Ruana Maíra Schneider

MÉTODOS DE MÁXIMO DECLIVE PARA MINIMIZAÇÃO QUADRÁTICA

Dissertação submetida ao Programa de Pós-Graduação em Matemática Pura e Aplicada para a obtenção do Grau de Mestre em Matemática.
Orientador: Prof. Dr. Clóvis Caesar Gonzaga

Florianópolis

2015

Ficha de identificação da obra elaborada pelo autor,
através do Programa de Geração Automática da Biblioteca Universitária da UFSC.

Schneider, Ruana Maíra
Métodos de máximo declive para minimização quadrática / Ruana Maíra Schneider ; orientador, Clóvis Caesar Gonzaga - Florianópolis, SC, 2015.
98 p.

Dissertação (mestrado) - Universidade Federal de Santa Catarina, Centro de Ciências Físicas e Matemáticas. Programa de Pós-Graduação em Matemática Pura e Aplicada.

Inclui referências

1. Matemática Pura e Aplicada. 2. Matemática aplicada. 3. Otimização. 4. Programação não-linear. 5. Minimização quadrática. I. Gonzaga, Clóvis Caesar . II. Universidade Federal de Santa Catarina. Programa de Pós-Graduação em Matemática Pura e Aplicada. III. Título.

Ruana Maíra Schneider

MÉTODOS DE MÁXIMO DECLIVE PARA MINIMIZAÇÃO QUADRÁTICA

Esta Dissertação foi julgada adequada para obtenção do Título de "Mestre em Matemática", e aprovada em sua forma final pelo Programa de Pós-Graduação em Matemática Pura e Aplicada.

Florianópolis, 11 de maio de 2015.

Prof. Dr. Daniel Gonçalves
Coordenador do Curso

Banca Examinadora:

Prof. Dr. Clóvis Caesar Gonzaga
Orientador
Universidade Federal de Santa Catarina

Prof.ª Dr.ª Elizabeth Wegner Karas
Universidade Federal do Paraná

Prof.ª Dr.ª Melissa Weber Mendonça
Universidade Federal de Santa Catarina

Prof. Dr. Juliano de Bem Francisco
Universidade Federal de Santa Catarina

Prof. Dr. Douglas Soares Gonçalves
Universidade Federal de Santa Catarina

AGRADECIMENTOS

À minha família, em especial à minha mãe, pelo apoio em todas as minhas decisões, pelo conforto nos momentos de insegurança e por sempre acreditar na minha capacidade.

Ao professor Clóvis, pela dedicação e disposição para me orientar durante o mestrado e nesta dissertação. Agradeço por todos os encontros nos quais discutimos os resultados deste trabalho, intercalando-os com as mais variadas histórias que se pode imaginar. Aprendi muito sobre Otimização, Matemática e, sobretudo, sobre humildade.

Aos professores membros da banca, Elizabeth, Juliano, Douglas e Melissa, os quais também estiveram presentes nos eventos em que participei e apresentei resultados desta pesquisa.

Aos meus amigos e colegas do mestrado, pelo apoio nos estudos, em especial, à Aline, ao Ado e ao Edson. Às amigas Thuysa, Juciara, Maria, Débora e Kelly, pela amizade e pelas palavras de incentivo. Em especial, ao Rafael.

À CAPES, pelo suporte financeiro concedido durante a pesquisa.

RESUMO

Neste trabalho apresentamos uma descrição detalhada do método de máximo declive para problemas quadráticos com busca unidirecional exata (método de Cauchy). Esse método é globalmente convergente, porém é ineficiente, pois é lento e apresenta um comportamento oscilatório, convergindo para uma busca no espaço gerado pelos autovetores associados ao maior e ao menor autovalor da matriz Hessiana do problema quadrático. Analisamos o comportamento oscilatório do gradiente da função objetivo no caso quadrático, bem como da sequência de passos gerados pelo método de Cauchy. Apresentamos o método de Barzilai-Borwein que, experimentalmente, exibe um desempenho melhor do que o método de Cauchy, e, também, algumas variantes do método de Barzilai-Borwein. Analisamos o comportamento do gradiente causado pela escolha de outros tamanhos de passos no método de máximo declive, o que nos permitiu propor uma nova escolha para o tamanho de passo. Com isso, propomos alguns novos algoritmos (*Cauchy-short, alternated Cauchy-short* e outros) que alternam o tamanho de passo entre passos de Cauchy e passos curtos. Adotamos, ainda, uma nova proposta que utiliza passos de tamanhos dados por raízes de um polinômio de Chebyshev de ordem adequada. Experimentalmente, os novos métodos apresentam um bom desempenho, superando inclusive o método de Barzilai-Borwein. Além do bom desempenho, os novos métodos têm a vantagem de gerar sequências monotonicamente decrescentes de valores da função objetivo.

Palavras-chave: Métodos de Máximo Declive. Minimização Quadrática. Método de Cauchy. Método de Barzilai-Borwein. Convergência Assintótica.

ABSTRACT

In this thesis we show a detailed description of the steepest descent method for quadratic problems with exact line searches (Cauchy Method). Although this method is globally convergent, it is inefficient because it is slow and it shows an oscillatory behavior, converging to a search in the space spanned by the eigenvectors associated with the largest and the smallest eigenvalue of the Hessian matrix of the quadratic objective. We analyze the oscillatory behavior of the gradient of the objective function in the quadratic case as well as the sequence of steps generated by the Cauchy method. We describe the Barzilai-Borwein method, which experimentally shows a better performance than the Cauchy method, and also some of its variations. We analyzed the behavior of the gradients due to the choice of different step sizes in the steepest descent method, which allowed us to come up with a new choice for the step size. Thus, we introduce a few new algorithms (Cauchy-short, alternated Cauchy-short and others) which alternate the step sizes between Cauchy steps and short steps. We also describe a new strategy based on step sizes given by the roots of a Chebyshev polynomial with suitable order. Experimentally, the new algorithms show a good enough performance, even better than the Barzilai-Borwein method. Besides the good performance, the new methods have the advantage of generating monotonically decreasing objective function values.

Keywords: Steepest Descent Methods. Quadratic Minimization. Cauchy Method. Barzilai-Borwein Method. Asymptotic Convergence.

LISTA DE FIGURAS

SUMÁRIO

INTRODUÇÃO

O método de máximo declive, também chamado de método do gradiente, é o algoritmo mais clássico na Otimização Contínua. Esse método, proposto por Cauchy [3], em 1847, é utilizado para resolver problemas de minimização irrestrita em Programação Não Linear. É possível mostrar que o método de máximo declive com tamanho de passo dado por busca unidirecional exata, converge para uma solução ótima para problemas de minimização quadrática. Entretanto, esse método é ineficiente pois é lento computacionalmente e apresenta um comportamento oscilatório, convergindo para uma busca no espaço gerado pelos autovetores associados ao maior e ao menor autovalor da matriz Hessiana do problema quadrático.

Em 1988, Barzilai e Borwein [2] propuseram uma nova escolha de tamanho de passo para o método de máximo declive. Para problemas quadráticos, na iteração k, o tamanho do passo proposto por Barzilai e Borwein coincide com o passo que teria sido calculado pela minimização exata na iteração $k-1$. Ao contrário do método de máximo declive com busca exata, o método de Barzilai-Borwein (BB) não garante o decréscimo da função objetivo a cada iteração. Entretanto, em testes computacionais, o desempenho do método de Barzilai-Borwein é muito melhor do que o do método de máximo declive com busca exata, exigindo menos esforço computacional. Esse resultado despertou o interesse da comunidade acadêmica em entender as propriedades do novo método e, possivelmente, propor um método de máximo declive eficiente para problemas de grande porte. Várias adaptações do método de Barzilai-Borwein foram propostas. Entre essas propostas estão o método de passo alternado e o método de Cauchy aleatório, que serão descritas adiante. Em 1993, Raydan [20] provou a convergência do método de Barzilai-Borwein para funções quadráticas estritamente convexas. Entretanto, para o caso geral, a possibilidade de se obter convergência superlinear foi descartada por Fletcher em [6]. Raydan propôs ainda, em [21] uma extensão do método para problemas de otimização irrestrita utilizando a busca não monótona de Grippo, Lampariello e Lucidi [12].

Esses novos métodos de máximo declive encorajaram a busca por um método de primeira ordem com complexidade melhor do que o método de máximo declive com busca exata. Esse resultado foi alcançado com sucesso por Gonzaga em [9]. Utilizando raízes de um

polinômio de Chebyshev, Gonzaga propõe uma maneira de resolver o problema quadrático com o método de máximo declive com o mesmo desempenho do pior caso do método dos gradientes conjugados, sendo esse o método de primeira ordem com o melhor desempenho possível para minimização quadrática.

A importância da busca por métodos eficientes para resolver problemas quadráticos se dá pelo fato de que uma solução ótima destes problemas é também uma solução de um sistema linear $Ax = b$, quando A é definida positiva. Desse modo, nosso objetivo é fazer um estudo detalhado sobre os métodos de máximo declive para minimização quadrática. Daremos ênfase à análise do comportamento da norma do gradiente no método de máximo declive com busca exata, baseando-nos nos resultados mostrados por Akaike [1], Forsythe [8], Nocedal, Sartenaer e Zhu [17]. Tais resultados garantem que o método de Cauchy converge assintoticamente para uma busca em um subespaço bidimensional. De Asmundis et al [5] propuseram em 2014 o método chamado *steepest descent with alignment* (SDA), que alterna o tamanho dos passos entre passos dados por busca exata e passos pequenos calculados de maneira eficiente. Após analisar o comportamento causado no gradiente devido à escolha de diferentes tamanhos de passo no método de máximo declive, vamos propor os métodos que chamaremos de *Cauchy-short* (CS) e *alternated Cauchy-short* (ACS), nos quais os tamanhos de passo alternam entre passos dados por busca exata e passos curtos, descritos por Gonzaga e Schneider em [10]. Além disso, apresentaremos algumas variações desses novos métodos utilizando um resultado proposto por Gonzaga em [9].

Para tanto, este trabalho está organizado em 5 capítulos. No primeiro capítulo apresentamos o método de máximo declive para o caso geral, definimos busca unidirecional e provamos a convergência global desse método. Enunciamos ainda o método de máximo declive para problemas de minimização quadrática e mostramos como é calculado o passo dado pela busca exata, que, nesse caso, chamaremos de passo de Cauchy. Ao final, realizamos uma mudança de variável na forma geral do problema quadrático. Essa mudança de variável será importante para a análise que será apresentada no Capítulo 4.

No segundo capítulo definimos a complexidade de um algoritmo e mostramos a complexidade do método de Cauchy para problemas quadráticos. Faremos uma breve introdução aos métodos de Krylov, enunciando sua complexidade no caso quadrático. Ainda, apresentamos o método proposto por Gonzaga em [9], o qual utiliza um polinômio de Chebyshev para construir um conjunto finito de tamanhos de passos a

serem dados no método de máximo declive. Ressaltamos que o desempenho deste algoritmo é da mesma ordem do desempenho do método de gradientes conjugados e, portanto, igual à complexidade do problema de programação quadrática.

No terceiro capítulo enunciamos o método de Barzilai-Borwein e algumas modificações propostas por outros pesquisadores, bem como o método de Barzilai-Borwein Global, proposto por Raydan [21].

No quarto capítulo, fazemos uma análise do comportamento do gradiente e das variáveis no método de máximo declive com busca exata para o problema quadrático (diagonalizado), classificando as variáveis do problema em leves e pesadas. Observamos o comportamento assintótico do método de máximo declive para funções quadráticas com busca exata. Além disso, mostramos que a sequência de comprimentos de passo calculados por busca exata a cada duas iterações também converge. Com isso, concluímos que o método de Cauchy gera sequências oscilatórias que se tornam ineficientes para resolver o problema de minimização quadrática.

No quinto capítulo avaliamos o efeito causado por passos, que definiremos como grandes e pequenos, no comportamento do gradiente da função quadrática. Com esse resultado, propomos novos métodos que utilizam passos de Cauchy intercalados com passos curtos. Aplicaremos, ainda, nestes novos métodos, a técnica de construir um conjunto finito de passos, utilizando raízes de um polinômio de Chebyshev. Ao final, exibimos os resultados de testes computacionais através de um gráfico de perfil de desempenho (*performance profile*), conforme proposto em [15], comparando os novos métodos com o método de Barzilai-Borwein.

Concluímos o trabalho evidenciando os resultados obtidos.

1 MÉTODO DE MÁXIMO DECLIVE

O método de máximo declive, também chamado de método do gradiente, é o algoritmo mais clássico em Otimização contínua. Este método, proposto por Cauchy [3], em 1847, é um método de primeira ordem, ou seja, utiliza apenas dados da função e do gradiente da função nos pontos obtidos a cada iteração. Neste capítulo apresentaremos o método de máximo declive no caso geral e no caso quadrático. Exibiremos dois tipos de busca unidirecional e mostraremos a convergência global no caso quadrático. Ao final, vamos definir o problema que será utilizado nos capítulos posteriores.

O problema de minimização diferenciável no caso geral é dado por:

$$\underset{x\in\Omega}{\text{minimize}}\, f(x)$$

em que $\Omega \subseteq \mathbb{R}^n$ é um conjunto fechado e $f : \mathbb{R}^n \longrightarrow \mathbb{R}$ é uma função diferenciável. A função f é chamada de função objetivo e o conjunto Ω é o conjunto viável.

Definição 1.1. *Dada uma função $f : \mathbb{R}^n \longrightarrow \mathbb{R}$ e um ponto $x^* \in \Omega$, dizemos que x^* é um minimizador local de f em Ω, se existe $\delta > 0$ tal que $f(x^*) \leq f(x)$ para todo $x \in B(x^*, \delta) \cap \Omega$. Se a condição for satisfeita para todo $x \in \Omega$, então x^* é um minimizador global. Se existe $\delta > 0$ tal que $f(x^*) < f(x)$ para todo $x \in B(x^*, \delta) \cap \Omega$, com $x \neq x^*$, então x^* é um minimizador estrito.*

Nesta dissertação, vamos nos ater ao caso irrestrito, ou seja, quando $\Omega = \mathbb{R}^n$:

$$\underset{x\in\mathbb{R}^n}{\text{minimize}}\, f(x) \tag{1.1}$$

com f de classe C^1.

1.1 MÉTODOS DE DESCIDA

Em um método de descida, a cada novo ponto obtido, escolhemos uma direção de descida, ou seja, um vetor de direção que pode proporcionar um decréscimo da função objetivo. Ainda, definimos a cada iteração um tamanho de passo $\lambda \in \mathbb{R}$ a ser dado na direção de descida escolhida, de maneira que o decrescimento da função seja garantido.

Definição 1.2. *Dada a função $f : \mathbb{R}^n \longrightarrow \mathbb{R}$, um ponto $x \in \mathbb{R}^n$ e uma direção $d \in \mathbb{R}^n \backslash \{0\}$, dizemos que d é direção de descida para f a partir de x se existe $\delta > 0$ tal que $f(x + \lambda d) < f(x)$ para todo $\lambda \in (0, \delta)$.*

Teorema 1.1. *Se $\nabla f(x)^T d < 0$, para algum $d \in \mathbb{R}^n$, então d é uma direção de descida para f a partir de x.*

Assim, podemos definir o algoritmo de descida no caso geral. Vamos denotar a norma Euclideana por $||.||$.

Algoritmo 1: Método de descida

Dados: $x^0 \in \mathbb{R}^n$, $k = 0$;

enquanto $||\nabla f(x^k)|| \neq 0$
- escolher $d^k \in \mathbb{R}^n$ tal que $\nabla f(x^k)^T d^k < 0$;
- escolher $\lambda_k > 0$ tal que $f(x^k + \lambda_k d^k) < f(x^k)$;
- $x^{k+1} = x^k + \lambda_k d^k$;
- $k = k + 1$;

fim

Resultado: x^k

Note que podemos obter vários métodos de descida, pois cada maneira de escolher a direção de descida e de determinar o tamanho de passo a ser dado define um algoritmo de descida diferente.

1.1.1 Direção de máximo declive

Dada uma função $f : \mathbb{R}^n \longrightarrow \mathbb{R}$ diferenciável e um ponto qualquer $x \in \mathbb{R}^n$, a direção de máximo declive de f no ponto x será a solução do seguinte problema:

$$\begin{array}{ll} \text{minimize} & f'(x, d) \\ \text{sujeito a} & ||d|| = 1 \end{array} \tag{1.2}$$

em que $f'(x, d)$ é a derivada direcional de $f(x)$ na direção d. Como f é diferenciável,

$$f'(x, d) = \langle \nabla f(x), d \rangle = \nabla f(x)^T d.$$

Afirmamos que a direção procurada é $d = \dfrac{-\nabla f(x)}{||\nabla f(x)||}$. Considere $d = \dfrac{-\nabla f(x)}{||\nabla f(x)||}$ e $v \in \mathbb{R}^n$ tal que $||v|| = 1$. Vamos mostrar que

$\nabla f(x)^T d \leq \nabla f(x)^T v$. De fato,

$$\nabla f(x)^T d = -\nabla f(x)^T \frac{\nabla f(x)}{||\nabla f(x)||} = -||\nabla f(x)||.$$

Pela desigualdade de Cauchy-Schwarz temos:

$$|\nabla f(x)^T v| \leq ||\nabla f(x)||.||v||$$

em que a igualdade vale se $\nabla f(x)$ e v forem colineares. Assim,

$$\nabla f(x)^T v \geq -||\nabla f(x)||.$$

Logo,

$$\nabla f(x)^T d \leq \nabla f(x)^T v$$

para qualquer $v \in \mathbb{R}^n$ tal que $||v|| = 1$.

Assim, a direção dada por $\frac{-\nabla f(x)}{||\nabla f(x)||}$ é a direção de máximo declive de f a partir de x.

1.1.2 Método de máximo declive

O método de máximo declive é um tipo de método de descida. Tal método foi proposto por Cauchy em 1847 [3] e é definido da seguinte maneira:

Algoritmo 2: Método de máximo declive

Dados: $x^0 \in \mathbb{R}^n$, $k = 0$;
enquanto $||\nabla f(x^k)|| \neq 0$
  encontrar um passo λ_k tal que
  $f(x^k - \lambda_k \nabla f(x^k)) < f(x^k)$;
  $x^{k+1} = x^k - \lambda_k \nabla f(x^k)$;
  $k = k + 1$;
fim
Resultado: x^k

Nesse método, a direção escolhida a cada iteração será a direção oposta ao gradiente da função no ponto x^k pois, como vimos, essa é a direção de máximo declive. É claro que, computacionalmente, não vamos obter $||\nabla f(x^k)|| = 0$. Na prática, encontramos um resultado aproximado. Para isso, escolhemos uma tolerância $\varepsilon > 0$ e algum critério

de parada, que pode depender de $f(x^k)$ ou $||\nabla f(x^k)||$. Por exemplo,

(i) $||\nabla f(x^k)|| \leq \varepsilon$

(ii) $f(x^k) - f(x^*) \leq \varepsilon$

(iii) $||x - x^*|| \leq \varepsilon$

(iv) $f(x^k) - f(x^*) \leq \varepsilon(f(x^0) - f(x^*))$

sendo x^* uma solução ótima. O primeiro critério é utilizado na prática. Como no segundo critério o valor ótimo da função deve ser conhecido e, nos dois últimos casos, as soluções ótimas do problema devem ser conhecidas, esses critérios são utilizados apenas para análise teórica dos algoritmos.

1.2 BUSCA UNIDIRECIONAL

A escolha do tamanho do passo λ a ser utilizado no algoritmo de descida, a partir de um ponto x^k, é outra decisão a ser tomada a cada iteração. Veremos que nem sempre o passo que proporciona o maior decrescimento a cada iteração será a melhor escolha para que o algoritmo resolva o problema com um menor número de iterações.

Em geral, definimos um critério para a escolha do passo λ, chamado de busca unidirecional. Existem vários tipos de busca. Uma delas é a busca exata, na qual o tamanho do passo proporciona o maior decrescimento possível da função f na direção de descida escolhida. Em muitos casos, a busca exata pode ser demorada ou demandar um alto custo computacional. Podemos realizar uma busca inexata, que também proporcionará decrescimento na função, porém não será necessariamente o maior possível.

O método de Armijo, por exemplo, é um tipo de busca inexata muito utilizado. A busca de Armijo consiste em encontrar um valor λ, dada uma direção d e um ponto x, tal que:

$$f(x + \lambda d) \leq f(x) + \eta\lambda\nabla f(x)^T d$$

com $\eta \in (0, 1)$. Em geral, utiliza-se $\eta < 0,5$.

De fato, podemos fazer essa busca devido ao próximo resultado[1].

Teorema 1.2. *Dada uma função $f : \mathbb{R}^n \longrightarrow \mathbb{R}$, um ponto $x \in \mathbb{R}^n$, uma direção de descida $d \in \mathbb{R}^n \backslash \{0\}$ e um parâmetro $\eta \in (0, 1)$, existe*

[1] A demonstração deste resultado pode ser encontrada em [13].

$\delta > 0$ tal que

$$f(x + \lambda d) \leq f(x) + \eta\lambda\nabla f(x)^T d$$

para todo $\lambda \in [0, \delta)$.

Na prática, a busca de Armijo é feita da seguinte forma: fixamos um valor $\sigma \in (0, 1)$ e iniciamos com $\lambda = 1$. Se para esse λ obtivermos

$$f(x + \lambda d) > f(x) + \eta\lambda\nabla f(x)^T d$$

fazemos $\lambda = \sigma\lambda$ quantas vezes for necessário até obtermos

$$f(x + \lambda d) \leq f(x) + \eta\lambda\nabla f(x)^T d.$$

1.3 CONVERGÊNCIA GLOBAL

Vamos mostrar que, para funções de classe C^1, o método de máximo declive converge globalmente segundo a próxima definição.

Definição 1.3. *Um algoritmo é dito globalmente convergente para o problema* (1.1) *se para qualquer sequência (x^k) gerada pelo algoritmo e qualquer ponto de acumulação $\bar{x}$ de (x^k), temos que $\bar{x}$ é estacionário, isto é, $\nabla f(\bar{x}) = 0$.*

Para demonstrar a convergência global do método de máximo declive com busca linear exata, seguiremos a demonstração apresentada em [13] e utilizaremos o seguinte resultado clássico de Análise:

Teorema 1.3. *Seja (y^k) uma sequência monótona em $\mathbb{R}$ que possui subsequência convergindo a $\bar{y}$, então, $y^k \to \bar{y}$.*

Teorema 1.4. *O método de máximo declive com busca exata é globalmente convergente para o problema* (1.1).

Demonstração: Seja (x^k) uma sequência gerada pelo algoritmo, $\bar{x}$ um ponto de acumulação de (x^k) e K um conjunto de índices que forma uma subsequência de (x^k) tal que $x^k \xrightarrow{K} \bar{x}$. Suponha, por contradição, que $\bar{x}$ não é estacionário, isto é, $\nabla f(\bar{x}) \neq 0$. Tome $\bar{d} = -\nabla f(\bar{x})$. Pelo Teorema 1.1, $\bar{d}$ é direção de descida. De fato,

$$-\nabla f(\bar{x})^T \nabla f(\bar{x}) = -||\nabla f(\bar{x})^T|| < 0.$$

Portanto, existem $\lambda > 0$ e $\beta > 0$ tais que

$$f(\overline{x}) - f(\bar{x} + \lambda\bar{d}) > \beta > 0.$$

Considere a função $h : \mathbb{R}^n \longrightarrow \mathbb{R}$ dada por $h(x) = f(x) - f(x - \lambda \nabla f(x))$. Como f e ∇f são contínuas, temos que h é contínua.
Assim, $h(x^k) \xrightarrow{K} h(\bar{x}) > \beta$. Portanto, para $k \in K$ suficientemente grande,

$$f(x^k) - f(x^k - \lambda \nabla f(x^k)) = h(x^k) \geq \frac{\beta}{2}.$$

Como λ_k foi obtido através de uma busca exata, então, λ_k é o minimizador de $f(x)$ na direção $-\nabla f(x^k)$, portanto, para $k \in K$, suficientemente grande,

$$f(x^{k+1}) = f(x^k - \lambda_k \nabla f(x^k)) \leq f(x^k - \lambda \nabla f(x^k)) \leq f(x^k) - \frac{\beta}{2},$$

$$f(x^k) - f(x^{k+1}) \geq \frac{\beta}{2}. \tag{1.3}$$

Sendo f contínua, temos que $f(x^k) \xrightarrow{K} f(\bar{x})$. Como $(f(x^k))$ é uma sequência decrescente, pelo Teorema 1.3, $f(x^k) \xrightarrow{\mathbb{N}} f(\bar{x})$, o que contradiz a desigualdade (1.3). Logo, $\overline{x}$ é ponto estacionário de f.

■

Note que esse resultado não garante que a sequência de pontos (x^k), gerada pelo algoritmo com busca exata, converge para $\bar{x}$. Mostramos apenas que os pontos de acumulação dessa sequência são estacionários.

1.4 MÉTODO DE MÁXIMO DECLIVE PARA QUADRÁTICAS

A partir de agora, vamos estudar o caso em que a função objetivo, denotada por $\bar{f}$, é uma função quadrática $\bar{f} : \mathbb{R}^n \longrightarrow \mathbb{R}$ da forma:

$$\bar{f}(z) = \frac{1}{2} z^T A z + b^T z \tag{1.4}$$

em que $b \in \mathbb{R}^n$ e $A \in \mathbb{R}^{n \times n}$ é uma matriz simétrica definida positiva. Assim, temos o seguinte problema de minimização:

$$\underset{z \in \mathbb{R}^n}{\text{minimize}} \frac{1}{2} z^T A z + b^T z. \tag{$\bar{P}$}$$

Como A é definida positiva, possui autovalores $d_1, \ldots, d_n$ todos positivos. Vamos supor, sem perda de generalidade, que

$$0 < d_1 < d_2 < \ldots < d_n.$$

Note que a função $\bar{f}$ é estritamente convexa e seu minimizador é a única solução da equação $Az = -b$.

1.4.1 Busca exata

O passo obtido pela busca exata no algoritmo de máximo declive foi estudado por Cauchy. Por isso, denotaremos esse passo por $\bar{\lambda}^C$. Quando a função a ser minimizada é quadrática, nas condições do problema $(\bar{P})$, o minimizador será o ponto $z \in \mathbb{R}^n$ que satisfaz:

$$\nabla \bar{f}(z) = Az + b = 0.$$

Para simplificar a notação no estudo do algoritmo de máximo declive, de agora em diante, utilizaremos $\nabla \bar{f}(z^k) = \bar{g}^k$ para todo $k \in \mathbb{N}$. Portanto, na iteração k do algoritmo de máximo declive, o passo $\bar{\lambda}_k^C$ será o valor que minimiza a função

$$\lambda \in \mathbb{R} \longmapsto \bar{f}(z^k - \lambda \bar{g}^k),$$

ou seja, $\lambda \in \mathbb{R}$ tal que $\dfrac{d}{d\lambda} \bar{f}(z^k - \lambda \bar{g}^k) = 0$. Assim, temos

$$\begin{aligned} 0 &= \frac{d}{d\lambda} \bar{f}(z^k - \lambda \bar{g}^k) = -\nabla \bar{f}(z^k - \lambda \bar{g}^k)^T \bar{g}^k \\ &= -(A(z^k - \lambda \bar{g}^k) + b)^T \bar{g}^k = -(Az^k + b)^T \bar{g}^k + \lambda (A \bar{g}^k)^T \bar{g}^k \\ &= -{\bar{g}^k}^T \bar{g}^k + \lambda {\bar{g}^k}^T A^T \bar{g}^k = -{\bar{g}^k}^T \bar{g}^k + \lambda {\bar{g}^k}^T A \bar{g}^k \end{aligned}$$

pois A é simétrica. Logo, o passo $\bar{\lambda}_k^C$ será dado por:

$$\bar{\lambda}_k^C = \frac{{\bar{g}^k}^T \bar{g}^k}{{\bar{g}^k}^T A \bar{g}^k}. \tag{1.5}$$

Note que ${\bar{g}^k}^T A \bar{g}^k \neq 0$, pois A é definida positiva e $\bar{g}^k \neq 0$, uma vez que, se $\bar{g}^k = 0$, então temos que z^k é solução ótima.

Algoritmo 3: Método de Cauchy

Dados: $z^0 \in \mathbb{R}^n$, $\bar{g}_0 = Az^0 + b$, $k = 0$;
enquanto $||\nabla f(z^k)|| \neq 0$
$\bar{\lambda}_k^C = \dfrac{\bar{g}^{k\,T}\bar{g}^k}{\bar{g}^{k\,T} A\,\bar{g}^k}$;
$z^{k+1} = z^k - \bar{\lambda}_k^C\,\bar{g}^k$;
$\bar{g}^{k+1} = Az^{k+1} + b$;
$k = k + 1$;
fim
Resultado: $z = z^k$

Esse é um exemplo do método de máximo declive com busca exata, no qual o passo $\bar{\lambda}_k^C$, calculado em cada iteração, é o que minimiza a função na direção do gradiente. De agora em diante, vamos nos referir a este método apenas como método de Cauchy. As diferentes propostas de escolha de passo receberão outros nomes.

Note que, no formato padrão do algoritmo, a cada iteração realizamos duas multiplicações matriciais, uma ao calcular o tamanho do passo e outra ao calcular o gradiente da função. Entretanto, é possível efetuar apenas uma multiplicação matricial por iteração, basta guardar o valor de $h^k = A\,\bar{g}^k$. Assim, temos para o passo de Cauchy,

$$\bar{\lambda}_k^C = \frac{\bar{g}^{k\,T}\bar{g}^k}{\bar{g}^{k\,T} h^k},$$

e para $\bar{g}^{k+1}$,

$$\bar{g}^{k+1} = Az^{k+1}+b = A(z^k - \bar{\lambda}_k^C\,\bar{g}^k)+b = Az^k+b-\bar{\lambda}_k^C A\,\bar{g}^k = \bar{g}^k - \bar{\lambda}_k^C h^k.$$

Além disso, não calculamos o valor da função nos pontos obtidos, que no problema quadrático, exige mais operações do que o cálculo do gradiente. Esta é uma observação importante, pois multiplicações matriciais são lentas computacionalmente, portanto, procuramos evitá-las.

1.4.2 Mudança de variável

Considere o problema na forma geral ($\bar{P}$):

$$\underset{z\in\mathbb{R}^n}{\text{minimize}}\ \frac{1}{2}z^TAz + b^Tz. \tag{$\bar{P}$}$$

Assim, para $z \in \mathbb{R}^n$, temos

$$\nabla\bar{f}(z) = Az + b \quad \text{e} \quad \nabla^2\bar{f}(z) = A.$$

Suponha que z^* seja uma solução desse problema. Dessa maneira,

$$\nabla\bar{f}(z^*) = Az^* + b = 0,$$

ou seja, $Az^* = -b$.

É possível transformar o problema quadrático ($\bar{P}$) no problema:

$$\underset{y\in\mathbb{R}^n}{\text{minimize}}\ \tilde{f}(y) = \frac{1}{2}y^TAy, \tag{1.6}$$

cuja solução é $y^* = 0$. De fato, se fizermos a substituição $y = z - z^*$ no problema ($\bar{P}$), obtemos que

$$\tilde{f}(y) = \bar{f}(z) - \bar{f}(z^*).$$

Como $\bar{f}(z^*)$ é um valor fixo, minimizar $\bar{f}(z)$ é equivalente a minimizar $\tilde{f}(y)$.

Faremos essa mudança de variável para estudar as propriedades dos algoritmos, uma vez que conhecemos a solução do problema tratado, neste caso, $y^* = 0$.

É possível simplificar ainda mais o problema. Para tanto, utilizaremos os seguintes resultados de álgebra linear:

Definição 1.4. *Duas matrizes quadradas A e D são similares se existe uma matriz inversível P tal que*

$$A = PDP^{-1}.$$

Definição 1.5. *Uma matriz quadrada A é diagonalizável se for similar a uma matriz diagonal D.*

Teorema 1.5. *Matrizes similares têm os mesmos autovalores.*

Teorema 1.6. *Se A é uma matriz simétrica, então autovetores asso-*

ciados a autovalores distintos de A são ortogonais.

Definição 1.6. *Uma matriz quadrada P é ortogonal se $P^{-1} = P^T$.*

Teorema 1.7. *As colunas (e linhas) de uma matriz quadrada P são ortonormais se e somente se a matriz é ortogonal.*

Teorema 1.8. *Se A é uma matriz simétrica, então existe uma matriz ortogonal P tal que*

$$P^T AP = D,$$

de maneira que D seja uma matriz diagonal. Além disso, a diagonal de D é composta pelos autovalores de A, e P é composta por autovetores ortonormais associados aos autovalores de A.

Como no problema $(\bar{P})$, a matriz A é simétrica definida positiva e já supomos que

$$0 < d_1 < \ldots < d_n,$$

em que $d_1, \ldots, d_n$ são os autovalores de A, temos

$$AP = PD.$$

A matriz P é composta por autovetores ortonormais associados aos autovalores de A. Vamos fazer a seguinte mudança de variável: $x = P^T y$, ou seja, $y = Px$. Portanto, o problema na forma reduzida se torna

$$\begin{aligned} \tilde{f}(y) &= \frac{1}{2} y^T A y = \frac{1}{2} y^T (PDP^T) y \\ &= \frac{1}{2} (P^T y)^T D P^T y = \frac{1}{2} x^T D x. \end{aligned}$$

Doravante estudaremos o problema de minimização na forma mais simplificada:

$$\underset{x \in \mathbb{R}^n}{\text{minimize}} \, \frac{1}{2} x^T D x \tag{P}$$

em que D é uma matriz diagonal. Desse modo, conhecemos os autovalores de D, que são os elementos da sua diagonal. Conhecer o maior e o menor autovalor da matriz D é interessante, pois assim saberemos seu número de condicionamento C, que será muito utilizado nos resultados de complexidade e é dado por:

$$C = \frac{d_n}{d_1}.$$

A partir de agora, denotaremos a função objetivo deste problema por $f(x)$, ou seja,

$$f(x) = \frac{1}{2}x^T Dx.$$

Observamos ainda que valem as seguintes igualdades:

(i) $||\nabla f(x)|| = ||\nabla \tilde{f}(y)||$
De fato,

$$\begin{aligned} ||\nabla \tilde{f}(y)||^2 &= \langle Ay, Ay\rangle \\ &= \langle PDP^T y, PDP^T y\rangle \\ &= \langle PDx, PDx\rangle \\ &= x^T D^2 x \\ &= ||\nabla f(x)||^2. \end{aligned}$$

(ii) $||x - x^*|| = ||y - y^*||$
Como $x^* = 0$ e $y^* = 0$, temos

$$\begin{aligned} ||x - x^*||^2 &= ||x||^2 \\ &= \langle P^T y, P^T y\rangle \\ &= \langle y, PP^T y\rangle \\ &= ||y||^2 = ||y - y^*||^2. \end{aligned}$$

(iii) $\tilde{f}(y) = f(x)$ De fato,

$$\begin{aligned} \tilde{f}(y) &= \frac{1}{2}y^T Ay \\ &= \frac{1}{2}y^T PDP^T y \\ &= \frac{1}{2}x^T Dx \\ &= f(x). \end{aligned}$$

Assim, para o estudo de propriedades dos algoritmos, podemos considerar o algoritmo de Cauchy para o caso simplificado, no qual

denotaremos $\nabla f(x^k)$ por g^k e o passo utilizado será dado por:

$$\lambda_k^C = \frac{{g^k}^T g^k}{{g^k}^T D\, g^k}.$$

Algoritmo 4: Método de Cauchy simplificado

Dados: $x^0 \in \mathbb{R}^n$, $g^0 = Dx^0$, $k = 0$;
enquanto $||\nabla f(x^k)|| \neq 0$
  $\lambda_k^C = \dfrac{{g^k}^T g^k}{{g^k}^T D\, g^k}$;
  $x^{k+1} = x^k - \lambda_k^C\, g^k$;
  $g^{k+1} = Dx^{k+1}$;
  $k = k + 1$;
fim
Resultado: $x = x^k$

Com a mudança de variáveis que escolhemos, temos $y = z - z^*$ e $x = Py$, logo, $x = Pz - Pz^*$. Ou seja,

$$z = P^{-1}x + z^*.$$

Desse modo, se aplicarmos o algoritmo de Cauchy para o problema $(\bar{P})$, a partir do ponto z^0, com passos calculados por

$$\bar{\lambda}_k^C = \frac{{\bar{g}^k}^T \bar{g}^k}{{\bar{g}^k}^T A\, \bar{g}^k}$$

e se aplicarmos o Algoritmo 4 para o problema (P) a partir do ponto $x^0 = Pz^0 - Pz^*$ com passos calculados por

$$\lambda_k^C = \frac{{g^k}^T g^k}{{g^k}^T D\, g^k},$$

então os passos $\bar{\lambda}_k^C$ e λ_k^C irão coincidir e os pontos x^k e z^k serão correspondentes.

Portanto, para estabelecer as propriedades de convergência do algoritmo de Cauchy, estudaremos as propriedades do Algoritmo 4.

A seguir, exibimos um exemplo em que utilizamos o método de Cauchy para resolver um problema quadrático na forma simplificada. A função objetivo é $f(x) = \frac{1}{2}x^T Dx$, de 1000 variáveis, em que D é uma matriz diagonal com autovalores não nulos e distintos entre si, distribuídos uniformemente. Nesse exemplo, o número de condicionamento de D é 1000. A figura 1 mostra a variação da função ao longo das iterações, em que foi usada escala logarítmica no eixo vertical.

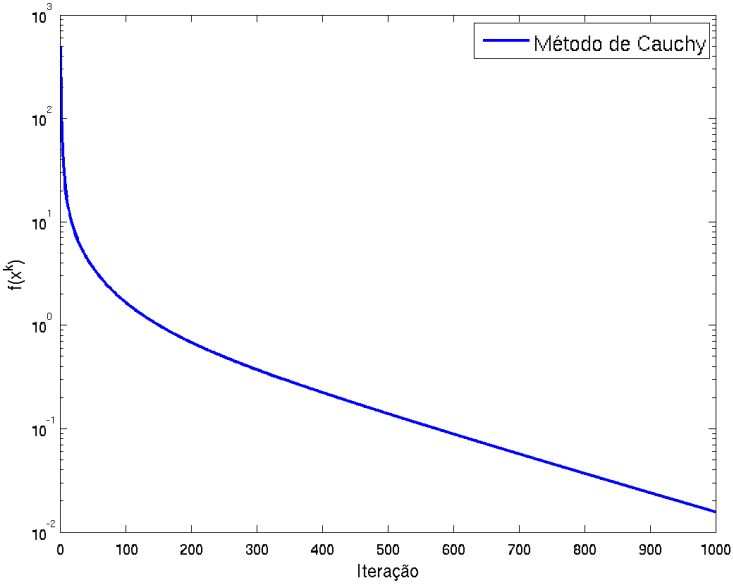

Figura 1 – Valores da função ao longo das iterações no método de Cauchy na resolução de uma função quadrática de 1000 variáveis com número de condicionamento igual a 1000.

Em geral, utilizamos como critério de parada

$$f(x^k) - f(x^*) \leq \varepsilon(f(x^0) - f(x^*)).$$

Como no problema diagonalizado $f(x^*) = 0$, utilizamos $f(x^k) \leq \varepsilon f(x^0)$. Para o exemplo apresentado, temos $f(x^0) = 500$ e $\varepsilon = 10^{-10}$, ou seja, o critério de parada seria $f(x^k) \leq 5 \times 10^{-8}$, porém utilizamos $k = 1000$ devido à lenta convergência do método.

2 RESULTADOS DE COMPLEXIDADE

A complexidade de um problema depende de duas situações: o tipo de oráculo e do desempenho do melhor algoritmo.

Chamamos de oráculo o tipo de informações disponíveis sobre a função objetivo em um ponto dado, que no caso estudado são $f(x)$ e $\nabla f(x)$. Como utilizamos apenas $f(x)$ e $\nabla f(x)$, esse oráculo é dito ser de primeira ordem. O desempenho de qualquer algoritmo será um limitante superior para a complexidade.

O estudo de desempenho de algoritmos resume-se em, dado um critério de tolerância ε para a solução do problema, analisar a quantidade máxima de iterações para que essa tolerância seja atingida em qualquer situação, ou seja, procuramos um número de iterações suficientes para que o problema seja resolvido no pior caso possível. Esse limitante para o número de iterações pode depender não apenas da tolerância ε, mas também do ponto inicial, da dimensão do problema, do número de condição, etc.

Enfim, chamamos de complexidade do problema, o desempenho do melhor algoritmo possível com determinado tipo de oráculo. A complexidade é conhecida para alguns problemas, porém, desconhecida para muitos.

Apresentaremos dois resultados clássicos de complexidade: o desempenho do método de Cauchy com busca exata para o problema quadrático diagonalizado (P), dado por

$$k \leq \left\lceil \frac{C}{4} \log\left(\frac{1}{\varepsilon}\right) \right\rceil \text{ [1]},$$

e o desempenho do método dos gradientes conjugados, dado por

$$k \leq \left\lceil \frac{\sqrt{C}}{2} \log\left(\frac{2}{\varepsilon}\right) \right\rceil \text{ [2]}.$$

Sabe-se que o método de gradientes conjugados é o melhor algoritmo de primeira ordem para funções quadráticas, ou seja, a complexidade do problema com oráculo de primeira ordem é $k \leq \left\lceil \frac{\sqrt{C}}{2} \log\left(\frac{2}{\varepsilon}\right) \right\rceil$.

[1] A função teto, denotada por $\lceil x \rceil$, é o menor inteiro maior do que x, para todo $x \in \mathbb{R}$.

[2] Esses resultados podem ser encontrados em [11].

Em 2014, Gonzaga mostrou em [9] que, conhecendo o maior e o menor autovalor da matriz A, é possível construir uma sequência de passos a serem dados pelo algoritmo de máximo declive que atinge a tolerância em $k \leq \left\lceil \frac{\sqrt{C}}{2} \log\left(\frac{2}{\varepsilon}\right) \right\rceil$ para qualquer ponto inicial. Além disso, se não forem conhecidos os autovalores de A, existe um algoritmo, cujo desempenho é $k \leq \left\lceil 3\sqrt{C} \log\left(\frac{2}{\varepsilon}\right) \right\rceil$.

2.1 MÉTODO DE CAUCHY

Vamos estudar o desempenho do algoritmo de Cauchy para funções quadráticas, utilizando o critério de parada

$$f(x^k) - f(x^*) \leq \varepsilon(f(x^0) - f(x^*)),$$

com $\varepsilon > 0$. Procuraremos um limitante para o número de iterações para resolver o problema no pior caso possível. Para isso, vamos utilizar os dois lemas a seguir, que estão provados em Luenberger e Ye [14, p.236].

Lema 2.1. *O algoritmo de Cauchy para o problema* (P) *satisfaz*

$$f(x^{k+1}) = \left(1 - \frac{({g^k}^T\, g^k)^2}{({g^k}^T\, D\, g^k)(g^k\, D^{-1}\, g^k)}\right) f(x^k).$$

Demonstração: Para facilitar a notação, vamos utilizar $\lambda_k^C = \lambda_k$. Assim, temos

$$\begin{aligned}
\frac{f(x^k) - f(x^{k+1})}{f(x^k)} &= \frac{\frac{1}{2}(x^k)^T D x^k - \frac{1}{2}(x^k - \lambda_k\, g^k)^T D (x^k - \lambda_k\, g^k)}{\frac{1}{2}(x^k)^T D x^k} \\
&= \frac{\frac{1}{2}(x^k)^T D \lambda_k\, g^k + \frac{1}{2}(\lambda_k\, g^k)^T D x^k - \frac{1}{2}\lambda_k^2\, {g^k}^T\, D\, g^k}{\frac{1}{2}(x^k)^T D x^k} \\
&= \frac{(\lambda_k\, g^k)^T D x^k - \frac{1}{2}\lambda_k^2\, {g^k}^T\, D\, g^k}{\frac{1}{2}(x^k)^T D x^k}.
\end{aligned}$$

Como $Dx^k = g^k$ e $\lambda_k = \dfrac{{g^k}^T\, g^k}{{g^k}^T\, D\, g^k}$, temos

$$\begin{aligned}\frac{f(x^k)-f(x^{k+1})}{f(x^k)} &= \frac{\dfrac{2({g^k}^T g^k)^2}{({g^k}^T D\, g^k)} - \dfrac{({g^k}^T g^k)^2}{({g^k}^T D\, g^k)}}{{g^k}^T D^{-1}\, g^k} \\ &= \frac{({g^k}^T g^k)^2}{({g^k}^T D\, g^k)({g^k}^T D^{-1}\, g^k)}.\end{aligned}$$

Logo,

$$f(x^{k+1}) = \left\{1 - \frac{({g^k}^T g^k)^2}{({g^k}^T D\, g^k)({g^k}\, D^{-1}\, g^k)}\right\} f(x^k).$$

■

Lema 2.2 (Desigualdade de Kantorovich)**.** *Seja $A \in \mathbb{R}^{n\times n}$ uma matriz simétrica definida positiva com autovalores $d_1, \ldots, d_n$ distintos. Então, para qualquer $x \in \mathbb{R}^n$ temos*

$$\frac{(x^Tx)^2}{(x^TAx)(x^TA^{-1}x)} \geq \frac{4d_1d_n}{(d_1+d_n)^2}$$

em que d_1 e d_n são o menor e o maior autovalores respectivamente.

Demonstração: Sejam $d_1, d_2, \ldots, d_n$ autovalores de A com

$$0 < d_1 < d_2 < \ldots < d_n.$$

Fazendo uma mudança de coordenadas (utilizando y), é possível diagonalizar a matriz A:

$$\frac{(x^Tx)^2}{(x^TAx)(x^TA^{-1}x)} = \frac{(\sum_{i=1}^n y_i^2)^2}{(\sum_{i=1}^n d_iy_i^2)(\sum_{i=1}^n (y_i^2)/d_i)}.$$

Denotando $\xi_i = \dfrac{y_i^2}{\sum_{i=1}^n y_i^2}$, temos

$$\frac{(x^Tx)^2}{(x^TAx)(x^TA^{-1}x)} = \frac{1/(\sum_{i=1}^n \xi_i d_i)}{(\sum_{i=1}^n \xi_i/d_i)}.$$

O valor mínimo dessa razão é atingido para algum $d = \xi_1 d_1 + \xi_n d_n$,

com $\xi_1 + \xi_n = 1$. Note que

$$\frac{\xi_1}{d_1} + \frac{\xi_n}{d_n} = \frac{(d_1 + d_n - \xi_1 d_1 - \xi_n d_n)}{d_1 d_n}.$$

Assim, essa razão é dada por

$$\frac{1/(\sum_{i=1}^{n} \xi_i d_i)}{(\sum_{i=1}^{n} \xi_i / d_i)} \geq \lim_{d_1 \leq d \leq d_n} \frac{(1/d)}{(d_1 + d_n - d)/(d_1 d_n)}$$

com o mínimo sendo atingido em $d = (d_1 d_n)/2$. Portanto,

$$\frac{1/(\sum_{i=1}^{n} \xi_i d_i)}{(\sum_{i=1}^{n} \xi_i / d_i)} \geq \frac{4 d_1 d_n}{(d_1 + d_n)^2}.$$

■

Observação: Note que esse mesmo resultado vale para a matriz diagonal D do problema (P).

Teorema 2.1. *Para todo ponto inicial $x^0 \in \mathbb{R}^n$, o método de Cauchy para o problema* (P) *converge para o único valor ótimo x^* de f; e para toda iteração k, temos*

$$f(x^{k+1}) \leq \left(\frac{d_n - d_1}{d_n + d_1}\right)^2 f(x^k).$$

Demonstração: Pelo Lema 2.1 e pela desigualdade de Kantorovich, temos

$$f(x^{k+1}) \leq \left(1 - \frac{4 d_1 d_n}{(d_n + d_1)^2}\right) f(x^k) = \left(\frac{d_n - d_1}{d_n + d_1}\right)^2 f(x^k).$$

Como $d_1 \neq 0$, temos

$$\frac{d_n - d_1}{d_n + d_1} < 1,$$

portanto,

$$f(x^{k+1}) < f(x^k).$$

Logo

$$f(x^k) = 1/2 (x^k)^T D x^k \to 0.$$

Além disso, como D é definida positiva, temos que $x^k \to 0$.

■

Desse modo, temos o seguinte resultado.

Teorema 2.2. *Considere o problema* (P) *e seja* $C = \dfrac{d_n}{d_1} > 1$ *o número de condicionamento da matriz D. Então, o desempenho do algoritmo de Cauchy, iniciando em um ponto* $x^0 \in \mathbb{R}^n$*, com o critério de parada* $f(x^k) - f(x^*) \leq \varepsilon(f(x^0) - f(x^*))$*, para* $\varepsilon > 0$ *fixado, é dado por*

$$k \leq \left\lceil \frac{C}{4} log\left(\frac{1}{\varepsilon}\right) \right\rceil.$$

Demonstração: Da desigualdade de Kantorovich, temos

$$f(x^{k+1}) \leq \left(\frac{d_n - d_1}{d_n + d_1}\right)^2 f(x^k),$$

ou seja,

$$f(x^k) \leq \left(\frac{C-1}{C+1}\right)^2 f(x^{k-1}).$$

Recursivamente, obtemos

$$\frac{f(x^k)}{f(x^0)} \leq \left(\frac{C-1}{C+1}\right)^{2k} = \left(\frac{C-1}{C+1}\right)^{(\frac{2k}{C})C}.$$

Como ambos os lados da desigualdade são positivos e o logaritmo é uma função estritamente crescente, temos

$$\log\left(\frac{f(x^k)}{f(x^0)}\right) \leq \frac{2k}{C} \log\left(\frac{C-1}{C+1}\right)^C.$$

Para $t \in (1, \infty)$, a função $t \mapsto \left(\dfrac{t-1}{t+1}\right)^t$ é crescente e

$$\left(\frac{t-1}{t+1}\right)^t \leq \lim_{t\to\infty} \left(\frac{t-1}{t+1}\right)^t = \frac{1}{e^2}.$$

Logo,

$$\log\left(\frac{f(x^k)}{f(x^0)}\right) \leq \frac{2k}{C} \log\left(\frac{1}{e^2}\right) = \frac{-4k}{C}.$$

Se na iteração k o critério $f(x^k) - f(x^*) \leq \varepsilon(f(x^0) - f(x^*))$ não for

atingido, temos $\frac{f(x^k)}{f(x^0)} > \varepsilon$, pois $f(x^*) = 0$. Logo,

$$\log(\varepsilon) < \log\left(\frac{f(x^k)}{f(x^0)}\right) \leq -\frac{4k}{C}.$$

Consequentemente,

$$k \leq \frac{C}{4}\log\left(\frac{1}{\varepsilon}\right).$$

■

2.2 PASSOS CURTOS

Vamos considerar agora o método de máximo declive com o passo λ_k fixado por $\lambda_k = 1/d_n$. É possível mostrar[3] que o desempenho desse algoritmo também é da ordem de $C\log\left(\frac{1}{\varepsilon}\right)$.

Teorema 2.3. *Considere o problema* (P) *e seja* $C = \frac{d_n}{d_1} \geq 1$ *o número de condicionamento da matriz D. Então, o desempenho do algoritmo de máximo declive com tamanho de passo fixado por* $\lambda_k = 1/d_n$*, para todo k, iniciando em um ponto* $x^0 \in \mathbb{R}^n$ *e com critério de parada* $f(x^k) - f(x^*) \leq \varepsilon(f(x^0) - f(x^*))$*, com* $\varepsilon > 0$ *fixado, é dado por*

$$k \leq \left\lceil \frac{C}{2} log\left(\frac{1}{\varepsilon}\right) \right\rceil.$$

2.3 MÉTODOS DE KRYLOV

Para minimização quadrática, dentre os algoritmos que utilizam apenas informações de primeira ordem, o algoritmo de Krylov é o melhor possível[4]. Vamos descrever brevemente a estrutura deste método que pode ser encontrado em [11].

Considere o problema

$$\underset{x\in\mathbb{R}^n}{\text{minimize}}\, f(x) = \frac{1}{2}x^T Dx \qquad \text{(P)}$$

[3] A demonstração desse resultado é semelhante à que foi apresentada no Teorema 2.2 e pode ser encontrada em [11].

[4] Este resultado pode ser encontrado em [19]

em que D é uma matriz diagonal cujos elementos da diagonal são não nulos e positivos.

O método de Krylov, para este problema, segue a seguinte estrutura: dado um ponto inicial x^0, definimos o espaço

$$V_1 = \{x^0 + \lambda \nabla f(x^0) : \lambda \in \mathbb{R}\} = x^0 + \text{span}\{\nabla f(x^0)\} = x^0 + \text{span}\{Dx^0\}.$$

Definimos x^1 como sendo a solução do problema auxiliar

$$x^1 = \underset{x \in V_1}{\text{argmin}}\, f(x) = x^0 + \lambda \nabla f(x^0) \qquad (P_1)$$

para algum $\lambda \in \mathbb{R}$. Nesse caso, λ é o passo de Cauchy, ou seja, $\lambda = -\lambda_1^C$.

Para o segundo passo, definimos o espaço

$$V_2 = x^0 + \text{span}\{\nabla f(x^0), \nabla f(x^1)\},$$

ou seja,

$$V_2 = x^0 + \text{span}\{Dx^0, D^2x^0\}.$$

Assim, definimos x^2 como a solução do problema bidimensional

$$x^2 = \underset{x \in V_2}{\text{argmin}}\, f(x).$$

Recursivamente, definimos o espaço

$$V_k = x^0 + \text{span}\{Dx^0, \ldots, D^k x^0\}$$

e o próximo ponto será definido pela solução do problema k-dimensional

$$x^k = \underset{x \in V_k}{\text{argmin}}\, f(x).$$

Note que pela maneira como foram definidos os pontos x^k, $\nabla f(x^k)$ é ortogonal ao espaço V_k, logo, é linearmente independente com

$$\{\nabla f(x^0), \ldots, \nabla f(x^{k-1})\}.$$

Assim, o problema será resolvido em no máximo n iterações, uma vez que $V_n = \mathbb{R}^n$.

Definição 2.1. *Seja $A \in \mathbb{R}^{n \times n}$ uma matriz definida positiva. Os vetores $v^1, v^2, \ldots, v^k \in \mathbb{R}^n \setminus \{0\}$ são ditos A-conjugados se*

$$(v^i)^T A v^j = 0$$

com $i, j = 0, 1, \ldots, k$ *e* $i \neq j$.

É possível mostrar que $(x^k - x^{k-1})$ definidos acima são direções D-conjugadas. O algoritmo de Fletcher-Reeves [7] é um método de gradientes conjugados que implementa o método de Krylov. A seguir enunciamos o método de Fletcher-Reeves para o problema quadrático geral $(\bar{P})$. O desenvolvimento detalhado do método de gradientes conjugados pode ser encontrado em [18].

Algoritmo 5: Fletcher-Reeves

Dados: $z^0 \in \mathbb{R}^n$, $\nabla \bar{f}(z^0) = Az^0 + b$, $k = 0$, $v^0 = -\nabla \bar{f}(z^0)$;

enquanto $||\nabla \bar{f}(z^k)|| \neq 0$

calcule λ_k através de busca linear exata;

$z^{k+1} = z^k + \lambda_k v^k$;

$\bar{g}^{k+1} = z^{k+1} - \lambda_k A\,\bar{g}^k$;

$\beta_{k+1} = \dfrac{\bar{g}^{k+1\,T}\,\bar{g}^{k+1}}{\bar{g}^{k\,T}\,\bar{g}^k}$;

$v^{k+1} = -\,\bar{g}^{k+1} + \beta_{k+1} v^k$;

$k = k + 1$;

fim

Resultado: $z = z^k$

Note que, para o problema quadrático, o passo λ_k calculado através de busca linear exata é o passo de Cauchy. Nesse caso, podemos guardar o valor da multiplicação matricial $h^k = A\,\bar{g}^k$ que será usado para o cálculo do passo de Cauchy e de $\bar{g}^{k+1}$. Dessa maneira, realizamos apenas uma multiplicação matricial por iteração.

A complexidade do método de Krylov para o problema $(\bar{P})$ com o critério de parada $\bar{f}(z^k) - \bar{f}(z^*) \leq \varepsilon(\bar{f}(z^0) - \bar{f}(z^*))$ é dada por:

$$k \leq \left\lceil \frac{\sqrt{C}}{2} \log\left(\frac{2}{\varepsilon}\right) \right\rceil$$

evidentemente melhor do que o método de Cauchy. Esse resultado é obtido utilizando Polinômios de Chebyshev, que definiremos na próxima seção, e está demonstrado em [16, 23].

Vale frisar que o método de Fletcher-Reeves não é um método de máximo declive. Além disso, no caso de problemas não quadráticos, este método depende de buscas exatas, que podem acumular erros e afetar o desempenho do algoritmo.

2.4 MÉTODO BASEADO EM POLINÔMIOS DE CHEBYSHEV

Já vimos que a complexidade do algoritmo de Cauchy é da ordem $\mathcal{O}(C \log \frac{1}{\varepsilon})$. Em [9], Gonzaga mostra que se o menor e o maior autovalor da matriz A forem conhecidos, existe um conjunto finito de k passos a serem dados, em qualquer ordem, no método de máximo declive para quadráticas, que satisfaz os quatro critérios de parada:

(i) $||\nabla f(x^k)|| \leq \varepsilon ||x^0 - x^*||$;

(ii) $f(x^k) - f(x^*) \leq \varepsilon$;

(iii) $||x - x^*|| \leq \varepsilon$;

(iv) $f(x^k) - f(x^*) \leq \varepsilon (f(x^0) - f(x^*))$.

Essa quantidade k de passos é limitada por

$$k = \left\lceil \frac{\cosh^{-1}\left(\frac{1}{\varepsilon}\right)}{\cosh^{-1}\left(1 + \frac{2}{C-1}\right)} \right\rceil \approx \left\lceil \frac{\sqrt{C}}{2} \log\left(\frac{2}{\varepsilon}\right) \right\rceil \tag{2.1}$$

em que $C = d_n/d_1$ é o número de condicionamento da matriz A. Os k passos, que denotaremos por $\lambda_i, i = 1, 2 \ldots, k$ são calculados pelo inverso das raízes de um polinômio de Chebyshev de ordem k, como veremos a seguir.

Os polinômios de Chebyshev foram definidos por Pafnuty Chebyshev, em 1854, como uma sequência de polinômios ortogonais que podem ser obtidos recursivamente da seguinte forma:

$$\begin{aligned} T_0(x) &= 1 \\ T_1(x) &= x \\ T_{n+1}(x) &= 2xT_n(x) - T_{n-1}(x). \end{aligned} \tag{2.2}$$

Podemos definir os polinômios de Chebyshev utilizando a seguinte relação:

Definição 2.2. *O polinômio de Chebyshev de ordem k,*

$$T_k : [-1, 1] \mapsto \mathbb{R},$$

é definido por

$$T_k(x) = \cos(k \cos^{-1}(x)).$$

É possível mostrar que esta definição satisfaz (2.2). As raízes do polinômio de Chebyshev, T_k, são:

$$x_j = \cos\left(\frac{1+2j\pi}{2k}\right),$$

com $j = 0, 1, \ldots, k-1$.

No método descrito em [9] precisamos determinar o polinômio de Chebyshev de menor grau cujas raízes estejam no intervalo $[d_1, d_n]$. Isto é obtido com uma mudança de variável para que possamos utilizar as raízes do polinômio da Definição 2.2 com ordem k dada por (2.1). Portanto, utilizaremos a seguinte relação:

$$w = \frac{d_n - d_1}{2}x + \frac{d_n + d_1}{2}.$$

Assim, $x = 0$ para $w = \dfrac{d_1 + d_n}{2}$, $x = -1$ para $w = d_1$ e $x = 1$ para $w = d_n$.

Sejam $d^-, d^+ \in \mathbb{R}_+$ tais que $d_1, d_n \in [d^-, d^+]$, com $d^- \neq 0$ e defina $\bar{C} = d^+/d^-$. Utilizaremos os passos $\lambda_j = 1/x_j$ dados por

$$x_j = \frac{d^+ - d^-}{2}\cos\left(\frac{1+2j\pi}{2k}\right) + \frac{d^+ + d^-}{2} \tag{2.3}$$

com $j = 0, 1, \ldots, k-1$, em que

$$k = \left\lceil \frac{\cosh^{-1}\left(\dfrac{1}{\varepsilon}\right)}{\cosh^{-1}\left(1 + \dfrac{2}{\bar{C}-1}\right)} \right\rceil. \tag{2.4}$$

A prova deste resultado pode ser encontrada em [9].

Note que, para utilizarmos o processo acima de maneira a obter o conjunto finito de passos, é necessário conhecer o maior e o menor autovalor de A. Também em [9], Gonzaga propõe um método adaptativo que utiliza raízes de polinômios de Chebyshev. Utiliza-se valores iniciais de d^- e d^+ fixados, não necessariamente no intervalo $[d_1, d_n]$, os quais serão atualizados sempre que for possível no decorrer do algoritmo. Dessa forma, obtém-se uma boa aproximação para d_1 e d_n, e, consequentemente, de C, o que nos permite utilizar o resultado acima e construir uma sequência finita de k passos, com k calculado por (2.4). Este método apresentado em [9] não é muito eficiente na prática pois

foi proposto para o estudo de complexidade, que nesse caso, é de

$$k \leq \left\lceil 12\sqrt{C}\log(4/\varepsilon) \right\rceil .$$

Utilizaremos um processo adaptativo semelhante ao apresentado em [9] nos algoritmos que serão propostos posteriormente.

3 MÉTODO DE BARZILAI-BORWEIN E VARIANTES

Considere o problema

$$\underset{z\in\mathbb{R}^n}{\text{minimize}}\ \bar{f}(z) = \frac{1}{2}z^T Az + b^T z. \qquad (\bar{P})$$

em que $b \in \mathbb{R}^n$ e $A \in \mathbb{R}^{n\times n}$ é uma matriz simétrica definida positiva.

No método de Cauchy, o tamanho de passo utilizado é calculado por $\bar{\lambda}_k^C = \dfrac{\bar{g}^{k\,T}\bar{g}^k}{\bar{g}^{k\,T}A\bar{g}^k}$, que resulta de uma busca exata. Como vimos, esta escolha é a que minimiza a função objetivo na direção $\nabla\bar{f}(z^k)$. Entretanto, podemos escolher um tamanho de passo de modo que o decrescimento da função não seja o maior possível. Podemos, ainda, permitir que a escolha do tamanho de passo proporcione um crescimento da função objetivo em algumas iterações, porém, de maneira controlada. Isso caracteriza um método de busca não monótona, como o proposto por Grippo, Lampariello e Lucidi em [12]. Neste capítulo apresentaremos o método de Barzilai-Borwein, um método de máximo declive não monótono. Veremos também várias propostas de tamanhos de passo baseadas na ideia proposta por Barzilai e Borwein para o método de máximo declive. Ao final, analisaremos o desempenho computacional dos algoritmos apresentados comparando os resultados.

3.1 MÉTODO DE BARZILAI-BORWEIN

Em 1988, Barzilai e Borwein [2], estudando problemas bidimensionais, propuseram dois novos tamanhos de passos para o algoritmo de máximo declive:

$$\lambda_k^{BB1} = \frac{\Delta z^T \Delta\bar{g}}{\Delta\bar{g}^T\Delta\bar{g}} \qquad (3.1)$$

e

$$\lambda_k^{BB2} = \frac{\Delta z^T \Delta z}{\Delta z^T\Delta\bar{g}} \qquad (3.2)$$

nos quais $\Delta z = z^{k+1} - z^k$ e $\Delta\bar{g} = \bar{g}^{k+1} - \bar{g}^k$. Esta escolha de passo não garante o decrescimento da função objetivo a cada iteração. Quando o problema tratado é de minimização quadrática, como no caso do problema $(\bar{P})$, o passo calculado em (3.2), na iteração k, coincide com

o passo $\bar{\lambda}_{k-1}^{C}$, que seria utilizado no método de Cauchy, na iteração $k-1$, ou seja,

$$\lambda_k^{BB2} = \bar{\lambda}_{k-1}^{C}.$$

Desse modo, a cada iteração do método de Barzilai-Borwein, estamos escolhendo o mesmo passo que seria dado na iteração anterior do algoritmo de Cauchy. Além disso, calculamos o passo λ_k^{BB2} somente com operações vetoriais, utilizando apenas informações da iteração anterior. O passo (3.2) pode ser obtido resolvendo o problema de minimizar a derivada direcional da função quadrática a partir do conhecimento de duas derivadas direcionais.

Lema 3.1. *Considere o problema* $(\bar{P})$*. O passo* (3.2)*, na iteração* $k+1$*, e o passo calculado no método de Cauchy, na iteração* k*, são iguais.*

Demonstração: Considere a direção h e sejam $\bar{g}_1^T h$ e $\bar{g}_2^T h$ duas derivadas direcionais, nos pontos z_1 e z_2, respectivamente, de modo que $z_2 = z_1 + \lambda_1 h$ em que λ_1 é um escalar qualquer.

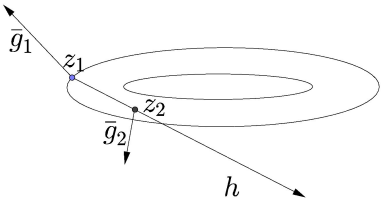

Figura 2 – Curvas de nível de $\bar{f}(z)$ e derivadas direcionais nos pontos z_1 e z_2.

Assim, podemos estabelecer uma relação entre a derivada direcional e o tamanho do passo λ, conforme a Figura 3.

A função que descreve essa relação é uma função linear dada por $F(\lambda) = a\lambda + b$, com $a = -\dfrac{\bar{g}_1^T h - \bar{g}_2^T h}{\lambda_1}$ e $b = \bar{g}_1^T h$. Portanto, procuramos λ_2 tal que $a\lambda_2 + b = 0$, pois procuramos o ponto em que a derivada direcional se anule, ou seja,

$$-\frac{\bar{g}_1^T h - \bar{g}_2^T h}{\lambda_1}\lambda_2 + \bar{g}_1^T h = 0.$$

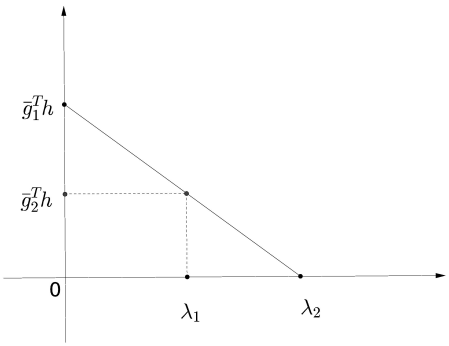

Figura 3 – Gráfico da função que calcula a derivada direcional no ponto $z + \lambda h$ em relação ao tamanho do passo λ.

Logo,

$$\lambda_2 = \frac{\lambda_1 \bar{g}_1^T h}{(\bar{g}_1 - \bar{g}_2)^T h}.$$

Note que

$$z_2 - z_1 = z_1 + \lambda_1 h - z_1 = \lambda_1 h.$$

Assim,

$$\lambda_2 = \frac{(z_2 - z_1)^T \bar{g}_1}{(\bar{g}_1 - \bar{g}_2)^T h}.$$

No caso em que $h = -\bar{g}_1$, temos

$$\lambda_2 = -\frac{(z_2 - z_1)^T \bar{g}_1}{(\bar{g}_1 - \bar{g}_2)^T \bar{g}_1} = \frac{(z_2 - z_1)^T \bar{g}_1}{(\bar{g}_2 - \bar{g}_1)^T \bar{g}_1}.$$

Observe que esse é o passo de Barzilai-Borwein dado por (3.2). De fato, temos

$$\begin{aligned} \Delta z &= z_2 - z_1 = \lambda_1 h = -\lambda_1 \bar{g}_1 \\ \Delta z^T \Delta z &= (z_2 - z_1)^T (-\lambda_1 \bar{g}_1) = -\lambda_1 (z_2 - z_1)^T \bar{g}_1 \\ \Delta g^T \Delta z &= (\bar{g}_2 - \bar{g}_1)^T (-\lambda_1 \bar{g}_1) = -\lambda_1 (\bar{g}_2 - \bar{g}_1)^T \bar{g}_1. \end{aligned}$$

Assim,

$$\frac{\Delta z^T \Delta z}{\Delta g^T \Delta z} = \frac{-\lambda_1 (z_2 - z_1)^T \bar{g}_1}{-\lambda_1 (\bar{g}_2 - \bar{g}_1)^T \bar{g}_1} = \frac{(z_2 - z_1)^T \bar{g}_1}{(\bar{g}_2 - \bar{g}_1)^T \bar{g}_1} = \lambda_2. \tag{3.3}$$

Além disso, se $\bar{g}_1 = Az_1 + b$ e $\bar{g}_2 = Az_2 + b$, então

$$\bar{g}_2 - \bar{g}_1 = A(z_2 - z_1) = A(-\lambda_1 \bar{g}_1) = -\lambda_1 A\bar{g}_1.$$

Portanto,

$$\bar{g}_1 - \bar{g}_2 = \lambda_1 A\bar{g}_1. \tag{3.4}$$

Substituindo (3.4) em (3.3), temos

$$\begin{aligned} \lambda_2 &= \frac{\lambda_1 \bar{g}_1^T \bar{g}_1}{(\bar{g}_1 - \bar{g}_2)^T \bar{g}_1} \\ &= \frac{\lambda_1 \bar{g}_1^T \bar{g}_1}{\lambda_1 (A\bar{g}_1)^T \bar{g}_1} \\ &= \frac{\bar{g}_1^T \bar{g}_1}{\bar{g}_1^T A^T \bar{g}_1} \\ &= \frac{\bar{g}_1^T \bar{g}_1}{\bar{g}_1^T A \bar{g}_1} \end{aligned}$$

que é o passo calculado pelo método de Cauchy $\bar{\lambda}^C$ na iteração anterior. ■

No método de Barzilai-Borwein é possível calcular o passo λ_k^{BB} somente a partir da segunda iteração. Desse modo, o passo inicial λ_0 será um valor fixado. Assim, o método de Barzilai-Borwein é definido no Algoritmo 6.

Note que, como no método de Cauchy, o método de Barzilai-Borwein utiliza apenas uma multiplicação matricial por iteração, ao calcular o valor do gradiente da função.

Na figura 4, exibimos um exemplo no qual utilizamos os métodos de Cauchy e de Barzilai-Borwein para resolver um problema quadrático na forma simplificada com o objetivo de fazer uma comparação. A função objetivo é $f(x) = \frac{1}{2} x^T D x$, com 1000 variáveis, em que D é uma matriz diagonal com autovalores não nulos e distintos entre si, distribuídos uniformemente. Nesse exemplo, o número de condicionamento de D é 1000.

Algoritmo 6: Barzilai-Borwein

Dados: $z^0 \in \mathbb{R}^n, \nabla \bar{f}(z^0) = \bar{g}^0$, $\lambda_0 = 1$, $k = 0$;
enquanto $||\nabla \bar{f}(z^k)|| \neq 0$
  $z^{k+1} = z^k - \lambda_k \bar{g}^k$;
  $\bar{g}^{k+1} = \nabla \bar{f}(z^{k+1})$;
  $\Delta z = z^{k+1} - z^k$;
  $\Delta \bar{g} = \bar{g}^{k+1} - \bar{g}^k$;
  $\lambda_{k+1} = \dfrac{\Delta z^T \Delta z}{\Delta \bar{g}^T \Delta z}$;
  $k = k + 1$;
fim
Resultado: z^k

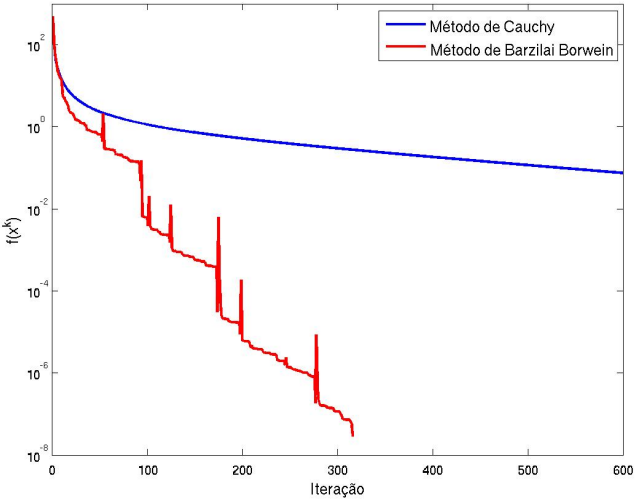

Figura 4 – Métodos de Cauchy e de Barzilai-Borwein para um problema quadrático simplificado.

Neste exemplo, o critério de parada utilizado foi $f(x^k) \leq \varepsilon f(x^0)$, em que $f(x^0) = 500$ e $\varepsilon = 10^{-10}$, ou seja, o critério de parada utilizado foi $f(x^k) \leq 5 \times 10^{-8}$. Paramos o método de Cauchy na iteração 600 para facilitar a comparação entre os métodos. Note que método de Barzilai-Borwein resolveu o problema em 232 iterações, enquanto que o método

de Cauchy sequer se aproxima da solução em $k = 600$. Além disso, podemos observar que o método de Barzilai-Borwein pode causar um aumento na função objetivo em algumas iterações, o que o caracteriza como um método com busca não monótona.

3.2 MÉTODO DE PASSO ALTERNADO

Raydan e Svaiter em [22] e Dai em [4], propõem um novo algoritmo, *alternated step*, que chamaremos de passo alternado. Esse algoritmo utiliza os passos $\bar{\lambda}_k^C$ e λ_k^{BB} alternadamente. A motivação desta escolha parte do comportamento em zigue-zague, frequentemente observado no método de Cauchy com busca exata, que analisaremos no próximo capítulo. Como esse comportamento acontece a cada duas iterações, na iteração $k+1$, repete-se o passo $\bar{\lambda}_k^C$ que foi calculado na iteração k:

$$\lambda_k = \begin{cases} \bar{\lambda}_k^C, & \text{se } k \text{ for par} \\ \lambda_k^{BB}, & \text{se } k \text{ for ímpar} \end{cases} \tag{3.5}$$

Uma vez que $\lambda_{k+1}^{BB} = \bar{\lambda}_k^C$, podemos reescrever esse passo como

$$\lambda_{2k} = \lambda_{2k+1} = \bar{\lambda}_{2k}^C.$$

Algoritmo 7: Passo alternado

Dados: $z^0 \in \mathbb{R}^n$, $k = 0$;
enquanto $||\nabla \bar{f}(z^k)|| \neq 0$

$$\bar{\lambda}_k = \frac{{\bar{g}^k}^T \bar{g}^k}{{\bar{g}^k}^T A\, \bar{g}^k};$$
$$z^{k+1} = z^k - \bar{\lambda}_k\, \bar{g}^k\,;$$
$$\bar{g}^{k+1} = \nabla \bar{f}(z^{k+1});$$
$$z^{k+2} = z^{k+1} - \bar{\lambda}_k\, \bar{g}^{k+1}\,;$$
$$\bar{g}^{k+2} = \nabla \bar{f}(z^{k+2});$$
$$k = k + 2;$$

fim
Resultado: z^k

Em [4], Dai apresenta também algumas variações deste algoritmo. A primeira delas consiste em, fixada uma quantidade m de

iterações, utilizar o método de Cauchy com busca exata nas primeiras $m-1$ iterações e tomar um passo de Barzilai-Borwein na próxima iteração, ou seja, repetir uma vez o último passo $\bar{\lambda}_k^C$ calculado. Assim, nessa variação do método do passo alternado, o passo será dado por

$$\lambda_{mk+1} = \begin{cases} \bar{\lambda}_{mk+i}^C, & \text{para } i = 1, \ldots, m-1 \\ \lambda_{mk+m}^{BB}, & \text{para } i = m \end{cases} \tag{3.6}$$

com $m \geq 1$. Claramente, a escolha (3.5) corresponde à escolha (3.6) para $m = 2$. Outra sugestão, é tomar um passo calculado pelo método de Cauchy, $\bar{\lambda}_k^C$, e repeti-lo $m-1$ vezes.

3.3 MÉTODO DE CAUCHY ALEATÓRIO

Em [22], Raydan e Svaiter fazem uma modificação no método de Cauchy. Esse novo algoritmo foi chamado de *randomly relaxed Cauchy method*, que chamaremos de método de Cauchy aleatório. Nesse método, a cada iteração, escolhemos um parâmetro $\theta_k \in [0, 2]$ a ser multiplicado ao passo calculado pela busca exata:

$$z^{k+1} = z^k - \theta_k \bar{\lambda}_k^C \bar{g}^k .$$

3.4 MÉTODO DE BARZILAI-BORWEIN GLOBAL

Como já mencionamos, o método de Barzilai-Borwein não é monótono, ou seja, não apresenta, necessariamente, um decrescimento a cada iteração. Dessa forma, o valor da função pode aumentar de uma iteração para a seguinte. Esse aumento nem sempre é prejudicial, uma vez que a função pode decair muito mais na iteração subsequente do que se tivéssemos exigido o decrescimento anteriormente. Este comportamento foi observado em testes computacionais. Desse modo, em 1997, Raydan [21] propôs o método de Barzilai-Borwein global, no qual ele utiliza uma maneira de controlar esse crescimento através de uma busca linear não monótona, proposta por Grippo, Lampariello e Lucidi[12]. Tal busca tem o objetivo de controlar o possível aumento da função objetivo em relação a uma quantidade fixada de iterações anteriores, impedindo, assim, grandes picos.

3.4.1 Busca linear não monótona

O objetivo principal da busca linear não monótona, proposta em [12], é: dado um inteiro $m > 0$, fazer uma "generalização" da condição de Armijo levando em consideração as m iterações anteriores, ou seja,

$$\bar{f}(z^k - \lambda_k \bar{g}^k) < \max_{j=k-i+1,\ldots,i} \{\bar{f}_j\} - \eta \lambda_k {\bar{g}^k}^T \bar{g}^k, \tag{3.7}$$

em que $i = \min\{k, m\}$ e $\eta \in (0, 1)$.

A escolha do valor de m é feita de maneira heurística.

No algoritmo de Barzilai-Borwein global, o passo escolhido λ_k a cada iteração é o passo de Barzilai-Borwein. Se a condição (3.7) não for satisfeita, calculamos $\sigma\lambda_k$, em que $\sigma \in (0, 1)$ é fixado inicialmente, e verificamos novamente. Repetimos esta operação até que a condição (3.7) seja satisfeita e prosseguimos.

Algoritmo 8: Método de Barzilai-Borwein global

Dados: $z^0 \in \mathbb{R}^n$, $\bar{g}_0 = \nabla f(z^0)$, $m > 0$, $\eta \in (0, 1)$, $\sigma \in (0, 1)$, $\lambda_0 > 0$, $k = 0$;

enquanto $||\nabla \bar{f}(z^k)|| \neq 0$

- *alternated stepi* $= \min\{k, m\}$;
- **se** $\bar{f}(z^k - \lambda_k \bar{g}^k) \geq \max_{j=k-i+1\ldots,i}\{\bar{f}_j\} - \eta \lambda_k {\bar{g}^k}^T \bar{g}^k$
  - $\lambda_k = \sigma \lambda_k$;
- **senão**
  - $z^{k+1} = z^k - \lambda_k \bar{g}^k$;
  - $\Delta z = z^{k+1} - z^k$;
  - $\Delta \bar{g} = \bar{g}^{k+1} - \bar{g}^k$;
  - $\lambda_k = \dfrac{\Delta z^T \Delta z}{\Delta z^T \Delta \bar{g}}$;
- **fim**
- $k = k + 1$;

fim

Resultado: z^k

3.5 COMPARAÇÃO ENTRE OS MÉTODOS

A seguir, exibimos um exemplo no qual utilizamos os métodos de Cauchy, Barzilai-Borwein, Passo alternado e Cauchy aleatório para

resolver um problema quadrático na forma simplificada com o objetivo de fazer uma comparação entre os métodos apresentados. A função objetivo é $f(x) = \frac{1}{2}x^T Dx$, em que D é uma matriz diagonal com autovalores não nulos e distintos entre si, distribuídos uniformemente. Nesse exemplo, o número de condicionamento de D é 1000.

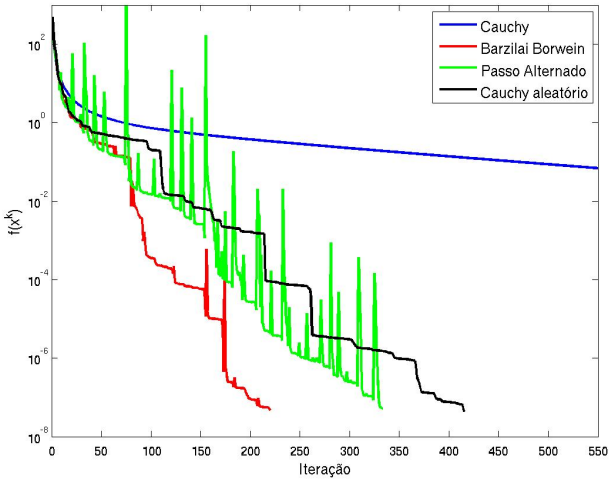

Figura 5 – Métodos de Cauchy, Barzilai-Borwein, Passo alternado e Cauchy aleatório para um problema quadrático simplificado.

Na figura 5 observamos o valor das funções ao longo das iterações nos métodos comparados. O critério de parada utilizado foi $f(x^k) \leq \varepsilon f(x^0)$, em que $f(x^0) = 500$ e $\varepsilon = 10^{-10}$, ou seja, o critério de parada utilizado foi $f(x^k) \leq 5 \times 10^{-8}$. Paramos o método de Cauchy na iteração 550 para facilitar a comparação entre os métodos. Note que os métodos apresentados neste capítulo apresentam um desempenho muito melhor do que o método de Cauchy. Utilizaremos apenas o método de Barzilai-Borwein para comparação de métodos que apresentaremos nos próximos capítulos.

4 PROPRIEDADES ASSINTÓTICAS

Consideremos o problema (P):

$$\underset{x\in\mathbb{R}^n}{\text{minimize}}\, f(x) = \frac{1}{2}x^T Dx \qquad \text{(P)}$$

em que D é uma matriz diagonal, cujos elementos da diagonal são $d_1, d_2, \ldots, d_n$ tais que $0 < d_1 < d_2 < \ldots < d_n$. Mostramos anteriormente que o método de Cauchy converge globalmente. Entretanto, esse algoritmo é conhecido por ser lento e apresentar um comportamento oscilatório. Neste capítulo, apresentaremos alguns resultados que explicam esse comportamento, mostrados, primeiramente, por Akaike em 1959 [1] e desenvolvidos, posteriormente, por Forsythe em [8]. Um estudo detalhado desses resultados pode ser encontrado em Nocedal, Sartenaer e Zhu [17] e Gonzaga e Schneider em [10]. Esses resultados garantem que o método de Cauchy converge assintoticamente para uma busca no subespaço bidimensional gerado pelos autovetores associados ao menor e ao maior autovalor de D, que nesse caso são e_1 e e_n respectivamente. Faremos uma análise detalhada sobre o comportamento de g^k, dos passos gerados pelo algoritmo de Cauchy e de como o tamanho dos passos dados pode interferir no comportamento de g^k.

4.1 RESULTADOS PRINCIPAIS

Para simplificar a notação, neste capítulo representaremos o passo de Cauchy λ_k^C apenas por λ_k. Note que a iteração $k+1$ do método de Cauchy pode ser escrita como:

$$x^{k+1} = x^k - \lambda_k\, g^k = x^k - \lambda_k D x^k (I - \lambda_k D) x^k.$$

Assim, para $i = 1, \ldots, n$,

$$x_i^{k+1} = (1 - \lambda_k d_i) x_i^k.$$

O mesmo ocorre no cálculo do gradiente g^k:

$$g^{k+1} = Dx^{k+1} = D(I - \lambda_k D)x^k = (I - \lambda_k D)\, g^k\ .$$

Logo, para $i = 1, \ldots, n$,

$$g_i^{k+1} = (1 - \lambda_k d_i) g_i^k.$$

Denotaremos por μ_k a sequência dos valores inversos dos passos calculados pelo método de Cauchy, ou seja,

$$\mu_k = \frac{1}{\lambda_k},$$

e chamaremos de y^k a sequência dos gradientes normalizados,

$$y^k = \frac{g^k}{|| \, g^k \, ||}.$$

Observe que se em alguma iteração do método de Cauchy tivermos $x_i^k = 0$, para algum i, então $x_i^j = 0$ para todo $j > k$. Portanto, essa variável não participará do comportamento assintótico. Por isso, vamos supor que $x_i^0 \neq 0$ para todo $i = 1, \ldots, n$.

Observação 4.1. *Os passos calculados pelo método de Cauchy, com $x_i^0 \neq 0$, para $i = 1, \ldots, n$, satisfazem*

$$\frac{1}{d_n} < \lambda_k < \frac{1}{d_1}.$$

De fato,

$$\lambda_k = \frac{{g^k}^T \, g^k}{{g^k}^T \, D \, g^k} = \frac{\sum_{i=1}^n (d_i x_i^k)^2}{\sum_{i=1}^n (d_i x_i^k)^2 d_i} < \frac{1}{d_1} \frac{\sum_{i=1}^n (d_i x_i^k)^2}{\sum_{i=1}^n (d_i x_i^k)^2} = \frac{1}{d_1},$$

uma vez que $d_1 < d_i, i = 2, \ldots, n$ e $x_i \neq 0$ para todo $i = 1, \ldots n$, o que implica que $g_1^k \neq 0$ e $g_n^k \neq 0$. Da mesma forma,

$$\lambda_k = \frac{{g^k}^T \, g^k}{{g^k}^T \, D \, g^k} = \frac{\sum_{i=1}^n (d_i x_i^k)^2}{\sum_{i=1}^n (d_i x_i^k)^2 d_i} > \frac{1}{d_n} \frac{\sum_{i=1}^n (d_i x_i^k)^2}{\sum_{i=1}^n (d_i x_i^k)^2} = \frac{1}{d_n}.$$

Logo, a sequência μ_k satisfaz

$$d_1 < \mu_k < d_n.$$

Consequentemente, obtemos que $g_1^k \neq 0$ e $g_n^k \neq 0$ para todo $k > 0$.

De fato, já é válido para $k = 0$. Suponha que vale para k. Assim,

por indução,

$$g_1^{k+1} = d_1 x_1^{k+1} = d_1(x_1^k - \lambda_k g_1^k) = d_1 x_1^k(1 - \lambda_k d_1) = g_1^k(1 - \lambda_k d_1).$$

Portanto, $g_1^{k+1} \neq 0$. Analogamente, mostramos que $g_n^{k+1} \neq 0$.

Os resultados principais que apresentaremos neste capítulo estão listados no teorema a seguir:

Teorema 4.1 (Teorema Principal)**.** *Considere as sequências x^k, g^k, $y^k = g^k / \| g^k \| \in \mathbb{R}^n, \lambda_k$ e $\mu_k = 1/\lambda_k \in \mathbb{R}$, geradas pelo método de Cauchy com ponto inicial $x^0 \in \mathbb{R}^n$, tal que $x_1^0, x_n^0 \neq 0$, com $n > 1$. Então, existem $\mu, \mu' \in (d_1, d_n)$, $r, r' \in \mathbb{R}^n$ e $\alpha \in (0,1)$ tais que*
(i) $\mu_{2k} \to \mu$, $\mu_{2k+1} \to \mu'$.
(ii) $y_i^k \to 0$ para $i = 2, 3, \ldots, n-1$.
(iii) $y^{2k} \to r$, $y^{2k+1} \to r'$, com $r, r' \in \mathcal{L}(e_1, e_n)$, espaço linear gerado pelos vetores e_1 e e_n.
(iv) $\lim\limits_{k\to\infty} \left| \dfrac{g_i^{k+2}}{g_i^k} \right| \leq 1$ para todo $i = 1, \ldots, n$ tais que $g_i^k \neq 0$ com $k \in \mathbb{N}$.
(v) $\mu + \mu' = d_1 + d_n$.
(vi) $\lim\limits_{k\to\infty} \dfrac{g_i^{k+2}}{g_i^k} = \lim\limits_{k\to\infty} \dfrac{\|g^{k+2}\|}{\|g^k\|} = \alpha$ para $i = 1$ e $i = n$, com

$$\alpha \geq 1 - 2\frac{d_1 d_n}{\tilde{d}^2 - \delta^2},$$

em que $\tilde{d} = (d_1 + d_n)/2$ e $\delta = \min_{i=1,\ldots,n}\{|d_i - \tilde{d}|\}$.
(vii) Os valores μ e μ' são limitados por

$$\tilde{d} - \sqrt{(\tilde{d}^2 + \delta^2)/2} \leq \mu, \mu' \leq \tilde{d} + \sqrt{(\tilde{d}^2 + \delta^2)/2}.$$

A demonstração detalhada desses resultados pode ser encontrada em Gonzaga e Schneider [10][1]. Os itens (i), (ii) e (iii) foram demonstrados inicialmente por Akaike [1] e Forsythe [8]. O item (iii) refere-se ao comportamento oscilatório do gradiente. Nocedal, Sartenaer e Zhu [17] fazem um estudo detalhado dessa oscilação, bem como, dos itens (i), (iv), (v), (vi) e (vii).

Faremos uma interpretação dos itens (i), (ii) e (iii) utilizando alguns exemplos. A demonstração em uma versão simplificada, em

[1] Selecionamos apenas os resultados mais relevantes para esta dissertação; além disso, os resultados foram ordenados da maneira mais conveniente para a demonstração.

[10], segue a estrutura da demonstração feita por Forsythe, em [8].

Para facilitar esta análise, vamos classificar as variáveis do problema de acordo com o autovalor de D correspondente. Chamaremos a variável x_i de variável leve se $d_i \leq \dfrac{d_n}{2}$ e chamaremos x_i de variável pesada se $d_i > \dfrac{d_n}{2}$.

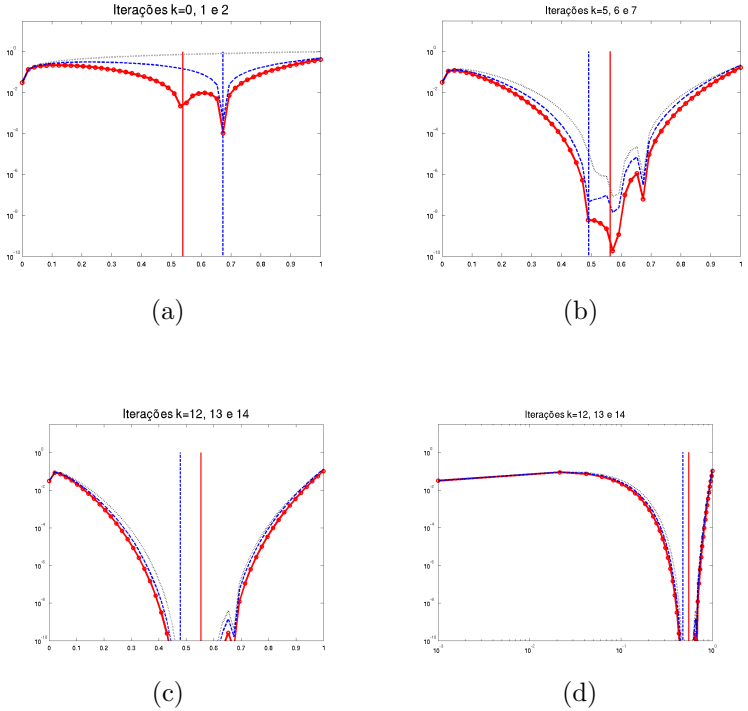

Figura 6 – Valor absoluto das componentes do gradiente em 3 iterações consecutivas e valores de $\mu_k = 1/\lambda_k$.

Na Figura 4.1, observamos as componentes $|g_i^k|$, com $i = 1, \ldots, n$, na iteração atual (em vermelho), na iteração anterior (em azul e tracejado) e na iteração $k-2$ (em preto e pontilhado). Na vertical, exibimos o valor de $1/\lambda_k$ (em vermelho) e o valor de $1/\lambda_{k-1}$ (em azul tracejado). Neste exemplo, o problema possui 50 variáveis, o número de condicionamento é igual a 100 e o ponto inicial x^0 é tal que $x_i^0 = 1/\sqrt{d_i}$, para $i = 1, \ldots, n$. Na Figura 4.1, representamos os autovalores no eixo x, em escala logarítmica, e os valores das componentes de g^k, em

módulo, no eixo y. As Figuras 4.1.a, 4.1.b e 4.1.c correspondem às iterações $k = 2, k = 7$ e $k = 14$, respectivamente, juntamente com as duas iterações anteriores em cada caso. A Figura 4.1.d também corresponde à iteração $k = 14$, porém, utiliza escala logarítmica nos dois eixos.

Notamos que as variáveis correspondentes a autovalores mais próximos de $\frac{d_n}{2}$ são reduzidas mais rapidamente em relação às variáveis extremas. Isso ocorre pois, segundo o item (ii) do Teorema 4.1 temos:

$$y_i^k \to 0 \quad \text{para} \quad i = 2, 3, \ldots, n-1$$

em que $y^k = g^k / \| g^k \|$. Portanto, o método de Cauchy converge assintoticamente para o subespaço gerado por e_1 e e_n que são os autovetores correspondentes aos autovalores d_1 e d_n, respectivamente, conforme afirma o item (iii) do Teorema 4.1. Na Figura 4.1.d, observamos que o algoritmo afeta as variáveis de valores intermediários a pesados, enquanto que as variáveis leves sofrem pouca alteração. Observe o comportamento de $|g_1^k|$ e $|g_n^k|$ no método de Cauchy na Figura 7.

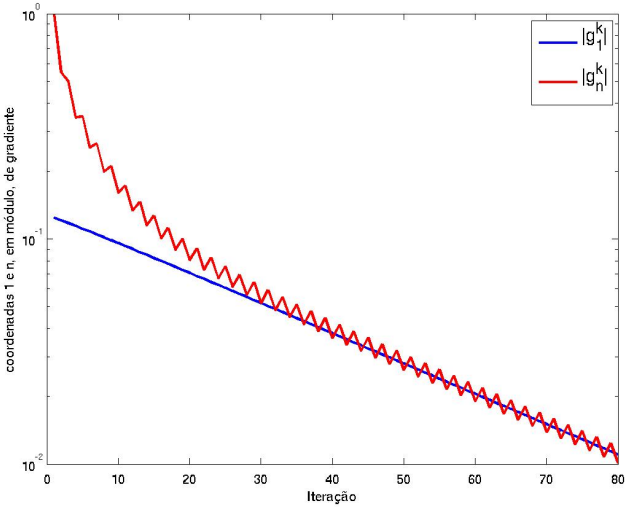

Figura 7 – Valores de g_1 e g_n no método de Cauchy para um problema quadrático.

Note que $|g_1^k|$, que está relacionada com a variável leve x_1, decresce lentamente enquanto que $|g_n^k|$, que está relacionada com a variável pesada x_n, oscila em torno de $|g_1^k|$.

4.2 SEQUÊNCIA DE PASSOS

A sequência de passos gerada pelo método de Cauchy também oscila. O item (i) do Teorema 4.1 afirma que as sequências μ_{2k} e μ_{2k+1} convergem para μ e $\mu' \in (d_1, d_n)$ respectivamente, em que $\mu_k = 1/\lambda_k$. Na Figura 8 observamos o comportamento da sequência μ_k no método de Cauchy para um problema de 1000 variáveis com número de condicionamento igual a 1000.

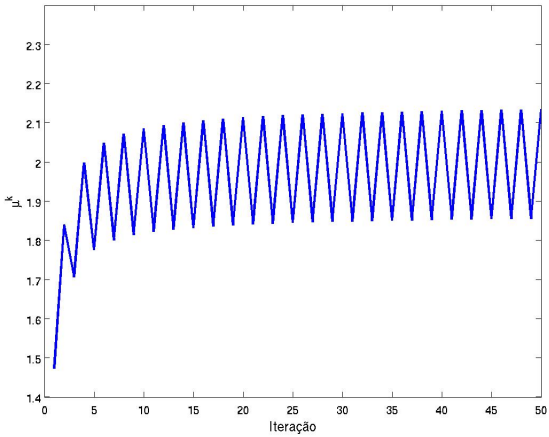

Figura 8 – Sequência μ_k dada pelo método de Cauchy para um problema quadrático.

Demonstraremos os itens (iv), (v), (vi) e (vii) do Teorema 4.1, conforme Gonzaga e Schneider [10].

Para demonstrar o item (iv), vamos considerar uma dupla iteração, supondo que k é grande:

$$\lim_{k\to\infty} \frac{g_i^{k+2}}{g_i^k} = \left(1 - \frac{d_i}{\mu}\right)\left(1 - \frac{d_i}{\mu'}\right) \leq \left(1 - \frac{d_n}{\mu}\right)\left(1 - \frac{d_n}{\mu'}\right), \qquad (4.1)$$

pois $d_i < d_n$ para todo $i = 1, \ldots, n-1$. Como $g_i^k \to 0$ para todo i,

devemos ter

$$\lim_{k\to\infty}\left|\frac{g_i^{k+2}}{g_i^k}\right| \leq 1 \tag{4.2}$$

o que prova o item (iv) do Teorema 4.1.

Sejam μ e μ' tais que $\mu_{2k} \to \mu$ e $\mu_{2k+1} \to \mu'$. Note que, para k grande, as reduções causadas em $|g_1|$ e $|g_n|$ são dadas respectivamente por $(1 - d_1/\mu)(1 - d_1/\mu')$ e $(1 - d_n/\mu)(1 - d_n/\mu')$. Pelo item (iv) do Teorema 4.1, temos que esses valores são iguais, ou seja,

$$(1 - d_1/\mu)(1 - d_1/\mu') = (1 - d_n/\mu)(1 - d_n/\mu').$$

Logo,

$$\begin{aligned} -d_1(\mu+\mu') + d_1^2 &= -d_n(\mu+\mu') + d_n^2 \\ (d_n - d_1)(\mu+\mu') &= d_n^2 - d_1^2 \\ \mu+\mu' &= \frac{(d_n - d_1)(d_n + d_1)}{d_n - d_1} = d_1 + d_n \end{aligned}$$

o que prova o item (v) do Teorema 4.1.

No item (vi), para mostrar que $\lim_{k\to\infty} \dfrac{g_i^{k+2}}{g_i^k} = \lim_{k\to\infty} \dfrac{||g^{k+2}||}{||g^k||}$, utilizamos o item (iii), que afirma que

$$r = \lim_{k\to\infty} \frac{g^{2k}}{\|g^{2k}\|} \text{ e } r' = \lim_{k\to\infty} \frac{g^{2k+1}}{\|g^{2k+1}\|}.$$

Definindo r^+ e r^{++} como

$$r^+ = \left(1 - \frac{1}{\mu}D\right) r \tag{4.3}$$

$$r^{++} = \left(1 - \frac{1}{\mu'}D\right) r^+ = \left(1 - \frac{1}{\mu'}D\right)\left(1 - \frac{1}{\mu}D\right) r, \tag{4.4}$$

obtemos a proporção

$$\frac{r_1^{++}}{r_n^{++}} = \frac{r_1}{r_n}.$$

Assim,

$$\frac{g_1^{2k+2}}{g_n^{2k+2}} = \frac{g_1^{2k}}{g_n^{2k}},$$

e a norma se mantém.

Observe que pela desigualdade (4.2), temos

$$\left(1-\frac{d_i}{\mu}\right)\left(1-\frac{d_i}{\mu'}\right)>-1, \quad i=1,\ldots,n. \tag{4.5}$$

Desenvolvendo essa desigualdade, obtemos

$$\mu\mu' \geq \frac{(\mu+\mu')d_i - d_i^2}{2} = \frac{(d_1+d_n)d_i - d_i^2}{2}, \quad \text{pois} \quad \mu+\mu' = d_1+d_n. \tag{4.6}$$

Vamos definir

$$\tilde{d} = (d_1+d_n)/2, \tag{4.7}$$

$$\delta_i = \tilde{d} - d_i, \quad i=1,\ldots,n \tag{4.8}$$

$$\delta = \min_{i=1,\ldots,n.} \{|\delta_i|\} \tag{4.9}$$

Note que

$$\tilde{d^2} - \delta_i^2 = \tilde{d^2} - (\tilde{d^2} - 2\tilde{d}d_i + d_i^2) = 2\left(\frac{d_1+d_n}{2}\right)d_i - d_i^2 = (d_1+d_n)d_i - d_i^2.$$

Substituindo em (4.6) temos:

$$\mu\mu' \geq \frac{\tilde{d}^2 - \delta_i^2}{2}, \quad \text{e } \mu+\mu' = 2\tilde{d}, \quad i=1,\ldots,n,$$

o que também vale para δ:

$$\mu\mu' \geq \frac{\tilde{d}^2 - \delta^2}{2}, \quad \text{e } \mu+\mu' = 2\tilde{d}, \quad i=1,\ldots,n. \tag{4.10}$$

A redução causada em $|g_1|$ e $|g_n|$, a cada duas iterações, é dada por $\alpha = (1-d_1/\mu)(1-d_1/\mu')$ e $\alpha' = (1-d_n/\mu)(1-d_n/\mu')$, respectivamente. Desenvolvendo α' e utilizando $\mu+\mu' = d_1+d_n$, obtemos:

$$\alpha' = 1 + \frac{d_n^2 - d_n(\mu+\mu')}{\mu\mu'} = 1 + \frac{d_n^2 - d_n(d_1+d_n)}{\mu\mu'} = 1 - \frac{d_1 d_n}{\mu\mu'}.$$

O mesmo vale para α. Assim, utilizando a desigualdade (4.10), concluímos que

$$\alpha \geq 1 - 2\left(\frac{d_1 d_n}{\tilde{d}^2 - \delta^2}\right), \tag{4.11}$$

o que prova o item (vi) do Teorema (4.1).

Para provar o item (vii), basta resolver o sistema 4.10, que se reduz a uma equação de segundo grau, cuja solução é

$$\bar{\mu}, \bar{\mu}' = \tilde{d} \pm \sqrt{(\tilde{d}^2 + \delta^2)/2}, \tag{4.12}$$

e, portanto, $\bar{\mu} \leq \mu, \mu' \leq \bar{\mu}'$.

Estes últimos dois itens, (vi) e (vii), referem-se à razão de convergência do método de Cauchy no "melhor caso", ou seja, para $|g_i| = 0$, com $i = 2, \ldots, n-1$.

Exemplo 4.1. *Suponha que $d_n = 1$ e $d_1 = 1/C << 1$, em que C é o número de condicionamento de A. Assim, $\tilde{d} \approx 0,5$. Logo,*

$$\bar{\alpha} = 1 - 2\left(\frac{d_1}{0,25 - \delta^2}\right).$$

Portanto, para $\delta \approx 0$, ou seja, se existir algum autovalor próximo a 0,5, obtemos $\bar{\alpha} = 1 - 8d_1 = 1 - 8/C$.

Por outro lado, supondo que $\delta = 0,9\tilde{d}$, ou seja, todos os autovalores estão próximos de d_1 ou d_n, obtemos $\bar{\alpha} \approx 1 - 40/C$.

Se o número de condicionamento C for igual a 1000, os valores de $\bar{\alpha}$ são 0,99 e 0,96. Portanto, a razão de convergência para os dois casos acima é ruim, pois é próximo de $\dfrac{C-1}{C+1} \approx 1 - \dfrac{2}{C}$, a razão de convergência para o pior caso para o método de Cauchy[2], que no caso da dupla iteração se aproxima de $1 - 4/C$.

Portanto, concluímos que a razão de convergência do método de Cauchy é ruim para todas as situações de distribuição de autovalores e a existência de autovalores próximos de $\tilde{d}$ tampouco interfere na razão de convergência do método.

4.3 OUTRAS PROPRIEDADES

Chamaremos o passo λ de passo pequeno se satisfizer $\lambda < \dfrac{2}{d_n}$. Observe que se $\lambda << \dfrac{1}{d_i}$, então,

$$x_i^{k+1} = (1 - \lambda d_i) x_i^k \approx x_i^k.$$

[2]Esse resultado pode ser encontrado em [13]

Em particular, se $\lambda = \dfrac{1}{d_n}$,

$$x_n^{k+1} = (1 - \frac{1}{d_n} d_n) x_i^k = 0$$

Assim, passos pequenos são eficientes para reduzir variáveis pesadas e reduzem todas as variáveis, porém, têm pouco efeito sobre as variáveis leves.

Por outro lado, chamaremos o passo λ de passo grande se satisfizer $\lambda > \dfrac{2}{d_n}$. Considere, por exemplo, o passo $\lambda = 500$ em que $d_1 = 0,001$ e $d_n = 1$. Assim,

$$x_1^{k+1} = (1 - 500\frac{1}{1000}) x_1^k = 0,5 x_1^k$$

$$x_n^{k+1} = (1 - 500) x_n^k = -499 x_n^k.$$

Portanto, passos muito grandes são eficientes para reduzir variáveis leves, porém, causam um aumento significativo nas variáveis pesadas. Mesmo assim, passos muito grandes são necessários; logo, devem ser utilizados com cautela.

Os passos de Cauchy têm tamanho médio e não são eficientes para reduzir variáveis leves, porém, reduzem variáveis intermediárias. Como vimos, os passos de Cauchy ficam limitados por

$$\frac{1}{d_n} < \lambda_k < \frac{1}{d_1},$$

e, como vimos, μ_k e μ_{k+1} convergem para μ e μ', respectivamente.

É possível mostrar também que, para k suficientemente grande, os valores de $|g_n|$ oscilam em torno dos valores de $|g_1|$, com $|g_i|$ decrescendo em todas as iterações, como observamos nas Figuras 7 e 4.1, apresentadas anteriormente. Além disso, temos que

$$\left| \frac{g_1^{k+1}}{g_n^{k+1}} \right| \approx \left| \frac{g_1^k}{g_n^k} \right|.$$

Portanto, a velocidade de convergência do método é determinada pela velocidade com a qual a variável associada ao menor autovalor decresce.

5 NOVOS ALGORITMOS

No capítulo anterior fizemos uma análise detalhada sobre o comportamento das variáveis e das componentes $|g_1^k|, |g_2^k|, \ldots, |g_n^k|$ do gradiente da função em cada iteração do método de Cauchy para o problema quadrático diagonalizado (P). Além disso, vimos que as sequências λ_{2k}^C e λ_{2k+1}^C convergem e classificamos as variáveis do problema em variáveis leves e pesadas: chamamos de variáveis leves, aquelas associadas aos menores autovalores de A; e de pesadas, aquelas associadas aos maiores autovalores de A.

Neste capítulo, vamos analisar o comportamento de $|g_i^k|$, com $i = 1 \ldots, n$, ao escolhermos outros comprimentos de passos no método de máximo declive. Nessa situação nos preocuparemos em avaliar o efeito causado pela escolha de passos pequenos e grandes. Da mesma maneira definida no capítulo anterior, dizemos que um passo λ_k é pequeno se $\lambda_k < 2/d_n$ e dizemos que um passo λ_k é grande se $\lambda_k > 2/d_n$. Após esta análise, apresentaremos o algoritmo *steepest descent with alignement* (SDA), proposto por De Asmundis et al em [5]. Finalmente, vamos propor dois novos algoritmos: *Cauchy-short* (CS) e *alternated Cauchy-short* (ACS), que também podem ser encontrados em Gonzaga e Schneider [10]. Apresentaremos, também, algumas variações desses novos algoritmos utilizando uma proposta descrita por Gonzaga em [9].

5.1 PASSOS GRANDES E PEQUENOS

Mostramos anteriormente que os passos calculados pelo método de Cauchy satisfazem

$$\frac{1}{d_n} < \lambda_k < \frac{1}{d_1}.$$

Vimos, também, que o método de Cauchy converge assintoticamente para uma busca no espaço bidimensional gerado pelos autovetores relacionados ao maior e ao menor autovalor da matriz A, do problema $(\bar{P})$. Assim, temos $\dfrac{g_i^k}{|| g^k ||} \longrightarrow 0$ para $i = 2, \ldots, n-1$. Além disso, as variáveis mais leves e mais pesadas são reduzidas mais lentamente em relação às intermediárias, que são aquelas relacionadas aos autovalores mais próximos de $(d_1 + d_n)/2$.

Note que, se escolhermos um passo $\lambda_k = \dfrac{1}{d_i}$, para algum i, a

componente g_i^k de g^k será reduzida a zero. De fato,

$$\begin{aligned} g_i^{k+1} &= (1-\lambda_k d_i) g_i^k \\ &= (1-1) g_i^k \\ &= 0. \end{aligned}$$

Como observamos no capítulo anterior, passos pequenos não são prejudiciais, pois reduzem todas as variáveis com pouca influência nas variáveis leves. Por outro lado, passos grandes são necessários pois reduzem variáveis leves, entretanto, podem proporcionar um grande aumento nas variáveis pesadas.

Na Figura 9, observamos o comportamento das componentes de g^k, em módulo, ao escolhermos um passo muito pequeno, ou seja, muito próximo de $1/d_n$. As imagens na Figura 9, mostram g^k, em módulo(no eixo vertical), nas iteração k, as componentes $|g_i^k|$ na iteração anterior (em azul tracejado) e as componentes de $|g_i^k|$ na iteração $k-1$ (em preto pontilhado). O traço vertical representa o valor de $1/\lambda_k$ (em vermelho) e de $1/\lambda_{k-1}$ (em azul tracejado). O problema possui 50 variáveis, número de condicionamento igual a 100 ($d_1 = 0,01$ e $d_n = 1$) e o ponto inicial é $x_i^0 = 1/\sqrt{d_i}$. Nesse exemplo, observamos o efeito causado por um passo pequeno, muito próximo de $\dfrac{1}{d_n}$, dado na iteração 11, e os passos de Cauchy seguintes, dados nas iterações 12, 13 e 14.

Observe que, ao darmos passos de Cauchy nas próximas três iterações, logo após termos dado um passo muito pequeno, obtemos uma redução nas variáveis leves.

Podemos perceber que as variáveis pesadas são reduzidas quando escolhemos um passo muito pequeno. Após esse passo pequeno, o passo de Cauchy será um passo grande, que reduz variáveis leves. Observe que, os passos de Cauchy calculados após o passo pequeno causam um aumento nas variáveis pesadas.

Entretanto, na terceira iteração, após o passo pequeno, o passo de Cauchy já atinge um valor intermediário. Assim, mesmo que um passo grande de Cauchy seja dado (logo após um passo muito pequeno), os passos de Cauchy voltam a ter valores intermediários nas próximas iterações, ou seja, os valores de λ_k voltam a ficar próximos de $2/d_n$.

Desse modo, é natural pensarmos em maneiras de encontrar passos pequenos, próximos de $1/d_n$, sem, de fato, conhecer os autovalores de D. A seguir, apresentaremos duas maneiras de obter passos pequenos nos algoritmos.

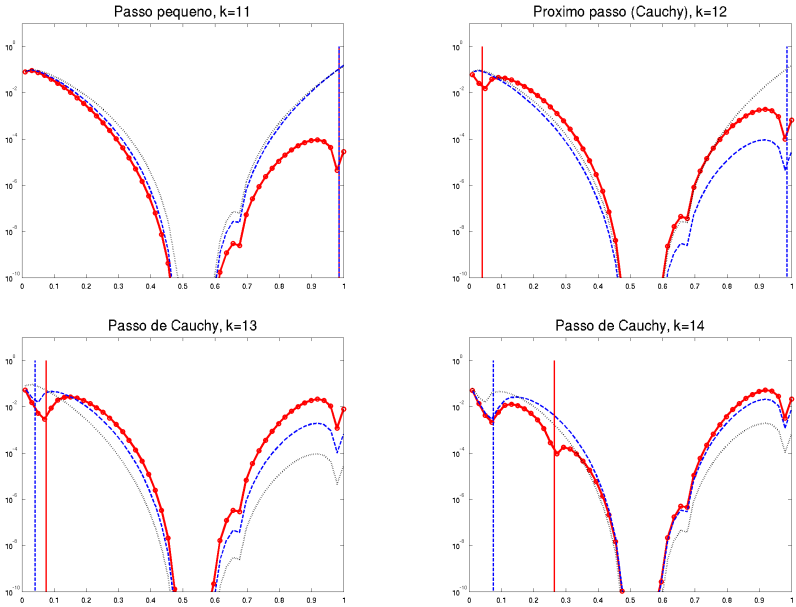

Figura 9 – Componentes do gradiente em módulo antes e depois de darmos um passo pequeno.

5.2 MÉTODO SDA

Como demonstramos no capítulo anterior, a sequência $\mu_k = 1/\lambda_k$ satisfaz:

$$\mu_{2k} \to \mu, \text{ e } \mu_{2k+1} \to \mu',$$

com $\mu + \mu' = d_1 + d_n$. Portanto,

$$\lim_{k\to\infty} \left(\frac{1}{\bar{\lambda}^C_{2k}} + \frac{1}{\bar{\lambda}^C_{2k-1}} \right) = d_1 + d_n.$$

Como, em geral, não sabemos quais são os autovalores da matriz A, De Asmundis et al [5], sugerem a escolha do passo,

$$\tilde{\lambda_k} = \left(\frac{1}{\bar{\lambda}^C_k} + \frac{1}{\bar{\lambda}^C_{k-1}} \right)^{-1},$$

pois, como vimos, esse valor se aproxima de $\dfrac{1}{d_1 + d_n}$. O novo algoritmo proposto foi chamado de *steepest descent with alignment* (SDA) e consiste em utilizar passos de Cauchy até que a sequência $\tilde{\lambda_k}$ se estabilize, ou seja, mantenha-se dentro de uma tolerância $\varepsilon_2 > 0$,

$$|\tilde{\lambda_k} - \tilde{\lambda}_{k-1}| < \varepsilon_2.$$

Depois disso, efetuam-se p passos do tipo $\tilde{\lambda_k}$, desde que essa escolha implique em um decrescimento da função. Caso contrário, escolhe-se o passo $2\bar{\lambda}^C_k$, o qual não altera o valor da função. Apresentamos o algoritmo SDA para o problema $(\bar{P})$ na forma geral no Algoritmo 9.

Observação 5.1. *Em todos os algoritmos apresentados neste capítulo, omitiremos a menção à regra de parada, que, nas implementações, deve ser testada logo após o cálculo de cada novo iterado. A regra usual é* $|| g^k || < \varepsilon$, *com a precisão* $\varepsilon > 0$ *dada. Nos testes para o problema diagonalizado, podemos calcular o valor da função na solução ótima* x^* *que, nesse caso, será zero. Assim, nos exemplos e nos testes computacionais, utilizaremos o critério* $f(x^k) - f(x^*) \leq \varepsilon(f(x^0) - f(x^*))$, *ou seja,* $f(x^k) < \varepsilon f(x^0)$.

Algoritmo 9: Método SDA

Dados: $z^0 \in \mathbb{R}^n, \quad \varepsilon_2 > 0, \quad p$ inteiro

$\bar{g}^0 = Az^0 + b$

$\bar{\lambda}_0 = \dfrac{{\bar{g}^0}^T \bar{g}^0}{{\bar{g}^0}^T A \bar{g}^0}; \quad z^1 = z^0 - \bar{\lambda}_0 \bar{g}^0; \quad \bar{g}^1 = Az^1 + b$

$\bar{\lambda}_1 = \dfrac{{\bar{g}^1}^T \bar{g}^1}{{\bar{g}^1}^T A \bar{g}^1}; \quad z^2 = z^1 - \bar{\lambda}_1 \bar{g}^1; \quad \bar{g}^2 = Az^2 + b$

$\tilde{\lambda}_1 = \dfrac{\bar{\lambda}_1 \bar{\lambda}_0}{\bar{\lambda}_1 + \bar{\lambda}_0}$

$k = 2;$

enquanto $||\nabla \bar{f}(z^k)|| \neq 0$

- $\bar{\lambda}_k = \dfrac{{\bar{g}^k}^T \bar{g}^k}{{\bar{g}^k}^T A \bar{g}^k}; \quad z^{k+1} = z^k - \bar{\lambda}_k \bar{g}^k;$
- $\bar{g}^{k+1} = \bar{g}^k - \bar{\lambda}_k A \bar{g}^k$
- $\tilde{\lambda}_k = \dfrac{\lambda_k \lambda_{k-1}}{\lambda_k + \lambda_{k-1}}$
- $k = k + 1;$
- **se** $|\tilde{\lambda}_{k-1} - \tilde{\lambda}_{k-2}| < \varepsilon_2$
  - $\lambda = min\{\tilde{\lambda}_k, 2\bar{\lambda}_k\}$
  - **para** $i = 1, h$
    - $z^{k+1} = z^k - \lambda \bar{g}^k;$
    - $\bar{g}^{k+1} = \bar{g}^k - \lambda A \bar{g}^k$
    - $k = k + 1;$
  - **fim**
- **fim**

fim

Resultado: z^k

5.3 MÉTODO CS

Com o mesmo objetivo de encontrar um valor de passo que seja próximo de $\frac{1}{d_n}$, vamos propor uma escolha diferente. Como já mencionamos no capítulo anterior, no método de Cauchy, as componentes $|g_2^k|, |g_3^k|, \ldots, |g_{n-1}^k|$ decrescem rapidamente enquanto $|g_1^k|$ decresce lentamente e $|g_n^k|$ oscila em torno de $|g_1^k|$. Desse modo, se tomarmos um passo L muito grande, tal que $L >> 1/d_1$, a componente $|g_n^k|$ sofrerá um acréscimo "muito maior" do que o acréscimo causado em $|g_1^k|$. Assim, o próximo passo de Cauchy será muito pequeno, o que proporcionará um decréscimo no valor de $|g_n^k|$. Esse passo de Cauchy será aproximadamente $\frac{1}{d_n}$.

O passo grande L será utilizado apenas para estimar o próximo passo de Cauchy, que será um passo pequeno, e não será aplicado no método. Assim, o passo de Cauchy dado após esse passo muito grande L será chamado de passo curto e denotado por λ^S. Além disso, como o passo curto é um passo de Cauchy, estará entre $\frac{1}{d_n}$ e $\frac{1}{d_1}$. Podemos utilizar ainda uma salvaguarda, ou seja, utilizar o passo λ^S apenas se este for inferior ao menor passo de Cauchy já calculado em todas as iterações anteriores.

Vamos mostrar a seguir que o passo de Cauchy, calculado após o passo muito grande, é, de fato, pequeno. Suponha que as componentes $|g_2^k|, |g_3^k|, \ldots, |g_{n-1}^k|$ já tenham sido reduzidas, que estejam próximas de zero e seja L um passo "muito grande", ou seja, tal que

$$\tilde{g}_i = (1 - Ld_i)g_i^k \approx -Ld_i g_i^k$$

para todo $i \in \{1, \ldots, n\}$. Assim, o próximo passo calculado pelo método de Cauchy será:

$$\lambda^C = \frac{\tilde{g}^T \tilde{g}}{\tilde{g}^T D \tilde{g}} = \frac{g^T D^2 g}{g^T D^3 g} \approx \frac{(g_n^k)^2 d_n^2}{(g_n^k)^2 d_n^3} = \frac{1}{d_n},$$

uma vez que as componentes $|g_2^k|, |g_3^k|, \ldots, |g_{n-1}^k|$ já estão próximas de zero e d_1 é o menor autovalor, o que torna $(g_1^k)^2 d_1^2 + (g_n^k)^2 d_n^2$ próximo de $(g_n^k)^2 d_n^2$.

Observe que, a cada iteração k, ao calcular o passo de Cauchy $\bar{\lambda}_k^C$, podemos guardar o valor da multiplicação matricial $h^k = A\,\bar{g}^k$. Assim, ao calcular o gradiente da função no ponto z^{k+1}, não precisaremos

efetuar mais uma multiplicação matricial. De fato,

$$\bar{\lambda}_k^C = \frac{\bar{g^k}^T \bar{g}^k}{\bar{g^k}^T h^k}$$

$$z^{k+1} = z^k - \bar{\lambda}_k^C \bar{g}^k$$

$$\bar{g}^{k+1} = Az^{k+1} + b = Az^k - \bar{\lambda}_k^C A\bar{g}^k + b = \bar{g}^k - \bar{\lambda}_k^C h^k$$

Assim, definimos o algoritmo CS (*Cauchy-short*), proposto por Gonzaga e Schneider em [10], que intercala uma quantidade m de passos de Cauchy e uma quantidade p de passos curtos. As quantidades m e p de passos são fixadas inicialmente. Geralmente, utilizamos $m = 6$ e $p = 2$. Além disso, antes de iniciar o processo descrito, daremos uma quantidade inicial de passos de Cauchy para reduzir as variáveis intermediárias. Para implementações, utilizaremos 10 passos de Cauchy antes de iniciar o algoritmo. O passo curto, por sua vez, é calculado da seguinte maneira: dada uma iteração k e $L \in \mathbb{R}$ "muito grande",

$$\begin{aligned} \tilde{g} &= (I - LA)\bar{g}^k \\ \lambda^S &= \frac{\tilde{g}^T \tilde{g}}{\tilde{g}^T A \tilde{g}} \end{aligned} \tag{5.1}$$

O algoritmo CS para o problema geral $(\bar{P})$ fica definido no Algoritmo 10.

De agora em diante, vamos comparar os métodos propostos, aplicando-os sempre no mesmo problema cuja função objetivo é

$$f(x) = \frac{1}{2} x^T D x,$$

com 1000 variáveis, número de condicionamento de D igual a 1000 e ponto inicial $x_i^0 = 1/\sqrt{d_i}, i = 1, \ldots, 1000$. Na Figura 10, observamos os valores da função ao longo das iterações para o algoritmo CS e para o método de Barzilai-Borwein.

Para fazer uma comparação com a análise apresentada no Capítulo 4, exibimos, na Figura 11, a sequência de passos gerada pelo método CS e as componentes $|g_1|$ e $|g_n|$ do gradiente, para o mesmo método. Podemos perceber como o comportamento oscilatório foi quebrado, tanto da sequência de passos quanto das componentes do gradiente.

Algoritmo 10: Método CS

Dados: $z^0 \in \mathbb{R}^n, p$ inteiro, m inteiro, $L \in \mathbb{R}$ "muito grande";
$\bar{g}^0 = Az^0 + b$;

Dar 10 passos de Cauchy;
$k = 10$;
enquanto $||\nabla \bar{f}(z^k)|| \neq 0$
  para $i = 1 : m$
    $h^k = A\bar{g}^k$;
    $\bar{\lambda}_k^C = \dfrac{\bar{g}^{k\,T}\bar{g}^k}{\bar{g}^{k\,T}h^k}$;
    $z^{k+1} = z^k - \bar{\lambda}_k^C \bar{g}^k$;
    $\bar{g}^{k+1} = \bar{g}^k - \bar{\lambda}_k^C h^k$;
    $k = k + 1$;
  fim

  Calcular o passo curto λ^S utilizando $\bar{g}^k$ e (5.1);

  para $i = 1 : p$
    $z^{k+1} = z^k - \lambda^S \bar{g}^k$;
    $\bar{g}^{k+1} = \bar{g}^k - \lambda^S \bar{g}^k$;
    $k = k + 1$;
  fim
fim
Resultado: z^k

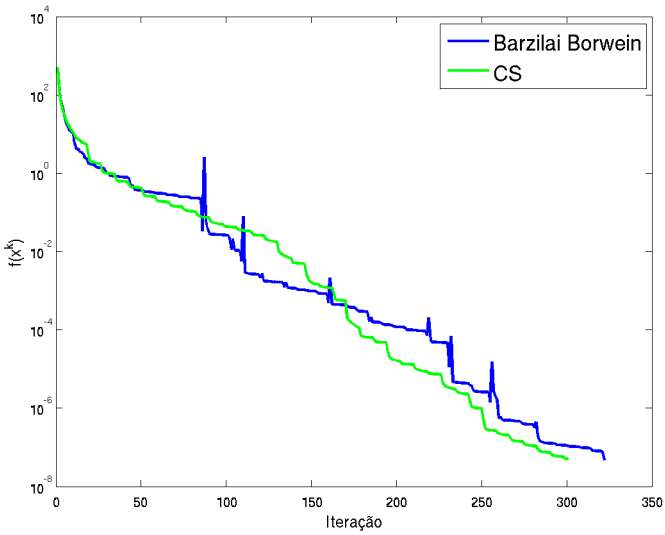

Figura 10 – Variação da função ao longo das iterações nos métodos CS e Barzilai-Borwein na resolução de um problema quadrático.

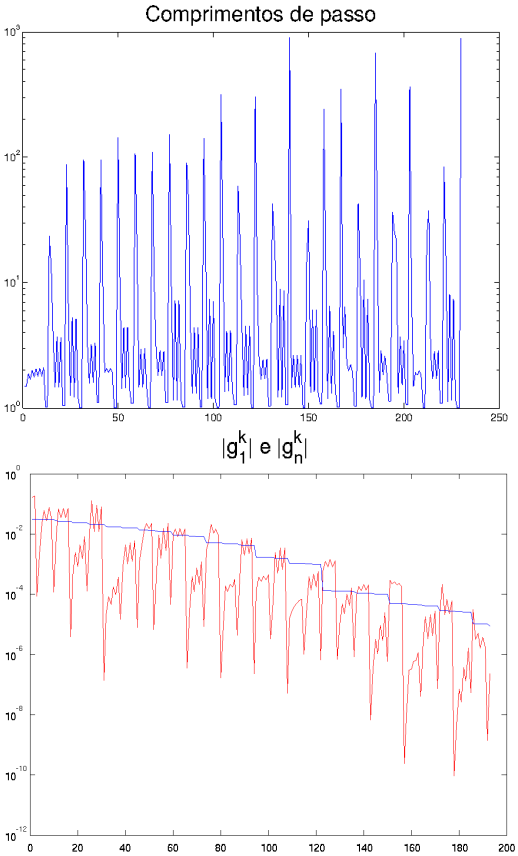

Figura 11 – Componentes do gradiente em módulo antes e depois do algoritmo dar um passo pequeno.

5.4 MÉTODO ACS

Como vimos no capítulo anterior, passos pequenos não são prejudiciais. No algoritmo CS, a cada m passos de Cauchy, damos p passos pequenos, que são os passos curtos. Agora, vamos propor que a cada passo de Cauchy, um passo curto seja dado logo em seguida. Para que o passo curto não seja calculado a cada iteração, vamos calculá-lo a cada m passos de Cauchy, ou seja, calculamos um passo curto e, a cada passo de Cauchy, damos, em seguida, esse passo curto. Faremos isso m vezes e, assim, temos $2m$ iterações. A cada $2m$ iterações, calculamos um novo passo curto e o repetimos p vezes. Chamaremos esse novo método de ACS (*alternated Cauchy-short*), que também pode ser encontrado em [10]. Para testes computacionais, utilizamos $m = 6$ e $p = 2$.

Seja λ^S um passo curto. Observe que se utilizarmos esse passo, o gradiente no novo ponto exigirá apenas uma multiplicação matricial:

$$z^{k+1} = z^k - \lambda^S \bar{g}^k\,,$$

$$\bar{g}^{k+1} = Az^{k+1} + b = Az^k + b - \lambda^S A\bar{g}^k = \bar{g}^k - \lambda^S A\bar{g}^k\,.$$

Assim, realizamos uma multiplicação matricial a cada iteração, da mesma maneira que o método de Cauchy. Cada passo pequeno, dado após o passo de Cauchy, será considerado como uma nova iteração.

Na Figura 5.4 observamos os valores da função ao longo das iterações, para o mesmo problema utilizado na seção anterior, para os métodos CS, Barzilai Borwein e ACS.

5.5 UTILIZANDO RAÍZES DE CHEBYSHEV

No Capítulo 2, apresentamos uma maneira de calcular um conjunto finito de k passos a serem dados pelo método de máximo declive, de modo a resolver o problema quadrático nessas k_c iterações. A quantidade k de iterações é dada por

$$k_c = \left\lceil \frac{\cosh^{-1}\left(\dfrac{1}{\varepsilon}\right)}{\cosh^{-1}\left(1 + \dfrac{2}{C-1}\right)} \right\rceil .$$

Para isso, é necessário que d_1 e d_n sejam conhecidos. Calculamos o conjunto Λ que possuirá k_c passos, em que cada passo λ_j será dado

Algoritmo 11: Método ACS

Dados: $z^0 \in \mathbb{R}^n, p$ inteiro, m inteiro, $L \in \mathbb{R}$ grande;
$\bar{g}^0 = Az^0 + b$;
Dar 10 passos de Cauchy;
$k = 10$;
Calcular o passo curto λ^S utilizando $\bar{g}^k$ e (5.1);
enquanto $||\nabla \bar{f}(z^k)|| \neq 0$
  para $i = 1 : 2m$
    $h^k = A\bar{g}^k$;
    $\bar{\lambda}_k^C = \dfrac{{\bar{g}^k}^T \bar{g}^k}{{\bar{g}^k}^T h^k}$;
    $z^{k+1} = z^k - \bar{\lambda}_k^C \bar{g}^k$;
    $\bar{g}^{k+1} = \bar{g}^k - \bar{\lambda}_k^C h^k$;
    $k = k + 1$;
    $z^{k+1} = z^k - \lambda^S \bar{g}^k$; *(utilizamos o passo curto λ^S já calculado)*
    $\bar{g}^{k+1} = \bar{g}^k - \lambda^S \bar{g}^k$;
    $k = k + 1$;
  fim
  Calcular um novo passo curto λ^S utilizando $\bar{g}^k$ e (5.1);
  para $i = 1 : h$
    $z^{k+1} = z^k - \lambda^S \bar{g}^k$;
    $\bar{g}^{k+1} = \bar{g}^k - \lambda^S A\bar{g}^k$;
    $k = k + 1$;
  fim
fim
Resultado: z^k

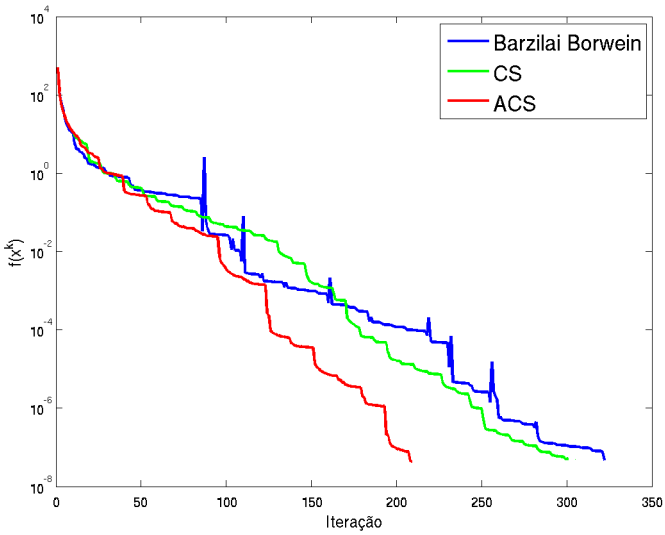

Figura 12 – Variação da função ao longo das iterações nos Métodos CS, ACS e Barzilai-Borwein na resolução de um problema quadrático.

por $\lambda_j = 1/x_j$, com x_j dados por

$$x_j = \frac{d_n - d_1}{2} \cos\left(\frac{1 + 2j\pi}{2k_c}\right) + \frac{d_n + d_1}{2}, \text{ para } j = 1, \ldots, k_c. \quad (5.2)$$

Vale ressaltar que esses k_c passos podem ser dados em qualquer ordem. Desse modo, podemos procurar a melhor maneira de ordená-los. Uma sugestão é: a cada iteração, calcula-se o passo que seria dado no método de Barzilai-Borwein e, escolhe-se, dentre os passos do conjunto Λ, aquele que seja o mais próximo do passo de BB; retirando o passo escolhido do conjunto Λ. Desse modo, os passos utilizados são passos calculados através das raízes do polinômio de Chebyshev na ordem determinada pelo algoritmo de Barzilai-Borwein. Vamos apresentar o algoritmo para o caso em que d_1 e d_n são conhecidos apenas para a análise do desempenho.

Algoritmo 12: Método Barzilai-Borwein com polinômio de Chebyshev

Dados: $z^0 \in \mathbb{R}^n, \lambda_0 \in \mathbb{R}^n$, d_1, d_n
$\bar{g}^0 = Az^0 + b$;
$z^1 = z^0 - \lambda_0 \bar{g}^0$;
$\bar{g}^1 = Az^1 + b$;
$k = 1$;
Calcular Λ; (conjunto de passos calculados por raízes de Chebyshev utilizando o processo descrito em (5.2)).

enquanto $||\nabla \bar{f}(z^k)|| \neq 0$
  $\Delta z = z^k - z^{k-1}$;
  $\Delta g = \bar{g}^k - \bar{g}^{k-1}$;
  $\lambda_k^{BB} = \dfrac{\Delta z^T \Delta z}{\Delta g^T \Delta z}$;
  Escolher o passo $\lambda_j \in \Lambda$ mais próximo a λ_k^{BB} e retirá-lo do conjunto Λ;
  $z^{k+1} = z^k - \lambda_j \bar{g}^k$;
  $\bar{g}^{k+1} = Az^{k+1} + b$;
  $k = k + 1$;
fim
Resultado: z^k

No exemplo apresentado na Figura 13, que utilizaremos apenas para esta seção, podemos observar a melhora no comportamento do algoritmo de máximo declive utilizando a reordenação dos passos descrita acima. Nesse exemplo, a função objetivo é $f(x) = \frac{1}{2}x^T Dx$, com 1000 variáveis, o número de condicionamento de D é igual a 1000 e o ponto inicial é $x_i^0 = 1/\sqrt{d_i}, i = 1, \ldots, 1000$. Neste caso, o conjunto Λ possui 193 tamanhos de passo. Chamaremos o algoritmo que utiliza os passos do conjunto Λ em ordem crescente de Chebyshev.

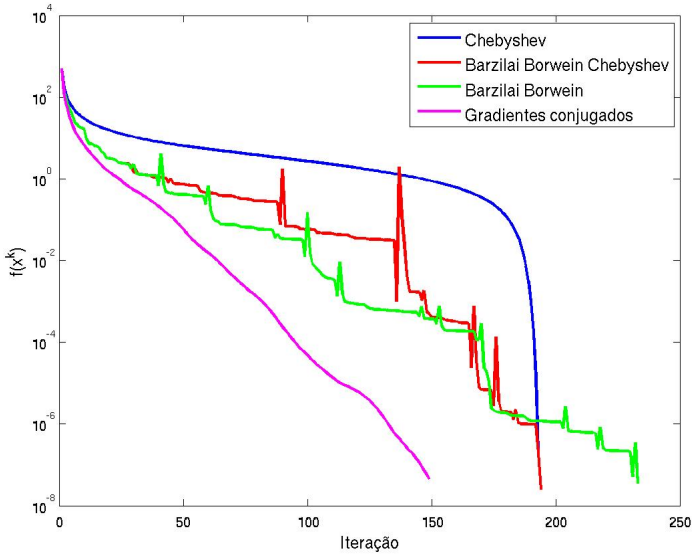

Figura 13 – Variação da função ao longo das iterações nos métodos Barzilai-Borwein, Chebyshev, Barzilai-Borwein Chebyshev e Gradientes conjugados na resolução de um problema quadrático.

Note que, apesar de ambos os algoritmos, Chebyshev e Barzilai-Borwein Chebyshev, terem resolvido o problema com a mesma quantidade de iterações (193), o algoritmo Barzilai-Borwein Chebyshev se aproxima da solução mais cedo do que o algoritmo Chebyshev. Uma vantagem de se utilizar raízes do polinômio de Chebyshev calculado utilizando o valor de C é que $|x_i^k| < \varepsilon |x_i^0|$ para todo k, ou seja, o método provoca uma redução da função na direção dada por cada autovetor de A.

5.6 MÉTODO CS COM POLINÔMIO DE CHEBYSHEV

Utilizaremos, agora, o mesmo processo apresentado na seção anterior aplicando-o ao método CS. Suponha que d_1 e d_n sejam conhecidos. Assim, é possível calcular o conjunto Λ de passos relacionados às raízes do polinômio de Chebyshev de ordem k. O novo algoritmo segue o mesmo padrão do algoritmo CS, entretanto, tanto os passos de Cauchy quanto os passos curtos serão substituídos pelos passos do conjunto Λ mais próximos a eles. Além disso, os passos do conjunto Λ serão utilizados apenas uma vez, ou seja, após o passo λ_j do conjunto Λ ser escolhido, o mesmo será excluído do conjunto Λ para que não seja utilizado novamente. Estamos supondo que d_1 e d_n são conhecidos apenas para a análise do desempenho do algoritmo. Posteriormente, apresentaremos um método adaptativo que utilizará raízes de polinômios de Chebyshev sem o conhecimento inicial dos autovalores de A.

O método CS utilizando raízes do polinômio de Chebyshev está definido no Algoritmo 13.

Na Figura 5.6 podemos observar o desempenho dos algoritmos ao resolverem o mesmo problema.

5.6.1 Método CS com polinômio de Chebyshev adaptativo

Como, na prática, não sabemos o número de condicionamento de A, vamos utilizar um processo adaptativo no método CS Chebyshev para obter uma aproximação dos valores de d_1 e d_n e, assim, calcular o conjunto Λ de passos calculados através das raízes do polinômio de Chebyshev. Assim como nos outros métodos, fazemos algumas iterações iniciais com passos de Cauchy para reduzir as variáveis intermediárias. Calculamos, então, um passo curto λ^S para aproximar o valor de $1/d_n$ e faremos $u = (1,2)\dfrac{1}{\lambda^S}$. Fazemos, então, $l = u/100$ para aproximar d_1. Assim, temos uma aproximação inicial l de d_1 e u de d_n. A cada iteração, caso tenhamos uma melhor aproximação para d_1 ou d_n, ou seja, se obtivermos algum passo de Cauchy λ_k^C tal que $1/\lambda_k^C > u$ ou $1/\lambda_k^C < l$, atualizamos os valores de u ou l multiplicando u por 1,2 ou dividindo l por 4. Essa atualização também será feita utilizando a mesma comparação para o passo curto λ^S. Se os valores de u ou l forem atualizados, construímos um novo conjunto Λ de raízes de polinômio de Chebyshev relacionado com o novo valor aproximado do número de condicionamento de A, ou seja, com o valor $(\tilde{C}) = \dfrac{u}{l}$.

Algoritmo 13: Método CS com polinômio de Chebyshev

Dados: $z^0 \in \mathbb{R}^n, p$ inteiro, m inteiro, $L \in \mathbb{R}$ "muito grande"; d_1, d_n;
$\bar{g}^0 = Az^0 + b$;
Dar 10 passos de Cauchy iniciais;
$k = 10$;
Calcular Λ utilizando $C = d_n/d_1$; (conjunto de passos calculados por raízes de Chebyshev)

enquanto $||\nabla \bar{f}(z^k)|| \neq 0$
  para $i = 1 : m$
    $h^k = A\,\bar{g}^k$;
    $\bar{\lambda}_k^C = \dfrac{\bar{g}^{k\,T}\bar{g}^k}{\bar{g}^{k\,T}h^k}$;
    Escolher o passo $\lambda_j \in \Lambda$ mais próximo a $\bar{\lambda}_k^C$ e retirá-lo do conjunto Λ;
    $z^{k+1} = z^k - \lambda_j\,\bar{g}^k$;
    $\bar{g}^{k+1} = \bar{g}^k - \lambda_j h^k$;
    $k = k + 1$;
  fim

  Calcular o passo curto λ^S utilizando $\bar{g}^k$ e (5.1);

  para $i = 1 : h$

    Escolher o passo $\lambda_j \in \Lambda$ mais próximo a λ^S e retirá-lo do conjunto Λ;

    $z^{k+1} = z^k - \lambda_j\,\bar{g}^k$;
    $\bar{g}^{k+1} = \bar{g}^k - \lambda_j A\,\bar{g}^k$;
    $k = k + 1$;
  fim
fim
Resultado: z^k

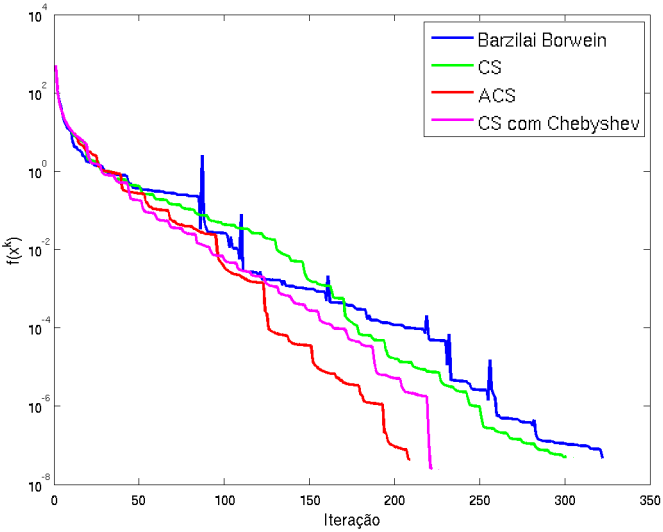

Figura 14 – Variação da função ao longo das iterações nos métodos Barzilai-Borwein, CS, ACS e CS com Chebyshev na resolução de um mesmo problema quadrático.

Observe que, se o número de condicionamento aproximado por $\tilde{C} = u/l$ for maior do que o número de condicionamento verdadeiro, o conjunto Λ, calculado com base no número de condicionamento aproximado, possuirá mais passos do que o caso em que utilizamos o número de condicionamento verdadeiro. Se o conjunto Λ possuir mais passos do que a quantidade necessária, tal excesso não será prejudicial. Devemos nos preocupar quando o conjunto Λ, calculado com o valor aproximado do número condicionamento $\tilde{C}$, possuir menos passos do que o necessário. Se isso acontecer, construímos o conjunto Λ novamente, utilizando o último valor aproximado $\tilde{C}$. A seguir, apresentamos a subrotina utilizada para atualização dos valores de u_i e l_i, que não será explicitada nos próximos algoritmos.

Algoritmo 14: Processo adaptativo para aproximação do número de condicionamento da A

Dados: λ_k, l_i, u_i, para algum i;
se $\dfrac{1}{\lambda_k} < l_i$
  $l_{i+1} = \dfrac{l_i}{4}$; $u_{i+1} = u_i$; $i+1$;
fim
se $\dfrac{1}{\lambda_k} > u_i$
  $u_{i+1} = (1,2)u_i$; $l_{i+1} = l_i$; $i+1$;
fim
se *l ou u foram atualizados*
  Calcular Λ usando u_i e l_i;
fim

Este processo adaptativo também será utilizado para os passos curtos. O método CS com polinômios de Chebyshev com o processo adaptativo fica definido no Algoritmo 15.

Na Figura 15, utilizamos o método CS com polinômio Chebyshev adaptativo para resolver o mesmo problema que foi utilizado para testar os algoritmos anteriores.

5.7 MÉTODO ACS COM POLINÔMIO DE CHEBYSHEV

Utilizaremos, agora, a técnica de retirar passos de um conjunto finito Λ, calculado através das raízes do polinômio de Chebyshev, aplicada ao método ACS.

Algoritmo 15: Método CS com polinômio de Chebyshev adaptativo

Dados: $z^0 \in \mathbb{R}^n, p$ inteiro, m inteiro, $L \in \mathbb{R}$ "muito grande";
$\bar{g}^0 = Az^0 + b$;
Dar 10 passos de Cauchy iniciais;
$k = 10$;
Calcular o passo curto λ^S utilizando $\bar{g}^k$;
$u_1 = (1,2)\dfrac{1}{\lambda^S}$; $\quad l_1 = u_1/100$;
Calcular Λ utilizando $C = \dfrac{u_1}{l_1}$;
enquanto $||\nabla \bar{f}(z^k)|| \neq 0$
  para $i = 1 : m$
    $h^k = A\,\bar{g}^k$;
    $\bar{\lambda}_k^C = \dfrac{{\bar{g}^k}^T \bar{g}^k}{{\bar{g}^k}^T h^k}$;
    Escolher o passo $\lambda_j \in \Lambda$ mais próximo a $\bar{\lambda}_k^C$ e retirá-lo do conjunto Λ;
    $z^{k+1} = z^k - \lambda_j\,\bar{g}^k$;
    $\bar{g}^{k+1} = \bar{g}^k - \lambda_j h^k$;
    $k = k + 1$;
  fim
  Calcular o passo curto λ^S utilizando $\bar{g}^k$;
  Verificar se é possível atualizar os valores de u_i e l_i;
  para $i = 1 : p$
    Escolher o passo $\lambda_j \in \Lambda$ mais próximo a λ^S e retirá-lo do conjunto Λ;
    $z^{k+1} = z^k - \lambda_j\,\bar{g}^k$;
    $\bar{g}^{k+1} = \bar{g}^k - \lambda_j\,\bar{g}^k$;
    $k = k + 1$;
  fim
fim
Resultado: z^k

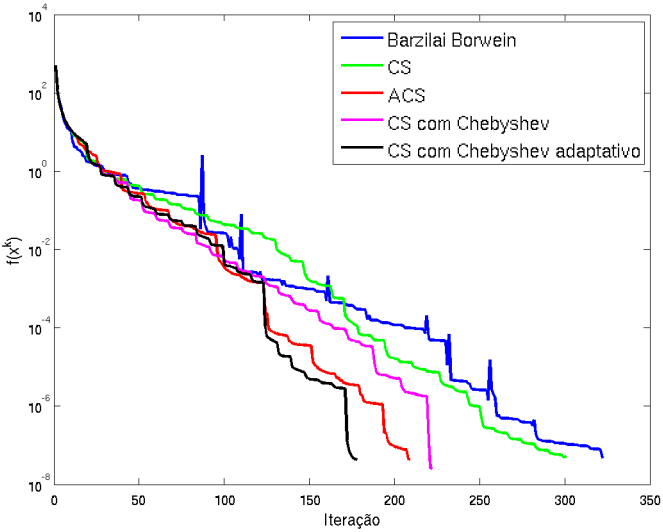

Figura 15 – Variação da função ao longo das iterações nos métodos Barzilai-Borwein, CS, ACS, CS Chebyshev, CS Chebyshev adaptativo na resolução de um mesmo problema quadrático.

Vamos supor, inicialmente, que o maior e o menor autovalores de A sejam conhecidos. O algoritmo é semelhante ao método ACS, entretanto, para cada passo de Cauchy $\bar{\lambda}_k^C$ e curto λ^S, iremos escolher os passos no conjunto Λ mais próximos e retirá-los do conjunto Λ para que não sejam utilizados novamente (Algoritmo 16).

Novamente, na Figura 16 apresentamos o desempenho do método ACS com polinômio de Chebyshev para o mesmo problema quadrático utilizado nos algoritmos anteriores.

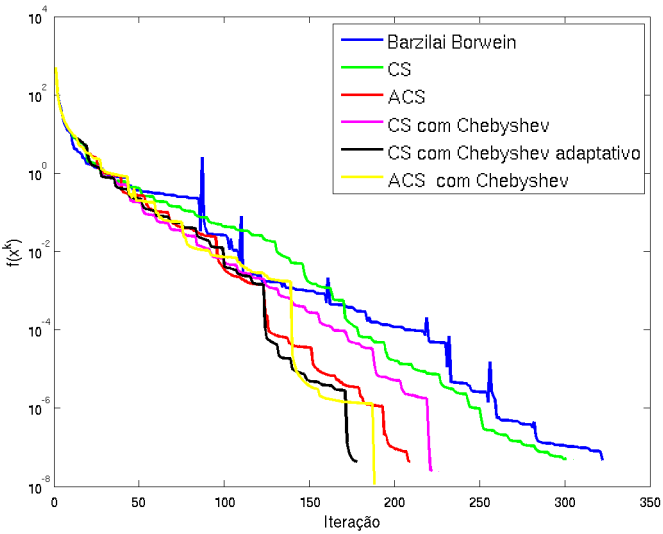

Figura 16 – Métodos Barzilai-Borwein, CS, ACS, CS Chebyshev, CS Chebyshev adaptativo e ACS Chebyshev para um mesmo problema quadrático.

5.7.1 Método ACS com polinômio de Chebyshev adaptativo

Novamente, como não sabemos os autovalores de A, vamos utilizar o processo adaptativo apresentado na seção 5.6.1, agora no algoritmo ACS com polinômio de Chebyshev. Assim, obtemos o Algoritmo 17.

Na Figura 17 apresentamos o método ACS com polinômio de Chebyshev adaptativo aplicado ao mesmo problema que foi utilizado

Algoritmo 16: Método ACS com polinômio de Chebyshev

Dados: $z^0 \in \mathbb{R}^n, p$ inteiro, m inteiro, $L \in \mathbb{R}$ "muito grande", d_1, d_n;
$\bar{g}^0 = Az^0 + b$;
Dar 10 passos de Cauchy iniciais;
$k = 10$;
Calcular Λ utilizando $C = \dfrac{d_1}{d_n}$; (*conjunto de passos calculados por raízes de Chebyshev*)
Calcular o passo curto λ^S utilizando $\bar{g}^k$;
enquanto $||\nabla \bar{f}(z^k)|| \neq 0$
  para $i = 1 : 2m$
    $h^k = A\,\bar{g}^k$; $\quad \bar{\lambda}_k^C = \dfrac{{\bar{g}^k}^T \bar{g}^k}{{\bar{g}^k}^T h^k}$;
    Escolher o passo $\lambda_j \in \Lambda$ mais próximo a $\bar{\lambda}_k^C$ e retirá-lo do conjunto Λ;
    $x^{k+1} = x^k - \lambda_j\,\bar{g}^k$;
    $\bar{g}^{k+1} = \bar{g}^k - \lambda_j h^k$;
    $k = k + 1$;
    Escolher o passo $\lambda_j \in \Lambda$ mais próximo a λ_k^S e retirá-lo do conjunto Λ;
    $x^{k+1} = x^k - \lambda_j\,\bar{g}^k$;
    $\bar{g}^{k+1} = \bar{g}^k - \lambda_j A\,\bar{g}^k$;
    $k = k + 1$;
  fim
  Calcular o passo curto λ^S utilizando $\bar{g}^k$;
  para $i = 1 : p$
    $h^k = A\,\bar{g}^k$;
    Escolher o passo $\lambda_j \in \Lambda$ mais próximo a λ^S e retirá-lo do conjunto Λ;
    $z^{k+1} = z^k - \lambda_j\,\bar{g}^k$;
    $\bar{g}^{k+1} = \bar{g}^k - \lambda_j A\,\bar{g}^k$;
    $k = k + 1$;
  fim
fim
Resultado: z^k

Algoritmo 17: Método ACS com polinômio de Chebyshev adaptativo

Dados: $z^0 \in \mathbb{R}^n, p$ inteiro, m inteiro, $L \in \mathbb{R}$ "muito grande";
$\bar{g}^0 = Az^0 + b$;
Dar 10 passos de Cauchy iniciais;
$k = 10$;
Calcular o passo curto λ^S utilizando $\bar{g}^k$ e (5.1);
$u_1 = (1, 20\dfrac{1}{\lambda^S}; \quad l_1 = \dfrac{u_1}{100}$;
Calcular Λ utilizando $C = \dfrac{u_1}{l_1}$;
enquanto $||\nabla \bar{f}(z^k)|| \neq 0$
  para $i = 1 : 2m$
    $h^k = A\bar{g}^k$;
    $\bar{\lambda}_k^C = \dfrac{\bar{g}^{k\,T}\bar{g}^k}{\bar{g}^{k\,T}h^k}$;
    Escolher o passo $\lambda_j \in \Lambda$ mais próximo a $\bar{\lambda}_k^C$ e retirá-lo do conjunto Λ;
    $z^{k+1} = z^k - \lambda_j \bar{g}^k$;
    $\bar{g}^{k+1} = \bar{g}^k - \lambda_j h^k$;
    $k = k + 1$;
    Verificar se é possível atualizar os valores de u_i e l_i;
    Escolher o passo $\lambda_j \in \Lambda$ mais próximo a λ_k^S e retirá-lo do conjunto Λ;
    $z^{k+1} = z^k - \lambda_j \bar{g}^k$;
    $\bar{g}^{k+1} = \bar{g}^k - \lambda_j A\bar{g}^k$;
    $k = k + 1$;
  fim
  Calcular o passo curto λ^S utilizando $\bar{g}^k$;
  Verificar se é possível atualizar os valores de u_i e l_i;
  para $i = 1 : p$
    $h^k = A\bar{g}^k$;
    Escolher o passo $\lambda_j \in \Lambda$ mais próximo a λ^S e retirá-lo do conjunto Λ;
    $z^{k+1} = z^k - \lambda_j \bar{g}^k$;
    $\bar{g}^{k+1} = \bar{g}^k - \lambda_j A\bar{g}^k$;
    $k = k + 1$;
  fim
fim
Resultado: z^k

nos exemplos anteriores.

Na Figura 18, apresentamos o desempenho dos métodos propostos (sem o conhecimento dos autovalores de A), comparando-os com o método SDA, Barzilai-Borwein e gradientes conjugados.

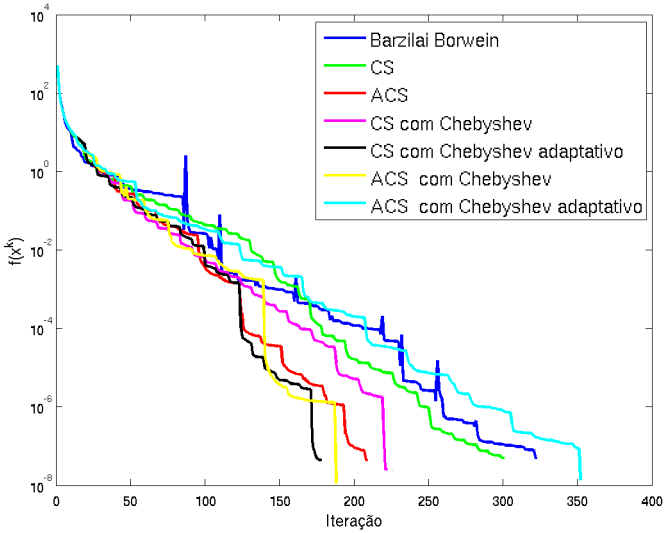

Figura 17 – Variação da função ao longo das iterações nos métodos Barzilai-Borwein, CS, ACS, CS Chebyshev, CS Chebyshev adaptativo, ACS Chebyshev e ACS Chebyshev adaptativo na resolução de um mesmo problema quadrático.

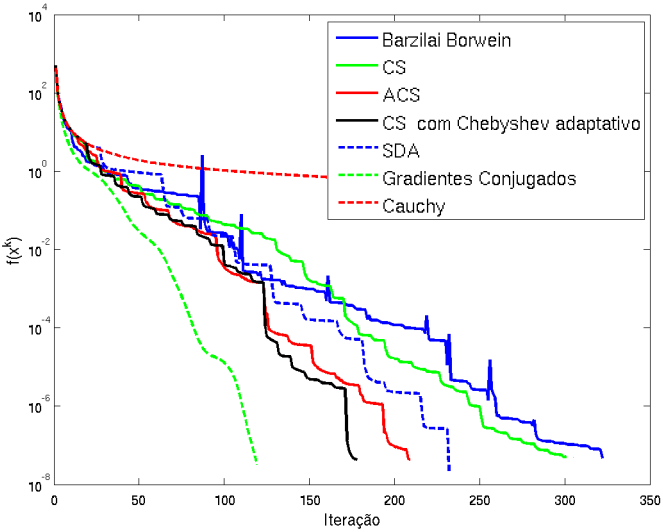

Figura 18 – Variação da função ao longo das iterações nos métodos Barzilai-Borwein, CS, ACS, CS Chebyshev adaptativo, ACS Chebyshev adaptativo, SDA e gradientes conjugados na resolução de um mesmo problema quadrático.

5.8 TESTES COMPUTACIONAIS

Para fazer uma comparação entre os métodos, implementamos, em MATLAB®, os seguintes algoritmos:

(i) Barzilai-Borwein, Algoritmo 6;

(ii) SDA, Algoritmo 9;

(iii) CS, Algoritmo 10;

(iv) ACS, Algoritmo 11;

(v) CS com Chebyshev adaptativo, Algoritmo 13.

Utilizamos 120 problemas quadráticos simplificados, com função objetivo dada por

$$f(x) = \frac{1}{2}x^T Dx$$

com 1000 variáveis, em que D é uma matriz diagonal cujos elementos da diagonal, $d_1, \ldots, d_n$ satisfazem $0 < d_1 < \ldots < d_n$. Cada conjunto de 30 problemas possui uma distribuição diferente de autovalores. Dentre os 30 problemas de cada tipo de distribuição, 10 problemas possuem número de condicionamento $C = 1000$, outros 10 problemas possuem número de condicionamento $C = 10000$, e os últimos 10 possuem número de condicionamento $C = 100000$. Em todos os casos, o ponto inicial escolhido foi $x_i^0 = 1/\sqrt{d_i}$. As distribuição utilizadas foram: uniforme, logarítmica e senoidal. Além disso, utilizamos uma distribuição na qual há muitos autovalores grandes e muitos autovalores pequenos, porém, existem alguns autovalores próximos de $\dfrac{d_1 + d_n}{2}$. Na Figura 19 apresentamos um exemplo de cada distribuição para o problema de 1000 variáveis em que o condicionamento de D é 1000. No eixo horizontal, representamos o índice do autovalor e, no vertical, representamos o valor de cada autovalor.

Utilizamos um gráfico de perfil de desempenho, como proposto em [15], comparando a quantidade de iterações que cada algoritmo necessitou para resolver cada problema, utilizando como critério de parada

$$f(x^k) - f(x^*) \leq \varepsilon(f(x^0) - f(x^*))$$

em que $f(x^*) = 0$ e $\varepsilon = 10^{-10}$.

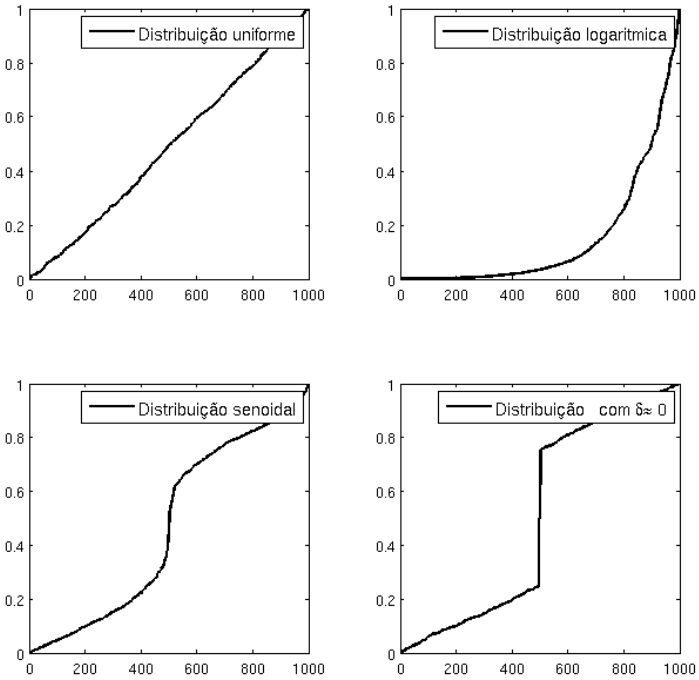

Figura 19 – Distribuição dos autovalores de D nos problemas utilizados nos testes computacionais.

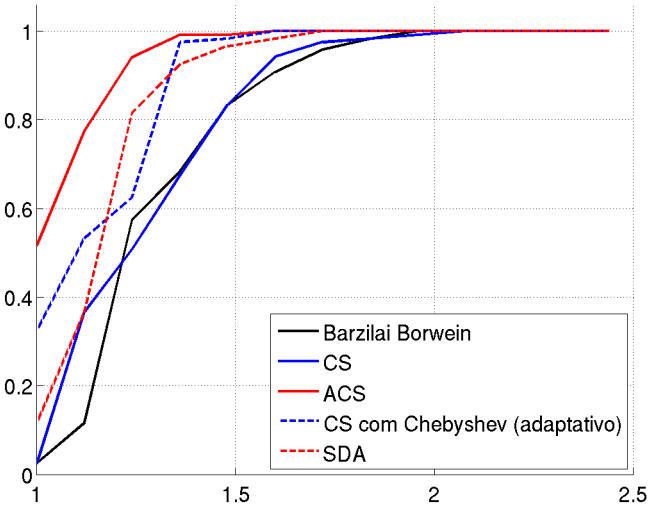

Figura 20 – Gráfico de perfil de desempenho dos métodos propostos juntamente com o método de Barzilai-Borwein.

Com esses testes, concluímos que o algoritmo ACS supera os outros algoritmos propostos e do método de Barzilai-Borwein tanto em eficiência quanto em robustez. Além disso, o algoritmo CS com polinômio de Chebyshev que utiliza o processo adaptativo para os autovalores de A também exibe um desempenho relevante.

CONCLUSÃO

Neste trabalho apresentamos o método de máximo declive e provamos a convergência global. Enunciamos o método de máximo declive com busca exata para problemas de minimização quadrática, o qual chamamos de método de Cauchy. Realizamos, ainda, uma mudança de variável no problema quadrático geral, o que contribuiu para uma análise simplificada do comportamento oscilatório do método, realizada no Capítulo 4. Mostramos o desempenho do método de Cauchy no pior caso possível e enunciamos resultados de complexidade para outros métodos. Apresentamos um método que utiliza raízes de um polinômio de Chebyshev, proposto em [9], que adaptamos para ser utilizado nos métodos que foram propostos neste trabalho e em [10].

Com o método de Barzilai-Borwein, e algumas variantes, apresentados no Capítulo 3, vimos que alguns métodos com busca linear não exata e não monótonos podem ser eficientes para problemas de minimização quadrática, pois não geram um comportamento oscilatório, frequentemente observado no método de Cauchy. Assim, passamos a nos preocupar em entender as propriedades do método de Cauchy para, possivelmente, propor uma maneira eficiente de quebrar esse comportamento oscilatório, porém, procurando manter a monotonicidade do método.

Desse modo, no Capítulo 4 apresentamos uma análise detalhada sobre o comportamento do gradiente da função objetivo bem como da sequência de passos calculados no método de Cauchy. Vimos que o método de Cauchy converge para uma busca linear no subespaço bidimensional gerado pelos autovetores correspondentes ao menor e ao maior autovalor da matriz A do problema quadrático. Com isso, percebemos que os passos de Cauchy são, geralmente, intermediários, portanto, não são eficientes para diminuir variáveis leves e pesadas, pois convergem para dois valores fixos. Essa análise, bem como o método SDA, proposto por De Asmundis et al [5], nos permitiram encontrar uma maneira eficiente de calcular passos pequenos que, na verdade, também são passos de Cauchy, porém, obtidos após um passo muito grande, que não é realizado.

Enfim, apresentamos novos algoritmos no Capítulo 5. Utilizamos passos de Cauchy para uma quantidade m de iterações e passos curtos, para outras p iterações no método de máximo declive e chamamos este método de CS. Sugerimos, também, que um passo pequeno fosse dado

com mais frequência, a cada duas iterações, o que caracterizou o método ACS. Para ambos os métodos, sugerimos uma adaptação que utiliza raízes de um polinômio de Chebyshev, primeiramente calculado com o conhecimento do maior e do menor autovalor de A, apenas para análise de desempenho, e, em seguida, calculado com o auxílio de um processo adaptativo para aproximação do maior e do menor autovalor. Todos os algoritmos citados também foram propostos em [10].

Para finalizar, realizamos alguns testes computacionais e apresentamos um gráfico de perfil de desempenho para os métodos propostos em comparação com o método de Barzilai-Borwein. Constatamos que os métodos propostos tiveram bom desempenho, semelhante ao Barzilai-Borwein. Destacaram-se, entre eles, os métodos ACS e CS com polinômio de Chebyshev adaptativo.

REFERÊNCIAS

1 AKAIKE, H. **On a successive transformation of probability distribution and its application to the analysis of the optimum gradient method**. Ann. Inst. Stat. Math: vol. 11, pp. 1-16. Tokyo, 1959.

2 BARZILAI, J.; BORWEIN, J. M. **Two point step size gradient methods**, IMA Journal of Numerical Analysis: vol. 8, pp. 141-148, 1988.

3 CAUCHY, A. **Méthode générale pour la résolution des systèmes d'équations simultanées.** C. R. Acad. Sci.: vol. 25, pp. 536-538. Paris, 1847.

4 DAI, Y. H. **Alternate step gradient method**, Optimization: vol 52, pp. 395-415, 2003.

5 DE ASMUNDIS, R.; DI SERAFINO, D.; RICCIO, F.; TORALDO, G. **On spectral properties of steepest descent methods**. IMA Journal of Numerical Analysis: vol. 33, pp. 1416-1435, 2013.

6 FLETCHER, R. **Low storage methods for unconstrained optimization,** Lectures in Applied Mathematics: vol. 26, pp. 165-179. American Mathematical Society, 1990.

7 FLETCHER, R.; REEVES, C.M. **Function minimization by conjugate gradients**. Computer J.: vol. 7, pp. 149-154, 1964.

8 FORSYTHE, G. R. **On the Asymptotic directions of the s-dimensional optimum gradient method**, Numer. Math: vol. 11, pp. 57-76, 1968.

9 GONZAGA, C. C. **Optimal performance of the steepest descent algorithm for quadratic functions**. Technical report. Universidade Federal de Santa Catarina, 2014.

10 GONZAGA, C. C.; SCHNEIDER, R. M. **The steepest descent algorithm for quadratic functions**. Technical report. Universidade Federal de Santa Catarina, 2015.

11 GONZAGA, C. C.; KARAS, E. W. **Complexity of first-order methods for differentiable convex optimization.** Special issue

on Nonlinear Programming, Pesquisa Operacional: vol. 34, pp. 395-419. Brazil, 2014.

12 GRIPPO; L.; LAMPARIELLO, F.; LUCIDI, S. **A nonmonotone line search technique for Newton's method**, SIAM Journal on Numerical Analysis: vol. 23, pp. 707-716, 1986.

13 KARAS, E. W.; RIBEIRO, A. A. **Otimização Contínua: aspectos teóricos e computacionais**. São Paulo: Cengage Learning, 2013.

14 LUENBERGER, D.G.; YE, Y. **Linear and Nonlinear Programming.** Third edition. New York: Springer,2008.

15 DOLAN, E. D.; MORÉ, J. J. **Benchmarking optimization software with performance profiles**. Mathematical Programming: vol. 91(2), pp. 201-213, 2002.

16 NEMIROVSKI, A. S.; YUDIN, D. B. **Problem Complexity and Method Efficiency in Optimization.** New York: John Wiley & Sons, 1983.

17 NOCEDAL, J.; SARTENAER, A.; Zhu, C. **On the Behavior of the Gradient Norm in the Steepest Descent Method**. Computational Optimization and Applications: vol. 22, pp. 5-35, 2002.

18 NOCEDAL, J.; WRIGHT, S. **Numerical Optimization**. Second edition. Springer Series in Operations Research. Springer: 2006.

19 POLYAK, B. T. **Introduction to Optimization**. Optimization Software Inc. New York, 1987.

20 RAYDAN, M. **On the Barzilai and Borwein choice of steplength for the gradient method**. IMA Journal of Numerical Analysis, vol. 13, pp. 321-326, 1993.

21 RAYDAN, M. **The Barzilai and Borwein Gradient Method for the Large Scale Unconstrained Minimization Problem**, SIAM Journal on Optimization: vol. 7, pp. 26-33, 1997.

22 RAYDAN, M.; SVAITER, B. F. **Relaxed Steepest Descent and Cauchy-Barzilai-Borwein Method**, Computational Optimization and Applications: vol. 21, pp. 155-167, 2002.

23 SHEWCHUK, J. R. **An introduction to the conjugate gradient method without the agonizing pain.** Technical report, School of Computer Science. Carnegie Mellon University, 1994.