

MÉTODO PRIMAL DUAL BARREIRA LOGARÍTMICA PREDITOR-CORRETOR APLICADO À *BASIS PURSUIT*

Paula Aparecida Kikuchi^{a*}, Aurelio Ribeiro Leite de Oliveira^a,
Daniela Renata Cantane^b

^a*Departamento de Matemática Aplicada
Universidade Estadual de Campinas - UNICAMP, Campinas-SP, Brasil*

^b*Departamento de Bioestatística
Universidade Estadual Paulista - UNESP, São Paulo-SP, Brasil*

Resumo

Vários são os métodos propostos para reconstrução de sinal, o enfoque deste trabalho é o método *Basis Pursuit*. Trabalhando com dicionários *overcomplete*, são inúmeras as combinações possíveis para a representação do sinal e *Basis Pursuit* encontra a mais esparsa. O problema em questão pode ser escrito como um problema de programação linear e o Método de Pontos Interiores Primal-Dual Barreira Logarítmica é apresentado para a resolução deste problema. Buscando maior eficiência propõe-se a inclusão das direções afim-escala, centragem e de correção no mesmo método, obtendo o Método Primal-Dual Barreira Logarítmica Preditor-Corretor e, além disso, uma variação do mesmo. Resultados computacionais aplicados em problemas reais sugerem a eficiência do método proposto.

Palavras-chave: Programação Linear, Métodos de Pontos Interiores, Processamento de Sinais.

Abstract

There are many proposed methods for signal reconstruction, this work deals with the *Basis Pursuit* method. When working with overcomplete dictionaries, there exist countless possible combinations to represent the signal and *Basis Pursuit* finds the sparsest one. The problem in question can be rewritten as a linear programming problem and the Primal-Dual Logarithmic Barrier Method is shown for the solution of this problem. Seeking higher efficiency, we present the affine scaling direction, the centering direction and the nonlinear correction direction in the same method, obtaining the Predictor-Corrector Primal-Dual Logarithmic Barrier Method, of which a variation is also implemented. Computational results with real life problems show the efficiency of the proposed method.

Keywords: Linear Programming, Interior Point Methods, Signal Processing.

*Autor para correspondência: e-mail: paulapkikuchi@gmail.com

¹Todos os autores assumem a responsabilidade pelo conteúdo do artigo.

1. Representação de Sinais

Algo que muitas vezes não percebemos, mas faz parte constante de nossas vidas, são os sinais. Seja na imagem reproduzida no televisor, fotos, a música que ouvimos de um rádio, o uso cotidiano do celular, entre outros, os sinais são os responsáveis pela reprodução e uso destes.

Podemos representar um sinal como uma combinação linear de *waveforms*, tais *waveforms* desta combinação são extraídas a partir de um conjunto chamado dicionário.

Definição 1 (Dicionário). Definimos dicionário como uma coleção de *waveforms* parametrizadas $D = (\phi_\gamma : \gamma \in \Gamma)$, sendo *waveforms* ϕ_γ sinais discretos de tempo chamados átomos, que nesse trabalho será visto como um vetor pertencente a \mathbb{R}^n , e Γ o conjunto de todos os parâmetros do dicionário.

Representação de sinais por meio de dicionários é uma alternativa que surge, além das representações usuais que utilizam superposição de senóides, como mencionado em Chen *et al.* (2001).

Considerando um sinal s como sendo um vetor pertencente a \mathbb{R}^n , sua decomposição em um dicionário é dada por:

$$s = \sum_{\gamma \in \Gamma} \alpha_\gamma \phi_\gamma \quad (1.1)$$

Considerando que temos um dicionário discreto de p *waveforms* e uma matriz Φ cujas colunas correspondem às p *waveforms* ($\Phi: n \times p$), podemos reescrever (1.1) como:

$$\Phi \alpha = s, \quad (1.2)$$

onde $\alpha = (\alpha_\gamma)$ é o vetor dos coeficientes em (1.1).

Veja que, se as *waveforms* são linearmente independentes, teremos uma matriz não singular, o que implica que Φ terá representação única. Em particular, se as *waveforms* forem ortonormais $\Phi^{-1} = \Phi^T$.

Dicionários são definidos como completos quando possuem exatamente n átomos, *overcomplete* quando possuem mais que n átomos, ou *undercomplete* quando possuem menos de n átomos. Exemplos de dicionários podem ser encontrados em Chen (1995).

Note que, em (1.1), se trabalhamos com um dicionário completo, s terá uma representação única. Já quando fazemos uso de dicionários *overcomplete*, sua representação

não será única, permitindo que escolhamos a que nos for mais conveniente. Como um dos objetivos que queremos alcançar nas representações de sinais é esparsidade, já podemos ter em mente que uma boa ideia seria minimizar $\sum_{\gamma \in \Gamma} |\alpha_\gamma|$.

Uma técnica eficiente em adquirir e reconstruir sinais é chamada *Compressive Sensing* (também conhecida como *Compressive Sampling*) (Candes e Wakin, 2008). *Compressive Sensing* é aplicado nas áreas de fotografia (Huang *et al.*, 2013), ressonância magnética (Lustig *et al.*, 2007), tomografia (Firooz e Roy, 2010), entre outras. Uma das formas de resolver os problemas obtidos por essa técnica, consiste em utilizar o Método *Basis Pursuit*.

1.1 O Método *Basis Pursuit*

Como mencionado em Kikuchi (2013), não são poucos os métodos propostos para representar sinais em dicionários *overcomplete*. Em especial temos o método de frames (MOF), que escolhe a representação cujos coeficientes tenham a norma 2 mínima, e *basis pursuit* (BP). Diferentemente do método de frames, BP irá escolher a representação cujos coeficientes tenham a norma 1 mínima. Chamamos essa decomposição do sinal de superposição “ótima” de elementos do dicionário. Chen *et al.* (2001) mostram várias vantagens de BP, dentre elas, a esparsidade. Note que o método BP baseia-se no seguinte modelo:

$$\begin{aligned} & \text{minimizar } \|\alpha\|_1 \\ & \text{sujeito a } \Phi\alpha = s \end{aligned} \tag{1.3}$$

Podemos reformular o problema de encontrar os coeficientes mais esparsos em (1.1), por meio de um problema de programação linear na forma padrão, dada por:

$$\begin{aligned} & \text{minimizar } c^T x \\ & \text{sujeito a } Ax = b, \\ & \quad x \geq 0 \end{aligned} \tag{1.4}$$

sendo A uma matriz de restrições pertencente a $\mathbb{R}^{m \times n}$, x um vetor coluna pertencente a \mathbb{R}^n , cujas componentes são denominadas variáveis primais, e b e c vetores coluna pertencentes a \mathbb{R}^m e \mathbb{R}^n , respectivamente, sendo c os custos associados aos elementos de x .

De fato, considere e o vetor pertencente a \mathbb{R}^p , cujos elementos são compostos por 1, escrevendo α em (1.3) como a diferença de dois vetores não-negativos u e v , pertencentes a \mathbb{R}^p , e sendo $\|\alpha\|_1$ limitante superior de $e^T u + e^T v$, pois estamos minimizando $\|\alpha\|_1$ e, nesse

PESQUISA OPERACIONAL PARA O DESENVOLVIMENTO

caso, $u_i v_i = 0$ para i variando de 1 a p , teremos $\Phi\alpha = \Phi(u - v)$ e $e^T u + e^T v = \|\alpha\|_1$, esta última segue do seguinte fato:

$$\begin{aligned} \sum_{\gamma=1}^p |\alpha_\gamma| &= \sum_{\gamma=1}^p |u_\gamma - v_\gamma| \leq \sum_{\gamma=1}^p |u_\gamma| + \sum_{\gamma=1}^p |v_\gamma| = \sum_{\gamma=1}^p u_\gamma + v_\gamma = e^T u + e^T v \leq \sum_{\gamma=1}^p |\alpha_\gamma| \Rightarrow \sum_{\gamma=1}^p |\alpha_\gamma| \\ &= e^T u + e^T v. \end{aligned}$$

Dessa forma obtemos o seguinte problema equivalente:

$$\begin{aligned} & \text{minimizar } e^T u + e^T v \\ & \text{sujeito a } \Phi(u - v) = s, \\ & \quad (u, v) \geq 0 \end{aligned}$$

que pode ser reformulado como um problema de programação linear na forma padrão. Como visto em Chen (1995), essa reformulação é obtida por meio das seguintes associações:

$$\begin{aligned} m &\Leftrightarrow 2p \\ x &\Leftrightarrow (u; v) \\ c &\Leftrightarrow (e; e) \\ A &\Leftrightarrow (\Phi, -\Phi) \\ b &\Leftrightarrow s. \end{aligned}$$

Como o problema BP pode ser reescrito como um problema linear, nosso enfoque será trabalhar com este problema aplicando Métodos de Pontos Interiores Barreira Logarítmica. Em Chen (1995), podemos encontrar a comparação do Método Simplex e do Método de Pontos Interiores, sendo constatado que o Método Primal-Dual Barreira Logarítmica obteve melhor desempenho. De certa forma isso já era esperado, visto que a precisão adotada não é rigorosa, geralmente a solução obtida pelos Métodos de Pontos Interiores Primal-Dual está mais próxima de uma solução ótima que a obtida pelo Método Simplex (Gondzio *et al.*, 2013).

Vamos analisar o método proposto em Chen (1995) para o problema BP. Nele resolvemos o problema linear perturbado:

$$\begin{aligned} & \text{minimizar } c^T x + \frac{1}{2} \|\gamma x\|_2^2 + \frac{1}{2} \|p\|_2^2 \\ & \text{sujeito a } Ax + \delta p = b, \\ & \quad x \geq 0 \end{aligned} \tag{1.6}$$

onde γ e δ são parâmetros de perturbação pequenos, da ordem de (10^{-4}) , utilizando o método primal-dual barreira logarítmica.

Um código referente a tal método (BP_Interior), que faz parte de um pacote denominado *Atomizer*, foi implementado em Matlab e pode ser encontrado em Chen *et al.* (2012).

2. Método Primal Dual Barreira Logarítmica Aplicado a BP

Aplicaremos o Método Primal-Dual Barreira Logarítmica a BP, mas antes reescreveremos o problema como um problema linear perturbado. A perturbação é incluída com o objetivo de regularizar o problema, obtendo dessa forma um melhor condicionamento dos sistemas lineares a serem resolvidos Gondzio (2012).

2.1 Formulação do Problema BP

Em Gill *et al.* (1991) é discutido o método de barreira para a resolução de problemas de programação lineares e quadráticos (PQ), expressos na forma padrão como:

$$\begin{aligned} & \text{minimizar } c^T x + \frac{1}{2} x^T Q x \\ & \text{sujeito a } Ax = b, \\ & \quad l \leq x \leq u \end{aligned} \tag{2.1}$$

onde A é uma matriz $m \times n$ ($m \leq n$) e Q é uma matriz simétrica semidefinida positiva. Note que, se $Q = 0$, (2.1) é um problema de programação linear (PL).

No artigo em questão, é descrito um algoritmo Primal-Dual PQ para resolver (2.1). Para tal resolução, os autores reapresentam o problema da seguinte forma:

$$\begin{aligned} & \text{minimizar } c^T x + \frac{1}{2} x^T Q x + \frac{1}{2} \|\gamma x\|_2^2 + \frac{1}{2} \|p\|_2^2 \\ & \text{sujeito a } Ax + \delta p = b, \\ & \quad x - s_1 = l \\ & \quad x + s_2 = u \end{aligned} \tag{2.2}$$

onde $s_1 \geq 0, s_2 \geq 0$ e os escalares γ e δ são parâmetros de regularização “pequenos”. Denota-se por π, z e $-y$ as variáveis duais associadas às três equações de restrições.

Na solução, z e y são não negativos.

O termo $\frac{1}{2} \|\gamma x\|_2^2$ é incluído para garantir que $\|\gamma x^*\|$ seja limitado (sendo x^* uma solução ótima), e o termo δp para que uma solução aproximada do sistema $Ax=b$ seja determinada através do método dos mínimos quadrados, caso a restrição não tenha solução factível (podemos associar ao caso do sinal ser perturbado).

Para mais detalhes desse problema, ver Gill *et al.* (1991).

2.2 Aplicação do Método à BP

A resolução da técnica *Compressive Sensing*, utilizando *Basis Pursuit*, pode ser abordado de outras formas, como feito em Fountoulakis (2015). Escolhemos em nosso trabalho, dar continuidade ao trabalho desenvolvido por Chen (1995). O objetivo principal é a busca da representação do sinal s através de *Basis Pursuit*. Dessa forma, como queremos a representação exata do sinal, vamos resolver o problema de programação linear (1.4), que é equivalente a (1.3). Para resolver tal problema, vamos fazer uso da formulação proposta por Gill *et al.* (1991), reescrevendo o problema como o problema de programação linear perturbado (1.6); e resolvendo-o assim como feito em Chen (1995).

Associamos ao problema linear perturbado (1.6) o subproblema barreira logarítmica:

$$\begin{aligned} & \text{minimizar } c^T x + \frac{1}{2} \|\gamma x\|_2^2 + \frac{1}{2} \|p\|_2^2 - \mu \sum_{i=1}^m \ln(x_i) \\ & \text{sujeito a } Ax + \delta p = b. \end{aligned}$$

A restrição de desigualdade $x > 0$ fica implícita. Como $\mu \rightarrow 0$ a solução converge para a solução do problema linear perturbado. Definimos o Lagrangiano como:

$$L(x, y, p) = c^T x + \frac{1}{2} \|\gamma x\|_2^2 + \frac{1}{2} \|p\|_2^2 - \mu \sum_{i=1}^m \ln(x_i) + y^T (b - Ax - \delta p).$$

Agora vamos calcular as condições necessárias de primeira ordem do problema de minimização da função Lagrangiana obtendo:

$$-\nabla_x L = A^T y - \gamma^2 x - c + \mu X^{-1} e = 0,$$

PESQUISA OPERACIONAL PARA O DESENVOLVIMENTO

onde e é o vetor de dimensão apropriada, cujos elementos são compostos por 1.

No artigo de McShane *et al.* (1989) quando o parâmetro do problema primal e do dual são iguais, $z = \mu X^{-1}e$. Assim, as condições necessárias de primeira ordem são dadas por:

$$\begin{aligned} -\nabla_x L &= A^T y - \gamma^2 x - c + z = 0 \\ -\nabla_p L &= \delta y - p = 0 \Rightarrow p = \delta y, \\ -\nabla_y L &= Ax + \delta p - b = Ax + \delta^2 y - b = 0 \end{aligned}$$

e por McShane *et al.* (1989) novamente:

$$ZXe - \mu e = 0.$$

Rearranjando as expressões, obtemos:

$$F(x, y, z) = \begin{pmatrix} A^T y - \gamma^2 x - c + z \\ Ax + \delta^2 y - b \\ ZXe - \mu e \end{pmatrix} = 0,$$

onde z é um vetor dual, e X e Z são as matrizes diagonais formadas pelos elementos dos vetores x e z , respectivamente.

Aplicamos o Método de Newton a fim de encontrar a solução do sistema não linear acima. As direções de Newton $(\Delta x, \Delta y, \Delta z)$ devem satisfazer:

$$F(x, y, z) + (\Delta x, \Delta y, \Delta z) = \begin{pmatrix} \frac{\delta F}{\delta x}(x, y, z) \\ \frac{\delta F}{\delta y}(x, y, z) \\ \frac{\delta F}{\delta z}(x, y, z) \end{pmatrix} = 0,$$

o que corresponde a:

$$\begin{pmatrix} -\gamma^2 \Delta x + A^T \Delta y + \Delta z \\ A \Delta x + \delta^2 \Delta y \\ Z \Delta x + X \Delta z \end{pmatrix} = \begin{pmatrix} c + \gamma^2 x - z - A^T y \\ b - Ax - \delta^2 y \\ \mu e - Zx \end{pmatrix} = \begin{pmatrix} t \\ r \\ q \end{pmatrix},$$

que fica reduzido ao seguinte sistema:

PESQUISA OPERACIONAL PARA O DESENVOLVIMENTO

$$(ADA^T + \delta^2 I)\Delta y = r - AD(X^{-1}q - t), \quad (2.3)$$

$$\Delta x = DA^T \Delta y + D(X^{-1}q - t), \quad (2.4)$$

$$\Delta z = X^{-1}q - X^{-1}Z\Delta x. \quad (2.5)$$

sendo $D = (X^{-1}Z + \gamma^2 I)^{-1}$. Para o cálculo do tamanho do passo (ρ_p, ρ_d) , vamos sempre escolher o maior possível, contanto que o ponto $x^{(k+1)}$ e $z^{(k+1)}$ sejam pontos interiores. Dessa forma teremos:

$$\rho_p = 0,99 \times \max\{\rho: x + \rho\Delta x \geq 0, \rho \leq 1\};$$

$$\rho_d = 0,99 \times \max\{\rho: z + \rho\Delta z \geq 0, \rho \leq 1\}.$$

A cada passo de Newton vamos decrescer o parâmetro de barreira μ monotonicamente e mais rapidamente se largos passos são tomados:

$$\mu \leftarrow (1 - \min(\rho_p, \rho_d, 0,99))\bar{\mu}. \quad (2.6)$$

sendo $\bar{\mu}$ o valor de μ do passo anterior.

O método converge quando $\|b - Ax - \delta^2 y\|_2$, $\|c + \gamma^2 x - z - A^T y\|_2$ e $z^T x$ são suficientemente pequenos. O algoritmo encontra-se em Chen *et al.* (2001).

Na resolução de sistemas do tipo:

$$Ax = b, \quad (2.7)$$

sendo A uma matriz simétrica e definida-positiva, frequentemente utilizamos a fatoração de Cholesky e determinamos o valor da variável x resolvendo dois sistemas triangulares. Como a fatoração de Cholesky tem complexidade de $O(n^3)$ e tendo em vista que estamos interessados em matrizes A de grande dimensão, essa fatoração acaba não sendo conveniente.

Em nosso caso o sistema que queremos resolver é dado pela equação (2.3), dessa forma sabendo que para certos dicionários, os quais serão os que iremos trabalhar, temos o “algoritmo implícito rápido”, o que significa que, para vetores arbitrários x , as operações Ax e $A^T x$ são calculadas em $O(n \log(n))$ ou $O(n)$, sem mesmo armazenar as matrizes A e A^T , e utilizando propriedades especiais da matriz para acelerar o cálculo da operação; teremos maior eficiência usando métodos iterativos, sendo o mais popular o Método dos Gradientes Conjugados e sendo este o utilizado neste trabalho.

Até agora, apresentamos os conceitos básicos do Método Primal-Dual Barreira Logarítmica aplicado ao problema não-linear (1.6). O programa BP_Interior desenvolvido por Chen, contido em *Atomizer* (Chen *et al.*, 2012), resolve o problema *Basis Pursuit* (1.3) seguindo o mesmo procedimento descrito no Método Primal-Dual Barreira Logarítmica. Maiores detalhes podem ser encontrados em Kikuchi (2013).

3. Desenvolvimento dos Métodos Propostos

Visando obter maior eficiência, propomos modificações no Método de Pontos Interiores Primal-Dual Barreira Logarítmica, do programa BP_Interior. Aplicaremos o Método Primal-Dual Barreira Logarítmica Preditor-Corretor (como foi sugerido em Chen, 1995), e uma variação deste último a (1.6).

3.1 Método Primal-Dual Barreira Logarítmica Preditor-Corretor

A fim de melhorar o Método Primal-Dual Barreira Logarítmica vamos introduzir as três componentes: Direção Afim-Escala, Direção de Centragem e Direção de Correção. Tendo em vista esses conceitos, e em busca de melhorar o desempenho de BP_Interior, façamos as correções dos termos não lineares do método descrito na seção anterior. Note que o termo não linear corresponde à condição de complementaridade. Assim, no primeiro passo consideraremos q tal que $q = -Zx$, e no segundo passo $q = \mu e - Zx - \Delta Z \Delta x e$, onde Z e X são as matrizes diagonais cujos elementos são as entradas dos vetores z e x respectivamente, que foram obtidos no primeiro passo.

O algoritmo pode ser encontrado em Kikuchi (2013).

3.2 Variante do Método Primal-Dual Barreira Logarítmica Preditor-Corretor

Anteriormente associamos ao problema linear perturbado (1.6) o subproblema barreira logarítmica e, tendo definido o Lagrangiano, calculamos as condições necessárias de primeira ordem do problema de minimização da função Lagrangiana, obtendo:

$$\begin{aligned} -\nabla_x L &= A^T y - \gamma^2 x - c + z = 0 \\ -\nabla_p L &= \delta y - p = 0 \Rightarrow p = \delta y, \\ -\nabla_y L &= Ax + \delta p - b = Ax + \delta^2 y - b = 0 \end{aligned} \tag{3.1}$$

e da condição de complementaridade:

$$ZXe - \mu e = 0.$$

Prosseguimos substituindo a segunda equação de (3.1) na terceira equação; agora, vamos propor uma variação do Método Primal-Dual Barreira Logarítmica Predictor-Corretor, onde não realizaremos tal substituição. Dessa forma, rearranjaremos as expressões das condições de primeira ordem como:

$$F(x, y, z, p) = \begin{pmatrix} A^T y - \gamma^2 x - c + z \\ \delta y - p \\ Ax + \delta p - b \\ ZXe - \mu e \end{pmatrix} = 0,$$

onde z é um vetor dual, X e Z são as matrizes diagonais formadas pelos elementos dos vetores x e z , respectivamente.

Aplicando o Método de Newton a $F(x, y, z, p) = 0$, obtemos:

$$F(x, y, z, p) + (\Delta x, \Delta y, \Delta z, \Delta p) = \begin{pmatrix} \frac{\delta F}{\delta x}(x, y, z, p) \\ \frac{\delta F}{\delta y}(x, y, z, p) \\ \frac{\delta F}{\delta z}(x, y, z, p) \\ \frac{\delta F}{\delta p}(x, y, z, p) \end{pmatrix} = 0,$$

que corresponde a:

$$\begin{pmatrix} -\gamma^2 \Delta x + A^T \Delta y + \Delta z \\ \delta \Delta y - I \Delta p \\ A \Delta x + \delta \Delta p \\ Z \Delta x + X \Delta z \end{pmatrix} = \begin{pmatrix} c + \gamma^2 x - z - A^T y \\ -\delta y + p \\ b - Ax - \delta p \\ \mu e - Zx \end{pmatrix} = \begin{pmatrix} t \\ r \\ q \\ w \end{pmatrix},$$

que fica reduzido ao seguinte sistema:

$$(ADA^T + \delta^2 I) \Delta y = r - AD(X^{-1}w - t) + \delta r, \quad (3.2)$$

$$\Delta x = DA^T \Delta y + D(X^{-1}q - t), \quad (3.3)$$

$$\Delta z = X^{-1}q - X^{-1}Z \Delta x, \quad (3.4)$$

$$\Delta p = \delta \Delta y - r, \quad (3.5)$$

sendo $D = (X^{-1}Z + \gamma^2 I)^{-1}$.

Adicionando a direção afim-escala, a direção de centragem, e a direção de correção dadas pelo Método Preditor-Corretor, vamos considerar no primeiro passo $w = -Zx$ e no segundo passo $w = \mu - Zx - \Delta Z \Delta X e$

O método inicializa-se pela solução do Método de Frames (MOF). Assim, nosso p inicial será dado por: $p = \frac{(b-Ax)}{\delta}$. Para a escolha do comprimento do passo (ρ_p, ρ_d) , vamos fazer do mesmo modo que para o Método Primal-Dual Barreira Logarítmica Preditor-Corretor, onde sempre escolhemos o maior passo possível, contanto que $x^{(k+1)}$ e $z^{(k+1)}$ sejam pontos interiores. A cada passo do Método de Newton, vamos decrescer o parâmetro de barreira monotonicamente e mais rapidamente se passos largos são tomados:

$$\mu \leftarrow (1 - \min(\rho_p, \rho_d, 0,99))\bar{\mu}.$$

sendo $\bar{\mu}$ o valor de μ do passo anterior.

O método converge quando $\|b - Ax - \delta p\|_2$, $\|c + \gamma^2 x - z - A^T y\|_2$, $\|p - \delta y\|_2$ e $z^T x$ são suficientemente pequenos (10^{-1}). O algoritmo encontra-se em Kikuchi (2013).

4. Testes Computacionais

A seguir, apresentamos os resultados obtidos para o Método BP_Interior de Chen (1995), e para os nossos métodos modificados propostos: Método Primal-Dual Barreira Logarítmica Preditor-Corretor, e nossa variação deste último, os quais serão chamados BP_InteriorPC e BP_InteriorPC1, respectivamente.

Os experimentos numéricos foram implementados em Matlab R2010a, com o sistema operacional Linux, distribuição Ubuntu 11.04, processador Intel® core i7 2600, 3.4 Ghz, 4 GB de memória DDR3, RAM clock 1333Mhz.

Na Tabela 1 listamos os sinais utilizados, com seus respectivos parâmetros, que são números ou vetores linha, o nome do dicionário que vamos utilizar para a representação e o tamanho do problema.

As Tabelas 2, 3 e 4 mostram os resultados obtidos para BP_Interior, BP_InteriorPC e BP_InteriorPC1. Nas Tabelas 2 e 3 temos os valores da função objetivo (FO), ou seja, o mínimo valor de $\|\alpha\|_1$ em (1.3), e o tempo decorrido até a obtenção da convergência (em segundos).

PESQUISA OPERACIONAL PARA O DESENVOLVIMENTO

Na Tabela 4 são apresentados o número de iterações realizadas por cada método, e o número de iterações realizadas pelo Método dos Gradientes Conjugados (ItGC).

A relação entre o número de iterações e o valor da função objetivo, obtidos nos métodos implementados, com os sinais da Tabela 1 podem ser encontrados em Kikuchi (2013).

Tabela 1: Dados do problema.

Sinal	Tamanho do Problema	Dicionário	par1	par2	par3
TwinSine-1	256	DCT	4	0	0
WernerSorrows	1024	CP	6	Seno	0
Carbon	1024	WP	10	Qmf	0
TwinSine-2	256	DCT	4	0	0
FM-Cosine	1024	CP	6	Seno	0
Gong	1024	CP	10	Seno	0
Dynamic-0	256	DCT e DIRAC	MekeList (4,0)	0	0
Dynamic-2	256	DCT e DIRAC	MekeList (4,0)	0	0
MultiGong	256	MDC	8	1	0

Tabela 2: Resultados computacionais para a função objetivo e tempo de processamento.

Sinal	BP_Interior	
	FO	Tempo
TwinSine-1	2,00933e+00	0,1
WernerSorrows	5,07482e+02	249,2
Carbon	6,00247e+00	19,8
TwinSine-2	2,01150e+00	0,1
FM-Cosine	2,52872e+02	237,3
Gong	4,73171e+00	1924,5
Dynamic-0	6,01964e+00	0,3
Dynamic-2	4,03672e+02	0,4
MultiGong	2,43810e+01	5,6

O tempo de convergência alcançado pelos três métodos foram similares, sendo as diferenças mais significativas obtidas para o sinal WernerSorrows, onde BP_Interior obteve

PESQUISA OPERACIONAL PARA O DESENVOLVIMENTO

249, 2 segundos para a convergência, praticamente o dobro de tempo que BP_InteriorPC (130, 0 segundos) e BP_InteriorPC1 (124, 1 segundos); e para o sinal Gong, obtendo tempos bem distintos para os três métodos em questão, para BP_Interior obtemos 1924, 5 segundos para a convergência, para BP_InteriorPC 3501, 3 segundos, e BP_InteriorPC1 11587, 9 segundos, um número bem maior em relação aos anteriores.

Tabela 3: Resultados computacionais para a função objetivo e tempo de processamento.

Sinal	BP_InteriorPC		BP_InteriorPC1	
	FO	Tempo	FO	Tempo
TwinSine-1	2,00934e+00	0,1	2,00934e+00	0,1
WernerSorrows	5,07587e+02	130,0	5,07576e+02	124,1
Carbon	6,00003e+00	19,6	6,00003e+00	20,2
TwinSine-2	2,01108e+00	0,1	2,01108e+00	0,1
FM-Cosine	2,52885e+02	210,9	2,52896e+02	208,5
Gong	4,73088e+00	3501,3	4,72132e+00	11587,9
Dynamic-0	6,01902e+00	0,4	6,01907e+00	0,5
Dynamic-2	4,02187e+02	0,7	4,02187e+02	0,7
MultiGong	2,44009e+01	7,2	2,43895e+01	7,3

Tabela 4: Número de iterações dos métodos.

Sinal	BP_Interior		BP_InteriorPC		BP_InteriorPC1	
	It	ItGC	It	ItGC	It	ItGC
TwinSine-1	11	59	9	90	9	90
WernerSorrows	18	20883	11	10850	11	10520
Carbon	8	85	6	99	6	99
TwinSine-2	9	43	9	97	9	98
FM-Cosine	17	20098	12	17891	12	18026
Gong	21	11510	19	21681	23	71812
Dynamic-0	7	40	5	54	5	54
Dynamic-2	7	52	7	91	7	91
MultiGong	19	853	15	1097	17	1457

5. Conclusões

Nota-se que BP_Interior, BP_InteriorPC e BP_InteriorPC1 possuem os valores da função objetivo muito próximos e, em alguns casos, idênticos. A diferença dos valores observada se deve à precisão adotada não ser muito rigorosa.

Quanto ao tempo de convergência, esta foi similar para os três métodos, com diferenças significativas apenas para dois sinais.

Quando o número de iterações realizadas por BP_InteriorPC não foi menor que o número realizado por BP_Interior, essa mostrou-se igual. Já BP_InteriorPC1 obteve mais iterações que BP_Interior apenas para o sinal Gong, com uma diferença de apenas duas iterações.

Em relação ao Método dos Gradientes Conjugados podemos notar pelos resultados obtidos em que o número de iterações do Método dos Gradientes Conjugados aumenta constantemente. Isso ocorre porque a solução inicial está próxima do centro da região factível. Assim, nas iterações iniciais, o sistema de equações (2.3) é bem condicionado e o método converge rapidamente. Mas, como $x^T z$ converge para 0, z/x convergirá para infinito ou 0. Assim, a matriz $D = (X^{-1}Z + \gamma^2 I)$ e a matriz $(ADA^T + \delta^2 I)$ tornam-se mais mal-condicionadas, e o método demora para convergir. Nos métodos implementados, não pré-condicionamos a matriz A. Por isso, obtivemos grande número de iterações no Método dos Gradientes Conjugados.

5.1 Perspectivas Futuras

Com a finalidade de obter melhor desempenho, pretendemos pré-condicionar a matriz de interesse, a fim de obter um número menor de iterações no Método dos Gradientes Conjugados.

Também pretendemos verificar a aplicação dos métodos para dicionários diferentes e de maior dimensão, verificando se esta abordagem é viável em aplicações reais.

Agradecimentos

Este trabalho foi parcialmente financiado pela FAPESP, CNPq e CAPES.

Referências

Baraniuk, R. G. (2007). Compressive sensing [Lecture notes]. IEEE Signal Processing Magazine, v. 24, n. 4, p. 118-121.

PESQUISA OPERACIONAL PARA O DESENVOLVIMENTO

Candes, E. J.; Wakin, M. B. (2008). An introduction to compressive sampling. *IEEE Signal Processing Magazine*, v. 25, n. 2, p. 21-30.

Chen, S. S. (1995). Basis pursuit. Tese (Doutorado) - Universidade Stanford, Stanford, CA.

Chen, S. S.; Donoho, D. L.; Saunders, M. A. (2001). Atomic decomposition by basis pursuit. *SIAM Review*, v. 43, n. 1, p. 129-159.

Chen, S. S.; Donoho, D. L.; Saunders, M. Atomizer for matlab5. x. Disponível em: <<http://www-stat.stanford.edu/~atomizer>>. Acesso em: 15 nov. 2012.

Firooz, M. H.; Roy, S. (2010). Network tomography via compressed sensing. In: *Global Telecommunications Conference (GLOBECOM 2010)*, 2010 IEEE, p. 1-5.

Fountoulakis, K. (2015). Higher-order methods for large-scale optimization. Tese (Doutorado) - Universidade de Edimburgo, Edimburgo, Reino Unido.

Gill, P.; Murray, W.; Ponceleon, D. B.; Michael, A. (1991). Solving reduced KKT systems in barrier methods for linear and quadratic programming. (Relatório Técnico) - Depto de Pesquisa Operacional, Universidade Stanford, Stanford, CA.

Gondzio, J. (2012). Interior point methods 25 years later. *European Journal of Operational Research*, v. 218, n. 3, p. 587-601.

Gondzio, J.; Gonzalez-Brevis, P.; Munari, P. (2013). New developments in the primal-dual column generation technique. *European Journal of Operational Research*, v. 244, n. 1, p. 41-51.

Huang, G.; Jiang, H.; Matthews, K.; Wilford, P. (2013). Lensless imaging by compressive sensing. In: *2013 IEEE International Conference on Image Processing*. IEEE, p. 2101-2105.

Kikuchi, P. A. (2013). Métodos de pontos interiores aplicados à basis pursuit. Dissertação (Mestrado em Matemática Aplicada) - IMECC, Universidade Estadual de Campinas, Campinas, SP.

Lustig, M.; Donoho, D.; Pauly, J. M. (2007). Sparse MRI: The application of compressed sensing for rapid MR imaging. *Magnetic Resonance in Medicine*, v. 58, n. 6, p. 1182-1195.

McShane, K. A.; Monma, C. L.; Shanno D. (1989). An implementation of a primal-dual interior point method for linear programming. *ORSA Journal on Computing*, v. 1, n. 2, p. 70-83.