definição de μ_{ij} em 2-16. Consequentemente, $\overline{c}_{\pi}(j) = -\gamma$. Assim, a imposição deste limite superior para λ_i e a execução do *Dual Ascent* respeitando esta condição aumentará a probabilidade de serem eliminados arcos na instância de teste. Na escolha da facilidade j para a fixação de λ_i em d_{ij}, analisamos todas as facilidades cujo custo reduzido seja superior ao *Gap* entre a melhor solução primal α^* conhecida e a solução dual corrente π. Dentre estes arcos, selecionamos aquele que possui $\overline{c}_{\pi}(j)$ máximo.

Implementação

A Figura 5.6 descreve o pseudocódigo do método *Fixação Ativa de Facilidades* cuja complexidade será discutida nesta seção. Neste pseudocódigo, há um *loop* envolvendo as linhas 3 a 19. Dentro deste *loop*, invocamos os métodos *Dual Ascent*, *Dual Adjustment*, o método construtivo Primal-Dual e uma busca local nas linhas 4, 5, 6 e 7, respectivamente. Conforme já analisado, o *Dual Adjustment* é o método mais custoso dentre estes com complexidade $O(\rho n^3)$ por iteração. As linhas 8 e 9 tratam da atualização da melhor solução primal e dual encontradas até o momento. Cada atualização possui complexidade $O(n)$. Na linha 10 é calculado o *Gap* em tempo constante. Na linha 11 é selecionado a facilidade que possui o custo reduzido $\overline{c}_{\pi}(j)$ mínimo em tempo $O(n)$. Nas linhas 12 a 14 é definido o valor máximo para as variáveis λ_i em tempo $O(n)$ no total. Nas linhas 15 a 18 é feita a fixação de facilidades por custo reduzido em tempo $O(n)$. Contudo, a complexidade do método *Fixação Ativa de Facilidades* é $O(\rho n^3)$ por iteração.

Procedimento *Fixação Ativa de Facilidades*

1. **Entrada:** Solução dual viável π, as melhores soluções primal α^* e dual π^* encontradas até o momento
2. **Saída:** Solução dual viável π
3. **enquanto** Critério de parada não satisfeito **faça**
4. $\pi \leftarrow$ *DualAscent*(π);
5. $\pi \leftarrow$ *DualAdjustment*(π);
6. $\alpha \leftarrow$ *PrimalDual*(π);
7. $\alpha \leftarrow$ *BuscaLocal*(α);
8. **se** $fO_\alpha < fO_{\alpha^*}$ **então** $\alpha^* \leftarrow \alpha$ **fim-se**
9. **se** $fO_\pi > fO_{\pi^*}$ **então** $\pi^* \leftarrow \pi$ **fim-se**
10. $Gap = fO_{\alpha^*} - fO_\pi$;
11. Selecionar a facilidade j que possui $\overline{c}_\pi(j)$ máximo dentre todas as facilidades que possuem $\overline{c}_\pi(j) > Gap$
12. **para** cada $i \in U$ **faça**
13. Definir o valor máximo permitido para λ_i como sendo d_{ij}
14. **fim-para**
15. **para** cada $j \in F$ **faça**
16. **se** $\overline{c}_\pi(j) > Gap$ **então** Fixar a facilidade j como fechada **fim-se**
17. **fim-para**
18. **fim-enquanto**
19. **retorne** (π^*, α^*);

Figura 5.6: *Algoritmo Fixação Ativa de Facilidades*

5.7 Resultados Computacionais

5.7.1 Soluções Obtidas

Dual Ascent e Dual Adjustment

Nesta seção são apresentados os resultados obtidos com o *Dual Ascent* e com o *Dual Adjustment*. A sequência de passos para a geração destes resultados é a seguinte: primeiramente, constrói-se soluções iniciais com todos os métodos implementados e considera a melhor solução para o passo seguinte. No próximo passo, aplica-se uma busca local sobre esta solução inicial. Os passos seguintes mudam um pouco para a coleta de dados para análise do *Dual Ascent* e do *Dual Adjustment* isoladamente. Para o *Dual Ascent* os passos são os seguintes: executa-se o método *Dual Ascent* seguido do método construtivo *Primal-Dual*. Sobre a solução primal fornecida pelo *Primal-Dual* aplica-se uma busca local. Para o *Dual Adjustment* os passos são os seguintes: executa-se o método *Dual Ascent*, o método *Dual Adjustment* e o método *Primal-Dual* utilizando a solução dual fornecida pelo *Dual Adjustment*. Sobre a solução primal fornecida pelo *Primal-Dual* aplica-se uma busca local. É importante ressaltar que os dados expostos nas tabelas a seguir com referência aos métodos *Dual Ascent* e *Dual Adjustment* não dizem respeito a estes métodos puramente, mas sim a execução de todos os passos descritos anteriormente para cada um destes métodos.

As tabelas 5.1 e 5.2 comparam os métodos *Dual Ascent* e *Dual Adjustment* quando aplicados sobre as instâncias da classe *OR-Library*. Para o *Dual Ascent*, das 40 soluções analisadas, 18 ficaram com um *Gap* menor do que 2% e 22 ficaram com um *Gap* entre 2% e 5%. Estas soluções foram geradas gastando um tempo menor do que 1 segundo para 34 instâncias e para as demais 6 instâncias foi gasto um tempo inferior a 2.3 segundos. Porém, para o *Dual Adjustment*, 30 soluções ficaram com um *Gap* menor do que 2% e 10 ficaram com um *Gap* entre 2% e 3.5%. Estas soluções foram geradas gastando um tempo menor do que 1 segundo para 10 instâncias e para as demais 30 instâncias foi gasto um tempo inferior a 84 segundos. Conforme disposto na tabela, o *Dual Adjustment* somente não melhorou o *Gap* fornecido pelo *Dual Ascent* para as instâncias *pmed4* e *pmed5*. Contudo, houve um acréscimo considerável no tempo de execução quando comparado com aquele gasto pelo *Dual Ascent*, fato justificado pela complexidade dos métodos (ver Tabela 5.20).

Instância	**(n,p)**	**Dual Ascent**						**Dual Adjustment**					
		dP_1	dP_2	**dD**	Gap_1	Gap_2	T_{DA}	dP_1	dP_2	**dD**	Gap_1	Gap_2	$T_D Adj$
pmed1	(100,5)	**0,00**	**0,00**	3,41	3,41	3,41	0,02	1,27	**0,00**	2,34	3,64	2,34	0,13
pmed2	(100,10)	**0,00**	**0,00**	1,59	1,59	1,59	**0,00**	**0,00**	**0,00**	1,49	1,49	1,49	0,08
pmed3	(100,10)	1,01	**0,00**	1,75	2,78	1,75	**0,00**	0,07	**0,00**	0,83	0,90	0,83	0,05
pmed4	(100,20)	0,13	0,13	0,20	0,33	0,33	**0,00**	0,13	0,13	0,20	0,33	0,33	0,06
pmed5	(100,33)	0,22	0,15	0,89	1,12	1,04	**0,00**	0,22	0,15	0,89	1,12	1,04	0,03
pmed6	(200,5)	0,12	**0,00**	3,56	3,68	3,56	0,03	0,12	**0,00**	2,18	2,30	2,18	0,92
pmed7	(200,10)	0,14	0,14	2,38	2,53	2,53	0,02	0,14	0,14	0,97	1,11	1,11	0,48
pmed8	(200,20)	0,76	0,20	1,39	2,17	1,60	0,03	2,05	0,20	0,43	2,49	0,63	0,47
pmed9	(200,40)	1,13	0,48	0,63	1,77	1,10	0,03	**0,00**	**0,00**	0,26	0,26	0,26	0,36
pmed10	(200,67)	0,72	0,64	0,40	1,12	1,04	0,06	0,72	0,64	0,32	1,04	0,96	0,39
pmed11	(300,5)	0,23	**0,00**	3,16	3,40	3,16	0,05	0,35	**0,00**	2,03	2,39	2,03	3,56
pmed12	(300,10)	0,21	**0,00**	2,46	2,67	2,46	0,06	0,21	**0,00**	1,27	1,48	1,27	1,84
pmed13	(300,30)	**0,00**	**0,00**	0,99	0,99	0,99	0,08	0,21	**0,00**	0,55	0,76	0,55	1,41
pmed14	(300,60)	0,61	0,10	0,71	1,32	0,81	0,08	0,47	0,10	0,34	0,81	0,44	1,56
pmed15	(300,100)	0,35	0,06	0,35	0,70	0,41	0,14	0,23	0,06	0,17	0,41	0,23	1,48
pmed16	(400,5)	**0,00**	**0,00**	3,70	3,70	3,70	0,08	1,19	**0,00**	2,54	3,76	2,54	10,73
pmed17	(400,10)	0,71	**0,00**	3,11	3,85	3,11	0,11	0,69	**0,00**	1,85	2,55	1,85	6,84
pmed18	(400,40)	0,12	0,04	2,47	2,60	2,51	0,14	0,17	0,04	1,16	1,33	1,20	4,69
pmed19	(400,80)	0,21	0,21	1,57	1,79	1,79	0,28	0,35	0,21	0,92	1,28	1,14	3,95
pmed20	(400,133)	0,34	**0,00**	0,79	1,13	0,79	0,42	0,28	**0,00**	0,56	0,84	0,56	4,16

Tabela 5.1: Resultados - *Dual Ascent e Dual Adjustment* aplicado as instâncias da classe *OR-Library*

Instância	**(n,p)**	**Dual Ascent**						**Dual Adjustment**					
		dP_1	dP_2	**dD**	Gap_1	Gap_2	T_{DA}	dP_1	dP_2	**dD**	Gap_1	Gap_2	$T_D Adj$
pmed21	(500,5)	**0,00**	**0,00**	2,08	2,08	2,08	0,19	**0,00**	**0,00**	0,85	0,85	0,85	13,42
pmed22	(500,10)	**0,00**	**0,00**	3,44	3,44	3,44	0,19	0,68	**0,00**	1,74	2,43	1,74	16,47
pmed23	(500,50)	0,48	**0,00**	1,92	2,41	1,92	0,30	1,15	**0,00**	0,94	2,10	0,94	9,45
pmed24	(500,100)	0,51	0,24	1,40	1,92	1,64	0,56	0,44	0,24	0,89	1,33	1,12	8,16
pmed25	(600,167)	0,49	0,33	1,39	1,89	1,72	0,95	0,88	0,33	0,94	1,82	1,27	9,38
pmed26	(600,5)	0,07	**0,00**	3,55	3,62	3,55	0,31	0,07	**0,00**	2,22	2,29	2,22	28,52
pmed27	(600,10)	0,61	**0,00**	2,68	3,31	2,68	0,30	1,56	**0,00**	1,55	3,14	1,55	22,67
pmed28	(600,60)	0,29	0,02	2,25	2,55	2,27	0,63	0,27	0,02	1,35	1,62	1,37	14,98
pmed29	(600,120)	0,59	0,03	1,85	2,45	1,88	1,08	0,69	0,03	0,86	1,56	0,90	17,03
pmed30	(600,200)	0,96	0,60	1,22	2,19	1,83	2,25	1,11	0,60	1,07	2,18	1,68	18,36
pmed31	(700,5)	0,43	**0,00**	4,57	5,02	4,57	0,34	0,70	**0,00**	3,09	3,81	3,09	59,98
pmed32	(700,10)	0,38	0,04	3,67	4,06	3,71	0,44	1,31	0,04	1,94	3,28	1,98	40,25
pmed33	(700,70)	0,60	0,09	2,26	2,87	2,35	1,05	0,38	0,09	0,99	1,38	1,07	25,89
pmed34	(700,140)	0,46	0,27	1,35	1,82	1,61	2,16	0,33	0,27	0,84	1,17	1,10	27,27
pmed35	(800,5)	0,06	**0,00**	4,73	4,79	4,73	0,47	0,05	**0,00**	2,85	2,90	2,85	53,84
pmed36	(800,10)	0,29	**0,00**	3,98	4,28	3,98	0,52	0,45	**0,00**	2,47	2,93	2,47	58,72
pmed37	(800,80)	0,24	0,02	2,45	2,69	2,47	1,47	0,42	0,02	1,34	1,76	1,36	35,97
pmed38	(900,5)	0,96	**0,00**	4,85	5,86	4,85	0,63	1,10	**0,00**	3,46	4,60	3,46	83,84
pmed39	(900,10)	0,06	**0,00**	3,79	3,86	3,79	0,59	0,17	**0,00**	2,09	2,26	2,09	83,19
pmed40	(900,90)	0,27	0,25	2,50	2,78	2,76	2,16	0,62	0,25	1,40	2,04	1,66	58,47

Tabela 5.2: Resultados - *Dual Ascent e Dual Adjustment* aplicado as instâncias da classe *OR-Library*

A Tabela 5.3 compara as soluções fornecidas pelo *Dual Ascent* e pelo *Dual Adjustment* com a solução ótima da relaxação linear do PMNC definida pela função objetivo (2-1) juntamente com as restrições (2-2)-(2-4), (2-6) e (2-7). Esta solução ótima é o melhor valor possível que podemos obter para o limite inferior fornecido por estes métodos. Para a coluna *Relaxação Linear*, apresentamos o desvio dual dD (2-20) considerando como valor de referência (FO_{PR}) o ótimo de cada instância. O ótimo da relaxação linear (FO_{DE}), que é utilizado no cálculo do dR, foi obtido utilizando o pacote comercial ILOG CPLEX 11.2 (ILOG, 2008). No entanto, além do dR, apresentamos também o tempo (T_{CPLEX}) gasto pelo ILOG CPLEX 11.2 para obter estas soluções. Para as colunas *dR Dual Ascent* e *dR Dual Adjustment*, revelamos o desvio da relaxação linear dR (2-21). Comparando a qualidade das soluções geradas pelo *Dual Ascent*, das 40 instâncias analisadas, 8 soluções possuem um desvio menor do que 1%, 10 soluções possuem um desvio entre 1% e 2%, 13 soluções possuem um desvio entre 2% e 3% e 9 soluções possuem um desvio entre 3% e 4%. Já para o método *Dual Adjustment*, 19 soluções possuem um desvio menor do que 1%, 18 soluções possuem um desvio entre 1% e 2% e apenas 3 soluções possuem um desvio entre 2% e $2,5\%$. Estas medidas dão uma análise mais precisa da qualidade das soluções geradas por estes métodos. Quanto ao tempo, o CPLEX encontra o ótimo das relaxações em menos de 1.1 segundo para todas as 40 instâncias.

A Tabela 5.4 compara os métodos *Dual Ascent* e *Dual Adjustment* para a instância *fl1400* da classe *TSP-Library*. Para o *Dual Ascent*, das 18 soluções analisadas, 7 ficaram com um *Gap* menor do que 2%, 8 ficaram com um *Gap* entre 2% e 5% e 3 ficaram com um *Gap* entre 5% e 7.3%. Estas soluções foram geradas gastando um tempo menor do que 20 segundos para 13 instâncias e para as demais 5 instâncias foi gasto um tempo inferior a 37 segundos. Para o *Dual Adjustment*, 11 soluções ficaram com um *Gap* menor do que 2%, 4 ficaram com um *Gap* entre 2% e 3% e 3 ficaram com um *Gap* entre 3% e 6%. Como podemos observar, o *Dual Adjustment* somente não melhora o *Gap* para a rodada considerando $p = 10$, onde a solução se manteve no mesmo patamar.

As tabelas 5.5 e 5.6 apresentam os resultados obtidos com a aplicação do *Dual Ascent* sobre as instâncias *pcb3038* e *rl5934* da classe *TSP-Library*. O *Dual Adjustment* não foi aplicado sobre estas instâncias devido ao fato de se gastar um tempo computacional proibitivo para a conclusão do procedimento. Analisando a Tabela 5.5, das 28 soluções analisadas, 4 ficaram com um *Gap* menor do que 2%, 22 ficaram com um *Gap* entre 2% e 3% e 2 ficaram com um *Gap* entre 3% e 3.4%. Estas soluções foram geradas gastando um tempo entre 66,44 e 411,55 segundos. Já para a instância *rl5934*, Tabela 5.6, também

Instância	**(n,p)**	**Relaxação Linear**		**dR Dual Ascent**	**dR Dual Adjustment**
		dD	T_{CPLEX}		
pmed1	(100,5)	0,00	0,00	3,41	2,34
pmed2	(100,10)	0,11	0,11	1,48	1,38
pmed3	(100,10)	0,22	0,22	1,52	0,60
pmed4	(100,20)	0,00	0,00	0,20	0,20
pmed5	(100,33)	0,00	0,00	0,89	0,89
pmed6	(200,5)	0,52	0,52	3,02	1,65
pmed7	(200,10)	0,00	0,00	2,38	0,97
pmed8	(200,20)	0,00	0,00	1,39	0,43
pmed9	(200,40)	0,00	0,00	0,63	0,26
pmed10	(200,67)	0,00	0,00	0,40	0,32
pmed11	(300,5)	0,03	0,03	3,13	1,99
pmed12	(300,10)	0,12	0,12	2,33	1,14
pmed13	(300,30)	0,00	0,00	0,99	0,55
pmed14	(300,60)	0,03	0,03	0,69	0,31
pmed15	(300,100)	0,00	0,00	0,35	0,17
pmed16	(400,5)	0,87	0,87	2,81	1,66
pmed17	(400,10)	0,44	0,44	2,66	1,41
pmed18	(400,40)	0,01	0,01	2,46	1,15
pmed19	(400,80)	0,00	0,00	1,57	0,92
pmed20	(400,133)	0,00	0,00	0,79	0,56
pmed21	(500,5)	0,00	0,00	2,08	0,85
pmed22	(500,10)	0,41	0,41	3,01	1,33
pmed23	(500,50)	0,00	0,00	1,92	0,94
pmed24	(500,100)	0,00	0,00	1,40	0,89
pmed25	(600,167)	0,00	0,00	1,39	0,94
pmed26	(600,5)	0,64	0,64	2,89	1,56
pmed27	(600,10)	0,06	0,06	2,62	1,49
pmed28	(600,60)	0,00	0,00	2,25	1,35
pmed29	(600,120)	0,00	0,00	1,85	0,86
pmed30	(600,200)	0,00	0,00	1,22	1,07
pmed31	(700,5)	0,60	0,60	3,95	2,47
pmed32	(700,10)	0,05	0,05	3,62	1,89
pmed33	(700,70)	0,00	0,00	2,26	0,99
pmed34	(700,140)	0,00	0,00	1,35	0,84
pmed35	(800,5)	0,95	0,95	3,75	1,88
pmed36	(800,10)	1,02	1,02	2,92	1,43
pmed37	(800,80)	0,00	0,00	2,45	1,34
pmed38	(900,5)	1,03	1,03	3,78	2,41
pmed39	(900,10)	0,63	0,63	3,14	1,45
pmed40	(900,90)	0,00	0,00	2,50	1,40

Tabela 5.3: Resultados - *Relaxação Linear, Dual Ascent e Dual Adjustment* aplicado as instâncias da classe *OR-Library*

p	**Dual Ascent**						**Dual Adjustment**					
	dP_1	dP_2	**dD**	Gap_1	Gap_2	T_{DA}	dP_1	dP_2	**dD**	Gap_1	Gap_2	T_{DAdj}
10	0,30	0,00	0,37	0,67	0,37	10,67	0,30	0,00	0,37	0,67	0,37	78,44
20	0,35	0,00	0,71	1,07	0,71	8,05	3,17	0,00	0,65	3,85	0,65	955,47
30	2,06	0,62	1,82	3,91	2,45	5,86	2,05	0,62	1,33	3,41	1,96	649,09
40	1,56	0,05	1,41	3,00	1,46	5,98	1,32	0,05	1,02	2,36	1,08	543,83
50	1,96	0,00	1,22	3,21	1,22	6,61	2,69	0,00	0,87	3,58	0,87	390,16
60	1,71	0,23	1,47	3,20	1,70	5,75	2,16	0,23	1,26	3,45	1,50	401,23
70	1,28	0,48	0,45	1,73	0,93	5,77	1,19	0,48	0,27	1,46	0,76	451,94
80	1,38	0,60	1,26	2,65	1,87	5,42	1,54	0,60	1,07	2,62	1,68	299,95
90	2,38	0,87	1,28	3,69	2,17	5,98	1,48	0,87	0,77	2,26	1,65	358,03
100	1,55	0,41	1,78	3,36	2,20	6,80	1,52	0,41	0,91	2,45	1,33	553,14
150	2,55	1,02	2,12	4,72	3,16	10,02	2,45	1,02	1,40	3,89	2,44	457,58
200	2,07	0,51	2,21	4,32	2,73	12,42	2,36	0,51	1,46	3,85	1,98	393,73
250	1,81	0,30	3,31	5,18	3,62	17,59	1,26	0,30	2,46	3,75	2,77	356,69
300	1,88	1,19	2,80	4,73	4,02	23,00	1,79	1,19	1,76	3,58	2,96	390,08
350	2,36	1,82	1,88	4,29	3,74	24,33	1,53	0,75	1,47	3,02	2,23	392,73
400	3,95	2,34	3,18	7,26	5,60	31,16	2,78	2,34	1,72	4,55	4,10	508,81
450	3,19	0,93	5,38	8,74	6,35	33,95	2,72	0,93	4,10	6,93	5,07	466,41
500	2,57	1,13	6,10	8,82	7,29	36,22	2,21	1,13	4,68	6,99	5,86	447,25

Tabela 5.4: Resultados - *Dual Ascent e Dual Adjustment* aplicado a instância *fl1400* da classe *TSP-Library*

das 28 soluções analisadas, 1 ficou com um *Gap* menor do que 2%, 24 ficaram com um *Gap* entre 2% e 3% e 3 ficaram com um *Gap* entre 3% e 3.3%. Estas soluções foram geradas gastando um tempo entre 485,61 e 3835,20 segundos. Estes tempos são bastante competitivos considerando a qualidade das soluções analisadas e, considerando também, o fato do pacote comercial ILOG CPLEX (ILOG, 2008) não conseguir obter o ótimo para instâncias desta dimensão no ambiente computacional utilizado em nossos experimentos.

p	**Dual Ascent**					
	dP_1	dP_2	**dD**	Gap_1	Gap_2	**T**
10	3,44	1,36	2,00	5,51	3,39	328,25
20	3,85	0,32	2,63	6,59	2,96	149,30
30	2,17	0,32	2,43	4,65	2,76	119,23
40	2,52	0,45	2,00	4,57	2,46	95,92
50	1,84	0,35	1,89	3,76	2,24	87,33
60	1,43	1,35	1,35	2,80	2,72	78,86
70	1,92	0,90	1,77	3,73	2,69	72,13
80	2,07	0,66	1,67	3,78	2,35	68,95
90	2,08	0,79	1,87	3,99	2,67	66,44
100	1,90	0,59	1,51	3,43	2,11	70,45
150	1,77	0,86	1,60	3,40	2,48	76,73
200	1,36	0,83	1,44	2,82	2,28	85,05
250	1,34	0,82	1,51	2,87	2,34	101,81
300	1,02	0,99	1,27	2,31	2,28	115,59
350	1,14	0,68	1,23	2,39	1,92	132,41
400	0,93	0,18	1,15	2,09	1,33	161,11
450	1,04	1,00	1,09	2,15	2,10	164,92
500	0,88	0,28	1,07	1,96	1,35	189,61
550	0,99	0,37	1,00	2,00	1,37	208,33
600	1,11	0,92	1,08	2,20	2,01	228,42
650	1,61	1,30	1,40	3,04	2,72	253,88
700	1,78	1,24	1,41	3,21	2,66	278,00
750	1,54	1,47	1,50	3,07	3,00	291,20
800	1,65	1,44	1,60	3,28	3,06	328,88
850	1,75	1,24	1,51	3,29	2,77	362,63
900	1,71	1,28	1,41	3,14	2,70	383,63
950	1,41	1,32	1,36	2,79	2,69	393,02
1000	1,17	1,11	1,20	2,39	2,33	411,55

Tabela 5.5: Resultados - *Dual Ascent* aplicado a instância *pcb3038* da classe *TSP-Library*

p	Dual Ascent					
	dP_1	dP_2	**dD**	Gap_1	Gap_2	**T**
10	3,79	0,03	3,30	7,22	3,33	2608,53
20	3,14	0,28	2,21	5,41	2,50	1383,13
30	2,52	0,32	2,33	4,90	2,66	964,84
40	1,59	0,27	1,72	3,34	1,99	805,16
50	2,45	0,36	2,55	5,06	2,92	603,58
60	2,67	0,96	2,52	5,25	3,51	566,64
70	2,26	0,71	2,46	4,78	3,19	537,86
80	1,84	0,55	1,65	3,52	2,21	549,36
90	2,34	0,59	1,93	4,32	2,52	501,48
100	1,62	0,62	1,90	3,55	2,53	485,61
150	1,77	0,74	1,82	3,62	2,57	531,17
200	1,42	0,81	1,69	3,14	2,52	621,41
250	1,41	0,64	1,46	2,89	2,11	673,48
300	1,34	0,66	1,77	3,13	2,44	792,45
350	1,54	0,58	1,93	3,51	2,52	909,20
400	1,60	0,61	1,85	3,49	2,47	1037,91
450	1,40	0,56	1,59	3,01	2,16	1106,33
500	1,13	0,65	1,43	2,58	2,09	1209,33
600	1,01	0,60	1,52	2,54	2,12	1411,30
700	1,03	0,64	1,46	2,50	2,12	1668,91
800	1,22	0,64	1,43	2,67	2,09	1865,64
900	1,06	0,87	1,31	2,39	2,19	2110,55
1000	1,17	0,72	1,29	2,47	2,02	2379,53
1100	1,30	0,71	1,32	2,64	2,04	2597,17
1200	1,42	0,73	1,52	2,96	2,26	2868,03
1300	1,74	0,77	1,75	3,52	2,54	3205,55
1400	1,94	0,85	1,94	3,92	2,81	3456,52
1500	1,81	0,88	1,88	3,73	2,78	3835,20

Tabela 5.6: Resultados - *Dual Ascent* aplicado a instância *rl5934* da classe *TSP-Library*

Dual Scaling

Nesta seção são apresentados os resultados obtidos com o *Dual Scaling*. A sequência de passos para a geração destes resultados é a seguinte: primeiramente, constrói-se soluções iniciais com todos os métodos implementados e refina cada uma destas soluções com a busca local. A melhor solução é considerada para o passo seguinte. No passo seguinte, executa-se o método *Dual Ascent*, o método *Dual Adjustment* e o método *Primal-Dual* utilizando a solução dual fornecida pelo *Dual Adjustment*. A solução primal obtida pelo *Primal-Dual* é refinada com a busca local. Dando continuidade, aplica-se o *Dual Scaling* utilizando como ponto de partida as soluções primal e dual obtidas no passos anteriores.

A Tabela 5.7 apresenta os resultados obtidos com a aplicação do *Dual Scaling* sobre as instâncias da classe *OR-Library*. Nesta tabela, exibimos os resultados deste método na versão aleatório e não aleatório. Para cada versão, temos o fator multiplicador β, o *Gap* (Gap_I) antes e o *Gap* (Gap_F) após a aplicação do método, o percentual de redução (R (%)) do *Gap* decorrente desta aplicação e o tempo (T), e segundos, gasto para a obtenção das novas soluções. As instâncias não citadas nesta tabela não foram beneficiadas com a aplicação deste método. O fator multiplicador β foi definido utilizando o algoritmo descrito na Figura 5.4. Para a versão aleatória, a menor e a maior redução do *GAP* foram 1,20% e 12,34%, respectivamente. Na versão não aleatória, estes valores foram 2,41% e 10,43%, respectivamente. Melhoras nas soluções duais ocorreram em 7 instâncias para a versão aleatória e em 9 instâncias para a versão não aleatória. No geral, para as instâncias onde ocorrem melhoras em ambas as versões, o método não aleatório obteve resultados mais satisfatórios. Apesar de serem poucas as instâncias beneficiadas com este método, estas reduções podem ser úteis para técnicas baseadas em enumeração implícita como aquela descrita na Seção 6.2.

Instância	**(n,p)**	**Aleatório**					**Não-Aleatório**				
		β	Gap_I	Gap_F	**R (%)**	**T**	β	Gap_I	Gap_F	**R (%)**	**T**
pmed1	(100,5)	0,9998	2,43	2,24	7,82	38,34	0,9999	2,43	2,24	7,82	83,41
pmed3	(100,10)	0,9998	0,83	0,82	1,20	6,11	0,9999	0,83	0,81	2,41	43,94
pmed6	(200,5)	0,9989	2,30	2,27	1,30	67,13	0,9999	2,30	2,06	10,43	30,73
pmed7	(200,10)	–	–	–	–	–	0,9999	1,11	1,06	4,50	40,00
pmed11	(300,5)	0,9993	1,88	1,80	4,26	10,45	0,9988	1,88	1,84	2,13	79,78
pmed19	(400,80)	–	–	–	2,91	–	0,9994	1,03	0,96	6,80	55,81
pmed22	(500,10)	–	–	–	–	–	0,9993	1,62	1,54	4,94	96,08
pmed27	(600,10)	–	–	–	–	–	0,9978	1,53	1,47	3,92	136,58
pmed31	(700,5)	0,9982	3,09	2,87	7,12	130,88	–	–	–	–	–
pmed32	(700,10)	–	–	–	–	–	0,9971	2,02	1,83	9,41	125,83
pmed34	(700,140)	0,9995	0,94	0,94	–	65,83	–	–	–	–	–
pmed37	(800,80)	0,9995	1,54	1,35	12,34	75,47	–	–	–	–	–

Tabela 5.7: Resultados - *Dual Scaling* aplicado as instâncias da classe *OR-Library*

A Tabela 5.8 compara as soluções fornecidas pelo *Dual Scaling* com a solução ótima da relaxação linear do PMNC para as instâncias que foram beneficiadas com a aplicação deste método.

Instância	**(n,p)**	**Relaxação Linear**		**dR Aleatório**	**dR Não-Aleatório**
		dP	**T**		
pmed1	(100,5)	0,00	0,00	2,29	2,24
pmed3	(100,10)	0,22	0,22	0,60	0,59
pmed6	(200,5)	0,52	0,52	1,74	1,53
pmed7	(200,10)	0,00	0,00	–	0,92
pmed11	(300,5)	0,03	0,03	1,84	1,81
pmed19	(400,80)	0,00	0,00	–	0,85
pmed22	(500,10)	0,41	0,41	–	1,13
pmed27	(600,10)	0,06	0,06	–	1,41
pmed31	(700,5)	0,60	0,60	2,47	–
pmed32	(700,10)	0,05	0,05	–	1,78
pmed34	(700,140)	0,00	0,00	0,84	–
pmed37	(800,80)	0,00	0,00	1,21	–

Tabela 5.8: Resultados - *Relaxação Linear e Dual Scaling* aplicado as instâncias da classe *OR-Library*

O *Dual Scaling* não apresentou melhoras para as instâncias da *TSP-Library* quando executado com um limite de tempo de 300 seg. Não é justificável aplicar este método considerando um tempo de execução superior a 300 seg. para estas instâncias devido ao pouco benefício obtido com o mesmo sendo, portanto, descartada a sua utilização nesta classe.

Fixação Ativa

Nesta seção serão apresentados os resultados obtidos com a fixação de arcos e facilidades por custo reduzido. A sequência de passos para a geração destes resultados é a seguinte: primeiramente, constroem-se soluções iniciais com todos os métodos implementados e, em seguida, é refinado cada uma destas soluções com a busca local. A melhor solução é considerada para o passo seguinte. No passo seguinte, aplica-se a *Fixação Ativa*.

Nas tabelas 5.9, 5.10, 5.11 e 5.12 serão analisadas tanto as fixações ocorridas após o término do método *Fixação Ativa* de arcos e facilidades (*FAA, FAF*) quanto as fixações ocorridas após a primeira iteração destes métodos (*FCR*). A sigla *FCR* denota Fixação por Custo Reduzido e as siglas *FAA* e *FAF* denotam, respectivamente, Fixação Ativa de Arcos e Fixação Ativa de Facilidades. Em cada agrupamento de dados (*FCR, FAA e FAF*) são expostos o *Gap*, o percentual de redução de arcos ou facilidades ($R\%$) e o tempo (T), em segundos, gasto para a obtenção destas soluções.

Iremos, agora, analisar o percentual de redução de arcos e facilidades quando comparamos os métodos *FCR* e *FAA* para os dados da Tabela 5.9. Observando a fixação de arcos, das 40 instâncias analisadas, 8 melhoram o percentual, 15 mantiveram o mesmo percentual e 17 não fixação nenhum arco nestes métodos. Observando a fixação de facilidades nesta mesma tabela, 20 melhoram o percentual, 5 mantiveram o mesmo percentual e 15 não fixação nenhuma facilidade nestes métodos. A explicação para a não fixação de arcos e facilidades serão argumentadas com o auxílio das tabelas 5.13 e 5.16, respectivamente.

Novamente, analisaremos o percentual de redução de arcos e facilidades quando comparamos os métodos *FCR* e *FAA* para os dados reportados na Tabela 5.10. Analisando a fixação de arcos, das 18 instâncias analisadas, 6 melhoram o percentual, 12 mantiveram o mesmo percentual e em todas as configurações houve fixação de arcos. Analisando a fixação de facilidades, 3 melhoram o percentual e 15 não fixação nenhuma facilidade nestes métodos. A explicação para a não fixação de facilidades será argumentada com o auxílio da Tabela 5.17 mais adiante.

Com relação ao percentual de redução de arcos e facilidades quando comparamos os métodos *FCR* e *FAA*, de acordo com a Tabela 5.11, analisando a fixação de arcos, das 28 instâncias analisadas, 12 melhoram o percentual, 4 mantiveram o mesmo percentual e 12 não fixação nenhum arco nestes métodos. Analisando a fixação de facilidades, uma configuração manteve o percentual de redução e 27 não fixação nenhuma facilidade nestes métodos. A explicação para a não fixação de arcos e facilidades serão argumentadas com o auxílio das

Instância	(n,p)	Fixação de Arcos				Fixação de Facilidades			
		FCR		FAA		FCR		FAF	
		R (%)	T	R (%)	T	R (%)	T	R (%)	T
pmed1	(100,5)	4,35	0,11	5,76	0,34	64,00	0,14	78,00	0,23
pmed2	(100,10)	65,46	0,06	65,81	0,22	40,00	0,10	68,00	0,30
pmed3	(100,10)	82,71	0,05	82,71	0,09	58,00	0,08	59,00	0,16
pmed4	(100,20)	95,80	0,06	95,88	0,14	47,00	0,09	54,00	0,16
pmed5	(100,33)	95,46	0,06	95,46	0,06	17,00	0,07	22,00	0,16
pmed6	(200,5)	0,00	0,98	0,00	0,97	53,00	1,35	63,50	2,88
pmed7	(200,10)	16,64	0,52	16,64	1,42	52,50	0,68	73,00	2,48
pmed8	(200,20)	87,53	0,45	87,58	1,11	64,50	0,59	78,00	1,33
pmed9	(200,40)	96,71	0,38	96,71	0,59	52,50	0,49	52,50	0,70
pmed10	(200,67)	97,13	0,39	97,13	0,61	16,00	0,51	35,50	1,06
pmed11	(300,5)	0,00	3,42	0,00	3,38	54,33	4,97	55,00	8,88
pmed12	(300,10)	1,02	1,97	1,02	3,86	59,00	2,82	73,33	10,63
pmed13	(300,30)	71,16	1,48	73,22	4,39	32,67	2,33	64,67	7,16
pmed14	(300,60)	95,03	1,59	95,03	2,67	20,67	2,40	22,67	5,84
pmed15	(300,100)	98,54	1,56	98,54	2,52	23,67	2,20	23,67	3,02
pmed16	(400,5)	0,00	9,95	0,00	10,67	60,50	14,76	60,50	17,16
pmed17	(400,10)	0,00	6,38	0,00	6,67	30,00	9,24	30,25	18,94
pmed18	(400,40)	1,01	4,78	1,02	14,14	0,00	6,63	0,00	4,73
pmed19	(400,80)	60,16	4,11	60,16	8,03	0,00	5,56	0,00	4,06
pmed20	(400,133)	96,84	4,34	96,84	6,81	0,00	5,93	0,00	4,36
pmed21	(500,5)	0,00	13,08	0,00	12,95	83,80	17,29	89,60	33,97
pmed22	(500,10)	0,00	16,88	0,00	16,88	36,80	22,57	36,80	31,95
pmed23	(500,50)	0,65	9,59	0,66	28,16	0,00	13,67	0,00	9,45
pmed24	(500,100)	22,80	8,55	22,83	25,50	0,00	11,86	0,00	8,59
pmed25	(600,167)	48,48	9,66	48,48	16,33	0,00	14,03	0,00	9,91
pmed26	(600,5)	0,00	23,86	0,00	23,67	61,00	40,75	69,00	72,84
pmed27	(600,10)	0,00	23,28	0,00	26,63	24,00	30,64	26,83	65,61
pmed28	(600,60)	0,14	14,83	0,14	30,28	0,00	19,83	0,00	14,72
pmed29	(600,120)	30,09	17,63	30,09	33,69	0,00	23,50	0,00	17,52
pmed30	(600,200)	12,52	18,61	12,52	34,16	0,00	24,24	0,00	18,63
pmed31	(700,5)	0,00	59,09	0,00	57,39	33,71	77,00	58,71	157,13
pmed32	(700,10)	0,00	37,61	0,00	37,28	0,00	49,47	0,00	36,78
pmed33	(700,70)	0,00	26,44	0,00	26,38	0,00	35,49	0,00	25,95
pmed34	(700,140)	14,19	27,83	14,19	54,38	0,00	37,22	0,00	27,59
pmed35	(800,5)	0,00	53,02	0,00	52,80	33,25	70,63	44,88	314,61
pmed36	(800,10)	0,00	56,44	0,00	55,86	0,00	75,25	0,00	55,47
pmed37	(800,80)	0,00	36,06	0,00	36,16	0,00	48,22	0,00	35,69
pmed38	(900,5)	0,00	86,80	0,00	86,67	51,89	115,77	51,89	155,94
pmed39	(900,10)	0,00	84,08	0,00	87,23	6,56	112,21	7,44	237,72
pmed40	(900,90)	0,00	60,34	0,00	63,13	0,00	83,07	0,00	60,34

Tabela 5.9: Resultados - *Fixação Ativa* de arcos e facilidades aplicado as instâncias da classe *OR-Library*

tabelas 5.14 e 5.18, respectivamente.

Com relação ao percentual de redução de arcos e facilidades quando comparamos os métodos *FCR* e *FAA*, de acordo com a Tabela 5.12, analisando a fixação de arcos, das 28 configurações de p analisadas, 10 melhoram o percentual de redução de arcos quando comparamos os métodos *FCR* e *FAA*, uma configuração manteve o mesmo percentual e 17 não fixação nenhum arco nestes métodos. Analisando a fixação de facilidades, uma configuração manteve o percentual de redução de facilidades quando comparamos os métodos *FCR* e *FAA* e 27 não fixação nenhuma facilidade nestes métodos. A explicação para a não fixação de arcos e facilidades serão argumentadas com o auxílio das tabelas 5.15 e 5.19, respectivamente.

As tabelas 5.13, 5.14 e 5.15 exibem, respectivamente, a maior distância

p	Fixação de Arcos				Fixação de Facilidades			
	FCR		FAA		FCR		FAF	
	R (%)	T	R (%)	T	R (%)	T	R (%)	T
10	78,59	88,52	78,59	155,94	78,29	165,18	81,29	597,36
20	80,40	932,84	83,83	2135,11	17,00	2149,48	89,29	5716,87
30	68,87	632,72	81,08	2229,86	0,00	1453,09	0,00	1530,46
40	78,69	574,66	79,70	1882,05	0,00	1266,31	0,00	1369,71
50	81,85	456,03	81,85	849,31	0,00	884,94	0,00	1035,73
60	81,45	491,06	83,46	1117,44	0,00	888,26	0,00	1113,56
70	87,18	446,13	87,18	765,56	0,43	837,79	1,93	3128,27
80	82,84	320,89	86,05	1036,30	0,00	572,68	0,00	706,16
90	84,83	385,03	87,75	821,64	0,00	689,68	0,00	839,24
100	84,63	643,78	84,63	879,34	0,00	1148,40	0,00	1393,34
150	84,16	477,78	84,16	766,19	0,00	782,00	0,00	1022,71
200	86,87	402,39	86,87	650,94	0,00	728,55	0,00	857,70
250	85,88	371,67	85,88	588,70	0,00	690,71	0,00	799,02
300	87,31	402,78	87,31	623,06	0,00	735,39	0,00	840,42
350	87,81	393,92	87,81	614,08	0,00	747,04	0,00	846,57
400	87,92	511,11	87,92	761,67	0,00	1046,74	0,00	1179,26
450	86,46	519,22	86,46	788,70	0,00	1129,31	0,00	1241,00
500	86,18	433,17	86,18	689,92	0,00	872,35	0,00	1068,64

Tabela 5.10: Resultados - *Fixação Ativa* de arcos e facilidades aplicado a instância *fl1400* da classe *TSP-Library*

p	Fixação de Arcos				Fixação de Facilidades			
	FCR		FAA		FCR		FAF	
	R (%)	T	R (%)	T	R (%)	T	R (%)	T
10	0,00	383,50	0,00	361,86	0,82	468,68	0,82	639,25
20	0,00	217,53	0,00	204,39	0,00	338,82	0,00	205,06
30	0,00	182,03	0,00	170,86	0,00	301,89	0,00	171,73
40	0,00	165,56	0,00	157,77	0,00	283,21	0,00	158,70
50	0,00	152,83	0,00	146,97	0,00	268,61	0,00	148,02
60	0,00	146,78	0,00	141,02	0,00	258,11	0,00	141,95
70	0,00	141,88	0,00	136,64	0,00	261,12	0,00	137,66
80	0,00	137,50	0,00	133,02	0,00	245,95	0,00	136,56
90	0,00	141,06	0,00	133,94	0,00	210,29	0,00	136,66
100	0,00	144,86	0,00	138,39	0,00	193,16	0,00	140,06
150	0,00	158,36	0,00	152,67	0,00	260,31	0,00	153,44
200	0,00	177,34	0,67	245,47	0,00	355,52	0,00	171,91
250	0,10	203,55	4,74	347,64	0,00	397,45	0,00	196,02
300	9,94	222,77	13,46	295,88	0,00	456,23	0,00	212,02
350	16,60	248,75	16,60	266,33	0,00	423,91	0,00	231,61
400	25,75	263,89	34,14	333,52	0,00	549,37	0,00	254,00
450	40,06	290,05	42,48	419,50	0,00	594,39	0,00	287,42
500	46,91	305,38	46,91	333,89	0,00	622,36	0,00	306,95
550	45,79	330,98	45,79	361,73	0,00	702,98	0,00	362,03
600	41,92	375,45	47,26	584,72	0,00	756,05	0,00	459,25
650	32,29	410,95	44,07	525,34	0,00	791,41	0,00	497,30
700	40,27	435,36	44,50	581,67	0,00	821,28	0,00	533,22
750	43,21	472,80	46,66	718,64	0,00	864,50	0,00	563,66
800	44,47	491,06	46,44	701,14	0,00	901,67	0,00	603,64
850	42,69	534,75	46,39	844,50	0,00	964,86	0,00	640,58
900	49,71	558,02	49,71	651,39	0,00	1071,59	0,00	673,59
950	54,01	566,08	54,96	798,97	0,00	1060,65	0,00	698,89
1000	58,79	596,02	65,16	817,61	0,00	1056,64	0,00	723,30

Tabela 5.11: Resultados - *Fixação Ativa* de arcos e facilidades aplicado a instância *pcb3038* da classe *TSP-Library*

p	Fixação de Arcos				Fixação de Facilidades			
	FCR		FAA		FCR		FAF	
	R (%)	T	R (%)	T	R (%)	T	R (%)	T
10	0,00	3314,59	0,00	2977,66	0,03	4648,26	0,03	7689,94
20	0,00	1864,98	0,00	1927,08	0,00	2373,79	0,00	1657,45
30	0,00	1478,30	0,00	1361,13	0,00	1785,22	0,00	1400,86
40	0,00	1314,13	0,00	1251,53	0,00	1625,89	0,00	1258,89
50	0,00	1118,98	0,00	1077,52	0,00	1383,26	0,00	1289,22
60	0,00	1077,30	0,00	1088,95	0,00	1368,58	0,00	1106,06
70	0,00	1087,55	0,00	1095,77	0,00	1352,66	0,00	1088,11
80	0,00	1049,36	0,00	1063,39	0,00	1325,96	0,00	1062,69
90	0,00	1053,69	0,00	1078,00	0,00	1328,19	0,00	1073,67
100	0,00	1066,77	0,00	1072,09	0,00	1367,02	0,00	1107,86
150	0,00	1237,14	0,00	1202,56	0,00	1494,61	0,00	1243,53
200	0,00	1392,14	0,00	1368,48	0,00	1690,49	0,00	1419,34
250	0,00	1595,33	0,00	1555,53	0,00	1865,54	0,00	1595,08
300	0,00	1783,03	0,00	1758,88	0,00	2175,60	0,00	1807,11
350	0,00	1937,95	0,00	1887,05	0,00	2392,43	0,00	2048,22
400	0,00	2143,34	0,00	2121,95	0,00	2598,97	0,00	2223,58
450	0,00	2310,56	0,00	2286,61	0,00	2895,86	0,00	2426,61
500	0,35	2466,14	2,27	3428,17	0,00	3119,62	0,00	2721,61
600	3,67	2855,31	8,56	3812,31	0,00	3633,05	0,00	3148,39
700	7,54	3562,38	15,46	4271,59	0,00	4003,16	0,00	3538,39
800	13,06	3598,38	21,11	4930,11	0,00	4582,56	0,00	4001,78
900	20,02	4094,72	25,63	5916,55	0,00	4855,86	0,00	4366,89
1000	31,23	4396,52	33,37	6759,70	0,00	5472,90	0,00	4782,98
1100	33,03	4850,67	40,27	7202,17	0,00	5927,70	0,00	5168,19
1200	29,14	5155,89	39,98	7657,80	0,00	7191,00	0,00	5549,33
1300	23,84	5614,42	38,07	8518,23	0,00	8978,33	0,00	6016,56
1400	23,32	6018,53	36,69	10434,66	0,00	10257,66	0,00	6301,48
1500	32,96	6358,34	32,96	7731,88	0,00	10727,65	0,00	6477,45

Tabela 5.12: Resultados - *Fixação Ativa* de arcos e facilidades aplicado a instância *rl5934* da classe *TSP-Library*

d_{ij}, juntamente com o respectivo arco (i, j), contida em cada um dos grafos que definem as instâncias da classe *OR-Library* e as instâncias *pcb3038* e *rl5934* da classe *TSP-Library*. Além, das triplas *(i, j, d_{ij})*, são apresentados os *Gaps* utilizados nas fixações. Nestas tabelas estão presentes apenas as instâncias/configurações de p onde nenhum arco foi fixado por custo reduzido. Analisando a Tabela 5.13 observamos que somente as instâncias *pmed6*, *pmed21*, *pmed33* e *pmed37* possuem $d_{ij} > Gap$. Para as demais instâncias, $d_{ij} < Gap$. Como o custo reduzido $\bar{c}_\pi(ij)$ do arco *(i,j)* é definido pela equação 2-12, estas instâncias nunca terão $\bar{c}_\pi(ij) > d_{ij}$. Para que *(i,j)* seja fixado, a condição $\bar{c}_\pi(ij) > Gap$ deve ser satisfeita (ver Seção 5.2.2) e, nestes casos, $\bar{c}_\pi(ij) < Gap$. Consequentemente, estas instâncias não terão nenhum arco fixado por custo reduzido. No entanto, no caso das instâncias *pmed6*, *pmed21*, *pmed33* e *pmed37*, a variável λ_i presente na equação 2-12 fez $\bar{c}_\pi(ij)$ ficar menor do que o *Gap*, não havendo a possibilidade de fixar nenhum arco nestas instâncias.

Na Tabela 5.14 somente $p = 200$ possui $d_{ij} > Gap$. Para as demais instâncias, $d_{ij} < Gap$. Consequentemente, estas demais instâncias não terão nenhum arco fixado por custo reduzido. Mas, no caso de $p = 200$, a variável λ_i presente na equação 2-12 fez $\bar{c}_\pi(ij)$ ficar menor do que o *Gap*, não havendo a possibilidade de fixar nenhum arco.

Instância Nome	(n,p)	Valor do *Gap*	Distância Máxima (i, j, d_{ij})		
pmed6	(200,5)	176,00	62	32	198,00
pmed11	(300,5)	142,00	299	187	134,00
pmed16	(400,5)	182,00	290	117	107,00
pmed17	(400,10)	127,00	307	34	105,00
pmed21	(500,5)	77,00	454	159	91,00
pmed22	(500,10)	137,00	459	393	113,00
pmed26	(600,5)	207,00	462	86	87,00
pmed27	(600,10)	125,00	599	462	91,00
pmed31	(700,5)	303,00	330	274	65,00
pmed32	(700,10)	184,00	699	134	124,00
pmed33	(700,70)	62,00	598	292	74,00
pmed35	(800,5)	288,00	799	319	74,00
pmed36	(800,10)	255,00	799	394	87,00
pmed37	(800,80)	77,00	256	39	78,00
pmed38	(900,5)	281,00	899	238	84,00
pmed39	(900,10)	209,00	899	726	115,00
pmed40	(900,90)	88,00	873	629	69,00

Tabela 5.13: Resultados - Maiores distâncias para as instâncias da classe *OR-Library* onde nenhum arco foi fixado por custo reduzido

p	Valor do *Gap*
10	23431,72
20	20666,78
30	20073,57
40	9121,23
50	11428,44
60	6506,09
70	8435,40
80	9363,91
90	9459,12
100	9342,87
150	5451,65
200	4517,31
(i, j, d_{ij})	(3036;0;4830,80)

Tabela 5.14: Resultados - Maiores distâncias para os valores de p na instância *pcb3038* da classe *TSP-Library* onde nenhum arco foi fixado por custo reduzido

Na Tabela 5.15 podemos observar que todos os valores de p possuem $d_{ij} < Gap$ implicando na não fixação de nenhum arco por custo reduzido nas diferentes configurações de p.

As tabelas 5.16, 5.17, 5.18 e 5.19 apresentam os valores dos *Gaps* e dos $-\gamma s$ utilizados nas fixações para as instâncias da classe *OR-Library* e para as diferentes configurações de p nas instâncias *fl1400*, *pcb3038* e *rl5934* da classe *TSP-Library* onde, em ambas as classes, nenhuma facilidade foi fixada por custo reduzido.

Ao analisarmos a Tabela 5.16, observamos que somente as instâncias *pmed20* e *pmed32* possuem $-\gamma > Gap$. Para as demais instâncias, $-\gamma < Gap$. Como o custo reduzido $\overline{c}_\pi(j)$ da facilidade j é definido pela equação 2-13, o valor máximo para $\overline{c}_\pi(j)$ é $-\gamma$. Para que a facilidade j seja fixado, a condição $\overline{c}_\pi(j) > Gap$ deve ser satisfeita (ver Seção 5.2.2) e, nestes casos, $\overline{c}_\pi(j) < Gap$. Consequentemente, estas instâncias não terão nenhum arco fixado por custo reduzido. No entanto, no caso das instâncias *pmed20* e *pmed32*, o valor da

p	Valor do *Gap*
10	270352,63
20	121531,88
30	135319,76
40	84904,34
50	118390,79
60	81795,15
70	74835,94
80	67368,18
90	71121,42
100	67430,08
150	50125,68
200	38687,07
250	34372,26
300	31379,82
350	28815,40
400	24863,90
450	22134,27
(i, j, d_{ij})	(5703;2944;21539,15)

Tabela 5.15: Resultados - Maiores distâncias para os valores de p na instância *rl5934* da classe *TSP-Library* onde nenhum arco foi fixado por custo reduzido

parcela $\sum_{i \in U} \mu_{ij}$ presente na equação 2-12 fez $\bar{c}_\pi(j)$ ficar menor do que o *Gap*, não havendo a possibilidade de fixar nenhum arco nestas instâncias.

Instância		Valor do *Gap*	$-\gamma$
Nome	(n,p)		
pmed18	(400,40)	60,00	57,00
pmed19	(400,80)	29,00	25,00
pmed20	(400,133)	11,00	16,00
pmed23	(500,50)	51,00	43,00
pmed24	(500,100)	34,00	22,00
pmed25	(600,167)	23,00	14,00
pmed28	(600,60)	67,00	35,00
pmed29	(600,120)	27,00	19,00
pmed30	(600,200)	37,00	11,00
pmed32	(700,10)	184,00	199,00
pmed33	(700,70)	62,00	33,00
pmed34	(700,140)	28,00	16,00
pmed36	(800,10)	255,00	224,00
pmed37	(800,80)	77,00	30,00
pmed40	(900,90)	88,00	26,00

Tabela 5.16: Resultados - Valor do *Gap* e do $-\gamma$ para as instâncias da classe *OR-Library* onde nenhuma facilidade foi fixada por custo reduzido

Examinado a Tabela 5.17, certificamos que as diferentes configurações para p geram um $-\gamma$ menor do que o *Gap*, exceto para *p=30,40 e 50*. Nestas configurações não ocorrem fixações pelos motivos explicados anteriormente.

As diferentes configurações para p na instância *pcb3038*, conforme exposto na Tabela 5.18, geram um $-\gamma$ menor do que o *Gap*. Consequentemente, nestas configurações não ocorrerão nenhuma facilidade fixada por custo reduzido.

Na Tabela 5.19 todas as diferentes configurações para p, exceto $p = 20$, geram um $-\gamma$ menor do que o *Gap*. Consequentemente, nestas configurações não ocorrerão nenhuma facilidade fixada por custo reduzido. Porém, para $p = 20$ não ocorrem fixações devido a contribuição da parcela $\sum_{i \in U} \mu_{ij}$,

p	**Valor do *Gap***	$-\gamma$
30	757,02	979,79
40	493,57	628,61
50	333,34	421,26
60	358,57	308,36
80	306,94	182,74
90	235,55	147,62
100	245,73	111,19
150	267,73	62,99
200	169,36	39,61
250	210,62	25,07
300	159,27	17,72
350	140,78	14,28
400	136,49	11,82
450	191,79	11,08
500	204,38	8,36

Tabela 5.17: Resultados - Valor do *Gap* e do $-\gamma$ para a instância *fl1400* da classe *TSP-Library* onde nenhuma facilidade foi fixada por custo reduzido

p	**Valor do *Gap***	$-\gamma$
20	20666,78	16460,06
30	20073,57	9788,21
40	9121,23	6823,76
50	11428,44	4376,79
60	6506,09	3440,87
70	8435,40	2672,58
80	9363,91	2120,55
90	9459,12	1815,84
100	9342,87	1530,33
150	5451,65	921,49
200	4517,31	612,60
250	4115,86	427,57
300	2855,21	350,44
350	2572,84	288,89
400	2287,24	238,89
450	1908,64	203,31
500	1740,72	175,09
550	1772,07	154,84
600	1872,87	136,52
650	2125,60	121,08
700	1920,84	110,65
750	1849,25	98,30
800	1820,51	87,82
850	1866,67	81,97
900	1693,42	77,10
950	1589,20	74,01
1000	1473,30	69,68

Tabela 5.18: Resultados - Valor do *Gap* e do $-\gamma$ para a instância *pcb3038* da classe *TSP-Library* onde nenhuma facilidade foi fixada por custo reduzido

presente na equação 2-12, na redução do $\overline{c}_\pi(j)$ a um valor menor do que o *Gap*.

p	Valor do *Gap*	$-\gamma$
20	121531,88	147907,92
30	135319,76	78536,18
40	84904,34	51138,23
50	118390,79	33181,64
60	81795,15	27870,21
70	74835,94	22535,04
80	67368,18	18347,47
90	71121,42	15087,12
100	67430,08	12731,53
150	50125,68	7419,59
200	38687,07	4821,91
250	34372,26	3457,12
300	31379,82	2599,44
350	28815,40	2126,66
400	24863,90	1724,65
450	22134,27	1472,00
500	17477,48	1273,48
600	14335,76	1002,83
700	12754,97	759,33
800	11314,43	636,93
900	10052,81	532,19
1000	8544,47	474,32
1100	8343,11	414,12
1200	8824,02	356,77
1300	9536,24	307,57
1400	9619,63	267,00
1500	8390,45	238,76

Tabela 5.19: Resultados - Valor do *Gap* e do $-\gamma$ para a instância *rl5934* da classe *TSP-Library* onde nenhuma facilidade foi fixada por custo reduzido

5.7.2 Complexidade

A Tabela 5.20 apresenta a complexidade de pior caso dos algoritmos duais implementados para este trabalho.

Algoritmo	**Complexidade de Pior Caso**
Dual Ascent	$O(n^2)$ por iteração
Dual Adjustment	$O(\rho n^3)$ por iteração
Dual Scaling	$O(\rho n^3)$ por iteração
Fixação Ativa de Arcos	$O(\rho n^3)$ por iteração
Fixação Ativa de Facilidades	$O(\rho n^3)$ por iteração

Tabela 5.20: Complexidade de pior caso dos algoritmos duais

5.8 Conclusão

O método *Dual Ascent* proposto no final dos anos 70 é bastante competitivo quando comparado aos atuais métodos de resolução para o PMNC. O *Dual Ascent* encontra em um tempo computacional não elevado soluções primais e duais com a grande maioria dos *GAPs* inferiores a 4%. Se compararmos as soluções primais obtidas com os ótimos ou com a melhor solução conhecida, o desvio fica bem menor do que 4%, chegando ao ótima para algumas instâncias. Se compararmos as soluções duais com o ótimo ou com o melhor conhecido, este percentual também reduz consideravelmente. Contudo, se aplicarmos uma busca local na solução primal obtida pelo *Dual Ascent*, o percentual de soluções ótimas encontradas aumenta bastante. Considerando a aplicação do método *Dual Adjustment*, grande parte dos *GAPs* encontrados ficam inferiores a 3%, mas ao custo de se elevar consideravelmente o tempo computacional. O *Dual Adjustment* também foi proposto no final dos anos 70, juntamente com o *Dual Ascent*, e este método pode ser interpretado como uma espécie de busca local. A sua aplicação melhora muito a qualidade das soluções encontradas pelo método *Dual Ascent*.

Obtemos melhoras nas soluções duais com o método *Dual Scaling* apenas para algumas instâncias da classe *OR-Library*. Na versão aleatória 7 instâncias foram beneficiadas e na versão não aleatória 9 instâncias foram beneficiadas. Porém, este benefício é modesto, sendo nulo para as instâncias da classe *TSP-Library*.

Comparando os métodos Fixação Ativa de Arcos (*FAA*) e de Facilidades (*FAF*), houve instâncias onde a *FAA* não eliminou nenhum arco enquanto a *FAF* conseguiu reduzir facilidades. Por outro lado, o inverso também aconteceu, ou seja, *FAF* não reduziu facilidades para uma instância e a *FAA* reduziu arcos nesta mesma instância. Nas instâncias onde ocorreram reduções em ambos os métodos, houve configurações onde a FAA reduziu mais e em outras configurações a FAF foi superior nas reduções. Para as instâncias da classe *OR-Library*, os resultados com ambas as fixações foram equilibrados. Porém, para as instâncias da classe *TSP-Library*, a FAA apresentou resultados superiores a FAF. Contudo, observamos que os valores da solução ótima atingem valores muito grandes quando o número de clientes ultrapassa um milhar e, neste caso, um GAP de 1% do valor ótimo corresponde a números grandes, que são superiores ao diâmetro da região ocupada pelos clientes, quando estes estão no plano e as distâncias consideradas são Euclidianas. Isto dificulta a redução das instâncias a partir de técnicas baseada nos custos reduzidos das variáveis. Para o método *Branch-and-Ascent* apresentado na seção 6.2, a FAF é mais promissora devido a forma como são criados os subproblemas neste método. Na seção 6.2 detalharemos melhor esta questão.