

Uma heurística baseada em valores de Ritz para a troca de fase do precondicionador híbrido

Petra Maria Bartmeyer

Unicamp
Cidade Universitária "Zeferino Vaz", Campinas-SP
ra160903@ime.unicamp.br

Silvana Bocanegra

UFRPE
Rua Dom Manoel de Medeiros, Recife-PE
silvana.bocanegra@gmail.com

Aurelio Ribeiro Leite de Oliveira

Unicamp
Cidade Universitária "Zeferino Vaz", Campinas-SP
aurelio@ime.unicamp.br

RESUMO

O método preditor corretor de Mehrotra é uma variante do método de pontos interiores para programação linear. A cada iteração deste método é necessária a resolução de dois sistemas lineares: o primeiro determina a direção preditora e o segundo a direção corretora. A maior parte do custo computacional do método consiste na solução desses sistemas lineares. Neste trabalho está sendo utilizado o método de gradientes conjugados precondicionado para resolver tais sistemas lineares. O precondicionamento é realizado de forma híbrida: nas primeiras iterações é utilizada a fatoração controlada de Cholesky e nas iterações finais o precondicionador separador. O ponto chave do precondicionador híbrido é determinar o momento para a troca. Estamos propondo o uso do número de condição, estimado por meio dos valores de Ritz, para o desenvolvimento de uma heurística para troca dos precondicionadores.

PALAVRAS CHAVE. valores de Ritz, precondicionador híbrido, método de pontos interiores.

ABSTRACT

The Mehrotra predictor corrector method is a variant of the interior point method for linear programming. At each iteration of this method is necessary to solve two linear systems: the first determines the predictor direction and the second the corrector direction. The solving of this linear systems holds most of the computational cost. In this work the preconditioned conjugate gradients method is applied to solve these linear systems. The preconditioning is performed by the hybrid preconditioner, where on the first iterations is used the controlled Cholesky factorization and on the later iterations the splitting preconditioner is applied. The key point of the hybrid preconditioner is to determine the moment for the phase exchange. We are proposing a heuristic for the change of preconditioners based upon the matrix condition number, which can be estimated by Ritz values.

KEYWORDS. Ritz values, hybrid preconditioner, interior point method.

1. Introdução

Um problema de programação linear de grande porte, em geral, é resolvido utilizando métodos de pontos interiores. A maior parte do custo computacional destes métodos envolve a determinação das direções de busca pois, a cada iteração um sistema linear esparso e mal condicionado é resolvido. A resolução desses sistemas lineares pode ser feita através de métodos diretos ou iterativos. Métodos diretos fornecem uma solução exata porém com custo computacional superior, já os métodos iterativos dependem das condições de convergência para obter a solução, mas demandam menos espaço de memória para armazenamento. Nesse trabalho é utilizado o método de gradientes conjugados para resolver tais sistemas lineares, pois ele preserva a esparsidade original da matriz e tem menor custo computacional, entretanto sua convergência está relacionada com o número de condição da matriz. Uma forma de melhorar o condicionamento da matriz é o preconditionamento.

Determinar um preconditionador que tenha bom desempenho em todas as iterações de pontos interiores é difícil pois o condicionamento da matriz vai se deteriorando no decorrer das iterações. Assim, abordagens híbridas com a combinação de preconditionadores têm apresentado resultados promissores. Nessas abordagens se assume a existência de fases durante as iterações de pontos interiores e em cada fase é usado um tipo de preconditionador. Nesse trabalho, como em [Bocanegra et al., 2007; Fontova et al., 2011], utiliza-se o preconditionador da Fatoração Controlada de Cholesky (FCC), que tem seu melhor desempenho nas iterações iniciais do método de pontos interiores, e o preconditionador separador, que apresenta melhor desempenho nas iterações finais. A escolha do momento da troca de fase é essencial para o sucesso das abordagens híbridas e, se bem feita, pode reduzir o tempo computacional e/ou o número de iterações. Algumas heurísticas têm sido propostas para determinar a troca de fases em métodos de pontos interiores que usam preconditionadores híbridos [Bocanegra, 2005; Bocanegra et al., 2007; Fontova et al., 2011]. No entanto, nenhuma dessas heurísticas leva em consideração o número de condição da matriz dos coeficientes, o qual está relacionado diretamente com o desempenho do preconditionador.

Na norma-2, o número de condição pode ser calculado usando o maior e o menor autovalores da matriz. Porém, o cálculo de autovalores é um procedimento com custo computacional elevado e portanto inviável quando se deseja reduzir o tempo de solução. A proposta desse trabalho é utilizar valores de Ritz para estimar o número de condição da matriz dos coeficientes. Os valores de Ritz são aproximações dos autovalores e podem ser obtidos a partir das projeções em subespaços de Krylov [Paige et al., 1995].

Esse trabalho está organizado da seguinte forma. Na Seção 2 é apresentada uma revisão sobre o método de pontos interiores. Na Seção 3 é destacado o método preditor-corretor de Mehrotra. Na Seção 4 estão descritos os métodos comumente usados para solução de sistemas lineares oriundos de pontos interiores: o método de gradiente conjugado preconditionado e a fatoração de Cholesky. A Seção 5 é destinado ao preconditionador híbrido, descrevendo o preconditionador FCC e o preconditionador separador. Fazem parte da Seção 6 as heurísticas de mudança de fase descritas na literatura e a proposta de heurística desse trabalho. A Seção 7 apresenta os testes numéricos realizados. Na Seção 8 estão as conclusões e perspectivas de trabalhos futuros. E por fim as Referências Bibliográficas.

2. Métodos de Pontos Interiores

O desenvolvimento de métodos de pontos interiores para a resolução de problemas de programação linear de grande porte se tornou um campo de pesquisa atrativo após a publicação do trabalho [Karmarkar, 1984]. Esses métodos tem como diferencial a complexidade do algoritmo, que é polinomial no pior caso, permitindo resolver problemas com dimensões maiores que os resolvidos pelo método Simplex.

Considere o seguinte problema de programação linear:

$$\begin{aligned} & \min c^T x \\ \text{sujeito a} & \quad Ax = b \\ & \quad x \geq 0, \end{aligned}$$

onde $A \in \mathbb{R}^{m \times n}$ é uma matriz de posto completo, $x \in \mathbb{R}^n$, $c \in \mathbb{R}^n$ e $b \in \mathbb{R}^m$. Esse problema é conhecido como problema primal e pode ser considerado no formato padrão. É possível trabalhar também com o problema dual:

$$\begin{aligned} & \max b^T y \\ \text{sujeito a} & \quad A^T y + z = c \\ & \quad z \geq 0, \end{aligned}$$

onde $y \in \mathbb{R}^m$ e $z \in \mathbb{R}^n$.

Existem métodos de pontos interiores primal, dual e primal-dual, dependendo dos problemas com os quais eles trabalham. As restrições dos problemas primal e dual geram um conjunto convexo, dentro desse conjunto estão os pontos factíveis e fora dele os infactíveis. Assim, um método factível trabalha com pontos dentro do conjunto convexo delimitado pelas restrições do problema e portanto esse ponto satisfaz todas as restrições. Já um método infactível trabalha com pontos que não satisfazem uma ou mais restrições e assim não se encontram no interior do conjunto.

A existência de métodos infactíveis se dá pela dificuldade de determinar um ponto factível em alguns problemas. Nesse trabalho está sendo utilizado o método predictor-corretor de [Mehrotra, 1992] que é um método primal-dual infactível.

3. Método Predictor-Corretor de Mehrotra

Em método de pontos interiores primal-dual as direções de busca são determinadas a partir da aplicação do Método de Newton às condições de otimalidade de Karush-Kuhn-Tucker (KKT). No método predictor-corretor de Mehrotra são resolvidos dois sistemas lineares a cada iteração: um para encontrar a direção preditora e outro para a direção corretora.

A direção preditora, também conhecida como direção afim-escala, consiste em um passo puro na direção de Newton. Já a direção corretora, ou direção de centragem, tenta criar uma trajetória mais central. A direção final (Δ) é composta pela direção preditora (Δ_a) e a corretora (Δ_c), as quais são obtidas resolvendo os seguintes sistemas lineares:

$$\begin{bmatrix} A & 0 & 0 \\ 0 & A^T & I \\ Z & 0 & X \end{bmatrix} \begin{bmatrix} \Delta_a x^k \\ \Delta_a y^k \\ \Delta_a z^k \end{bmatrix} = \begin{bmatrix} r_p \\ r_d \\ r_a \end{bmatrix} \quad (1)$$

e

$$\begin{bmatrix} A & 0 & 0 \\ 0 & A^T & I \\ Z & 0 & X \end{bmatrix} \begin{bmatrix} \Delta_c x^k \\ \Delta_c y^k \\ \Delta_c z^k \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ r_c \end{bmatrix}, \quad (2)$$

onde o resíduo primal é dado por $r_p = b - Ax$, o resíduo dual é dado por $r_d = c - A^T y - z$, o resíduo afim é dado por $r_a = -XZe$ e o resíduo da complementaridade relaxado é dado por $r_c = \mu e - XZe - \Delta X \Delta Z e$.

Assim, a direção final será $\Delta = \Delta_a + \Delta_c$. Os sistemas (1) e (2) podem ser reescritos como o sistema aumentado ou ainda o sistema de equações normais. Enquanto a matriz do sistema aumentado é esparsa, simétrica e indefinida, a matriz do sistema de equações normais é mais densa, simétrica e definida positiva. A escolha entre o sistema aumentado ou as equações normais vai determinar o uso de métodos diretos, como a fatoração de Cholesky, ou métodos iterativos, como o método de gradientes conjugados. Cada um deles apresenta aspectos positivos e negativos conforme o sistema a ser resolvido. Como a resolução desses sistemas lineares representa a maior parte do esforço computacional dos métodos de pontos interiores, o uso de métodos iterativos é adequado, pois esses preservam a esparsidade e requerem menos espaço para armazenamento.

4. Método de Gradientes Conjugados

O método de gradientes conjugados foi proposto por [Hestenes e Stiefel, 1952]. Esse método ao ser aplicado a matrizes simétricas positivas definidas e em aritmética exata, converge em n passos, onde n é o número de autovalores distintos da matriz A . Contudo, na aritmética de ponto flutuante isso não é necessariamente verdade, uma vez que a convergência também depende do número de condição da matriz, do espectro da matriz e da existência de agrupamento dos autovalores. Uma técnica para melhorar o condicionamento da matriz e portanto melhorar a convergência do método de gradientes conjugados é o preconditionamento. Nesse trabalho é utilizado o preconditionador híbrido, o qual é descrito a seguir.

4.1. Precondicionamento

O preconditionamento é uma estratégia para melhorar as características de convergência da solução de um sistema linear. Consiste em multiplicar a matriz do sistema $Ax = b$ por uma matriz preconditionadora, conhecida como preconditionador. Esse preconditionamento pode ocorrer multiplicando a matriz pela direita, pela esquerda ou por ambos os lados.

Neste trabalho será utilizada uma estratégia de preconditionamento híbrida, que envolve duas técnicas diferentes para a construção da matriz preconditionadora. O uso de técnicas de preconditionamento distintas visa encontrar preconditionadores mais adequados ao número de condição da matriz, uma vez que esse apresenta grandes mudanças no decorrer das iterações do método de pontos interiores.

4.2. Precondicionador Híbrido

O preconditionador híbrido é uma combinação entre a FCC, que tem bom desempenho nas iterações iniciais, e o preconditionador separador, que foi desenvolvido de forma específica para matrizes mal-condicionadas e assim apresenta melhor desempenho nas últimas iterações de pontos interiores.

4.2.1. Fatoração Controlada de Cholesky (FCC)

O preconditionador FCC, desenvolvido por [Campos, 1995], é uma variação da fatoração incompleta de Cholesky [Lin e Moré, 1999; Kershaw, 1978]. A FCC é construída por colunas através de uma heurística de minimização da norma de Frobenius entre o fator exato de Cholesky (L) e o fator controlado de Cholesky (\tilde{L}). Na FCC o fator de Cholesky será calculado de forma parcial e o número de entradas não nulas será determinado previamente pela análise de memória disponível. Nesse caso, antes de iniciar a fatoração são analisados quantas entradas podem ser criadas em cada coluna do preconditionador.

Sendo $E = L - \tilde{L}$, a FCC minimiza a norma de Frobenius de E .

$$\begin{aligned} \min \|E\|_F^2 &= \min \sum_{j=1}^m |l_{ij} - \tilde{l}_{ij}|^2 \\ &= \min \sum_{k=1}^{t_j+\eta} |l_{ikj} - \tilde{l}_{ikj}|^2 + \sum_{k=t_j+\eta+1}^m |l_{ikj}|^2 \end{aligned}$$

- \tilde{l}_{ij} maiores valores de l em módulo;
- t_j entradas não nulas da coluna j da matriz L ;
- $\eta \in \mathbb{Z}$ número de entradas que podem ser criadas em cada coluna do preconditionador.

Portanto, para valores de η positivos o preconditionador se aproxima do fator completo de Cholesky, enquanto que para valores negativos de η o preconditionador tende a ser diagonal. O preconditionador pode ser construído por colunas evitando o cálculo das equações normais.

4.2.2. Preconditionador Separador

O preconditionador separador foi proposto por [Oliveira e Sorensen, 2005]. Ele foi desenvolvido para matrizes mal condicionadas e assim tem bom desempenho na iterações finais do método de pontos interiores. Entretanto, quando aplicado a matrizes não tão mal condicionadas, seu desempenho é ruim. É construído da seguinte forma:

Seja $A \in \mathbb{R}^{m \times n}$ uma matriz de posto completo. O preconditionador consiste em encontrar uma partição $B \in \mathbb{R}^{m \times m}$ tal que $A = [B \ N]$, em que B tenha colunas linearmente independentes. Dessa forma é possível particionar o sistema:

$$A\Theta A^T = B\Theta_B B^T + N\Theta_N N^T.$$

Multiplicado a esquerda por $\Theta_B^{-\frac{1}{2}} B^{-1}$ e à direita pela transposta dessa matriz:

$$\Theta_B^{-\frac{1}{2}} B^{-1} (A\Theta A^T) B^{-T} \Theta_B^{-\frac{1}{2}} = I + WW^T,$$

onde $W = \Theta_B^{-\frac{1}{2}} B^{-1} N \Theta_N^{\frac{1}{2}}$.

Se B for escolhido de forma adequada, próximo da solução, as normas de Θ_B^{-1} e Θ_N serão pequenas. Para determinar B usa-se minimização da norma de W . As colunas linearmente independentes que formaram B serão escolhidas sendo as m primeiras colunas com norma-1 mínima da matriz $A\Theta$. O número de colunas linearmente dependentes visitadas pode elevar o custo computacional de forma significativa.

5. Troca de fases

Como o desempenho do preconditionador depende do espectro da matriz, o número de condição pode ser um bom indicativo para a troca de fase. Porém, o número de condição da matriz sofre grandes variações durante as iterações de pontos interiores e, entre diferentes problemas, pode apresentar variações superiores a 10^{10} , dificultando a criação de uma heurística para troca de fase.

5.1. Número de Condição

O número de condição de uma matriz, segundo a norma p , é definido como: $k(A)_p = \|A^{-1}\|_p \|A\|_p$. Ele é uma medida de estabilidade do sistema linear, ou seja, quanto a solução é afetada por uma mudança pequena na matriz do sistema. Matrizes bem condicionadas tem número de condição próximo da unidade. Nesse trabalho usaremos a norma-2 ao determinar o número de condição, o qual pode ser calculado usando os autovalores da matriz, conforme Teorema 5.1 e Teorema 5.2.

Teorema 5.1 *Seja $A \in \mathbb{M}_n$ uma matriz simétrica com entradas reais. Então:*

$$\|A\|_2 = \lambda_{Max} \quad \|A^{-1}\|_2 = \frac{1}{\lambda_{Min}}$$

sendo $\lambda_{Max} = \max|\lambda_j|$ e $\lambda_{Min} = \min|\lambda_j|$, onde $\lambda_1, \dots, \lambda_n$ são os autovalores de A .

Teorema 5.2 *Seja $A \in \mathbb{M}_n$ uma matriz simétrica inversível. Então:*

$$k_2(A) = \frac{\lambda_{Max}}{\lambda_{Min}}$$

onde λ_{Max} é o maior autovalor de A e λ_{Min} o menor.

A demonstração dos teoremas 5.1 e 5.2 estão em [Luenberger e Ye, 1984].

5.2. O método de Lanczos e a relação entre os coeficientes do método de gradientes conjugados e os valores de Ritz

O método de Lanczos é um método de ortogonalização que permite encontrar uma base ortonormal para um espaço, em especial para um espaço de Krylov. Os espaços de Krylov são construídos a partir de uma matriz A e um vetor r . Define-se o espaço de Krylov de dimensão i , como $K^i(A, r) = r, Ar, \dots, A^{i-1}r$. Nesse trabalho o resíduo é definido por $r = b - Ax$.

Como o método de gradientes conjugados é um método de espaços de Krylov, é possível determinar uma relação entre os coeficientes do método de gradientes conjugados e os valores de Ritz [Van der Sluis e van der Vorst, 1986].

Quando a matriz do sistema linear é simétrica [Kushida, 2015]:

$$AV_i = V_{i+1}T_{i+1,i}, \quad (3)$$

onde T é uma matriz tridiagonal e V é uma base ortonormal para o espaço de Krylov. Multiplicando a equação (3) à esquerda pela hermitiana de V_i , tem-se:

$$\begin{aligned} V_i^H AV_i &= V_i^H V_{i+1} T_{i+1,i} \\ &= I_{i,i+1} T_{i+1,i} \\ &= T_{i,i}. \end{aligned}$$

Suponha agora um autopar (z, θ) , onde $\|z\| = 1$, e define-se um resíduo do autopar como $r = Az - \theta z$. Deseja-se encontrar um valor tal que:

$$Az - \theta z = r \perp \mathbb{K}^i(A, v), \quad (4)$$

para um determinado vetor inicial $v \in \mathbb{R}^n$. Em particular pode-se considerar $v = r_0$. O vetor z será reescrito como combinação linear dos elementos da base e portanto $z = V_i y$. Usando as relações (3) e (4):

$$V_i^H (AV_i y_j - \theta_j^i V_i y_j) = 0,$$

ou ainda,

$$T_{i,i}y_j = \theta_j^i y_j.$$

Portanto, o valor de θ_j^i , que é o autovalor de $T_{i,i}$, será o valor de Ritz da matriz A e o vetor de Ritz dela é dado por $V_i y_j$. A dimensão de $T_{i,i}$ aumenta a cada iteração do método de gradientes conjugados e assim a quantidade valores de Ritz calculados também aumenta.

6. Heurísticas para troca de fase

Neste trabalho serão comparadas três heurísticas para a troca de fase, a proposta por [Bocanegra et al., 2007], a de [Fontova et al., 2011] e a nova que estamos propondo, baseada em valores de Ritz. Nas três heurísticas o valor inicial para η é estimado a partir da média de elementos não nulos em cada linha da matriz, definidos por [Bocanegra, 2005]. Assim:

$$Mel = \frac{|A\Theta A^T|}{m},$$

onde $|A\Theta A^T|$ representa o número de elementos não nulos da matriz e m é a dimensão da matriz.

A heurística proposta em [Bocanegra et al., 2007] está descrita no Algoritmo 1. A troca de fases ocorre quando o gap inicial for reduzido por um fator de 10^6 , o qual indica proximidade da solução ótima, ou o gradiente conjugado preconditionado pela FCC estiver perdendo eficiência e o η já tiver atingido o valor máximo estabelecido (η_{max}). O valor η_{max} é atribuído de acordo com a memória disponível.

Entrada: gap ($x_k z_k$), $gap_0 = x_0^T z_0$, iterGC (iterações do gradiente conjugado), η , η_{max}

Saída: η , troca (variável booleana, 1 indica troca)
se $gap < 10^{-6} gap_0$ ou $(2 * iterGC \leq m$ e $\eta = \eta_{max})$ **então**
 | troca=1
 | Troca de fase
fim
se $4 * iterGC \geq m$ e $\eta < \eta_{max}$ **então**
 | $\eta = \eta + 10$
fim

Algoritmo 1: Heurística de troca de fase [Bocanegra et al., 2007].

Em [Fontova et al., 2011] é proposta uma heurística de mudança de fase que leva em consideração o número de iterações do método de gradientes conjugados ($iterGC$), a dimensão da matriz (m) e no valor de η_{max} . Esta heurística está descrita no Algoritmo 2.

Entrada: η_{max} , η , iterGC, m

Saída: η , troca

se $6 * iterGC \geq m$ **então**

| **se** $\eta < \eta_{max}$ **então**
 | | $\eta \leftarrow \eta + 10$

| **fim**

| **senão**

| | troca=1

| | Troca de fase

| **fim**

fim

Algoritmo 2: Heurística de troca de fase [Fontova et al., 2011].

Nesse trabalho propomos uma nova heurística na qual a troca de fase utiliza a estimativa do número de condição da matriz (κ), o número de iterações do método de gradientes conjugados ($iterGC$) e a dimensão da matriz (m). O número de condição por sua vez será estimado usando o maior e o menor valor de Ritz da matriz preconditionada, portanto $k_2(A) \approx \frac{\theta_{max}}{\theta_{min}}$.

A nova heurística está apresentada no Algoritmo 3. A troca de fase ocorre se o número de condição for superior a 10^β e a $c \cdot 10^\gamma$, onde $\beta, \gamma \in \mathbb{N}$ e $c \in \mathbb{R}$. Os valores de c, β e γ foram estimados empiricamente. Foram testados valores de $c = 1, 3, 5$ e $7, \beta = 4, 5$ e 6 e $\gamma = 4, 5, 6$ e 7 . Os melhores valores foram obtidos com $c = 5, \gamma = 5$ e $\beta = 5$.

Entrada: $iterGC, \eta, m, \kappa$
Saída: η , troca
se $\kappa > 10^\beta$ **então**
 se $\kappa > c * 10^\gamma$ **então**
 troca=1
 Troca de fase
 fim
 se $3 * iterGC > m$ **então**
 $\eta \leftarrow \eta + 10$
 fim
fim

Algoritmo 3: Heurística proposta para troca de fases.

Número de condição: O número de condição apresenta relação com o número de iterações do método de gradientes conjugados. É esperado que quanto maior o número de condição mais iterações do método de gradientes conjugados sejam necessárias para a convergência da solução. A Figura 1 ilustra essa situação.

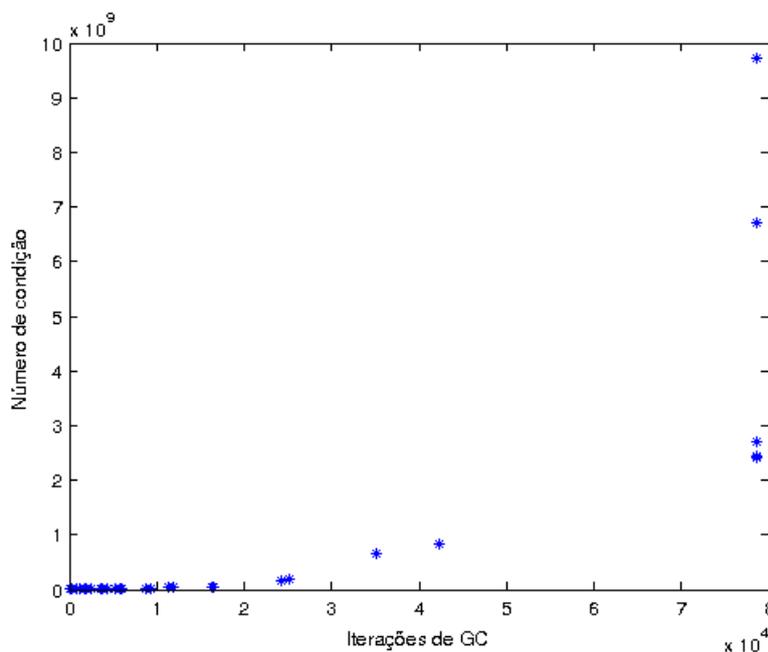


Figura 1: Instância ken-18 - o aumento no número de condição está relacionado com o aumento no número de iterações do método de gradiente conjugados. Resultado usando $\gamma = 5, tola = 10^{-8}, \beta = 5$ e $c = 5$.

O valor de η : Quando valores maiores de η são permitidos a qualidade do preconditionador tende a aumentar e portanto menos iterações do método de gradientes conjugados são necessárias. Esse comportamento se verifica antes da troca de fase e está representado na Figura 2.

Para maiores valores de η o número de iterações do método de gradiente conjugados é menor.

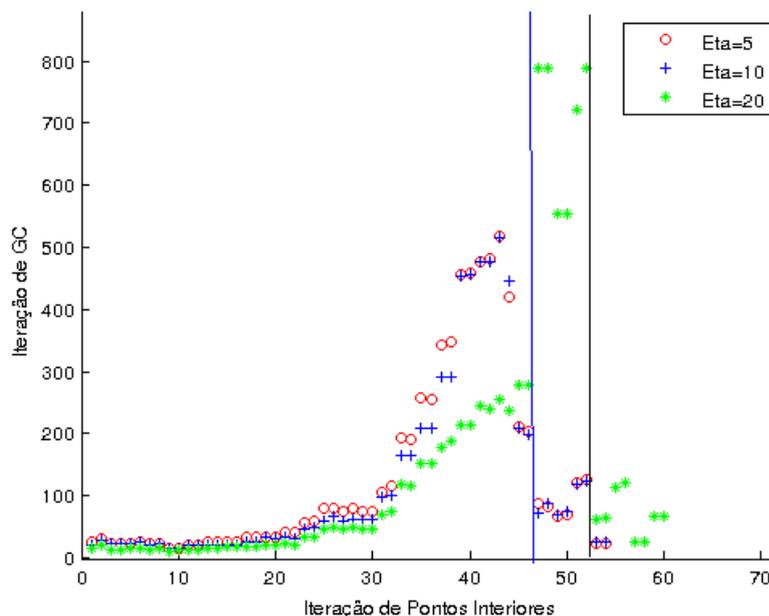


Figura 2: Instância 25fv47 - influência do valor de η no número de iterações do método de gradientes conjugados. A linha azul representa o momento de troca de fase para $\eta = 5$ e 10 e a linha preta para $\eta = 20$.

7. Resultados Computacionais

Os experimentos numéricos foram realizados utilizando a versão do código PCx modificada [Bocanegra, 2005]. O PCx [Czyzyk et al., 1999] é um código que utiliza o método preditor-corretor de pontos interiores para resolução de problemas de programação linear. Foi desenvolvido por *Optimization Technology Center*, em parceria com *Argonne National Laboratory* e a *Northwestern University*. Os problemas utilizados foram retirados das bibliotecas [Browne et al., 1995; Maros e Mészáros, 1999]. Os experimentos computacionais foram realizados em um *desktop* com processador Intel i7 com 8Gb de memória, em ambiente Linux.

Na versão utilizada os sistemas lineares são resolvidos a partir do método de gradientes conjugados preconditionados e o preconditionamento é feito de forma híbrida, sendo utilizado a FCC nas primeiras iterações e o preconditionador separador nas últimas [Bocanegra, 2005]. A mudança de preconditionador é realizada a partir do Algoritmo 3.

Os valores de Ritz foram calculados com a subrotina *dsterf* do LAPACK, uma biblioteca de rotinas para resolução de problemas de álgebra linear [Anderson et al., 1999]. Essa rotina calcula todos os autovalores da matriz T . A qualidade da aproximação entre os autovalores da matriz e os valores de Ritz foi analisada para problemas pequenos. Os autovalores foram obtidos a partir da rotina *eig* do *software* Matlab e a comparação foi feita a partir do erro relativo.

Os parâmetros η e a tolerância do método de gradientes conjugados (*tola*) foram alterados. Foram testadas valores de *tola* = -4, -6 e -8 e $\eta = 0, 5, 10$ e 20. Os melhores desempenhos foram encontrados para *tola* = 10^{-8} e $\eta = 5$.

Na Tabela 1 as colunas Heurística 1, Heurística 2 e Heurística proposta representam os resultados referentes as heurísticas apresentadas nos algoritmos 1, 2 e 3, respectivamente. São comparados o tempo computacional total em segundos (colunas Tempo(s)) e a iteração em que ocorreu a troca de fases (colunas Troca).

Note que o problema ken-18 mesmo não trocando de fase segundo a heurística proposta apresenta tempo computacional inferior as demais, sendo 36 vezes menor do que na heurística 1 e

Problema	Heurística 1		Heurística 2		Heurística proposta	
	Troca	Tempo(s)	Troca	Tempo(s)	Troca	Tempo(s)
25fv47	16	1,99	15	5,97*	17	1,4
80bau3b	-	3,72*	-	2,03*	11	10,07
agg2	-	0,27	-	0,39	17	0,36
agg3	-	0,41*	-	0,66*	16	0,3
bnl2	25	4,97	-	1,66*	25	3,2
cre-b	28	65,06	25	49,12	15	32,24
cre-d	28	41,7	25	33,55	7	18,93
cycle	9	6,31	7	15,8*	8	5,49
finnis	17	0,22	19	0,2	14	0,18
grow15	-	0,12	-	0,06	16	0,82*
grow22	-	0,27	-	0,12	19	1,6*
ken-13	24	222,46	22	81,06	13	19,79
ken-07	-	0,35	-	0,07	13	0,12
ken-18	31	3591,49	-	930,91	-	175,57
pds-06	-	26,12	-	14,8	26	11,41
pds-10	-	55,19	-	30,14	43	26,4
pds-20	-	607,42	-	355,13	34	188,93
pilotnov	-	1,04*	-	0,8	26	1,45
scr15	10	9,91	11	14,07	11	13,28
scr20	12	109,85*	13	91,52	12	82,27
scrs8	-	0,16*	-	0,04*	20	0,06

* representa os problemas que não atingiram o ótimo

- representa os problemas em que não trocaram de fase

Tabela 1: Comparação do tempo computacional, em segundos, e a iteração que ocorreu troca de fase, considerando $tol = 10^{-8}$, $\eta = 5$ e $\eta_{max} = 20$.

5 vezes menor do que na heurística 2. Outro comportamento a ser destacado é do problema pds-20. Nas heurísticas 1 e 2 não há troca de fase e na heurística proposta isso ocorre reduzindo o tempo computacional em 3,2 vezes quando comparado com a heurística 1 e em 1,8 vezes quando comparado com a heurística 2.

Contudo, problemas que haviam sido resolvidas pelas heurísticas 1 e 2 como grow15 e grow22 não atingiram o ótimo na heurística proposta. Em contra partida, problemas como 25fv47, 80bau3b, agg2, agg3, bnl2, cycle, finnis, scr15 e scrs8, não atingiram o ótimo com a heurística 1 ou 2 e foram resolvidos pela heurística proposta.

Os resultados do conjunto dos 86 problemas se encontram na Tabela 2. A heurística proposta quando comparada a Heurística 1, apresenta redução de 70% no número de iterações do gradiente conjugado, e em relação a heurística 2 apresenta aproximadamente 40%. Ao analisar o tempo computacional, o da Heurística 1 é 13 vezes maior que o da Heurística proposta e o da Heurística 2 é 3 vezes maior. O número de problemas em que ocorreu a troca de fases aumentou em aproximadamente 57% com a heurística proposta.

8. Conclusões

Neste trabalho a heurística proposta foi comparada com heurísticas apresentadas em [Fontova et al., 2011] e [Bocanegra et al., 2007] considerando o tempo computacional e o *status* da solução encontrada. Os resultados encontrados pela heurística proposta são superiores na maioria dos problemas. Considerando o conjunto de problemas testados os resultados foram sempre supe-

	Heurística 1	Heurística 2	Heurística proposta
Problemas testados que atingiram o ótimo	81	80	84
Problemas testados que trocaram de fase	40	47	74
Iterações de gradiente conjugado	2 070 769	681 771	422 467
Tempo (segundos)	4903.3	1674.78	546.05

Tabela 2: Comparação dos resultados do conjunto dos 86 problemas resolvidos pela heurística 1, 2 e pela heurística proposta. Foram usados os parâmetros $\beta = 5$ e $tola = -8$.

riores no que se refere ao tempo computacional e quantidade de problemas resolvidos no ótimo. Assim, o número de condição se mostrou um bom indicativo para a mudança de fase, reduzindo o número de iterações do método de gradientes conjugados e em alguns problemas até o número de iterações de pontos interiores. Uma heurística que considere o número de iterações do método de gradientes conjugados além do número de condição está sendo implementada.

Agradecimentos

À CAPES, CNPq e Fapesp pelo apoio financeiro.

Referências

- Anderson, E., Bai, Z., Bischof, C., Blackford, S., Dongarra, J., Du Croz, J., Greenbaum, A., Hammarling, S., McKenney, A., e Sorensen, D. (1999). *LAPACK Users' guide*, volume 9. Siam.
- Bocanegra, S. (2005). Algoritmos de newton-krylov preconditionados para métodos de pontos interiores. *Impresso, Universidade Federal de Minas Gerais, Belo Horizonte*.
- Bocanegra, S., Campos, F. F., e Oliveira, A. R. (2007). Using a hybrid preconditioner for solving large-scale linear systems arising from interior point methods. *Computational Optimization and Applications*, 36(2-3):149–164.
- Browne, S., Dongarra, J., Grosse, E., e Rowan, T. (1995). The netlib mathematical software repository. *D-Lib Magazine, Sep*.
- Campos, F. F. (1995). *Analysis of conjugate gradients-type methods for solving linear equations*. PhD thesis, University of Oxford.
- Czyzyk, J., Mehrotra, S., Wagner, M., e Wright, S. J. (1999). Pcx: An interior-point code for linear programming. *Optimization Methods and Software*, 11(1-4):397–430.
- Fontova, M. I. V., Oliveira, A. R. L. d., e Campos, F. F. (2011). Heuristics for implementation of a hybrid preconditioner for interior-point methods. *Pesquisa Operacional*, 31(3):579–591.
- Hestenes, M. R. e Stiefel, E. (1952). *Methods of conjugate gradients for solving linear systems*, volume 49. NBS.
- Karmarkar, N. (1984). A new polynomial-time algorithm for linear programming. In *Proceedings of the sixteenth annual ACM symposium on Theory of computing*, p. 302–311. ACM.
- Kershaw, D. S. (1978). The incomplete cholesky—conjugate gradient method for the iterative solution of systems of linear equations. *Journal of Computational Physics*, 26(1):43–65.
- Kushida, N. (2015). Condition number estimation of preconditioned matrices. *PloS one*, 10(3).
- Lin, C.-J. e Moré, J. J. (1999). Incomplete cholesky factorizations with limited memory. *SIAM Journal on Scientific Computing*, 21(1):24–45.

- Luenberger, D. G. e Ye, Y. (1984). *Linear and nonlinear programming*, volume 2. Springer.
- Maros, I. e Mészáros, C. (1999). A repository of convex quadratic programming problems. *Optimization Methods and Software*, 11(1-4):671–681.
- Mehrotra, S. (1992). On the implementation of a primal-dual interior point method. *SIAM J. Optim.* 2, p. 575–601.
- Oliveira, A. R. e Sorensen, D. C. (2005). A new class of preconditioners for large-scale linear systems from interior point methods for linear programming. *Linear Algebra and its applications*, 394:1–24.
- Paige, C. C., Parlett, B. N., e Van der Vorst, H. A. (1995). Approximate solutions and eigenvalue bounds from krylov subspaces. *Numerical linear algebra with applications*, 2(2):115–133.
- Van der Sluis, A. e van der Vorst, H. A. (1986). The rate of convergence of conjugate gradients. *Numerische Mathematik*, 48(5):543–560.