# INFLUÊNCIA DO REORDENAMENTO NO DESEMPENHO DE MÉTODO DE PONTOS INTERIORES PARA PROGRAMAÇÃO LINEAR

**Daniele Costa Silva**
Coordenação de Matemática, COMAT, UTFPR-CP
86300-000, Cornélio Procópio, PR
danielesilva@utfpr.edu.br

**Aurelio R. L. de Oliveira**
Depto de Matemática Aplicada, IMECC, UNICAMP
13083-859, Campinas, SP
aurelio@ime.unicamp.br

## RESUMO

A solução de sistemas lineares é o passo mais caro computacionalmente dos métodos de pontos interiores, tornando a busca por mecanismos que torne este procedimento mais eficiente imprescíndivel para um bom desempenho destes métodos. Neste trabalho, analisaremos a influência de heurísticas de reordenamento na resolução destes sistemas por métodos iterativos.
**PALAVRAS-CHAVE.** Métodos de Pontos Interiores. Sistemas Lineares. Heurísticas de Reordenamento.

## ABSTRACT

The solution of linear systems is the most computationally expensive step of interior point methods, making the search for mechanisms that make this procedure more efficient essential for a good performance of these methods. In this paper, we analyze the influence of reordering heuristics in solving these systems by iterative methods.
**KEYWORDS.** Interior-Point Methods. Linear Systems. Reordering Heuristics.

# 1 Introdução

O trabalho de Karmarkar (1984), revolucionou a área da programação linear, com a publicação de um algoritmo com complexidade polinomial e bom desempenho quando aplicado a problemas práticos. Esse fato desencadeou um novo campo de pesquisa: os métodos de pontos interiores. Desde então, códigos computacionais sofisticados baseados nessas ideias vêem se firmando como alternativas eficientes para solução de problemas de grande porte com estruturas genéricas (Adler et al. (1989), Bocanegra et al. (2007), Czyzyk et al. (1999), Gondzio (1996), Lusting et al. (1992), Oliveira e Sorensen, (2005)).

Cada iteração de um método de pontos interiores envolve a solução de um ou mais sistemas lineares (Gondzio (1996), Lusting et al. (1992), Mehrotra (1992)). Este é o passo mais caro destes métodos do ponto de vista computacional. Em aplicações reais esses sistemas quase sempre possuem dimensões elevadas e alto grau de esparsidade.

Usualmente métodos diretos são utilizados para resolver esses sistemas. A abordagem mais utilizada nas implementações existentes é a fatoração de Cholesky no sistema de equações normais (Adler et al. (1989), Czyzyk et al. (1999), Gondzio (1996), Lustig et al. (1992)). No entanto, por limitações de tempo e memória seu uso torna-se proibitivo em muitos problemas de grande porte, fazendo com que abordagens iterativas sejam mais adequadas (Bergamaschi et al. (2007), Bocanegra et al. (2007), Chai e Toh (2007), Oliveira e Sorensen (2005)).

Na resolução destes sistemas por métodos iterativos a utilização de precondicionadores se faz necessária para que uma convergência rápida possa ser obtida. O desenvolvimento de precondicionadores eficazes para métodos iterativos tem despertado o interesse de diversos pesquisadores ao longo dos últimos anos. Um interessante estudo sobre o estado da arte do precondicionamento em métodos iterativos, de uma forma geral, por ser visto em Benzi (2002). Já no contexto da programação linear, segundo Gondzio (2012), os primeiros precondicionadores bem sucedidos foram desenvolvidos para problemas especialmente estruturados decorrentes da otimização de redes (Resende e Veiga (1993)) e uma classe de precondicionadores aplicáveis em um cenário mais geral foi analisada por Gill et al. (1992), Oliveira (1997), Oliveira e Sorensen (2005).

Em particular, precondicionadores adaptativos a estes sistemas lineares foram aplicados em conjunto com o método dos gradientes conjugados em Bocanegra et al. (2007) e Velazco et al. (2010) obtendo bons resultados computacionais ao serem integrados ao PCx (Czyzyk (1999)). O precondicionador denominado fatoração controlada de Cholesky desenvolvido em Campos e Birkett (1998) é utilizado nas iterações iniciais e o precondicionador separador (Oliveira e Sorensen (2005)) especialmente desenvolvido para as iterações finais é utilizado em uma segunda etapa. No entanto, esta abordagem não utiliza uma técnica que pode ser bastante benéfica para precondicionadores baseados em fatoração incompleta, como é o caso da fatoração controlada de Cholesky, heurísticas de reordenamento. Exemplos de como o reordenamento pode interferir na qualidade deste tipo de precondicionadores podem ser vistos em D'Azevedo et al. (1992), Duff e Meurant (1989), Clift (1995), Notay (1993), Bridson e Tang (2000), Carmo (2005) e Ghidetti (2010).

Neste contexto, a proposta deste trabalho é avaliar a influência de heurísticas de reordenamento na resolução de problemas de programação linear por métodos de pontos interiores.

A organização deste trabalho é a que vem a seguir. Na Seção 2 apresentamos uma breve descrição dos métodos de pontos interiores primal-dual, destacando os sistemas lineares provenientes destes. Na Seção 3 apresentamos os precondicionadores que serão utilizados junto com o método dos gradientes conjugados para solucionar os sistemas oriundos do método de pontos interiores. Na Seção 4 são descritas brevemente as heurísticas utilizadas neste trabalho. Na Seção 5 são apresentados os resultados computacionais. E, finalmente, na Seção 6, nossas conclusões.

# 2 Métodos de Pontos Interiores Primal-Dual

Consideremos o seguinte problema de programação linear na forma padrão:

$$\begin{array}{lll} \text{minimizar} & c^t x & \\ \text{s.a} & Ax & = b, \\ & x & \geq 0, \end{array} \tag{1}$$

onde $A$ $m \times n$ é uma matriz de posto completo $m$, $c, b$ e $x$ são vetores coluna de dimensão apropriada. Associado a este problema temos o problema dual:

$$\begin{array}{lll} \text{maximizar} & b^t y & \\ \text{s.a} & A^t y + z & = c, \\ & z & \geq 0, \end{array} \tag{2}$$

onde $y$ é um vetor coluna de $m$ variáveis livres e $z$ é o vetor coluna de $n$ variáveis de folgas duais. O gap dual é dado por $\gamma = c^t x - b^t y$ que se reduz a $\gamma = x^t z$ para pontos primais e duais factíveis.

Para solucionar (1) os métodos de pontos interiores primais-duais consistem em, a partir de uma tripla $(x^0, y^0, z^0)$, construir uma sequência de triplas, $(x^i, y^i, z^i)$, dada por:

$$\begin{aligned} x^{i+1} &= x^i + \alpha \Delta x^i \\ y^{i+1} &= y^i + \alpha \Delta y^i \\ z^{i+1} &= z^i + \alpha \Delta z^i \end{aligned}$$

onde $i \geq 0$ e $\alpha \in (0, 1]$, que tende para a tripla $(x^*, y^*, z^*)$ solução de (1) e (2).

A direção afim nos métodos de pontos interiores primais-duais, $(\Delta x^i, \Delta y^i, \Delta z^i)$, é dada por:

$$\begin{pmatrix} A & 0 & 0 \\ 0 & A^t & I \\ Z & 0 & X \end{pmatrix} \begin{bmatrix} \Delta x^i \\ \Delta y^i \\ \Delta z^i \end{bmatrix} = \begin{bmatrix} r_p \\ r_d \\ r_c \end{bmatrix}, \tag{3}$$

onde $X = diag(x^i)$, $Z = diag(z^i)$ e os resíduos são dados por $r_p = b - Ax^i$, $r_d = c - A^t y^i - z^i$ e $r_c = -XZe$ e $e$ representa o vetor de uns.

Eliminando as variáveis $\Delta z^i$ de (3), obtemos o seguinte sistema linear, denominado sistema aumentado ou indefinido:

$$\begin{pmatrix} -D & A^t \\ A & 0 \end{pmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = \begin{bmatrix} r_d - X^{-1} r_c \\ r_p \end{bmatrix}, \tag{4}$$

onde: $D = X^{-1} Z$.

Utilizando a primeira equação de (4) para eliminar $\Delta x^i$, obtemos o seguinte sistema de equações normais:

$$(AD^{-1}A^t)\Delta y^i = AD^{-1}(r_d - X^{-1} r_c) - r_p. \tag{5}$$

Uma vez determinada as variáveis $\Delta y^i$ as demais soluções ($\Delta x^i$, $\Delta z^i$) podem ser facilmente calculadas.

A matriz do sistema de equações normais é simétrica e definida-positiva, o que possibilita a resolução de (5) através da fatoração de Cholesky ou do método dos gradientes conjugados.

# 3 Precondicionadores

Como visto na seção anterior, um dos passos dos métodos de pontos interiores primais-duais consiste na solução de um sistema linear simétrico definido positivo. Ao optarmos por resolvê-lo pelo método dos gradientes conjugados é essencial o uso de bons precondicionadores.

Em Oliveira e Sorensen (2005) é desenvolvido o precondicionador separador:

$$D_B^{\frac{1}{2}}B^{-1}(AD^{-1}A^t)B^{-t}D_B^{\frac{1}{2}} = I + D_B^{\frac{1}{2}}B^{-1}ND_N^{-1}N^tB^{-t}D_B^{\frac{1}{2}}$$

.

Onde $PDP^t = \begin{pmatrix} D_N & 0 \\ 0 & D_B \end{pmatrix}$ e $P$ um matriz de permutação das colunas de $A$.

Este precondicionador é específico para os sistemas lineares oriundos dos métodos de pontos interiores e quando aplicado ao método dos gradientes conjugados evita o cálculo da matriz de equações normais e sua fatoração. Este precondicionador apresenta muito bons resultados próximo a solução do problema com uma escolha adequada de $P$ mas não é eficiente nas primeiras iterações.

Por outro lado, a fatoração controlada de Cholesky (Campos e Birkett (1998)) apresenta bons resultados nestas iterações com uma fatoração incompleta que apresenta um alto grau de esparsidade.

Considere $AD^{-1}A^t = LL^t = \widetilde{L}\widetilde{L}^t + R$ onde $L$ é o fator completo de Cholesky, $R$ é a matriz resto da fatoração incompleta $\widetilde{L}$ e a matriz precondicionada $\widetilde{L}^{-1}AD^{-1}A^t\widetilde{L}^{-t} = (\widetilde{L}^{-1}L)(\widetilde{L}^{-t}L)^t$. Definindo $E = L - \widetilde{L}$ temos $\widetilde{L}^{-1}AD^{-1}A^t\widetilde{L}^{-t} = (I + \widetilde{L}^{-1}E)(I + \widetilde{L}^{-t}E)^t$.

O número de iterações do método dos gradientes conjugados está diretamente relacionado com a norma de $R = LE^t + E\widetilde{L}^t$ (Duff e Meurant (1989)). A fatoração controlada de Cholesky foi desenvolvida buscando minimizar $||E||_F^2 = \sum_{j=1}^{m} c_j$, onde $c_j = \sum_{i=1}^{m} |l_{ij} - \widetilde{l}_{ij}|^2$.

Considere $c_j = \sum_{k=1}^{m_j+\eta} |l_{i_kj} - \widetilde{l}_{i_kj}|^2 + \sum_{k=m_j+\eta+1}^{m} |l_{i_kj}|^2$ onde $m_j$ representa o número de elementos não nulos abaixo da diagonal da coluna $j$ de $AD^{-1}A^t$; $\eta$ representa o número adicional de elementos não nulos permitido na fatoração incompleta.

O problema de minimização pode ser abordado pelas seguintes heurísticas:

- Aumentando o parâmetro $\eta$ permitindo maior preenchimento.
- Escolhendo os $m_j + \eta$ maiores elementos de $\widetilde{L}$ em valor absoluto para $\eta$ fixo. Desta forma, os maiores elementos ficam na primeira somatória diminuindo $||E||_F$.

Estes precondicionadores foram agregados ao código PCx resultando em uma abordagem eficiente que obteve melhores resultados onde a fatoração de Cholesky falha ou é mais lenta em diversos problemas, principalmente de grande porte, em que a fatoração é densa (Bocanegra et al. (2007), Velazco et al. (2010)).

# 4 Heurísticas de Reordenamento

A influência do reordenamento no desempenho de métodos iterativos precondicionados tem sido discutida por um grande número de pesquisadores. Em Carmo (2005), é feita uma análise dessa influência sob o método dos gradientes conjugados precondicionado pela fatoração controlada de Cholesky utilizando as seguintes heurísticas de reordenamento: contagem de colunas, Cuthill McKee reverso (Cuthill e McKee (1969), George (1971)) e mínimo grau (George e Liu (1980), George e Liu (1981)). Conclui-se que as heurísticas Cuthill McKee reverso e mínimo grau podem ser benefícas ao método acelerando sua taxa de convergência e reduzindo seu tempo de processamento. Já o uso da contagem de colunas não é recomendado.

Na sequência apresentaremos uma breve descrição das heurísticas Cuthill McKee reverso e mínimo grau.

### 4.1 Cuthill McKee reverso

Cuthill e McKee (1969) propuseram uma heurística de reordenação cujo objetivo é reduzir a largura de banda de uma matriz simétrica. A largura de banda $\beta$ de uma matriz $A$ é dada pela maior distância de um elemento não nulo à diagonal principal, ou seja, $\beta(A) = max_{a_{ij} \neq 0}|i - j|$.

Uma vez que a largura de banda é mantida na fatoração de Cholesky, a reordenação também tende a reduzir o preenchimento.

Posteriormente, George (1971) mostrou que a solução do Cuthill Mckee pode ser melhorada simplesmente invertendo a ordem de numeração. Foi então definida a heurística Cuthill Mckee reverso (RCM), que passou a ser umas das heurísticas mais utilizadas para o problema de minimização da largura de banda.

### 4.2 Mínimo Grau

A heurística mínimo grau é a heurística mais utilizada para reordenação de matrizes simétricas. Seu objetivo é construir uma ordenação que reduza o preenchimento na fatoração dessas matrizes. Para tanto, modela-se a matriz simétrica como um grafo não direcionado, conhecido como grafo de eliminação. A cada iteração elimina-se o vértice de menor grau deste grafo, fazendo com que seus vizinhos se tornem adjacentes.

Apesar da ideia simples a heurística possui um histórico de difícil implementação. As primeiras implementações exigiam um enorme tempo de processamento (George e Liu (1989)). Felizmente, diversas variantes desta heurística já foram desenvolvidas cujos tempos de execução são bastante razoáveis em comparação com o custo da fatoração subseqüente. Bons exemplos dessas variações podem ser vistos em Amestoy et al. (1989), George e Liu (1981), Liu (1985) e Rothberg (1996).

## 5 Experimentos Computacionais

### 5.1 Implementação

O PCx (Czyzyk (1999)) é um solver de programação linear que utiliza o método primal-dual preditor-corretor (Mehrotra (1992)) com múltiplas correcções (Gondzio (1996)) e o método mínimo grau múltiplo (George e Liu (1981)), uma variação da heurística mínimo grau, para o reordenamento da matriz $AD^{-1}A^t$ . Já o código PCx modificado é uma nova versão do PCx em que a solução dos sistemas lineares é realizada de forma iterativa, por meio do método dos gradientes conjugados precondicionado. Uma abordagem de pré-condicionamento híbrida é considerada. Inicialmente a fatoração controlada de Cholesky é utilizada e em uma segunda etapa é feita a troca pelo precondicionador separador. Nessa nova versão não é utilizado o reordenamento de matrizes.

A fim de avaliar a influência de heurísticas de reordenamento no desempenho do método de pontos interiores integramos as heurísticas Cuthill McKee reverso e mínimo grau ao PCx modificado. Visto que estas heurísticas apresentaram bons resultados computacionais junto ao método dos gradientes conjugados precondicionado pela fatoração controlada de Cholesky, principal precondicionador da versão modificada do PCx.

A implementação da heurística mínimo grau é a mesma utilizada no PCx em sua versão original. Para a heurística RCM utilizamos o patch desenvolvido para o PCx cuja implementação encontra-se disponível em https://github.com/lpoo/PCx.

Os testes computacionais foram realizados em uma plataforma D 2,93 GHz Intel Core i7 com 16 GB de memória RAM, rodando o sistema operacional Linux de 64 bits, usando os compiladores gcc e gfortran.

## 5.2 Problemas teste

Neste trabalho, utilizou-se 25 problemas teste, os quais são apresentados na Tabela 1. Nesta tabela as colunas Linhas, Colunas e Nnulos mostram respectivamente, o número de linhas, colunas e elementos não nulos da matriz de restrições de cada problema. Todos estes problemas são de domínio público e podem ser encontrados em MISC[1], NETLIB[2], PDS[3] e QAP (Burkard et al. (1991)). Optamos pelos problemas de maior porte dessas coleções, já que é em problemas de grande escala que o uso de métodos de pontos interiores e métodos iterativos se torna atrativo.

Tabela 1: Problemas testes

| Problema | Linhas | Colunas | Nnulos | Coleção |
|---|---|---|---|---|
| DBIC1 | 34205 | 174457 | 819746 | MISC |
| DBIR1 | 14025 | 38763 | 1015478 | MISC |
| DBIR2 | 14100 | 388838 | 1096585 | MISC |
| F2177 | 9662 | 9908 | 30526 | MISC |
| FROM-LP-FILE | 24617 | 38601 | 156466 | MISC |
| LPL1 | 34037 | 89383 | 263081 | MISC |
| NSCT2 | 15341 | 26512 | 630662 | MISC |
| P10 | 8971 | 17861 | 113591 | MISC |
| RADIO | 57322 | 61268 | 317503 | MISC |
| ROUTING | 20779 | 43019 | 203672 | MISC |
| SC205 | 17612 | 31222 | 61638 | MISC |
| ZZ | 19760 | 36380 | 186590 | MISC |
| KEN-13 | 22365 | 36561 | 82191 | NETLIB |
| KEN-18 | 78538 | 128434 | 297886 | NETLIB |
| OSA60 | 10243 | 243209 | 849356 | NETLIB |
| STOCFOR3 | 15362 | 22228 | 62960 | NETLIB |
| PDS-70 | 111885 | 386238 | 822526 | PDS |
| PDS-80 | 126109 | 430800 | 916852 | PDS |
| PDS-90 | 139741 | 471538 | 1002902 | PDS |
| PDS-100 | 152289 | 498530 | 1060567 | PDS |
| CHR25a | 8149 | 15325 | 53725 | QAP |
| STE36a | 27683 | 13076 | 512640 | QAP |
| STE36b | 27683 | 13076 | 512640 | QAP |
| STE36c | 27683 | 13076 | 512640 | QAP |
| ROU20 | 7359 | 37640 | 152980 | QAP |

## 5.3 Resultados computacionais

Nas tabelas a seguir são apresentados os resultados obtidos através dos testes computacionais. Os campos *** indicam os casos em que não foi possível determinar a solução ótima.

Nos concentramos em analisar a influência do reordenamento na taxa de convergência e no tempo de processamento do método dos gradientes conjugados para solucionar os sistemas lineares provenientes do método preditor corretor (Tabela 2), visto que este é o passo mais caro do método, consumindo em média 65,58% de seu tempo computacional para os problemas testados (sem o reordenamento). E consequentemente, verificarmos como esta mudança interfere em todo o processo de resolução dos problemas,em especial em seu tempo de processamento (Tabela 3).

[1] Miscellaneous LP models, Hungarian Academy of Sciences OR Lab, Online at http://www.sztaki.hu/meszaros/public-ftp/lptestset/mist.
[2] Netlib collection LP test sets, Netlib lp repository, Online at http://www.netlib.org/lp/data.
[3] http://plato.asu.edu/ftp/lptestset/pds/.

Na Tabela 2 a coluna Iterações mostra o número total de iterações do método dos gradientes conjugados sem o uso do reordenamento e as colunas PRGMG e PRGRCM trazem o percentual de redução deste número quando utilizadas as heurísticas mínimo grau e Cuthill McKee reverso, respectivamente. Percentuais negativos indicam aumento. Já as colunas TGSR, TGMG e TGRCM apresentam o tempo total de processamento do método, em segundos, sem o reordenamento, com o reordenamento feito pela heurística mínimo grau e com o reordenamento obtido pela heurística Cuthill McKee reverso, respectivamente.

Tabela 2: Iterações e tempo de processamento (em segundos) - método dos gradientes conjugados

| **Problema** | **Iterações** | **PRGMG** | **PRGRCM** | **TGSR** | **TGMG** | **TGRCM** |
|---|---|---|---|---|---|---|
| DBIC1 | 157702 | -5,10% | **4,72%** | 1301,42 | 1322,41 | **1140,53** |
| DBIR1 | *** | - | *** | *** | **849,98** | *** |
| DBIR2 | 95604 | -11,24% | **1,29%** | 822,41 | 871,26 | **756,01** |
| F2177 | 5054 | **-0,02%** | -0,91% | **2,10** | 2,11 | 2,16 |
| FROM-LP-FILE | 132475 | **2,44%** | -5,09% | 406,76 | **387,89** | 511,66 |
| LPL1 | 236010 | **-112,13%** | -112,54% | 899,19 | **700,5** | 827,69 |
| NSCT2 | 305962 | -29,69% | **-1,80%** | **1912,96** | 2495,76 | 2032,71 |
| P10 | 11978 | **4,31%** | 1,19% | 10,09 | **9,89** | 10,25 |
| RADIO | 9121 | **0,57%** | -6,00% | **35,05** | 36,54 | 39,47 |
| ROUTING | 3935 | -4,35% | **0,00%** | **6,63** | 7,05 | 6,93 |
| SC205 | 17490 | -4,98% | **-0,71%** | 14,17 | 14,76 | **14,14** |
| ZZ | 732 | 0,55% | **0,68%** | 1,12 | **1,11** | 1,13 |
| KEN-13 | 56535 | **10,84%** | 9,35% | 118,87 | **86,13** | 98,29 |
| KEN-18 | 156656 | **3,14%** | -24,76% | 1165,4 | **951,92** | 1511,57 |
| OSA60 | *** | - | *** | *** | **7,69** | *** |
| STOCFOR3 | 76965 | 20,91% | **32,07%** | 107,6 | **97,04** | 98,29 |
| PDS-70 | 74257 | 1,63% | **13,59%** | 1184,86 | 1221,46 | **998,86** |
| PDS-80 | 80751 | 23,29% | **18,17%** | 1446,17 | 1174,97 | **1152,93** |
| PDS-90 | 94874 | 24,13% | **26,57%** | 1887,19 | 1530,76 | **1354,52** |
| PDS-100 | 125800 | 2,21% | **21,55%** | 2670,05 | 2805,59 | **2055,91** |
| CHR25a | 36689 | -46,66% | **-10,17%** | **41,04** | 85,47 | 48,27 |
| STE36a | 306685 | **14,58%** | -24,84% | 3139,41 | **2859,17** | 3971,40 |
| STE36b | 384651 | **15,07%** | *** | 4515,22 | **3567,45** | *** |
| STE36c | *** | - | *** | *** | **4077,12** | *** |
| ROU20 | 52237 | 10,34% | **21,71%** | **283,91** | 371,84 | 346,44 |

Através da Tabela 2 podemos notar que para 12 dos problemas teste com o uso da heurística mínimo grau reduziu-se, em média, 11,07% o número de iterações. E para 3 dos problemas as alterações foram pouco significativas. Já com o uso da heurística Cuthill McKee reverso, a redução foi, em média, de 15,02% para 10 dos problemas testados e para 4 dos problemas teste as alterações não foram significativas. Assim, ambas as heurísticas reduziram ou praticamente mantiveram o número de iterações para a maioria dos problemas. No entanto, também ocorreram casos de aumento bastante significativos, como por exemplo, o caso do problema LPL1 em que o aumento foi superior a 100% para ambas as heurísticas.

Em relação ao tempo de processamento podemos observar que para 9 dos problemas teste, a heurística mínimo grau reduziu o tempo de execução do método dos gradientes conjugados, com destaque para o problema STE36b, no qual a redução foi de 15,80 minutos, e em 5 dos problemas as alterações não foram significativas. Utilizando a heurística Cuthill McKee reverso, também foi possível reduzir o tempo de processamento para 9 dos problemas teste, evidenciando os problemas PDS90 e PDS100, nos quais ocorreram, respectivamente, uma redução de 8,88 e 10,24 minutos; e praticamente manter o tempo para 5 dos problemas teste.

Dentre os casos de aumento do tempo de processamento do método dos gradientes conjugados os mais significativos foram os dos problemas NSCT2 (mínimo grau) e STE36a (Cuthill McKee reverso).

Ao compararmos os dados referentes à iterações com os referentes a tempo, é possível notar que a redução/aumento no número de iterações do gradientes conjugados implica na redução/aumento de seu tempo de processamento. Exceto para o problema LPL1, no qual houve redução de tempo independente do acréscimo no número de iterações e para os problemas PDS70, PDS100 e ROU20 que com o uso da heurística mínimo grau o inverso ocorre, ou seja, reduziu-se o número de iterações, mas aumentou o tempo de processamento. Para o problema ROU20 este fato também se repete com o uso da heurística Cuthill McKee reverso.

A Tabela 3 expõe os dados referentes à influência do reordenamento no tempo total de processamento para resolução dos problemas. Nesta tabela, as colunas TTSR, TTMG e TTRCM mostram o tempo total de processamento, em segundos, quando os problemas são solucionados sem o reordenamento e com o reordenamento feito pela heurística mínimo grau e pela heurística Cuthill McKee reverso, respectivamente. Na Tabela 3 também é exibido o tempo de execução da heurística mínimo grau (coluna TMG) e da heurística Cuthill McKee reverso (coluna TRCM).

Tabela 3: Tempo total de processamento e tempo de reordenamento (em segundos)

| **Problema** | **TTSR** | **TTMG** | **TTRCM** | **TMG** | **TRCM** |
|---|---|---|---|---|---|
| DBIC1 | 1858,03 | 1324,58 | **1258,37** | **4,88** | 52,36 |
| DBIR1 | *** | **927,21** | *** | **3,44** | *** |
| DBIR2 | 892,65 | 920,46 | **809,43** | **3,76** | 11,24 |
| F2177 | **2,71** | 2,79 | 6,99 | **0,02** | 4,21 |
| FROM-LP-FILE | **552,15** | 574,98 | 708,31 | **0,18** | 9,71 |
| LPL1 | 2328,40 | 2109,98 | **1806,31** | **0,27** | 7,08 |
| NSCT2 | 2621,88 | 2671,70 | **2238,49** | **13,69** | 40,18 |
| P10 | **11,23** | 12,38 | 13,68 | **0,10** | 1,67 |
| RADIO | **76,68** | 91,80 | 150,66 | **6,48** | 72,26 |
| ROUTING | **11,13** | 16,29 | 17,96 | **0,05** | 1,86 |
| SC205 | **18,52** | 18,55 | 44,46 | **0,12** | 27,09 |
| ZZ | 8,18 | 9,49 | **7,64** | **0,12** | 0,41 |
| KEN-13 | 129,52 | **94,75** | 107,89 | **0,07** | 2,17 |
| KEN-18 | 1314,68 | **1106,46** | 1637,30 | **0,52** | 22,90 |
| OSA60 | *** | **28,43** | *** | 3,11 | *** |
| STOCFOR3 | 121,21 | 108,13 | **107,98** | **0,01** | 1,64 |
| PDS-70 | 1541,83 | 1871,48 | **1381,93** | 251,89 | **91,45** |
| PDS-80 | 1885,97 | 1902,24 | **1623,22** | 264,95 | **114,01** |
| PDS-90 | 2403,70 | 2297,47 | **1912,10** | 201,71 | **137,45** |
| PDS-100 | 3268,37 | 3288,26 | **2711,35** | 203,11 | **157,28** |
| CHR25a | **52,68** | 94,08 | 54,70 | **0,26** | 0,36 |
| STE36a | 14802,46 | **9575,58** | 17734,17 | 16,62 | **4,39** |
| STE36b | 16591,70 | **15239,28** | *** | **16,66** | *** |
| STE36c | *** | **16198,62** | *** | **16,66** | *** |
| ROU20 | **1217,20** | 1342,67 | 1260,75 | **0,14** | 0,47 |

Por meio da Tabela 3 é possível observar que com o uso da heurística mínimo grau houve redução no tempo total de processamento para 8 dos problemas teste, com destaque para os problemas STE36A e STE36B, nos quais houve uma redução de 87,11 e 22,54 minutos respectivamente. Para 4 dos problemas a alteração foi pouco significativa. Também é importante salientar que os problemas OSA60, DBIR2 e STE36C só convergiram para a solução ótima com o uso desta heurística. Dentre os casos em que ocorreram acréscimo no tempo total os mais significativos foram nos problemas PDS70 e ROU20.

Já a heurística Cuthill McKee reverso reduziu o tempo total para 10 dos problemas teste com destaque para os problemas DBIC1, LPL1 e os problemas da coleção PDS e praticamente manteve o tempo de execução para 3 dos problemas. No entanto, ocorreram casos de acréscimos bastante significativos nos problemas KEN18 e STE36A.

Em relação ao tempo utilizado para efetuar o reordenamento podemos notar que este constitui uma pequena porção do tempo total, exceto para os problemas RADIO, SC205 e F2177 quando utilizada a heurística Cuthill McKee reverso. A heurística mínimo grau foi a que apresentou os melhores resultados de tempo para o reordenamento.

Ao compararmos as Tabelas 2 e 3 notamos que na maioria dos casos em que houve redução no tempo de execução do método dos gradientes conjugados houve redução no tempo total de solução. Mas nem sempre na mesma proporção. Sinal de que o reordenamento interfere em outros processos da solução que não foram contemplados neste trabalho. O que justifica os casos em que mesmo com um tempo de processamento menor para os gradientes conjugados (ou alterações pouco significativas) ocorreram acréscimos no tempo total, ou até mesmo a situação inversa.

## 6 Conclusão

O objetivo deste trabalho era analisar a influência de heurísticas de reordenamento na resolução de sistemas lineares oriundos de métodos de pontos interiores e consequentemente, analisar a influência destas no processo de solução de problemas de programação linear por estes métodos. Nesse sentido, concluímos que heurísticas de reordenamento podem ser bastante benéficas. Visto que dentre os 25 problemas testados em 17 ao menos uma das heurísticas reduziram o tempo total de processamento, e em alguns casos reduções bastante significativas como as destacadas ao longo do texto. Outra contribuição importante é a possibilidade de solucionar problemas que não haviam sido resolvidos por outra abordagem com o uso do reordenamento, como nos casos dos problemas OSA60, DBIR2 e STE36C.

Como trabalho futuro, propomos a análise da influência do reordenamento em outros processos da resolução dos problemas, como por exemplo, a construção dos precondicionadores e o estudo com outras heurísticas.

**Agradecimentos**
Ao CNPq e à FAPESP pelo apoio financeiro.

## Referências

**Adler, I., Resende, M.G.C., Veiga, G. e Karmakar, N.** (1989), An implementation of Karmarkars algorithm for linear programming, *Mathematical Programming*, 44, 297-335.

**Amestoy, P., Davis, T. e Duff, I.**, An approximate minimum degree ordering algorithm, *Technical report TR-94-039*,University of Florida, 1994.

**Benzi, M.** (2002), Preconditioning techniques for large linear systems: a survey, *Journal of Computational Physics*, 182, 418-477.

**Bergamaschi, L., Gondzio, J., Venturin, M. e Zilli, G.** (2007), Inexact constraint preconditioners for linear systems arising in interior point methods, *Computational Optimization and Applications*, 36, 137-147.

**Bocanegra, S., Campos, F. F. e Oliveira, A. R. L.** (2007), Using a hybrid preconditioner for solving large-scale linear systems arising from interior point methods, *Computation Optimization and Applications*, 36, 149-164.

**Bridson, R. e Tang, W.** (2000), A structural diagnosis of some IC orderings, *SIAM Journal on Scientific Computing*, 22 (5), 1527-1532.

**Burkard, R. S., Karisch, S. e Rendl, F.** (1991), Qaplib - a quadratic assignment problem library, *European Journal of Operations Research*, 55, 115-119.

**Campos, F. F. e Birkett, N. R. C.** (1998),An efficient solver for multiright hand side linear systems based on the CCCG(*eta*) method with applications to implicit time-dependent partial differential equations, *SIAM Journal on Scientific Computing*, 19, 126-138.

**Carmo, F. C.** Análise da influência de algoritmos de reordenação de matrizes esparsas no desempenho do método CCCG(*eta*), *Dissertação de Mestrado*, Universidade Federal de Minas Gerais, Belo Horizonte, 2005.

**Chai, J. S. e Toh, K. C.** (2007),Preconditioning and iterative solution of symmetric indefinite linear systems arising from interior point methods for linear programming, *Computational Optimization and Applications*, 36, 221-247.

**Clift, S. S. e Tang, W.** (1995), Weighted graph based ordering techniques for preconditioned conjugate gradient methods, *BIT Numerical Mathematics*, 35 (1), 30-47.

**Cuthill, E. e McKee, J.** (1969), Reducing the bandwidth of sparse symmetric matrices, *In Proceedings of the 24th National Conference ACM*, 157-172.

**Czyzyk, J., Mehrotra, S., Wagner, M. e Wright, S. J.** (1999), PCx an interior point code for linear programming, *Optimization Methods Software*, 11-2, 397-430.

**D'Azevedo, E. F., Forsyth, P. A. e Tang W.** (1992), Ordering methods for preconditioned conjugate gradient methods applied to unstructured grid problems, *SIAM Journal on Matrix Analysis and Applications*, 13 (3), 944-961.

**Duff, I. S. e Meurant G. A.** (1989), The effect of ordering on preconditioned conjugate gradients, *BIT*, 29, 635-657.

**George J. A.**, Computer implementation of the finite element method, *PhD Thesis*, Departament of Computer Science, Stanford University, 1971.

**George A. e Liu, J.**, (1980), A fast implementation of the minimum degree algorithm using quotient graphs, *ACM Transactions on Mathematical Software*, 6 (3), 337-358.

**George, A. e Liu, J.**,*Computer solution of large sparse positive definite systems*, Prentice Hall Series in Computational Mathematics, New Jersey, 1981.

**George, A. e Liu, J.**, (1989), The evolution of the minimum degree ordering algorithm, *SIAM Review*, 31 (1), 1-19.

**Ghidetti, K., Catabriga, L., Boeres, M. C. e Rangel, M. C.**, (2010), A study of the influence of sparse matrices reordering algorithms on Krylov-type preconditioned iterative methods, *Mecánica Computacional*, 29, 2323-2343.

**Gill, P. E., Murray, M. A., Ponceléon, D. B. e Saunders, M. A.**, (1992), Preconditioners for indefinite systems arising in optimization, *SIAM Journal on Matrix Analysis and Applications*, 13, 292-311.

**Gondzio, J.** (1996), Multiple centrality corrections in a primal-dual methods for linear programming, *Computation and Optimization and Applications*, 6, 137-156.

**Gondzio, J.** (2012), Interior point methods 25 years later, *European Journal of Operational Research*, 218, 587-601.

**Karmarkar, N. K.** (1984), A new polynomial?time algorithm for linear programming, *Combinatorica*, 4, 373-395.

**Liu, J.** (1985), Modification of the minimum degree algorithm by multiple elimination, *ACM Transactions on Mathematical Software (TOMS)*, 11 (2), 141-153.

**Lustig, I. J., Marsten, R. E. e Shanno, D. F.** (1992), On implementing Mehrotra's predictor-corrector interior point method for linear programming, *SIAM Journal on Optimization*, 2, 435-449.

**Mehrotra, S.** (1992), On implementation of a primal-dual point method, *SIAM Journal on Optimization*, 2, 575-601.

**Notay, Y.**, Ordering methods for approximate factorization preconditioning, *Report IT/IF/14-11*, Université Libre de Bruxelles, 1993.

**Oliveira, A. R. L.**, A new class of preconditioners for large-scale linear systems from interior point methods for linear programming, *PhD Thesis*, Department of Computational and Applied Mathematics, Rice University, Houston, 1997.

**Oliveira, A. R. L. e Lyra, C.** (1991), Implementação de um método de pontos interiores para programação linear, *SBA: Controle Automação*, 3, 370-382.

**Oliveira, A. R. L. e Sorensen, D. C.** (2005), A new class of preconditioners for large-scale systems from interior point methods for linear programming, *Linear Algebra Its Applications*, 394, 1-24.

**Resende, M. G. C. e Veiga G.** (1993), An implementation of the dual affine scaling algorithm for minimum cost flow on bipartite uncapacitated networks, *SIAM Journal on Optimization*, 3, 516-537.

**Rothberg, E.**, Ordering sparse matrices using approximate minimum local fill, *Technical report*, Silicon Graphics, Inc. Mountain View, CA 94043, 1996.

**Velazco, M. I., Oliveira, A. R. L. e Campos, F. F.** (2010), A note on hybrid preconditioner for solving large-scale normal equations arising from interior point methods, *Optimization Methods and Software*, 25 2 , 321-332.