UNIVERSIDADE ESTADUAL DE CAMPINAS
INSTITUTO DE MATEMÁTICA, ESTATÍSTICA E COMPUTAÇÃO CIENTÍFICA
DEPARTAMENTO DE MATEMÁTICA APLICADA

GABRIEL RODRIGUES SILVA GRILLO

# Métodos de primeira ordem acelerados e buscas adaptativas para minimização suave

Campinas
08/12/2021

GABRIEL RODRIGUES SILVA GRILLO

# Métodos de primeira ordem acelerados e buscas adaptativas para minimização suave*

Monografia apresentada ao Instituto de Matemática, Estatística e Computação Científica da Universidade Estadual de Campinas como parte dos requisitos para obtenção de créditos na disciplina Projeto Supervisionado, sob a orientação do(a) Profa. Dra. Sandra Augusta Santos.

*Este trabalho foi financiado pela FAPESP, projeto 13946-3/2020.

# Resumo

Os desafios contemporâneos provenientes do tratamento de informações com grande quantidade de dados têm renovado o interesse por métodos que combinem baixo custo e eficiência para resolver problemas de otimização suave, em contextos inadequados para o uso de informações de segunda ordem. Motivado por tais desafios, o projeto em tela aborda o estudo de estratégias de gradiente acelerado, visando analisar possíveis relações entre os esquemas iterativos discretos e suas contrapartidas contínuas, que envolvem equações diferenciais ordinárias associadas.

# Abstract

Contemporary challenges arising from the treatment of information based on huge amount of data have generated renewed interest for first-order methods. Enriched with acceleration schemes, these methods may combine low cost and efficiency for solving smooth optimization problems, unsuited to second-order approaches. Motivated by such challenges, this project addresses the study of accelerated gradient strategies, aiming to analyze possible relationships between the discrete iterative schemes and their continuous counterparts, which involve related ordinary differential equations

# Conteúdo

# 1 Introdução

O presente trabalho se propõe a estudar métodos numéricos para minimização irrestrita de funções suaves utilizando-se apenas informações de primeira ordem. Diferentemente dos métodos de segunda ordem, como o método de Newton, os métodos tratados neste trabalho dispensam o cálculo e o armazenamento da matriz hessiana da função objetivo. Em dimensões grandes, essa característica se torna uma vantagem, visto que o simples armazenamento da matriz hessiana pode ser um desafio devido a restrições de memória.

Todavia, métodos de segunda ordem possuem rápida convergência, principalmente quando comparados com o método de Cauchy (também conhecido como método da máxima descida ou método do gradiente descendente), que possui convergência lenta. Assim, métodos capazes de alcançar maior velocidade de convergência sem a necessidade do uso da matriz hessiana passam a ser fundamentais para a solução de problemas de minimização em grandes dimensões.

Esses problemas surgem naturalmente na área de Aprendizado de Máquinas, visto que o treinamento de um modelo preditivo é, muitas vezes, um problema de minimização, em que a dimensão dos problemas cresce conforme a quantidade de dados disponíveis cresce. No contexto do *Big Data*, portanto, métodos de segunda ordem e o método de Cauchy se tornam inadequados, este pela lentidão da convergência e aquele pela restrição de memória.

Com isso, os métodos de primeira ordem acelerados ganham destaque, por balancearem características de eficiência e custo computacional. Em especial, os métodos com características estocásticas tendem a ser os escolhidos para os processos de treinamento com muitos dados, como é o caso do método *Adam* [6], amplamente usado em pacotes populares de Ciência de Dados, como `Keras` e `PyTorch`.

O método *Adam* se vale da ideia do uso de inércia: a direção tomada é uma combinação convexa dos gradientes calculados no ponto corrente e no ponto passado. O uso de inércia já estava presente em [15], em que métodos de passos múltiplos gerais são estudados. Vemos, então, o recente resgate de métodos de primeira ordem acelerados no contextos dos desafios atuais.

Além das aplicação das releituras estocásticas desses métodos, também existem trabalhos recentes sobre aspectos teóricos dos mesmos, como [19], em que os autores estudam a contrapartida contínua do método de Nesterov de 1983 [7].

O estudo das equações diferenciais ordinárias associadas aos métodos de otimização é uma estratégia para compreender esses métodos a partir do ferramental da análise numérica de solução de PVIs. Isso traz um ganho na intuição e amplia o repertório de estudo dos métodos.

Sendo assim, o presente trabalho se propõe a estudar o problema de minimização irrestrita suave e métodos de primeira ordem acelerados para a solução desse problema, o que é feito na Seção 2. Também, realizar a implementação dos métodos em `Matlab` e a experimentação computacional comparativa, o que é feito na Seção 3. Por fim, estudar suas contrapartidas contínuas, o que é feito na Seção 4. Os pseudocódigos nos quais as implementações computacionais estão baseadas se encontram no Apêndice A.

# 2 Fundamentos e resultados preliminares

Consideraremos o problema de minimização irrestrita

$$\begin{aligned} &\min f(x) \\ &\text{s.a } x \in \mathbb{R}^n, \end{aligned} \tag{2.1}$$

em que $f : \mathbb{R}^n \to \mathbb{R}$ é continuamente diferenciável ($f \in \mathcal{C}^1$) e *convexa*, i.e. vale a relação

$$f(x) \geq f(y) + \nabla f(y)^T (x - y) \tag{2.2}$$

para quaisquer $x, y \in \mathbb{R}^n$. Na Figura 1 vemos a interpretação geométrica de (2.2): a função $f \in \mathcal{C}^1$ é convexa quando o gráfico de $f$ nunca está abaixo de suas retas (no caso $n = 1$) tangentes.

Além disso, consideraremos funções $f$ *$L$-fortemente suaves*, $L \geq 0$, i.e. tais que

$$f(x) \leq f(y) + \nabla f(y)^T (x - y) + \frac{L}{2} \|x - y\|_2^2, \tag{2.3}$$

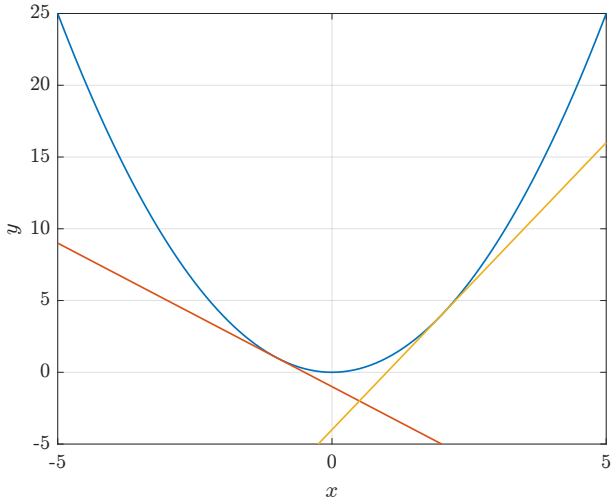



Figura 1: Exemplo de gráfico de uma função $f : \mathbb{R} \to \mathbb{R}$ diferenciável e convexa, juntamente com duas retas tangentes. (Ilustração com $f(x) = x^2$)

para quaisquer $x, y \in \mathbb{R}^n$. Quando $f$ é convexa, a $L$-suavidade forte é equivalente ao gradiente ser $L$-Lipschitz contínuo [8, Teorema 2.1.5], i.e.

$$\|\nabla f(x) - \nabla f(y)\|_2 \leq L\|x - y\|_2, \tag{2.4}$$

para quaisquer $x, y \in \mathbb{R}^n$.

Também, consideraremos funções $f$ *$\mu$-fortemente convexas*, $\mu > 0$, i.e. tais que

$$f(x) \geq f(y) + \nabla f(y)^T(x - y) + \frac{\mu}{2}\|x - y\|_2^2, \tag{2.5}$$

para quaisquer $x, y \in \mathbb{R}^n$.

Do ponto de vista geométrico, conforme exemplo da Figura 2, podemos interpretar uma função $f \in \mathcal{C}^1$ $L$-fortemente suave como sendo aquela que o seu gráfico *nunca está acima* de uma quadrática simples $q$, tal que $q(y) = f(y)$, $\nabla q(y) = \nabla f(y)$ e $\nabla^2 q(y) = LI_n$, para um ponto $y \in \mathbb{R}^n$ qualquer.

Analogamente, podemos interpretar uma função $f \in \mathcal{C}^1$ $\mu$-fortemente convexa como sendo aquela que o seu gráfico *nunca está abaixo* de uma quadrática simples $p$, tal que $p(y) = f(y)$, $\nabla p(y) = \nabla f(y)$ e $\nabla^2 p(y) = \mu I_n$, para um ponto $y \in \mathbb{R}^n$ qualquer.

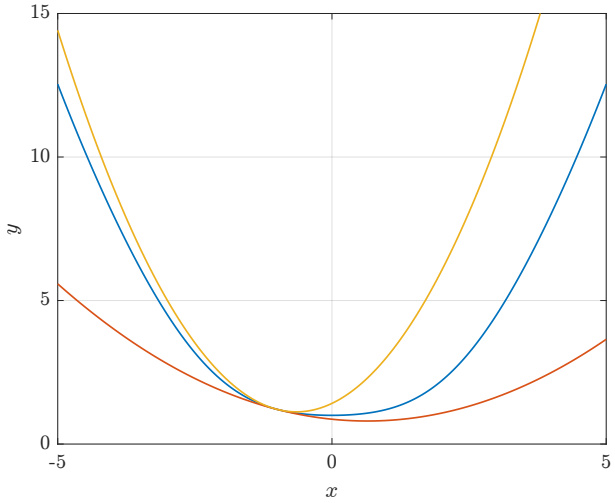



Figura 2: Exemplo do gráfico de uma função $f : \mathbb{R} \to \mathbb{R}$ continuamente diferenciável, fortemente convexa e fortemente suave, em azul, juntamente com o gráfico das quadráticas $q$, em amarelo, e $p$, em vermelho (Ilustração com $f(x) = \left(\frac{\sin x}{x}\right)^2 + \frac{x^2}{2}$)

A partir das constantes estruturais $\mu$ e $L$ de $f$, definimos o *número de condição* do problema $\kappa$, como $\kappa := \frac{L}{\mu}$. Veja que, por (2.3) e (2.5), $L \geq \mu$, logo $\kappa \geq 1$. Além disso, quando $f \in \mathcal{C}^2$, vale

$$\mu I_n \preceq \nabla^2 f(x) \preceq L I_n,$$

para todo $x \in \mathbb{R}^n$ [8, Teoremas 2.1.6 e 2.1.11]. Com isso, $\kappa$ é um limitante superior para o número de condição da matriz hessiana $\nabla^2 f(x)$ em um ponto $x \in \mathbb{R}^n$ qualquer.

Caso as escolhas de $\mu$ e $L$ sejam tais que esses valores estejam próximos do menor e maior autovalores de $\nabla^2 f(x)$, respectivamente, considerando todo $x \in \mathbb{R}^n$, então $\kappa$ é uma aproximação por cima para $\max_{x \in \mathbb{R}^n} \kappa_2(\nabla^2 f(x))$.

Além da associação de $\kappa$ com o número de condição da matriz $\nabla^2 f(x)$, também é possível associar esse valor a discrepâncias na variação de $f$: $\kappa \gg 1$ indica que em algumas regiões a função terá variações muito mais abruptas do que em outras.

Veja ainda que o número de condição caracteriza o problema de forma relativa: $\mu = 0.01$ e $L = 1$ implicam em $\kappa = 100$, assim como $\mu = 1$ e $L = 100$. Apesar de nas duas situações termos $\kappa = 100$, quando $\mu = 0.01$ temos o indicativo que a função tem alguma região em que varia pouco, como se seu gráfico fosse quase um platô. Por outro lado, quando $L = 100$, então temos uma região em que a função terá uma variação mais intensa.

Retomando o problema de minimizar $f$, considerando as características apre-

sentadas da função objetivo, pela Proposição 2.1, vemos que o problema (2.1) possui solução. Sendo assim, estudar métodos para determinar o minimizador $x^*$, que será único, passa a ser a nossa tarefa.

**Proposição 2.1.** *Seja $f : \mathbb{R}^n \to \mathbb{R}$ $\mu$-fortemente convexa, então existe um minimizador global único para $f$ em $\mathbb{R}^n$.*

*Demonstração.* Veja, por exemplo, o Corolário 3.4.20 de [5]. □

Veja que a hipótese de $f$ ser $L$-fortemente suave não é necessária para a existência e unicidade do minimizador, mas será uma característica importante para a aplicação de métodos de primeira ordem.

## 2.1 Método de Cauchy

O *método de Cauchy* (ou método do gradiente descendente) é o método de primeira ordem mais simples, sendo o iterado $x^{k+1}$ calculado apenas como um deslocamento do iterado $x^k$ na direção de $-\nabla f(x^k)$ considerando um passo $\alpha_k > 0$. Assim, dado $x^0 \in \mathbb{R}^n$, a sequência $\{x^k\}_{k=0}^{\infty}$ é gerada pelo processo

$$x^{k+1} = x^k - \alpha_k \nabla f(x^k), \tag{2.6}$$

em que o passo $\alpha_k$ pode ser pré-definido ou obtido a cada iteração com alguma estratégia de busca. A seguinte proposição apresenta o comportamento desse método quando o tamanho de passo é escolhido aproveitando-se informações da estrutura da função.

**Proposição 2.2** (Teorema 2.1.15 de [8] adaptado)**.** *Seja $f : \mathbb{R}^n \to \mathbb{R}$ tal que $f \in \mathcal{C}^2$ é $\mu$-fortemente convexa e $L$-fortemente suave. Então, $f$ possui minimizador global $x^* \in \mathbb{R}^n$ e, dado $x^0 \in \mathbb{R}^n$ qualquer, o método de Cauchy com passo constante $\alpha_k = \frac{2}{L+\mu}$ gera sequência $\{x^k\}_{k=0}^{\infty}$ tal que*

$$\|x^k - x^*\|_2 \leq \left(\frac{\kappa - 1}{\kappa + 1}\right)^k \|x^0 - x^*\|_2, \text{ em que } \kappa \coloneqq \frac{L}{\mu}.$$

No caso de as informações de $L$ e $\mu$ não estarem disponíveis ou as constantes não existirem, é possível determinar o tamanho de passo de uma iteração do método de Cauchy pela *regra de Armijo*. Essa regra estabelece um decréscimo suficiente da função a cada iteração, em que $\alpha_k > 0$ deve ser selecionado de forma que

$$f(x^k - \alpha_k \nabla f(x^k)) \leq f(x^k) - \sigma \alpha_k \nabla f(x^k)^T \nabla f(x^k), \tag{2.7}$$

em que $\sigma \in (0, 1)$ é um hiperparâmetro de controle de exigência do decréscimo. Para mais detalhes, veja, por exemplo, o Capítulo 3 de [13]

Veja na Proposição 2.2 como o número de condição $\kappa$ afeta a taxa de convergência do método de Cauchy: o método converge independentemente de $\kappa$, mas quanto maior for $\kappa$, mais próximo o quociente $\frac{\kappa-1}{\kappa+1}$ será de 1, de forma que $\left(\frac{\kappa-1}{\kappa+1}\right)^k$ irá para 0 mais lentamente. Com isso, quão pior for o condicionamento do problema, mais lenta será a convergência da sequência gerada pelo método de Cauchy.

Na prática, a taxa de convergência apresentada na Proposição 2.2 é muito sensível a aumentos de $\kappa$, de forma que a convergência se torna proibitivamente lenta mesmo em problemas com número de condição intermediários.

Dessa forma, vemos então a necessidade de se estudar métodos de primeira ordem acelerados, em que a taxa de convergência seja menos sensível a aumentos de $\kappa$ e possamos aproximar $x^*$ em tempo hábil para aplicações.

## 2.2 Método *Heavy-ball*

O método *Heavy-ball* (ou método da *bola pesada*) é um dos primeiros métodos de primeira ordem acelerados conhecidos, e consiste em acrescentar à iteração do método de Cauchy um termo de inércia, com iterações dadas por

$$x^{k+1} = x^k - \alpha_k \nabla f(x^k) + \beta_k (x^k - x^{k-1}), \tag{2.8}$$

em que $\alpha_k > 0$ e $\beta_k > 0$. A direção do método da bola pesada é, então, uma combinação linear entre a direção de Cauchy (menos o gradiente) e o passo dado na última iteração (termo de inércia).

Nesse método temos os tamanhos de passo $\alpha_k$ e $\beta_k$, que podem ser pré-definidos ou selecionados a cada iteração, de forma semelhante à seleção de $\alpha_k$ no método de Cauchy. A seguinte proposição apresenta tamanhos de passos fixos que garantem a convergência local da sequência $\{x^k\}_{k=0}^{\infty}$.

**Proposição 2.3** (Item (3) do Teorema 9 de [15] adaptado)**.** *Seja $f : \mathbb{R}^n \to \mathbb{R}$ tal que $f \in \mathcal{C}^2$ é $\mu$-fortemente convexa e $L$-fortemente suave. Então, $f$ possui minimizador global $x^* \in \mathbb{R}^n$ e, dados $x^0, x^1 \in \mathbb{R}^n$ suficientemente próximos de $x^*$, o método da bola pesada com passos $\alpha_k = \left(\frac{2}{\sqrt{L}+\sqrt{\mu}}\right)^2$ e $\beta_k = \frac{\sqrt{L}-\sqrt{\mu}}{\sqrt{L}+\sqrt{\mu}}$ gera sequência $\{x^k\}_{k=0}^{\infty}$ tal que*

$$\|x^k - x^*\|_2 \leq c_\delta \left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1} + \delta\right)^k \sqrt{\|x^0 - x^*\|_2^2 + \|x^1 - x^*\|_2^2},$$

*em que $\kappa := \frac{L}{\mu}$, $\delta > 0$ é suficientemente pequeno para que $\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1} + \delta < 1$ e, uma vez fixado $\delta$, $c_\delta$ é uma constante positiva.*

Veja que a taxa de convergência do método *heavy-ball* é semelhante à taxa apresentada na Proposição 2.2 para o método de Cauchy. Entretanto, a dependência do número de condição apresentada na Proposição 2.3 leva em consideração $\sqrt{\kappa}$ na posição onde existe $\kappa$ na Proposição 2.2. Assim, vemos que a taxa de convergência com o método *heavy-ball* é menos sensível a aumentos de $\kappa$.

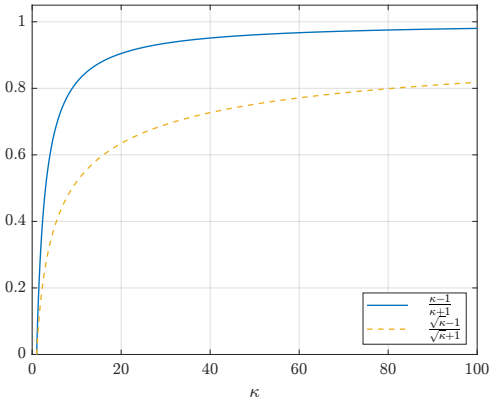



Figura 3: Comparativo entre as taxas de convergência das Proposições 2.2 e 2.3

Note pela Figura 3 que o quociente $\frac{\kappa-1}{\kappa+1}$ rapidamente se aproxima de 1, enquanto o quociente $\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}$ cresce mais lentamente, indicando que o método *heaby-ball* terá convergência mais rápida, em especial no cenário mal condicionado $\kappa \gg 1$.

Uma característica interessante do método da bola pesada é que, mesmo gerando uma sequência $\{x^k\}_{k=0}^{\infty}$ que converge mais rapidamente do que o método de Cauchy, a sequência $\{f(x^k)\}_{k=0}^{\infty}$ pode ser não monótona com as escolhas de passo dadas na Proposição 2.3. Isso não ocorre no método de Cauchy com a escolha do tamanho de passo dada na Proposição 2.2. Assim, vemos que a não monotonicidade dos valores funcionais, que é uma característica local do método, i.e., de iteração para iteração, pode gerar um resultado global positivo, que é a convergência mais rápida.

Como ilustração disso, podemos observar a Figura 4. Nela vemos a trajetória traçada pelos pontos obtidos com os métodos de Cauchy e *heavy-ball* utilizando-se os tamanhos de passos das Proposições 2.2 e 2.3, respectivamente, no processo de minimização de $f(x_1, x_2) = x_1^2 + 100x_2^2$ a partir do ponto $x^0 = (5, 5)$.

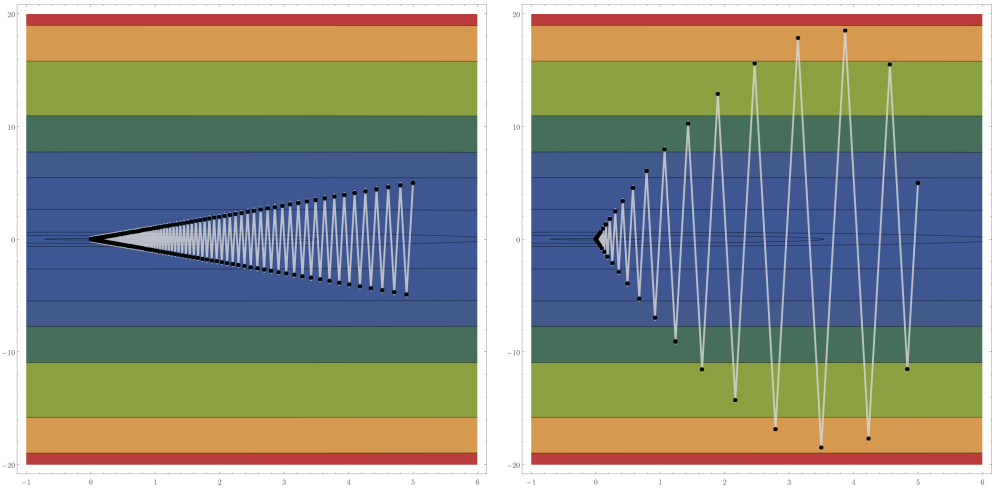


Figura 4: Trajetória traçada pelos pontos obtidos com os métodos de Cauchy (esquerda) e *heavy-ball* (direita) no processo de minimização de $f(x_1, x_2) = x_1^2 + 100x_2^2$

Veja que na trajetória obtida com o método de Cauchy os valores funcionais são sempre decrescentes. Por outro lado, os valores funcionais dos iterados obtidos com o método *heavy-ball* no começo da trajetória crescem e, depois de um determinado momento, passam a decrescer.

Além disso, vemos que a trajetória obtida com o método de Cauchy apresenta o comportamento típico de zigue-zague desse método. No caso de funções quadráticas no $\mathbb{R}^2$, os iterados oscilam entre um par de retas concorrentes que se interceptam no minimizador $x^*$ (veja, por exemplo, [1, pp.389–392]), que é um comportamento que podemos ver na

Figura 4. A trajetória obtida com o método *heavy-ball*, por sua vez, possui o zigue-zague atenuado, visto que a direção de inércia sempre deflete a direção de menos o gradiente.

Essas diferenças impactam na quantidade de iterações necessárias para que façamos a interrupção por convergência (determinada quando $\|\nabla f(x^k)\|_2 < 1e{-}6$). O método de Cauchy precisou de 1002 iterações, enquanto que o método *heavy-ball* precisou de 128. Assim, vemos a grande aceleração obtida utilizando-se o método *heavy-ball*, o que era esperado conforme as taxas de convergência das Proposições 2.2 e 2.3 e o que foi observado na Figura 4.

## 2.3 Uma proposta adaptativa para o método *Heavy-ball*

Sobre os passos $\alpha_k$ e $\beta_k$ do método *heavy-ball*, vemos na Proposição 2.3 que eles dependem da existência e conhecimento de constantes estruturais da função objetivo. Para que seja possível aplicar esse método nos contextos em que as constantes não são conhecidas ou não existam, propomos nesse trabalho a determinação de $\alpha_k$ e $\beta_k$ de maneira adaptativa.

A ideia do processo adaptativo proposto é desacoplar a análise de $\alpha_k$ e $\beta_k$, de forma a determinar um de cada vez. Para isso, nos baseamos em estabelecer decréscimo suficiente em cada uma das direções do método. Veja que a direção de inércia pode não ser de descida em alguns momentos, assim precisamos lidar com esse caso.

Outro ponto chave para o processo apresentado é se valer dos passos dados na iteração passada para a determinação dos passos na iteração corrente. Para isso, usamos a mesma estratégia de contração e dilatação apresentada por Nesterov em [9] para o processo de determinação adaptativa de $L$. A ideia é, então, inicializar os passos na iteração corrente como dilatações dos passos usados na iteração passada e realizar *backtracking* contraindo os valores de passo para que haja decréscimo suficiente, primeiro na direção de menos o gradiente, e depois na direção de inércia.

As dilatações realizadas produzem não monotonicidade dos tamanhos de passo, permitindo que em regiões favoráveis os passos possam ser grandes, mesmo depois de termos passado por regiões em que os passos precisaram ser pequenos.

O processo completo pode ser visto no Algoritmo 4, mas aqui vamos detalhar algumas etapas. Abaixo vemos o bloco para determinação de $\alpha_k$. Veja que selecionamos

$\alpha_k$, então, conforme a regra de Armijo (2.7).

---

**enquanto** $f(x^k - \alpha_k \nabla f(x^k)) > f(x^k) - \sigma \alpha_k \|\nabla f(x^k)\|_2^2$ **faça**
| $\alpha_k \leftarrow \alpha_{\text{contr}} \alpha_k$
**fim**
$\alpha_{k+1} \leftarrow \alpha_{\text{dil}} \alpha_k$

---

Utilizar uma dilatação de $\alpha_{k-1}$ como inicialização para o *backtracking* é uma forma de trazermos a memória do passo selecionado anteriormente, o que reduz a quantidade de avaliações da função objetivo e preserva a ideia do método da bola pesada de usar informação da iteração passada na iteração corrente.

Já o tamanho de passo na direção de inércia ($\beta_k$), como vemos abaixo, possui a sua seleção dividida em dois casos, a depender se essa direção é, ou não, de descida.

---

**se** $\nabla f(x^k)^T(x^k - x^{k-1}) \geq 0$ **então**
| $x^{k+1} \leftarrow x^k - \alpha_k \nabla f(x^k) + \delta \beta_k (x^k - x^{k-1})$
| $\beta_{k+1} \leftarrow \beta_k$
**senão**
| **enquanto** $f(x^k + \beta_k (x^k - x^{k-1})) > f(x^k) + \sigma \beta_k \nabla f(x^k)^T(x^k - x^{k-1})$ **faça**
| | $\beta_k \leftarrow \beta_{\text{contr}} \beta_k$
| **fim**
| $x^{k+1} \leftarrow x^k - \alpha_k \nabla f(x^k) + \beta_k (x^k - x^{k-1})$
| $\beta_{k+1} \leftarrow \beta_{\text{dil}} \beta_k$
**fim**

---

Quando a direção de inércia não é de descida, não é possível pedir decréscimo suficiente. Assim, não é realizada busca linear e sim um "amortecimento" do passo $\beta_k$ com um hiperparâmetro $0 \leq \delta \ll 1$. Veja que quando $\delta = 0$, descartamos a direção de inércia da iteração. Note também que nesse caso não há contração nem dilatação de $\beta_k$, de forma que a próxima iteração, inicializará $\beta_{k+1}$ como $\beta_{k-1}$.

Entretanto, se a direção de inércia for de descida, então aplicamos a seguinte regra de Armijo para determinar $\beta_k$:

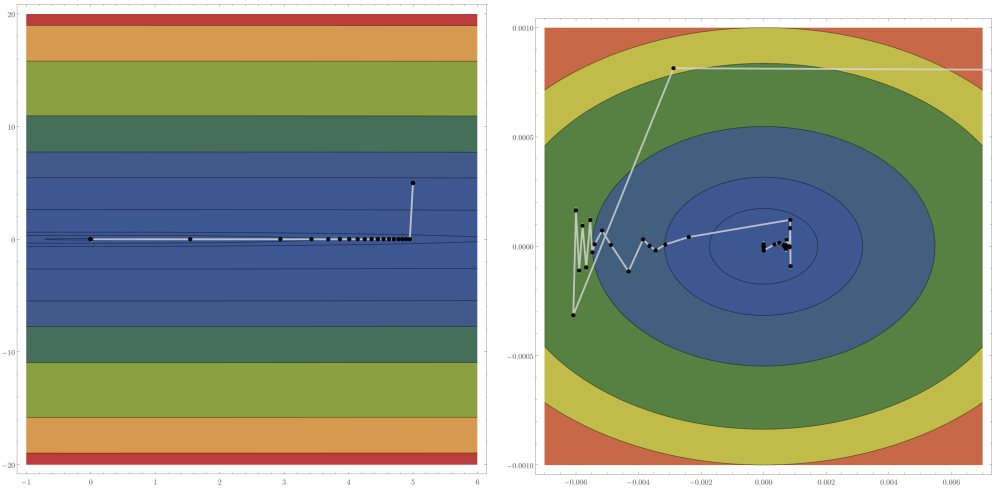
$$f(x^k + \beta_k(x^k - x^{k-1})) \leq f(x^k) + \sigma\beta_k\nabla f(x^k)^T(x^k - x^{k-1}). \tag{2.9}$$

Nesse caso, o processo é análogo ao realizado para determinar $\alpha_k$.

A estratégia adotada busca tirar proveito da direção de inércia sempre que essa

for de descida, e permite que os tamanhos de passo sejam maiores do que os passos utilizados na Proposição 2.3 por conta dos termos de dilatação, com a garantia de decréscimo suficiente. Na situação da direção de inércia não ser de descida, ainda é possível considerar essa direção, mesmo que com um amortecimento, para evitar um passo puramente do método de Cauchy.

Para ilustrar o comportamento da proposta apresentada, podemos acompanhar a Figura 5. Veja que a função sendo minimizada e o ponto inicial são os mesmo utilizados na Figura 4. Na visão global da Figura 5 o comportamento de zigue-zague é imperceptível, mostrando grande vantagem em relação ao método *heavy-ball* com passos dados pela Proposição 2.3.



(a) Visão global (b) Zoom na região próxima de $x^*$

Figura 5: Trajetória traçada pelos pontos obtidos com a proposta adaptativa para o método *heavy-ball* no processo de minimização de $f(x_1, x_2) = x_1^2 + 100x_2^2$

Note que a trajetória rapidamente se alinha com o eixo do autovetor associado ao maior autovalor de $\nabla^2 f(x)$, que no caso é constante. Feito o alinhamento, vemos que rapidamente são realizados passos longos, que são possíveis graças à dilatação do tamanho de passo de inércia.

Com o zoom feito próximo de $x^*$, vemos que a trajetória passa para pontos com $x_1 < 0$, o que não ocorre nas trajetórias da Figura 4. Podemos interpretar isso como os iterados estarem se deslocando muito rapidamente para $x^*$, de forma que ultrapassam a reta $x_1 = 0$, fazem algo como uma frenagem e realinhamento, e depois continuam de forma parecida. Veja que a frenagem e realinhamento ocorreram duas vezes na imagem,

e caso seja feito um zoom maior, poderemos ver isso acontecendo mais vezes antes de interrompermos o processo por convergência. Note que o zigue-zague do método ocorre especialmente no momento de realinhamento, e de forma breve e atenuada.

Como consequência do exposto, o processo de minimização com a proposta adaptativa para o método *heavy-ball* é interrompido após 65 iterações (utilizando-se o mesmo critério daquele da Figura 4). Isso corresponde a cerca da metade das iterações com o método *heavy-ball* e 6% das iterações com o método de Cauchy nas versões usadas para a Figura 4. Experimentos computacionais mais detalhados e gerais serão feitos na Seção 3.

Os hiperparâmetros da proposta adaptativa usados na geração da Figura 5 foram $\alpha_0 = \beta_0 = 0.01$, $\alpha_{\text{contr}} = 0.5$, $\alpha_{\text{dil}} = 1.1$, $\beta_{\text{contr}} = 0.2$, $\beta_{\text{dil}} = 2$ e $\delta = 1\text{e}{-3}$, que são os mesmos usados na Seção 3. Esses valores podem ser calibrados para tipos de problemas específicos. Nesse trabalho não nos concentramos em determinar esses hiperparâmetros, sendo que os utilizados nos experimentos computacionais foram fruto de uma pequena exploração com alguns problemas dos tipos que resolvemos.

## 2.4 Método de Nesterov de 1983

Este é um método da família de métodos ótimos, no sentido em que alcança a melhor taxa de convergência possível para métodos de primeira ordem (veja, por exemplo, [8, Seção 2.2]). O método é apresentado no contexto de funções convexas diferenciáveis com gradiente $L$-Lipchitz contínuo (o que é equivalente a funções convexas e $L$-fortemente suaves). A informação de convexidade forte da função objetivo, quando existir e estiver disponível, é usada pelo método.

De maneira geral, o método precisa de $L > 0$, constante de Lipchitz do gradiente (ou constante de suavidade forte de $f$), e de $\mu \geq 0$, constante de convexidade forte da função objetivo, que no caso de $\mu = 0$ indica apenas a convexidade da função. Definindo

$q \coloneqq \frac{\mu}{L}$ e dados $x^0 \in \mathbb{R}^n$, $y^0 = x^0$ e $\alpha_0 \in (0,1)$, o processo iterativo é definido por:

$$\begin{cases} x^{k+1} = y^k - \dfrac{1}{L}\nabla f(y^k), \\ \alpha_{k+1} = \dfrac{q - \alpha_k^2 + \sqrt{(q-\alpha_k^2)^2 + 4\alpha_k^2}}{2}, \\ y^{k+1} = x^{k+1} + \dfrac{\alpha_k(1-\alpha_k)}{\alpha_k + \alpha_{k+1}^2}(x^{k+1} - x^k). \end{cases} \tag{2.10}$$

Esse é o esquema iterativo simplificado para o método, em que o passo de gradiente para obtenção de $x^{k+1}$ possui tamanho fixo. Dessa maneira é possível eliminar uma sequência numérica e uma sequência auxiliar no $\mathbb{R}^n$. Em [8, Subseção 2.2.1], é possível acompanhar a versão mais geral do método, assim como sua dedução. A seguinte proposição apresenta a taxa de convergência da sequência $\{x^k\}_{k=0}^{\infty}$ obtida via (2.10).

**Proposição 2.4** (Teorema 2.2.3 de [8] adaptado)**.** *Seja $f : \mathbb{R}^n \to \mathbb{R}$ tal que $f \in \mathcal{C}^1$ é $\mu$-fortemente convexa e seu gradiente é $L$-Lipchitz contínuo, tal que $L \geq \mu > 0$. Então, $f$ possui minimizador global $x^* \in \mathbb{R}^n$ e, dado $x^0 \in \mathbb{R}^n$ qualquer, o esquema iterativo (2.10) com $\alpha_0$ dado pela maior raiz de $\alpha_0^2 + \left(1 - \frac{1}{\kappa}\right)\alpha_0 - 1 = 0$ gera sequência $\{x^k\}_{k=0}^{\infty}$ tal que*

$$\|x^k - x^*\|_2 \leq \sqrt{2\kappa \min\left\{\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}}\right)^k, \frac{4}{(k+2)^2}\right\}}\|x^0 - x^*\|_2.$$

Veja que na Proposição 2.4 assume-se $f$ $\mu$-fortemente convexa com $\mu > 0$. O caso $\mu = 0$, que indicaria apenas a convexidade de $f$, não é considerado, pois nele o minimizador pode não ser único. Entretanto, conforme o Teorema 2.2.3 de [8], quando $\mu = 0$ também há convergência para algum minimizador.

É interessante notar a semelhança do método de Nesterov de 1983 com o método da bola pesada: podemos reescrever a atualização de $x^{k+1}$ em (2.10) como

$$x^{k+1} = x^k + \frac{\alpha_{k-1}(1-\alpha_{k-1})}{\alpha_{k-1} + \alpha_k^2}(x^k - x^{k-1}) - \frac{1}{L}\nabla f(y^k). \tag{2.11}$$

Com isso, comparando (2.11) com o esquema (2.8), vemos que o gradiente é calculado no

ponto $y^k$, que é o ponto obtido após o movimento a partir de $x^k$ na direção de inércia. Além disso, o tamanho de passo na direção de menos o gradiente é fixo e depende de $L$ e o tamanho de passo na direção de inércia depende de uma sequência numérica definida recursivamente em (2.10), que tem os seus valores associados a $\mu$ e $L$.

Com essas características e uma grande exploração das constantes estruturais do problema, o método de Nesterov de 1983 apresenta uma das melhores taxas de convergência garantidas.

Por outro lado, as constantes estruturais $\mu$ e $L$ nem sempre são conhecidas. Além disso, o método como exposto usa as informações estruturais globais, mas para algumas funções pode ser interessante analisar essas informações localmente via estratégias adaptativas.

A intuição e o entendimento de que as informações locais do problema poderiam ajudar a acelerar os processos de minimização foram fruto do estudo de algumas implementações do pacote `TFOCS` (*Templates for First-Order Conic Solvers*) [2]. Nele, a menos que seja dito explicitamente para não se fazer buscas lineares (Subseção 2.4.3 da documentação do pacote `TFOCS`[†]), o pacote tentará determinar uma constante de Lipschitz local $L_k$ para o gradiente, conforme [2, Seção 5.3], de forma a tentar se adaptar a mudanças na curvatura local.

Para lidar com essas dificuldades, algumas propostas foram feitas por diversos autores. Em [9] é introduzido um esquema adaptativo para o ajuste da constante de Lipschtiz do gradiente a cada iteração. Já em [4], um método com uma diferente escolha para $\alpha_k$, que independe de $L$, e uma estratégia adaptativa para seleção de $\mu$ a cada iteração são introduzidos. Por fim, em [14], uma estratégia de reinício do método de Nesterov de 1983 é apresentada para lidar com o não conhecimento de $\mu$ e $L$.

Neste trabalho foi realizada a implementação do método de Gonzaga e Karas [4] de forma próxima ao que é indicado pelos autores. Também foi realizada uma implementação do método de Nesterov de 2007 [9], em que aplicamos a estratégia adaptativa de seleção de $L$ ao método de Nesterov de 1983 e usamos o reinício adaptativo baseado em gradiente apresentado em [14] para lidar com o desconhecimento de $\mu$.

[†]Documentação disponível em `http://cvxr.com/tfocs/doc/`

# 3 Experimentos Computacionais

No presente trabalho, todos os métodos estudados foram implementados em linguagem `MATLAB`, versão *R2020b Update 3 (9.9.0.1538559)*, em uma máquina com processador *Intel(R) Core(TM) i3-6006U CPU 2.00GHz* e 4Gb de RAM (64-bit). Os pseudocódigos dos respectivos métodos estão apresentados no Apêndice A. Para análise dos resultados foi utilizada a técnica de *performance profile* proposta em [3].

Para os experimentos computacionais foram utilizadas funções objetivo quadráticas simples e a *pior função do mundo de Nesterov* [8, p. 67]. Em ambas, os valores das respectivas constantes de Lipschitz para o gradiente (suavidade forte) e de convexidade forte foram escolhidos como $L = 1000$ e $\mu = 1$.

Adicionalmente, também consideramos a função de Rosenbrock para testar a proposta adaptativa do método *heavy-ball* apresentada no caso em que temos uma função que não é fortemente convexa e também não é fortemente suave.

Quando foi realizada algum tipo de busca de Armijo ((2.7) ou (2.9)), consideramos $\sigma = 1\text{e}{-4}$.

## 3.1 Funções quadráticas

Foram consideradas funções quadráticas $f : \mathbb{R}^n \to \mathbb{R}$ da forma

$$f(x) = \frac{1}{2} \sum_{i=1}^{n} d_i x_i^2, \tag{3.1}$$

em que $d \in \mathbb{R}^n$, $d_1 = \mu = 1$, $d_n = L = 1000$ e $1 < d_i < 1000$, $\forall i \in \{2, \ldots, n-1\}$.

As funções quadráticas são funções de teste importantes no contexto de métodos aplicados a funções fortemente convexas e fortemente suaves. Isso porque, conforme discutido anteriormente e ilustrado na Figura 2, essas funções são limitadas, em qualquer ponto, por cima e por baixo por quadráticas simples. Ao redor de um ponto qualquer, portanto, o comportamento da função se aproximará ao de uma quadrática. Além disso, veja que se a função for suficientemente suave, a expansão de Taylor de segunda ordem será uma aproximação quadrática local para a função objetivo.

Ainda, veja que as quadráticas (3.1) consideradas são quadráticas simples (com

hessiana diagonal). Em tal característica não há perda de generalidade, visto que quadráticas com matriz hessiana positiva definida não diagonal apenas diferem das consideradas nesse trabalho por rotações de eixos. Assim, fixamos os eixos preferenciais como aquele alinhados com a base canônica do $\mathbb{R}^n$ e nos preocupamos apenas com os autovalores $d_i$ da matriz hessiana.

Além do destacado, uma motivação para o uso dessas funções é que elas foram usadas por Gonzaga e Karas [4] nos experimentos computacionais. Em [4] os autores geraram aleatoriamente os autovalores $d_i$ no intervalo $[1, 1000]$. Neste trabalho, entretanto, foi utilizada uma sistemática diferente na gerações dos autovalores, a fim de se obter maior variabilidade dos problemas teste.

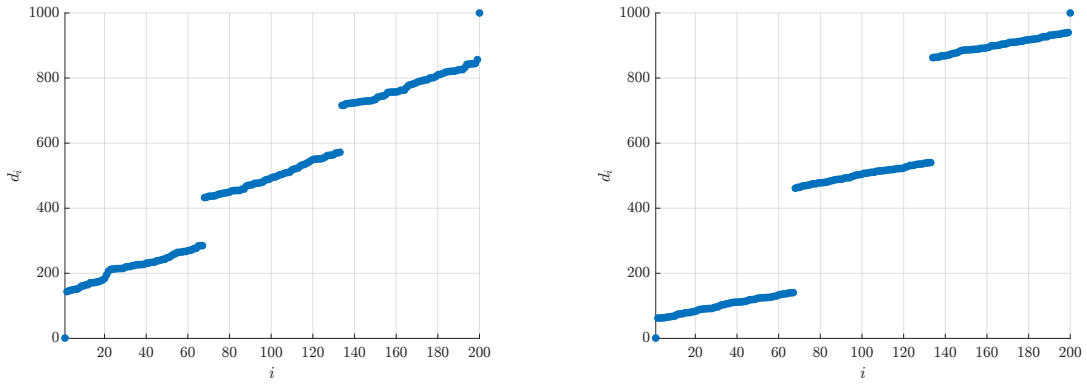
A escolha dos valores $d_i$, $i \in \{2, \ldots, n-1\}$, foi feita por distribuições uniformes em 1 até 5 *clusters*, que por sua vez também são uniformemente distribuídos no intervalo $[\mu, L]$. Além disso, foram considerados os casos em que $d_1$ e/ou $d_n$ estão isolados.

Além dessas características para construção de $d$, também foi considerado um fator de condensação $t$ dos valores em cada *cluster* ao redor de seu ponto central. Se $t = 0$ temos o *cluster* original, e se $t = 1$ todos os valores do *cluster* são iguais ao seu valor central.

Na Figura 6 temos quatro exemplos da distribuição dos valores de $d$ com $n = 200$, 3 *clusters*, com $d_1$ e/ou $d_n$ isolados ou não e com $t = 0$ ou $t \neq 0$.

Usando-se a sistemática apresentada para geração dos valores de $d$, i.e. dos autovalores de matriz Hessiana $\nabla^2 f(x)$, podemos variar diversos parâmetros para gerar uma grande gama de problemas teste. Para isso, consideramos a dimensão $n \in \{10, 200, 500, 1000, 2000\}$, uma quantidade de *clusters* variando de 1 até 5, o isolamento ou não de $d_1$ e $d_n$ e o fator $t \in \{0, 0.3, 0.6, 0.9\}$. Com isso, temos $5 \cdot 5 \cdot 2 \cdot 2 \cdot 4 = 400$ possibilidades para geração de problemas. Ademais, para cada problema gerado, três pontos iniciais $x^0$ foram aleatoriamente obtidos utilizando-se a distribuição uniforme no intervalo $[-1, 1]$ de forma que $\|x^0\|_\infty = 1$. Portanto, foram realizadas 1200 rodadas de testes.

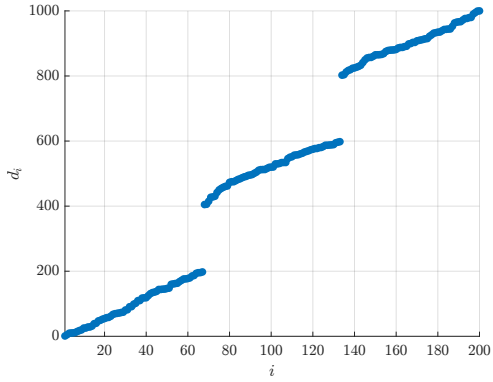
Visto que algumas implementações possuem opções de entrada, como informar ou não os valores de $L$ e $\mu$ e pedir que haja uma determinada busca linear ou não, foram realizados testes comparando essas opções antes que fosse feita uma comparação entre os métodos. Por exemplo, o método *Heavy-ball*, apresentado na Subseção A.2, possui duas



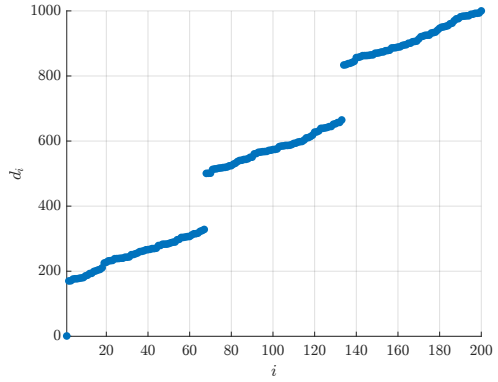

(a) $t = 0$ com $d_1$ e $d_n$ isolados




(b) $t = 0.6$ sem $d_1$ e $d_n$ isolados




(c) $t = 0$ sem $d_1$ e $d_n$ isolados




(d) $t = 0$ com apenas $d_1$ isolado

Figura 6: Exemplos de distribuição dos valores de $d$ para $n = 200$ com 3 *clusters*, incluindo casos com $d_1$ e/ou $d_n$ isolados ou não e com $t = 0$ e $t \neq 0$

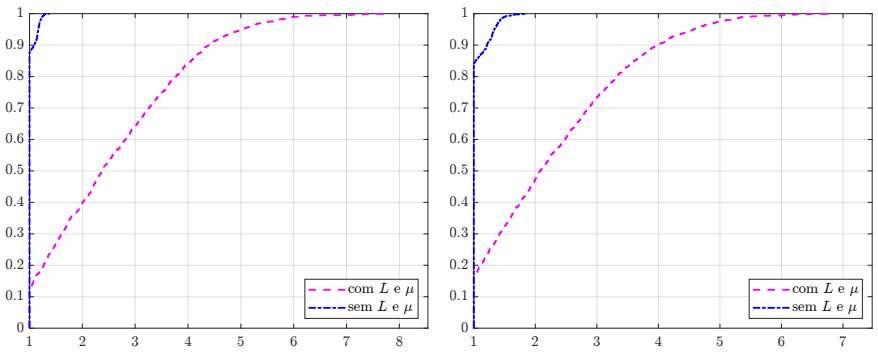
opções (os Algoritmos 3 e 4), então comparamos os resultados entre essas duas variantes. Aquelas implementações com melhores resultados foram usadas para comparar os métodos entre si.

Como o método do gradiente descendente, apresentado na Subseção A.1, precisa de uma grande quantidade de iterações para que tenhamos convergência, esse método não foi considerado nas comparações.

A convergência foi estabelecida quando obtivemos $x^k$ tal que $\|\nabla f(x^k)\|_2 \leq$ 1e$-$6. Além desse critério de parada, também foi definido como orçamento a quantidade máxima de 2000 iterações.

Na Figura 7 acompanhamos os perfis de desempenho tanto de tempo quanto de iterações obtidos aplicando o método *Heavy-ball* aos problemas com funções quadráticas geradas. Na figura, a legenda '*com* $L$ *e* $\mu$' indica que foram informados esses valores, e portanto o algoritmo calculou os valores para $\alpha_k$ e $\beta_k$ conforme a Proposição 2.3. A

legenda '*sem* $L$ *e* $\mu$', por sua vez, indica que esses valores não são informados, e então o Algoritmo 4 é aplicado e os hiperparâmetros usados são $\alpha_0 = \beta_0 = 0.01$, $\alpha_{\text{contr}} = 0.5$, $\alpha_{\text{dil}} = 1.1$, $\beta_{\text{contr}} = 0.2$, $\beta_{\text{dil}} = 2$ e $\delta = 1\text{e}{-3}$.



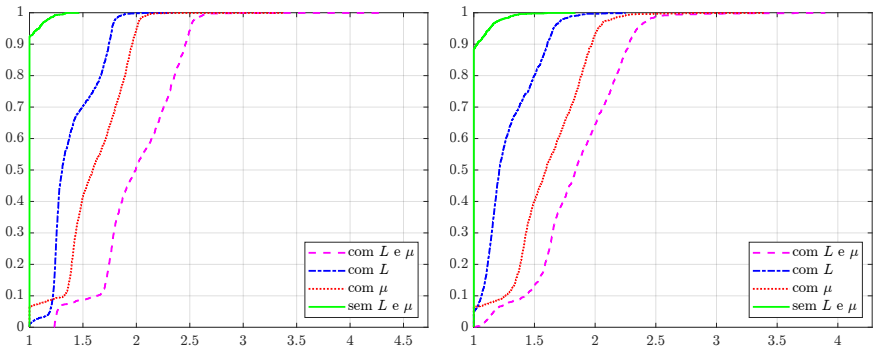

(a) perfil baseado em iterações (b) perfil baseado em tempo

Figura 7: Perfis baseados em iterações e tempo para problemas com funções quadráticas utilizando-se o método *Heavy-ball*

Veja que tanto em relação ao número de iterações, quanto em relação ao tempo de processamento, a nossa proposta adaptativa para o método *heavy-ball* (que denotaremos por `HB adapt`) é o mais rápido entre 80% e 90% dos problemas, enquanto que o mesmo método com os valores de $\alpha_k$ e $\beta_k$ escolhidos conforme a Proposição 2.3 é o mais rápido entre 10% e 20%. Veja também que ambas as opções são robustas no conjunto de problemas, mas a opção com buscas lineares se destaca, visto que resolve 100% dos problemas em menor quantidade de tempo e iterações do que a outra opção.

Na Figura 8 acompanhamos os mesmos perfis, mas agora para o método de Nesterov de 2007 (que denotaremos por `N07`), descrito na Subseção A.4. Note que a opção '*com* $L$ *e* $\mu$' equivale ao método de Nesterov de 1983. Fizemos essa opção para avaliar o custo do processo adaptativo para se estimar $L$. No conjunto de problemas tratado, como a avaliação de $\nabla f(\cdot)$ não é custosa ($\mathcal{O}(n)$) e a implementação da função $\nabla f(\cdot)$ é feita de forma vetorizada, essa busca não tem custo elevado. De qualquer forma, na média, para os problemas tratados são feitas 2.7 avaliações de gradiente extra por iteração. Assim, também não temos uma grande quantidade extra de avaliações. Os hiperparâmetros usados foram: $L_{\text{dil}} = 1.5$ e $L_{\text{contr}} = 0.5$ .

Além desse ponto, é possível destacar que quando existe pelo menos um processo adaptativo, seja ele de busca para determinação de $L$ ou de reinício para tratar o




(a) perfil baseado em iterações (b) perfil baseado em tempo

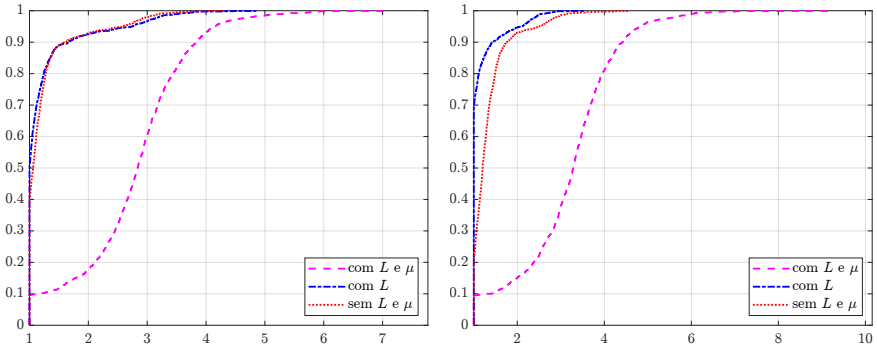
Figura 8: Perfis baseados em iterações e tempo para problemas com funções quadráticas utilizando-se o método de Nesterov de 2007

caso de $\mu$ indisponível, existe uma melhora em relação ao método de Nesterov de 1983. Finalmente, quando os dois processos adaptativos são realizados paralelamente, obtemos o melhor desempenho, com mais de 90% dos problemas sendo resolvidos mais rapidamente em termos de iteração e pouco menos de 90% em termos de tempo. De qualquer forma, observamos que todas as opções são robustas no conjunto de problemas.

Na Figura 9 acompanhamos os perfis para as opções do método de Gonzaga e Karas (que denotaremos por `GK13`), descritas na Subseção A.5, em que foi considerado, conforme sugerem os autores do método, $\beta = 1.02$. Além desse hiperparâmetro, nas chamadas para o Algoritmo 2, consideramos $\alpha_0 = 0.1$, $\alpha_{\text{contr}} = 0.5$ e $\alpha_{\text{dil}} = 1.1$. Em termos de iterações, as opções '*com* $L$' e '*sem* $L$ *e* $\mu$' apresentam grande similaridade de desempenho. Entretanto, quanto ao tempo de execução, a opção '*com* $L$' se destaca, com cerca de 70% dos problemas resolvidos mais rapidamente. De fato, a informação de $L$ implica que não é feita busca para a realização do passo de gradiente no método. Além disso, as buscas feitas para obtenção de $\theta_k$ (Algoritmo 9) são distintas. Ainda, note que as opções são robustas no conjunto de problemas.

Por fim, na Figura 10 vemos os perfis comparando os resultados das melhores opções dos métodos testados, juntamente com o método de Nesterov de 1983 (que denotaremos por `N83`) (Algoritmo 5). Assim como observado em [4] em relação a iterações, `GK13` se destaca em relação aos métodos `N07` e `N83`. O método `HB adapt`, que não foi considerado em [4], apresenta desempenho competitivo com o método `GK13`.

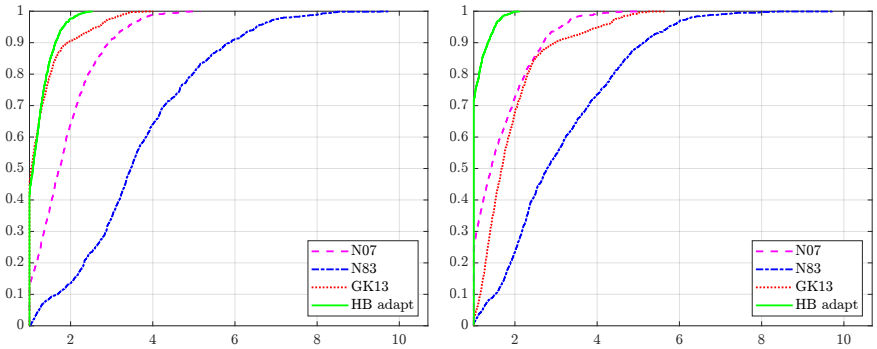
Já em relação ao tempo de execução, veja que a grande quantidade de buscas




(a) perfil baseado em iterações (b) perfil baseado em tempo

Figura 9: Perfis baseados em iterações e tempo para problemas com funções quadráticas utilizando-se o método de Gonzaga e Karas de 2013

existentes no método `GK13` resulta em um processamento mais lento, o que faz com que o método `N07` tenha melhor desempenho. Ademais, o método `HB adapt` supera o método `N07` e resolve pouco mais de 70% dos problemas mais rapidamente.

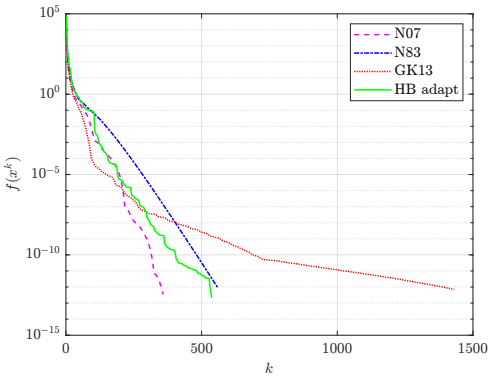



(a) perfil baseado em iterações (b) perfil baseado em tempo

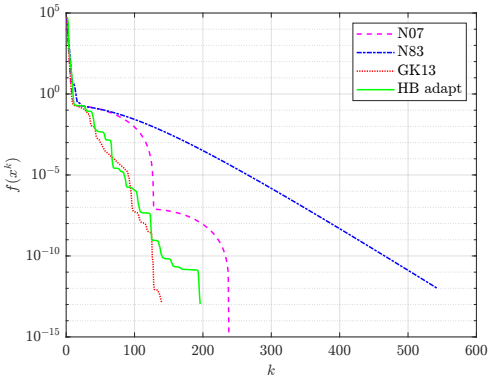
Figura 10: Perfis baseados em iterações e tempo para problemas com funções quadráticas utilizando-se as melhores opções dos métodos *Heavy-ball*, de Nesterov 2007, de Nesterov 1983 e de Gonzaga e Karas de 2013

Na Figura 11 vemos exemplos da evolução dos valores funcionais em uma rodada de testes utilizando os métodos estudados com as opções selecionadas como aquelas que obtiveram os melhores desempenhos.

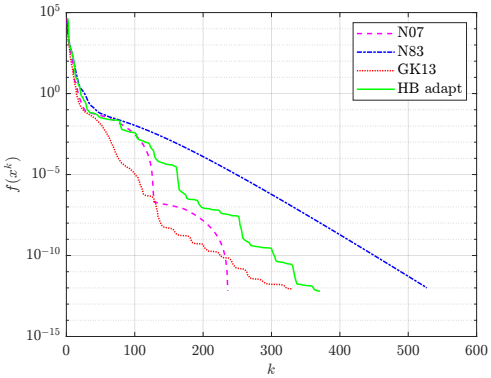
Observe que o método `GK13` apresenta uma grande quantidade de iterações no caso (a) da Figura 11, o que se inverte no caso (b), sendo o método mais rápido. Entretanto, a única diferença entre os casos (a) e (b) é que consideramos $t = 0.3$ em (b) e $t = 0$ em (a), i.e. em (b) os autovalores da quadrática estão mais concentrados, o que foi capaz de influenciar uma mudança significativa na evolução dos iterados do método



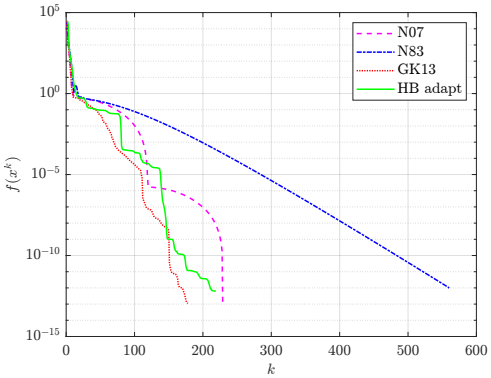

(a) 1 *cluster*, $t = 0$, $d_1$ e $d_n$ não isolados




(b) 1 *cluster*, $t = 0.3$, $d_1$ e $d_n$ não isolados




(c) 3 *clusters*, $t = 0.3$, $d_1$ não isolado e $d_n$ isolado




(d) 5 *clusters*, $t = 0.6$, $d_1$ e $d_n$ isolados

Figura 11: Exemplos da evolução dos valores funcionais ao decorrer das iterações realizadas com os métodos estudados e as opções selecionadas como na Figura 10 (as características do vetor $d$ são dadas nas figuras)

`GK13`, e, em menores proporções, também nos iterados do método `HB adapt`.

Além disso, nota-se nas evoluções obtidas com a implementação do método `N07` (Algoritmo 7) os momentos em que existe um reinício dados pelos "bicos" no gráfico. É possível notar para os exemplos apresentados, e na média dos problemas gerados, que são realizados 2 ou 3 reinícios em cada rodada. Esse é um número relativamente pequeno em comparação com a quantidade de iterações, mas os resultados são consideravelmente produtivos: maior velocidade de convergência sem a exigência do conhecimento de $\mu$.

As buscas e processos adaptativos existentes tanto no método `GK13` quanto no método `HB adapt` se expressam na evolução ruidosa dos valores funcionais ao decorrer das iterações. Por fim, veja que, em todos os casos, o método de Nesterov de 1983 converge em cerca de 500 iterações, evidenciando sua convergência linear a partir da iteração 200.

## 3.2 A pior função do mundo de Nesterov

Considerando $L > \mu > 0$, define-se a *pior função do mundo* de Nesterov (veja, por exemplo, [8, p. 67]), $f : \mathbb{R}^n \to \mathbb{R}$, como

$$f(x) = \mu\left(\frac{\kappa - 1}{8}\right)\left(x_1^2 + \sum_{i=1}^{n-1}(x_i - x_{i+1})^2 - 2x_1\right) + \frac{\mu}{2}\|x\|_2^2, \tag{3.2}$$

em que $\kappa = \frac{L}{\mu}$ é o número de condição de $f$. Note que (3.2) é uma quadrática, que possui termos mistos, com hessiana tridiagonal.

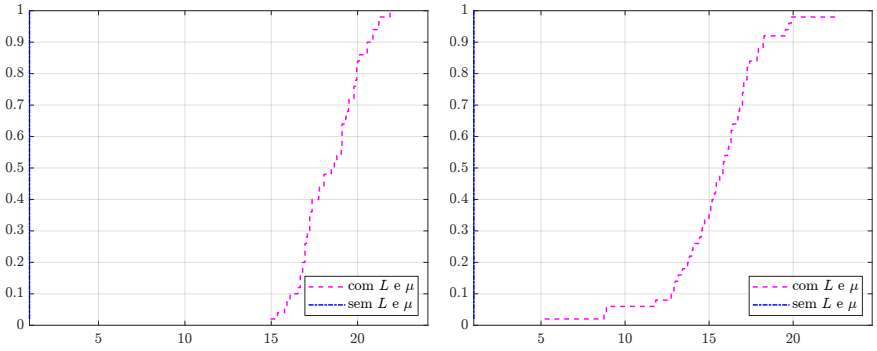
Em [8, p. 67], o autor constrói (3.2) em dimensão infinita a fim de apresentar um exemplo de função fortemente convexa e fortemente suave que satisfará uma cota inferior de complexidade de qualquer método de primeira ordem. Assim, (3.2) é projetada para apresentar dificuldade para esse tipo de método.

Os experimentos computacionais usando a função (3.2) consideraram $L = 1000$ e $\mu = 1$. Além disso, as dimensões consideradas foram, assim como nos testes com a função objetivo (3.1), $n \in \{10, 200, 500, 1000, 2000\}$. Os critérios de parada foram também os mesmos.

Novamente, os pontos iniciais foram gerados com as entradas distribuídas uniformemente no intervalo $[-1, 1]$ e foram tais que $\|x^0\|_\infty = 1$. Para cada dimensão foram gerados 10 pontos iniciais distintos. Logo, foram realizadas $5 \cdot 10 = 50$ rodadas para cada método. Repetindo o procedimentos aplicados para as funções da forma (3.1), analisamos inicialmente os diversos métodos com suas diferentes opções e depois selecionamos as de melhor desempenho entre tais opções para comparar os métodos entre si.

Na Figura 12 acompanhamos a comparação do desempenho das mesmas duas opções do método *Heavy-ball* utilizadas para as funções do tipo (3.1). Note a vantagem da versão com buscas adaptativas na soluções dos problemas, tendo em todas as rodadas a menor quantidade de iterações e tempo de processamento. Apesar disso, ambas opções são robustas no conjunto de problemas.

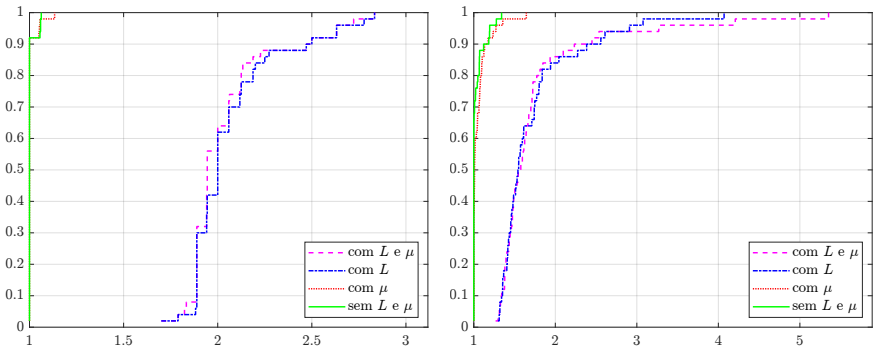
Os perfis gerados para o método de Nesterov de 2007 podem ser acompanhados na Figura 13. Note que, tanto em iterações, quanto em tempo, as opções em que informamos apenas $\mu$ e que não informamos nenhuma das constantes se destacam. No quesito iterações, as duas opções ficam praticamente empatadas, com pequena vantagem




(a) perfil baseado em iterações (b) perfil baseado em tempo

Figura 12: Perfis baseados em iterações e tempo para problemas com função objetivo da forma (3.2) utilizando-se o método *Heavy-ball*.

para a opção '*sem* $L$ *e* $\mu$', visto que consegue completar a solução de todas as rodadas um pouco mais rápido. Já em termos de tempo de processamento, a opção '*sem* $L$ *e* $\mu$' se destaca em relação à opção '*com* $\mu$', resolvendo mais rapidamente pouco menos de 70% dos problemas.




(a) perfil baseado em iterações (b) perfil baseado em tempo

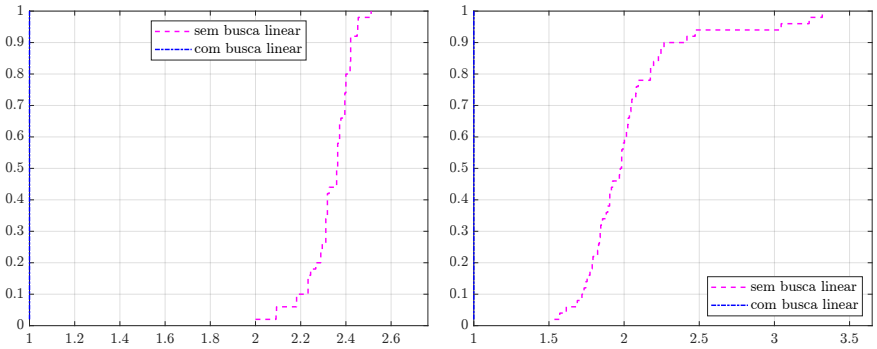
Figura 13: Perfis baseados em iterações e tempo para problemas com função objetivo da forma (3.2) utilizando-se o método de Nesterov de 2007.

Note na Figura 13 que as opções em que $L$ é informado, e portanto não existe busca adaptativa para determinar essa constante do problema, apresentam um desempenho inferior aos demais com a busca adaptativa. De fato, a busca adaptativa permite que $L_k < L$, o que representa localmente um modelo melhor condicionado para os problemas, além de o tamanho de passo de gradiente dado a partir de $y^k$ ser maior, aspectos que aceleram a convergência.

Para os problemas com a função (3.2), comparamos as opções do método de

Nesterov de 1983 apresentadas nos Algoritmos 5 e 6. Na implementação do Algoritmo 5 o passo de gradiente é fixo igual a $\frac{1}{L}$, e no Algoritmo 6 existe uma busca linear adaptativa para determinar o passo em cada iteração, garantindo que $f(y^k - \nu_k \nabla f(y^k)) \leq f(y^k) - \frac{1}{2L}\|\nabla f(y^k)\|_2^2$, que é o decréscimo exigido em [8, Esquema (2.2.6)]. Nos testes foram usados $\nu_0 = \frac{10}{L}$, $\nu_{contr} = 0.5$ e $\nu_{dil} = 1.1$.

Na Figura 14 vemos que a opção com busca linear adaptativa apresenta grande vantagem, resolvendo todos os problemas em menor quantidade de tempo e de iterações. Além disso, vemos que a opção '*sem busca linear*' precisa de pelo menos o dobro de iterações para atingir a convergência. O melhor desempenho com busca linear é fruto de $\nu_k > \frac{1}{L}$ na maioria das iterações, o que produz passos maiores por iteração. As duas opções, entretanto, são robustas no conjunto de testes.

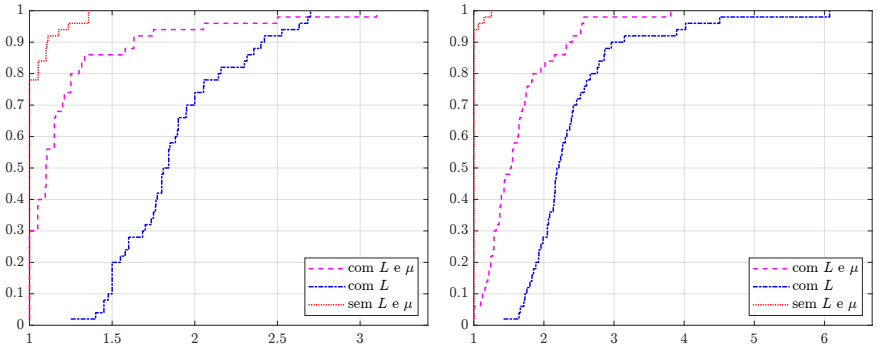



(a) perfil baseado em iterações (b) perfil baseado em tempo

Figura 14: Perfis baseados em iterações e tempo para problemas com função objetivo da forma (3.2) utilizando-se o método de Nesterov de 1983.

Em relação ao método de Gonzaga e Karas, podemos acompanhar na Figura 15 os perfis para as funções objetivo da forma (3.2). Vemos uma grande superioridade da opção '*sem $L$ e $\mu$*', tendo a solução mais rápida em termos de iterações em pouco menos de 80% dos problemas e em termos de tempo de processamento em pouco mais de 90%. Veja que, ao contrário do caso da função objetivo (3.1), nesse caso, fazer buscas adaptativas para a seleção do tamanho do passo de gradiente a partir de $y^k$ representa uma vantagem. De qualquer forma, novamente, todas as opções são robustas no conjunto de problemas testados.

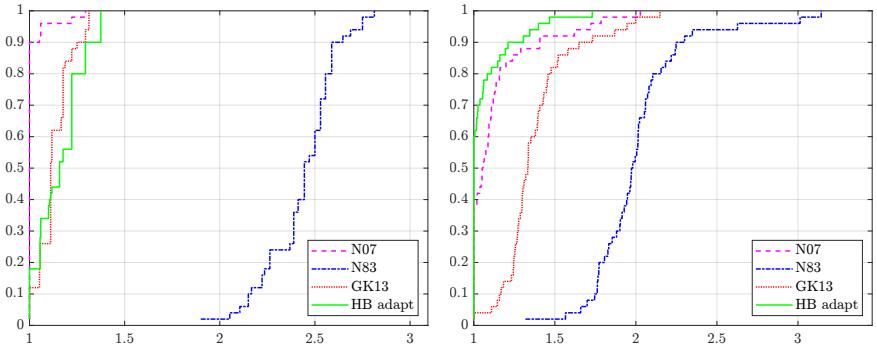
Finalmente, na Figura 16 vemos os perfis obtidos com as melhores opções de cada um dos métodos apresentados. Note que, em termos de iterações, o método `N07`




(a) perfil baseado em iterações (b) perfil baseado em tempo

Figura 15: Perfis baseados em iterações e tempo para problemas com função objetivo da forma (3.2) utilizando-se o método de Gonzaga e Karas.

apresenta grande vantagem, concluindo a solução mais rapidamente em cerca de 90% dos problemas. Além disso, os métodos `GK13` e `HB adapt` apresentam desempenhos similares. O método `N83`, por outro lado, apresenta um desempenho inferior, ficando distante dos demais métodos.




(a) perfil baseado em iterações (b) perfil baseado em tempo

Figura 16: Perfis baseados em iterações e tempo para problemas com função objetivo da forma (3.2) utilizando-se as melhores opções dos métodos *Heavy-ball*, de Nesterov 2007, de Nesterov 1983 e de Gonzaga e Karas de 2013

Em relação ao tempo de processamento, o método `HB adapt` se destaca, superando o método `N07` e resolvendo cerca de 60% dos problemas mais rapidamente. O método `N07` resolve cerca de 40% dos problemas mais rapidamente, sendo então esses dois os métodos mais velozes. O método `GK13`, que em termos de iterações foi competitivo com o método `HB adapt`, apresenta um desempenho inferior em termos de tempo, o que pode ser explicado pelas buscas para determinar $\theta_k$ e $\nu_k$. O método `HB adapt`, que também faz buscas lineares, precisa de cerca de 2.5 avaliações de função por iteração, enquanto `GK13`

precisa de cerca de 5 avaliações por iteração, sendo essa uma explicação para a diferença de desempenho comparativo em termos de iterações e tempo. Por fim, veja que o método `N83` também apresenta um desempenho inferior aos demais no quesito tempo.

A observação que fazemos tanto no contexto das funções objetivo (3.1) quanto (3.2) de que o método `N83` apresenta um desempenho inferior é um indicativo que as estratégias adaptativas são capazes de acelerar os esquemas, quando avaliações de função e gradiente não são caras computacionalmente. Apesar de o resultado apresentado na Figura 16 usar uma estratégia adaptativa para o passo de gradiente $\nu_k$, o método usa as constantes estruturais globais da função ($L$ e $\mu$) ao invés de buscar uma estratégia de tirar proveito de características locais do problema.

Na Figura 17 vemos um exemplo de resultado da evolução dos valores funcionais para um problema com função objetivo (3.2). Nesse exemplo, vemos que os métodos `N07`, `HB adapt` e `GK13` apresentam desempenho semelhante, com `N07` obtendo convergência mais rapidamente. Já o método `N83` se distancia desses métodos, apresentando convergência mais lenta.

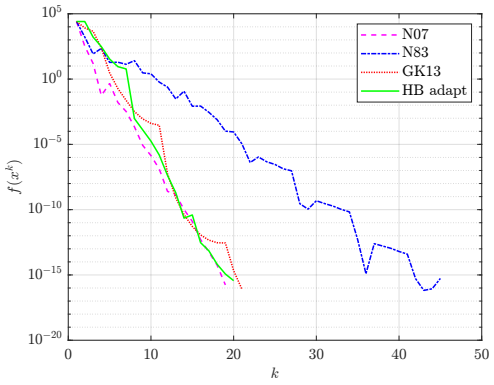



Figura 17: Exemplo da evolução dos valores funcionais ao decorrer das iterações realizadas com os métodos estudados e as opções selecionadas como na Figura 16.

## 3.3 Função de Rosenbrock

A função de Rosenbrock, introduzida em [16], é uma função-teste muito utilizada em problemas de minimização irrestrita. Em duas dimensões, ela pode ser definida como $f : \mathbb{R}^2 \to \mathbb{R}$, tal que

$$f(x_1, x_2) = (a - x_1)^2 + b(x_2 - x_1^2)^2, \tag{3.3}$$

em que $b > 0$ e $a \in \mathbb{R}$.

Usando o ambiente de computação simbólica do `Mathematica` para o auxílio nas manipulações algébricas, podemos analisar propriedades de (3.3). Primeiramente, vejamos que o ponto $(a, a^2)$ é um minimizador global estrito da função de Rosenbrock. Note que

$$\nabla f(x_1, x_2) = \begin{bmatrix} -2(a - x_1) - 4bx_1(x_2 - x_1^2) \\ 2b(x_2 - x_1^2) \end{bmatrix},$$

assim, um ponto estacionário satisfaz o sistema:

$$\begin{cases} -2(a - x_1) - 4bx(x_2 - x_1^2) = 0, & \text{(3.4i)} \\ 2b(x_2 - x_1^2) = 0. & \text{(3.4ii)} \end{cases}$$

Resolvendo-o, temos por (3.4ii), como $b > 0$, que vale $x_2 - x_1^2 = 0$. Substituindo $x_2 = x_1^2$ em (3.4i):

$$\begin{aligned} -2(a - x_1) - 4bx(x_1^2 - x_1^2) = 0 \\ \Rightarrow x_1 = a. \end{aligned}$$

Assim, vale $x_2 = a^2$. Logo o ponto $(a, a^2)$ é o único ponto estacionário para a função (3.3).

Computando $\nabla^2 f(x_1, x_2)$, obtemos

$$\nabla^2 f(x_1, x_2) = \begin{bmatrix} 2 + 4b(3x^2 - x_2) & -4bx \\ -4bx & 2b \end{bmatrix}.$$

Avaliando $\nabla^2 f(x_1, x_2)$ no ponto estacionário $(a, a^2)$, temos

$$\nabla^2 f(a, a^2) = \begin{bmatrix} 2 + 8ba^2 & -4ab \\ -4ab & 2b \end{bmatrix},$$

cujos autovalores são:

$$\lambda_1 = 1 + b + 4a^2b - \sqrt{(1 + b + 4a^2b)^2 - 4b},$$
$$\lambda_2 = 1 + b + 4a^2b + \sqrt{(1 + b + 4a^2b)^2 - 4b}.$$

Veja que $\lambda_1, \lambda_2 \in \mathbb{R}$, pois $\nabla^2 f(a, a^2)$ é simétrica. Assim, como $b > 0$ e $a \in \mathbb{R}$, vale $\lambda_2 > 0$, pois $\lambda_2$ é a soma de parcelas maiores ou iguais a zero com pelo menos uma parcela estritamente maior que zero. Além disso, como $(1 + b + 4a^2b)^2 - 4b \geq 0$, vale que

$$\lambda_1\lambda_2 = (1 + b + 4a^2 + b)^2 - ((1 + b + 4a^2 + b)^2 - 4b) = 4b > 0.$$

Com isso, temos que $\lambda_1 > 0$. Dessa forma, temos que $\nabla^2 f(a, a^2)$ é definida positiva. Portanto, pelas condições suficientes de segunda ordem de otimalidade [13, Teorema 2.4], podemos afirmar que $(a, a^2)$ é um minimizador local estrito de (3.3). Note que $(a, a^2)$ é único ponto estacionário da função (3.3), que é contínua, logo não existem minimizadores ou maximizadores locais, assim como pontos de sela.

Ainda, como $b > 0$, vale $f(x_1, x_2) \geq 0$, $\forall (x_1, x_2) \in \mathbb{R}^2$, pois $(a - x_1)^2 \geq 0$ e $(x_2 - x_1^2)^2 \geq 0$ para $(x_1, x_2) \in \mathbb{R}^2$. Com isso, como $f(a, a^2) = 0$, o ponto $(a, a^2)$ é um minimizador global de $f$. Ademais, veja que $f(x_1, x_2) = 0$ resulta em uma equação quadrática em $x_2$, cuja solução é dada por

$$x_2 = x_1^2 \pm \frac{\sqrt{-b(a - x_1)^2}}{b},$$

visto que $b \neq 0$. Veja que a única solução em que $(x_1, x_2) \in \mathbb{R}^2$ é tal que $(a - x_1)^2 = 0$, o que implica em $x_1 = a$, e assim temos $x_2 = a^2$. Logo, além do ponto $(a, a^2)$ ser minimizador local estrito, é também um minimizador global estrito.

Ainda analisando as propriedades de (3.3), é possível notar que essa função não é convexa. Veja que, calculando o valor do determinante de $\nabla^2 f(x_1, x_2)$, obtemos

$$\det \nabla^2 f(x_1, x_2) = 4b(1 + 2b(x_1^2 - x_2)),$$

de forma que, por exemplo, para $x_1 = 0$ e $x_2 = \frac{1}{b}$ vale que $\det \nabla^2 f(0, \frac{1}{b}) < 0$, e assim os

autovalores de $\nabla^2 f(0, \frac{1}{b})$ deverão ter sinais opostos, implicando que $\nabla^2 f(0, \frac{1}{b})$ é indefinida. Sendo assim, de fato, (3.3) não é convexa.

Além disso, a função (3.3) não possui gradiente Lipschitz contínuo, pois a fração

$$\frac{\|\nabla f(x_1, x_2) - \nabla f(a, a^2)\|_2^2}{\|(x_1, x_2) - (a, a^2)\|_2^2} = 2\frac{b\Big(x_1^2 - x_2\Big)^2 + \Big(a - x_1\big(1 + 2b(x_1^2 - x_2)\big)\Big)^2}{(a - x_1)^2 + (a^2 - x_2)^2},$$

definida em $(x_1, x_2) \in \mathbb{R}^2 \backslash \{(a, a^2)\}$, é ilimitada, já que no numerador temos $x_1$ elevado à sexta potência, enquanto que no denominador temos $x_1$ ao quadrado.

Com isso, a função em discussão não satisfaz algumas das propriedades básicas consideradas nos métodos numéricos apresentados nesse projeto: convexidade e gradiente Lipschitz contínuo. Assim, a aplicação do método de Nesterov, por exemplo, se torna inadequada, bem como a aplicação do método de Gonzaga e Karas e do método *heavy-ball* os tamanhos de passo dados pela Proposição 2.3. Entretanto, os métodos de Cauchy com regra de Armijo e a proposta adaptativa para o método *heavy-ball* ainda são alternativas para solução numérica do problema de minimização irrestrita de (3.3).

Para a experimentação computacional, definiremos $a = 1$ e $b = 100$ em (3.3):

$$f(x_1, x_2) = (1 - x_1)^2 + 100(x_2 - x_1^2)^2. \tag{3.5}$$

Como $a \in \mathbb{R}$ e $b > 0$, todas as propriedades que verificamos para o caso geral valem para (3.5). Em particular, o minimizador global estrito de $f$ é o ponto $(1, 1)$.

Na Figura 18 podemos ver parte do gráfico da função de Rosenbrock (3.5). Por meio da figura é possível perceber a não convexidade dessa função. Veja que no corte $x_2 = 2.5$ da figura temos uma parte da função que é côncava.

Como discutido, aplicamos o método de Cauchy (Algoritmo 2 com $\alpha_0 = 1$, $\alpha_{\text{contr}} = 0.5$ e $\alpha_{\text{dil}} = 1.1$) e a proposta adaptativa para o método *heavy-ball* (Algoritmo 4 com os mesmos hiperparâmetros usados para minimização de (3.1)) para solução do problema de minimização irrestrita de (3.5). O ponto inicial usado é o clássico para o problema: $x^0 = (-1.2, 1)$. A convergência é estabelecida quando $\|\nabla f(x_1^k)\|_2 < 1\text{e}{-6}$ e foi definido o número máximo de iterações como 1000.

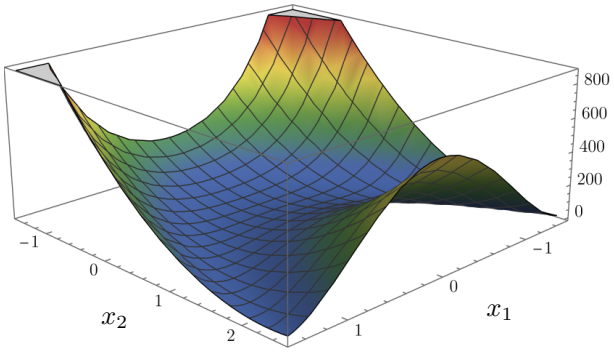



Figura 18: Gráfico da função (3.5) para $x_1 \in [-1.5, 1.5]$ e $x_2 \in [-1.5, 2.5]$

Na Figura 19 podemos acompanhar o valor de (3.5) durante as iterações de uma rodada dos métodos citados. Veja que o método do gradiente descendente não alcança a convergência, realizando o número máximo de iterações. O valor funcional ao final do processo é 1.39e−2, foram realizadas 2146 avaliações de $f$ e o tempo total de processamento foi de $\approx$ 0.2 segundos. Por outro lado, o método `HB adapt` alcança a convergência em 138 iterações, realizando ao total 425 avaliações de $f$ e com um tempo total de processamento de $\approx$ 0.03 segundos. O valor funcional alcançado pelo método `HB adapt` foi 6.79e−14.

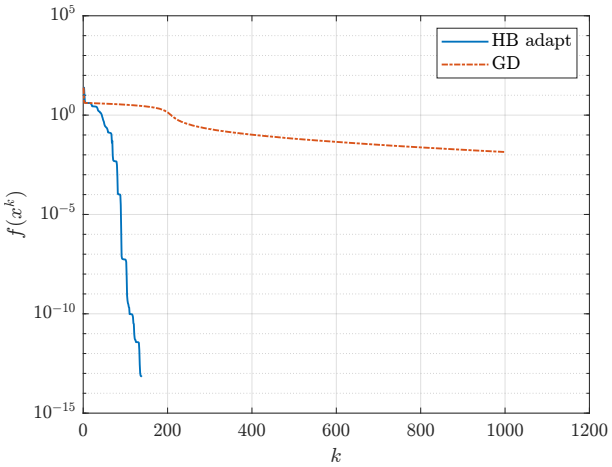
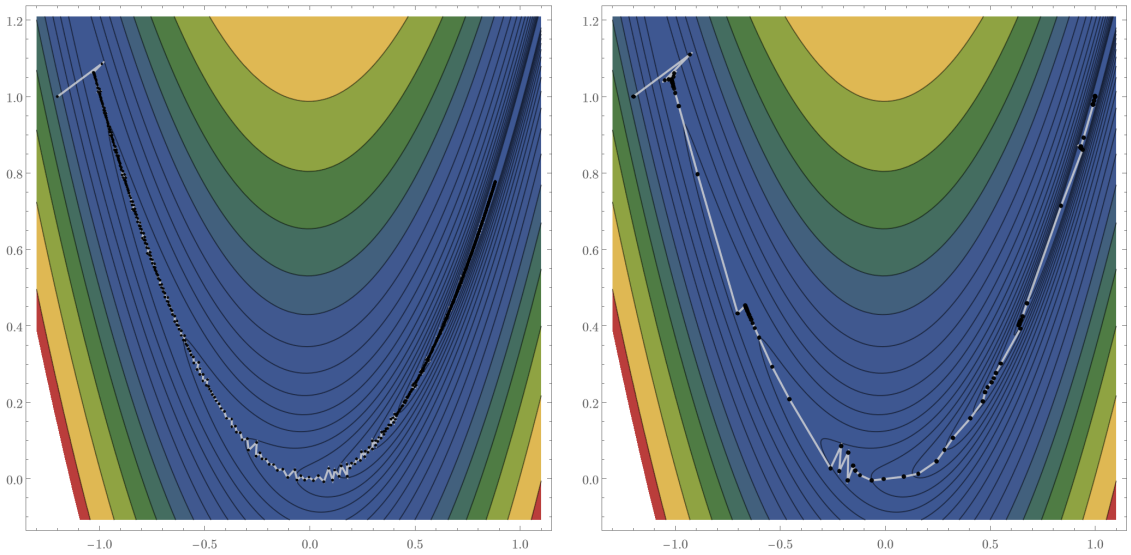



Figura 19: Evolução do valor funcional de (3.5) ao longo das iterações do método do gradiente descendente (`GD`) e da proposta adaptativa para o método *heavy-ball* (`HB adapt`)

Com essas informações e a Figura 19 vemos que o método *heavy-ball* é superior ao método do gradiente descendente para esse problema. Na Figura 20, vemos as trajetórias obtidas com os dois métodos. Na Figura 20a é possível observar que os pontos pretos, que representam os iterados, são próximos um do outro ao decorrer de todo o

processo, e que quanto mais próximos os pontos ficam da solução $x^* = (1, 1)$, menor é o passo dado de uma iteração para a seguinte. Além disso, conseguimos ver o movimento de zigue-zague que a trajetória apresenta, que é uma caraterística do método e que se acentua nesse problema.

Já na Figura 20b, vemos que o método `HB adapt` é capaz de dar passos substancialmente maiores, e ainda assim seguir trajetória parecida com a descrita pelos pontos obtidos com o método do gradiente descendente. A grande aceleração dada no método `HB adapt` pode ser vista na Figura 20b pelos largos passos no decorrer na trajetória aliados de momentos de desaceleração e ajuste da direção.




(a) Gradiente descendente (b) proposta adaptativa para o método *heavy-ball*

Figura 20: Curvas de nível da função (3.5) juntamente com as trajetórias traçadas pelos pontos obtidos com o método do gradiente descendente e com a proposta adaptativa para o método *heavy-ball*

Note que em alguns momentos temos grandes passos seguidos de regiões de condensação de iterados. Isso se dá porque na região de condensação dos pontos a direção de inércia já não é mais de descida, logo o hiperparâmetro $\delta$ age sobre essa direção. Depois dessa iteração, que tende a ser dominada pela direção de gradiente, a direção de inércia volta a ser de descida. Na primeira iteração subsequente, pode acontecer do passo de inércia ser contraído, mas logo nas iterações seguintes ele volta a ser dilatado e novamente são alcançados passos longos. Paralelamente, podemos observar esse comportamento na Figura 19, em que temos trechos de rápida diminuição do valor funcional seguidos de

algumas iterações em que a função varia pouco.

Destacamos que o comportamento do método `HB adapt` que acompanhamos na Figura 20 pôde ser visto também na Figura 5. Assim, vemos que mesmo em uma função que não está no contexto teórico deste trabalho, a proposta adaptativa para o método *heavy-ball* funciona de forma satisfatória.

# 4 Equações diferenciais ordinárias associadas aos métodos estudados

Uma característica interessante de diversos métodos de minimização para solução de (2.1) é a sua conexão com equações diferenciais ordinárias (EDOs). Interpretando a função objetivo $f$ como um potencial (no sentido físico, podemos pensar em um potencial gravitacional para melhor intuição) e os iterados $x^k$ como posições de uma partícula nos instantes $t_k$, podemos interpretar os esquemas iterativos dos métodos como discretizações de equações diferenciais ordinárias que regem o movimento de uma partícula sob influência do potencial $f$.

Para o método de Cauchy essa conexão é direta, dado que o processo iterativo

$$x^{k+1} = x^k - \alpha_k \nabla f(x^k),$$

partindo de $x^0$, pode ser visto como a aplicação do método de Euler explícito com tamanho de passo $\alpha_k$ na $k$-ésima iteração para a integração numérica do problema de valor inicial (PVI)

$$\begin{cases} \dot{X}(t) = -\nabla f(X(t)), \\ X(0) = x^0. \end{cases} \tag{4.1}$$

A equação diferencial em (4.1) é chamada de *fluxo do gradiente* e indica que a velocidade da partícula sempre aponta na direção de menos o gradiente.

Veja que, $t_0 = 0$ e $t_k = \sum_{i=0}^{k-1} \alpha_i$. Quando o tamanho de passo é fixo, i.e, $\alpha_k = \alpha$, então $t_k = k\alpha$. Em geral, por simplicidade na análise das discretizações, o tamanho de passo é escolhido como fixo. Além disso, no contexto de solução de EDO's, é mais comum que se utilize a letra $s$ como tamanho de passo no tempo. Essas considerações

foram seguidas neste trabalho.

A Figura 21 ilustra a conexão entre o método de Cauchy (2.6) e o PVI (4.1). Veja que quanto menor é o tamanho de passo $s$ usado no método de Euler explícito para integração de (4.1) (o que equivale a aplicar o método de Cauchy com passo fixo $s$), mais próxima a trajetória dos iterados está da solução exata. Ademais, como houve convergência para $x^* = (0, 0)$ a despeito do tamanho de passo escolhido, todas as trajetórias se aproximam entre si antes de chegar em $x^*$. Veja na Figura 21 que para $x_1 < 0.2$ todas as trajetórias já são muito próximas umas das outras.

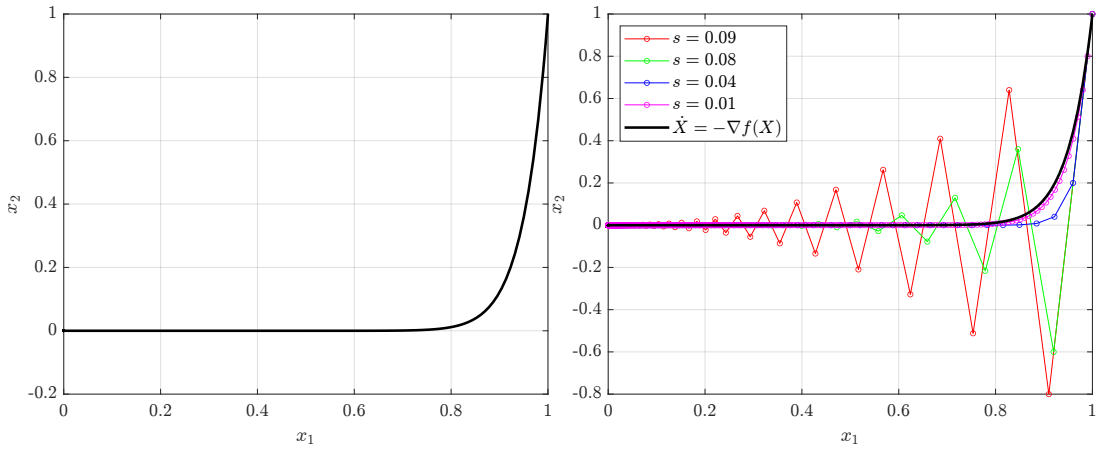



Figura 21: Solução de (4.1) (esquerda) e iterados obtidos pelo método de Cauchy (2.6) com diferentes valores de $s$ (direita) considerando $f(x) = x_1^2 + 20x_2^2$ e $x^0 = (1, 1)$

Os métodos de primeira ordem acelerados também possuem suas contrapartidas contínuas. Por exemplo, o PVI

$$\begin{cases} \ddot{X}(t) + 2\sqrt{\mu}\dot{X}(t) + \nabla f(X(t)) = 0, \\ X(0) = x^0, \\ \dot{X}(0) = 0, \end{cases} \tag{4.2}$$

modela o método *heavy-ball* com uma *EDO de baixa resolução*[‡] [18, Equação (2.3)]. Interpretando (4.2) do ponto de vista físico, sua EDO ($\ddot{X} = -2\sqrt{\mu}\dot{X} - \nabla f(X)$) pode ser lida como a aplicação da segunda lei de Newton para o estudo do movimento de um corpo com massa unitária em um potencial $f$ (que gera força $-\nabla f(X)$) e sujeito a uma força

[‡]Terminologia em contrapartida a *EDOs de alta resolução*, as quais modelam fenômenos com maior acurácia, e portanto caracterizam com maior precisão os métodos acelerados (cf. [17]).

de arrasto proporcional à sua velocidade ($-2\sqrt{\mu}\dot{X}$).

É interessante notar que, assim como discutido na Subseção 2.2, a atualização (2.8) do método *heavy-ball* é uma combinação linear entre a direção usada na última iteração (de inércia) e a direção de menos o gradiente. Isso aparece no modelo contínuo (4.2), já que temos a introdução de um termo de arrasto.

Muitas vezes o termo de arrasto é chamado de termo de amortecimento, visto que a solução de (4.2) possui característica oscilatória, e esse termo reduz gradativamente a amplitude de oscilação, colaborando com a convergência para o ponto estacionário $x^*$. A Figura 22 (esquerda) ilustra a solução de (4.2) para a mesma função e ponto inicial considerados na Figura 21. Veja que a solução apresenta oscilação sob um regime de amortecimento.

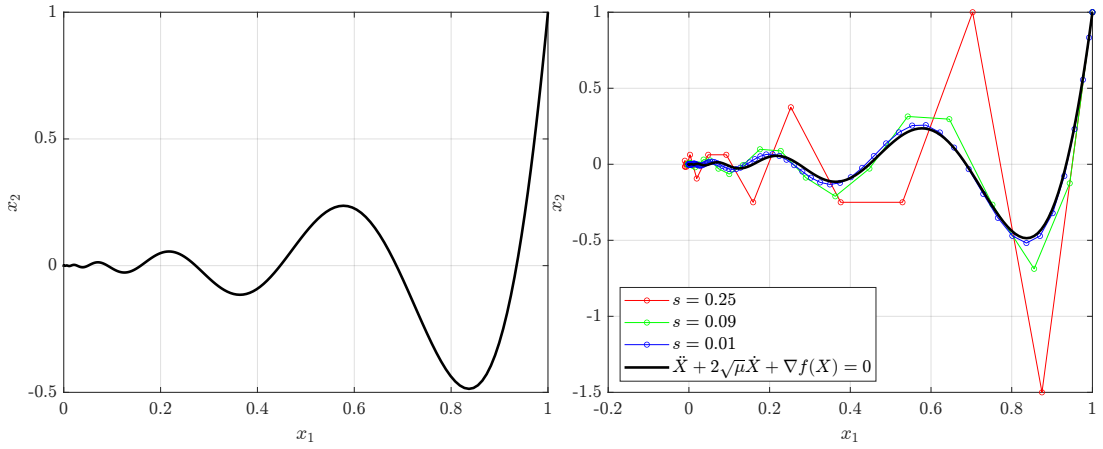



Figura 22: Solução de (4.2) (esquerda) e iterados obtidos pelo esquema (4.3) com diferentes valores de $s$ (direita) considerando $f(x) = x_1^2 + 20x_2^2$ e $x^0 = (1, 1)$

É possível identificar (4.2) como um modelo contínuo para o método *heavy-ball* pois podemos transitar de (4.2) para o esquema discreto (2.8) se aplicarmos o método de Euler simplético para discretizarmos a EDO de (4.2). Dessa forma, seguindo a estratégia exposta em [18], obtemos

$$x^{k+1} = x^k - \frac{s}{1 + 2\sqrt{\mu s}}\nabla f(x^k) + \frac{1}{1 + 2\sqrt{\mu s}}(x^k - x^{k-1}), \tag{4.3}$$

como esquema iterativo para a aproximação $X(k\sqrt{s}) \approx x^k$, e as condições iniciais implicam em $x^0 = x^1$. Se compararmos (2.8) e (4.3), vemos que os passos $\alpha_k$ e $\beta_k$ são constantes

dados por $\alpha = \frac{s}{1+2\sqrt{\mu s}}$ e $\beta = \frac{1}{1+2\sqrt{\mu s}}$.

Na Figura 22 (direita) vemos os iterados obtidos pelo esquema (4.3) com diferentes valores de $s$. Assim como visto na Figura 21, todas as trajetórias convergem para o otimizador $x^* = (0, 0)$, e quanto menor $s$, mais próximos os iterados estão da solução do seu modelo contínuo.

Entretanto, em comparação com a Figura 21, foi possível considerar alguns passos maiores na Figura 22. Isso impacta na velocidade de convergência para $x^*$: os maiores passos considerados nas Figuras 21 e 22, $s = 0.09$ e $s = 0.25$, produziram sequencias que convergiram em 147 e 52 iterações, respectivamente. De qualquer forma, mesmo comparando tamanhos de passo iguais, $s = 0.09$, na Figura 22 esse tamanho de passo produziu uma sequencia que convergiu em 70 iterações, cerca da metade das iterações necessárias para a convergência do método de Cauchy.

Outro método de primeira ordem acelerado estudado neste trabalho foi o método de Nesterov de 1983. O seu modelo contínuo é semelhante ao modelo do método *heavy-ball*, visto que esses métodos são semelhantes na forma de atualização dos iterados $x^k$, como pode ser visto por (2.11). De fato, o modelo (4.2) pode ser usado como contrapartida contínua com uma EDO de baixa resolução para o método de Nesterov de 1983, conforme [18]. Para o estudo de modelos que distinguem entre o método *heavy-ball* e de Nesterov de 1983 são necessárias as EDOs de alta resolução [17].

Outro modelo com EDO de baixa resolução para o método de Nesterov de 1983 é o PVI

$$\begin{cases} \ddot{X}(t) + \dfrac{3}{t}\dot{X}(t) + \nabla f(X(t)) = 0, \\ X(0) = x^0, \\ \dot{X}(0) = 0, \end{cases} \tag{4.4}$$

que apresenta propriedades análogas à sua contrapartida discreta [19].

Veja que a interpretação física dos modelos (4.2) e (4.4) é análoga. A principal diferença está no coeficiente de amortecimento $(\frac{3}{t})$ de (4.4), que depende do inverso de $t$. Sendo assim, existe uma transição no regime de amortecimento: quando $t$ é pequeno, o sistema é superamortecido e quando $t$ cresce, o sistema passa a ser subamortecido.

Na Figura 23 vemos a solução do modelo (4.4) considerando a mesma função e ponto inicial usados nas Figuras 21 e 22. Veja que, assim como no modelo para o

método *heavy-ball*, temos um comportamento oscilatório. Entretanto, é possível observar a transição do sistema superamortecido para o sistema subamortecido: veja que no início da trajetória, a solução oscila menos do que a solução de (4.2), mas quando $t$ é grande a solução passa a oscilar muito nas proximidades do ponto estacionário $x^* = (0, 0)$. Note que o subamortecimento passa a ser tão marcante que mesmo com o zoom feito, ainda podemos observar oscilações.

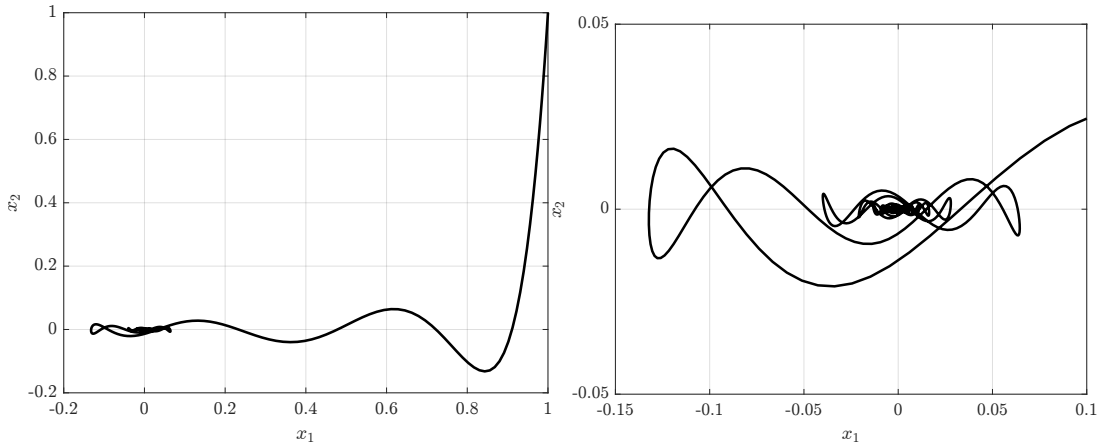



Figura 23: Solução de (4.4) (esquerda) e zoom (direita) considerando $f(x) = x_1^2 + 20x_2^2$ e $x^0 = (1, 1)$

Uma discretização do modelo (4.4) não nos fará reconhecer o método de Nesterov de 1983 pois a avaliação de gradiente em (2.10) é feita em $y^k$, ponto obtido após o movimento na direção de inércia. Entretanto, conforme [19, Teorema 2], se substituirmos o tamanho de passo de gradiente $\frac{1}{L}$ por $s$ em (2.10), quando $s \to 0$, vale que os iterados do método de Nesterov de 1983 convergem para a solução de (4.4).

Do ponto de vista teórico também existem outros ganhos quando analisamos o método de Nesterov de 1983 por meio do modelo contínuo (4.4). Por exemplo, a taxa de convergência do modelo contínuo ($f(X(t)) - f^* \leq \mathcal{O}(1/t^2)$) é análoga à do método discreto ($f(x^k) - f^* \leq \mathcal{O}(1/(\sqrt{s}k)^2)$). Esse paralelismo nos dá a intuição de que os esquemas evoluem de forma parecida.

Além disso, os autores em [19] se valem de funcionais de energia para a demonstração dessas taxas de convergência, inclusive para o caso discreto. Essa é uma maneira mais rápida de se provar a taxa de convergência do método de Nesterov de 1983, que é uma alternativa à estratégia de prova construtiva apresentada em [8], que se vale

da ideia de sequências estimadoras e um grande volume manipulações algébricas.

Esse é um exemplo de que o uso de modelos contínuos para os métodos de otimização pode trazer vantagens na análise teórica dos métodos, tanto no que se refere à convergência quanto no estabelecimento de propriedades. Também é um exemplo de que esses modelos podem ser úteis para compreender melhor os métodos discretos e ganharmos mais intuição sobre eles.

Entretanto, o estudo de equações diferenciais que possuem como ponto estacionário o minimizador $x^*$ procurado não se sobrepõe ao estudos de métodos iterativos para minimização de maneira clássica. Isso porque, apesar do grande arcabouço teórico e prático criado no estudo de solução de PVIs, na minimização temos um objetivo diferente, que é encontrar $x^*$, enquanto que os métodos de solução de PVIs visam a determinação de toda a trajetória até que se alcance a ponto estacionário $x^*$. A busca por essa trajetória traz dificuldades e custos adicionais desnecessários para a solução do problema de minimização.

Veja que, como discutido nas Figuras 21 e 22, para todos os tamanhos de passos tivemos convergência para $x^*$. A diferença entre as trajetórias é o quanto elas são fiéis à solução dos respectivos modelos contínuos. Entretanto, para o problema de minimização, a fidelidade a essa curva solução não é importante no decorrer de todo o processo. Precisamos, de fato, que os iterados convirjam para $x^*$, que é o ponto estacionário dos modelos contínuos. Com isso, não precisamos estar próximos da curva solução desse modelo o tempo todo, o que inclusive traria maior lentidão ao processo de otimização. Precisamos, na verdade, nos aproximar dessa curva no final da integração, como ocorre na trajetória com $s = 0.09$ da Figura 21 e na trajetória com $s = 0.25$ da Figura 22.

Sendo assim, é necessários fazer ponderações para utilizar uma discretização de um modelo contínuo como método de minimização. Por exemplo, esse método não precisa seguir fielmente a sua solução do modelo contínuo, apenas deve estar suficientemente próximo a ela para que não haja divergência. Do ponto de vista de desempenho, por outro lado, os iterados devem estar suficientemente longe para que o tamanho de passo não seja proibitivamente pequeno, e possam percorrer mais espaço mais rapidamente. Nesse sentido, o estudo da região de estabilidade das discretizações pode ser um caminho para determinarmos tamanhos de passo apropriados.

# 5 Conclusão

Neste trabalho estudamos métodos de primeira ordem acelerados para problemas de minimização suave irrestrita. Recentemente, esses métodos passaram a ser amplamente usados e estudados devido a necessidades contemporâneas, como tratamento de grande quantidade de dados. Além das acelerações, também nos debruçamos sobre o estudo de determinação adaptativa dos tamanhos de passos nos diferentes métodos analisados. Como resultado, pudemos propor uma estratégia adaptativa para o método *heavy-ball*, que apresentou resultados satisfatórios nos experimentos realizados.

Sobre os experimentos computacionais, o uso de *performance profile* e de uma grande variabilidade de problemas testes foram fundamentais para obtermos resultados robustos. De fato, a variabilidade introduzida nas funções quadráticas (3.1) nos permitiu identificar um erro de sinal na versão do artigo [4] usada, que não aparecia em rodadas isoladas.

O estudo de contrapartidas contínuas para os métodos de minimização é um tópico interessante e que possui diversas contribuições recentes, como [19] em 2016, [18] em 2019 e [17] em 2021. Essa área de pesquisa possui perguntas em aberto, como '*é possível, de maneira sistemática e teoricamente robusta, obtermos novos métodos acelerados via discretizações de EDOs?*' [18]. Além disso, o desenvolvimento de uma teoria que mapeie propriedades de modelo contínuos nas suas contrapartidas discretas é uma tarefa importante e em aberto [19].

Para trabalhos futuros, objetivamos um estudo teórico detalhado de convergência da proposta adaptativa para o método *heavy-ball*, além de testá-la em uma maior gama de funções. Para funções não convexas e/ou com gradiente não Lipschitz contínuo, a nossa proposta pode ser uma estratégia interessante, como vista para a função de Rosenbrock (3.3). Paralelamente, objetivamos continuar o estudo de contrapartidas contínuas para os métodos de otimização.

Destacamos, ainda, três trabalhos publicados neste ano por Nesterov [10, 11, 12], em que o autor continua estudando estratégias aceleradas, agora se valendo de acelerações em ordem superior e inexatidão. Esses trabalhos indicam, também, direções para estudos futuros.

# Referências


[1] M. S. Bazaraa, H. D. Sherali, and C. M. Shetty, *Nonlinear programming: theory and algorithms*, John Wiley & Sons, 3 ed., 2013.

[2] S. R. Becker, E. J. Candès, and M. C. Grant, *Templates for convex cone problems with applications to sparse signal recovery*, Mathematical Programming Computation, 3 (2011), p. 165.

[3] E. D. Dolan and J. J. Moré, *Benchmarking optimization software with performance profiles*, Mathematical Programming, 91 (2002), pp. 201–213.

[4] C. C. Gonzaga and E. W. Karas, *Fine tuning Nesterov's steepest descent algorithm for differentiable convex programming*, Mathematical Programming, 138 (2013), pp. 141–166.

[5] A. Izmailov and M. Solodov, *Otimização, Volume 1: Condições de Otimalidade, Elementos de Análise Convexa e de Dualidade*, IMPA, Rio de Janeiro, 3 ed., 2014.

[6] D. P. Kingma and J. Ba, *Adam: A method for stochastic optimization*, CoRR, abs/1412.6980 (2015).

[7] Y. Nesterov, *A method of solving a convex programming problem with convergence rate $O(1/k^2)$*, Soviet Mathematics Doklady, 27 (1983), pp. 372–376.

[8] ——, *Introductory Lectures on Convex Optimization: A Basic Course*, vol. 87 of Applied Optimization, Springer Science & Business Media, New York, 2003.

[9] ——, *Gradient methods for minimizing composite objective function*, Discussion paper 76, CORE, UCL, Bélgica, 2007.

[10] ——, *Inexact accelerated high-order proximal-point methods*, Mathematical Programming, (2021), pp. 1–26.

[11] ——, *Inexact high-order proximal-point methods with auxiliary search procedure*, SIAM Journal on Optimization, 31 (2021), pp. 2807–2828.

[12] ——, *Superfast second-order methods for unconstrained convex optimization*, Journal of Optimization Theory and Applications, 191 (2021), pp. 1–30.

[13] J. Nocedal and S. Wright, *Numerical Optimization*, Springer Science & Business Media, 2006.

[14] B. O'Donoghue and E. Candes, *Adaptive restart for accelerated gradient schemes*, Foundations of Computational Mathematics, 15 (2015), pp. 715–732.

[15] B. T. Polyak, *Some Methods of Speeding Up the Convergence of Iteration Methods*, USSR Computational Mathematics and Mathematical Physics, 4 (1964), pp. 1–17.

[16] H. Rosenbrock, *An automatic method for finding the greatest or least value of a function*, The Computer Journal, 3 (1960), pp. 175–184.

[17] B. Shi, S. S. Du, M. I. Jordan, and W. J. Su, *Understanding the acceleration phenomenon via high-resolution differential equations*, Mathematical Programming, (2021), pp. 1–70.

[18] B. Shi, S. S. Du, W. Su, and M. I. Jordan, *Acceleration via symplectic discretization of high-resolution differential equations*, in Advances in Neural Information Processing Systems, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, eds., vol. 32, Curran Associates, Inc., 2019.

[19] W. Su, S. Boyd, and E. J. Candès, *A Differential Equation for Modeling Nesterov's Accelerated Gradient Method: Theory and Insights*, The Journal of Machine Learning Research, 17 (2016), pp. 1–43.

# Apêndice A Pseudocódigos

## A.1 Gradiente Descendente

**Algoritmo 1:** Método do Gradiente Descendente com sequência $\{\alpha_k\}_{k=0}^{\infty}$ pré-definida

**Dados:** $x^0 \in \mathbb{R}^n$, $\{\alpha_k\}_{k=0}^{\infty}$, $\epsilon > 0$, $k_{\max} > 0$, $k = 0$
**enquanto** $\|\nabla f(x^k)\|_2 \geq \epsilon$ **e** $k < k_{max}$ **faça**
  $x^{k+1} \leftarrow x^k - \alpha_k \nabla f(x^k)$
  $k \leftarrow k + 1$
**fim**

**Algoritmo 2:** Método do Gradiente Descendente com Regra de Armijo

**Dados:** $x^0 \in \mathbb{R}^n$, $\alpha_0 > 0$, $\alpha_{\text{contr}} < 1$, $\alpha_{\text{dil}} \geq 1$, $\sigma > 0$, $\epsilon > 0$, $k_{\max} > 0$, $k = 0$
**enquanto** $\|\nabla f(x^k)\|_2 \geq \epsilon$ **e** $k < k_{max}$ **faça**
  **enquanto** $f(x^k - \alpha_k \nabla f(x^k)) > f(x^k) - \sigma \alpha_k \|\nabla f(x^k)\|_2^2$ **faça**
    $\alpha_k \leftarrow \alpha_{\text{contr}} \alpha_k$
  **fim**
  $x^{k+1} \leftarrow x^k - \alpha_k \nabla f(x^k)$
  $\alpha_{k+1} \leftarrow \alpha_{\text{dil}} \alpha_k$
  $k \leftarrow k + 1$
**fim**

## A.2 *Heavy-ball*

**Algoritmo 3:** Método *Heavy-ball* com sequências $\{\alpha_k\}_{k=0}^{\infty}$ e $\{\beta_k\}_{k=0}^{\infty}$ pré-definidas

**Dados:** $x^0, x^1 \in \mathbb{R}^n$, $\{\alpha_k\}_{k=0}^{\infty}$, $\{\beta_k\}_{k=0}^{\infty}$, $\epsilon > 0$ $k_{\max} > 1$, $k = 1$
**enquanto** $\|\nabla f(x^k)\|_2 \geq \epsilon$ **e** $k < k_{max}$ **faça**
  $x^{k+1} \leftarrow x^k - \alpha_k \nabla f(x^k) + \beta_k (x^k - x^{k-1})$
  $k \leftarrow k + 1$
**fim**

**Algoritmo 4:** Método *Heavy-ball* com buscas lineares

**Dados:** $x^0, x^1 \in \mathbb{R}^n$, $\alpha_0 > 0$, $\alpha_{\text{contr}} < 1$, $\alpha_{\text{dil}} \geq 1$, $\beta_0 > 0$, $\beta_{\text{contr}} < 1$, $\beta_{\text{dil}} \geq 1$, $\sigma > 0$, $0 \leq \delta \ll 1$, $\epsilon > 0$, $k_{\max} > 1$, $k = 1$
**enquanto** $\|\nabla f(x^k)\|_2 \geq \epsilon$ **e** $k < k_{max}$ **faça**

  **enquanto** $f(x^k - \alpha_k \nabla f(x^k)) > f(x^k) - \sigma \alpha_k \|\nabla f(x^k)\|_2^2$ **faça**
    $\alpha_k \leftarrow \alpha_{\text{contr}} \alpha_k$
  **fim**
  $\alpha_{k+1} \leftarrow \alpha_{\text{dil}} \alpha_k$

  **se** $\nabla f(x^k)^T (x^k - x^{k-1}) \geq 0$ **então**
    $x^{k+1} \leftarrow x^k - \alpha_k \nabla f(x^k) + \delta \beta_k (x^k - x^{k-1})$
    $\beta_{k+1} \leftarrow \beta_k$
  **senão**
    **enquanto** $f(x^k + \beta_k (x^k - x^{k-1})) > f(x^k) + \sigma \beta_k \nabla f(x^k)^T (x^k - x^{k-1})$ **faça**
      $\beta_k \leftarrow \beta_{\text{contr}} \beta_k$
    **fim**
    $x^{k+1} \leftarrow x^k - \alpha_k \nabla f(x^k) + \beta_k (x^k - x^{k-1})$
    $\beta_{k+1} \leftarrow \beta_{\text{dil}} \beta_k$
  **fim**

  $k \leftarrow k + 1$
**fim**

## A.3 Nesterov (1983)

**Algoritmo 5:** Método de Nesterov de 1983 [7]

**Dados:** $x^0 \in \mathbb{R}^n$, $L > 0$, $\mu \geq 0$, $\alpha_0 \in (0, 1]$, $\epsilon > 0$, $k_{\max} > 0$, $k = 0$

$q \leftarrow \dfrac{\mu}{L}$

$y^0 \leftarrow x^0$

**enquanto** $k < k_{max}$ **faça**

  **se** $\|\nabla f(y^k)\|_2 < \epsilon$ **então**

    $x^{k+1} \leftarrow y^k$

    **interrompa**

  **fim**

  $x^{k+1} \leftarrow y^k - \dfrac{1}{L}\nabla f(y^k)$

  $\alpha_{k+1} \leftarrow \dfrac{q - \alpha_k^2 + \sqrt{(q - \alpha_k^2)^2 + 4\alpha_k^2}}{2}$

  $y^{k+1} \leftarrow x^{k+1} + \dfrac{\alpha_k(1 - \alpha_k)}{\alpha_k + \alpha_{k+1}^2}(x^{k+1} - x^k)$

  $k \leftarrow k + 1$

**fim**

**Algoritmo 6:** Método de Nesterov de 1983 [7] com busca linear

**Dados:** $x^0 \in \mathbb{R}^n$, $L > 0$, $\mu \geq 0$, $\gamma_0 \geq \mu$, $\nu_0 > 0$, $\nu_{\text{contr}} < 1$, $\nu_{\text{dil}} \geq 1$, $\epsilon > 0$, $k_{\max} > 0$, $k = 0$

$z^0 \leftarrow x^0$

**enquanto** $k < k_{max}$ **faça**

- $\alpha_k \leftarrow \frac{\mu - \gamma_k + \sqrt{(\gamma_k - \mu)^2 + 4L\gamma_k}}{2L}$
- $\gamma_{k+1} \leftarrow (1 - \alpha_k)\gamma_k + \alpha_k \mu$
- $y^k \leftarrow \frac{\alpha_k \gamma_k z^k + \gamma_{k+1} x^k}{\gamma_k + \alpha_k \mu}$
- **se** $\|\nabla f(y^k)\|_2 < \epsilon$ **então**
  - $x^{k+1} \leftarrow y^k$
  - **interrompa**
- **fim**
- **enquanto** $\nu_k > \frac{1}{L}$ **e** $f(y^k - \nu_k \nabla f(y^k)) > f(y^k) - \frac{1}{2L}\|\nabla f(y^k)\|_2^2$ **faça**
  - $\nu_k \leftarrow \nu_{\text{contr}}\ \nu_k$
- **fim**
- **se** $\nu_k < \frac{1}{L}$ **então**
  - $\nu_k \leftarrow \frac{1}{L}$
- **fim**
- $x^{k+1} \leftarrow y^k - \nu_k \nabla f(y^k)$
- $z^{k+1} \leftarrow \frac{1}{\gamma_{k+1}}((1 - \alpha_k)\gamma_k z^k + \alpha_k \mu y^k - \alpha_k \nabla f(y^k))$
- $\nu_{k+1} \leftarrow \nu_{\text{dil}}\ \nu_k$
- $k \leftarrow k + 1$

**fim**

## A.4 Nesterov (2007)

**Algoritmo 7:** Método de Nesterov de 2007 [9] com recomeço adaptativo baseado em gradiente [14]

**Dados:** $x^0 \in \mathbb{R}^n$, $\alpha_0 \in (0,1]$, $L_{\text{contr}} < 1$, $L_{\text{dil}} \geq 1$, $\epsilon > 0$, $k_{\max} > 0$, $k = 0$

**se** $\mu$ *não é dado* **então**
  $\mu \leftarrow 0$
**fim**
**se** $L$ *não é dado* **então**
  $\bar{x} \leftarrow x^0 - \frac{1\text{e}{-3}}{\|\nabla f(x^0)\|_2} \nabla f(x^0)$
  $L_0 \leftarrow \frac{1}{2} \frac{\|\nabla f(x^0) - \nabla f(\bar{x})\|_2}{\|x^0 - \bar{x}\|_2}$
**senão**
  $L_0 \leftarrow L$
**fim**
$q_0 \leftarrow \dfrac{\mu}{L_0}$
$y^0 \leftarrow x^0$
**enquanto** $k < k_{max}$ **faça**
  **se** $\|\nabla f(y^k)\|_2 < \epsilon$ **então**
    $x^{k+1} \leftarrow y^k$
    **interrompa**
  **fim**
  $T^k \leftarrow y^k - \frac{1}{L_k} \nabla f(y^k)$
  **enquanto** $\nabla f(T^k)^T \nabla f(y^k) < \|\nabla f(T^k)\|_2^2$ **faça**
    $L_k \leftarrow L_{\text{dil}}\, L_k$
    $T^k \leftarrow y^k - \frac{1}{L_k} \nabla f(y^k)$
  **fim**
  $x^{k+1} \leftarrow T^k$
  $q_k \leftarrow \frac{\mu}{L_k}$
  $L_{k+1} \leftarrow L_{\text{contr}} L_k$
  **se** $\nabla f(y^k)^T (x^{k+1} - x^k) > 0$ **então**
    $\alpha_{k+1} \leftarrow \alpha_0$
    $y^{k+1} \leftarrow x^{k+1}$
  **senão**
    $\alpha_{k+1} \leftarrow \dfrac{q_k - \alpha_k^2 + \sqrt{(q_k - \alpha_k^2)^2 + 4\alpha_k^2}}{2}$
    $y^{k+1} \leftarrow x^{k+1} + \dfrac{\alpha_k(1 - \alpha_k)}{\alpha_k + \alpha_{k+1}^2}(x^{k+1} - x^k)$
  **fim**
  $k \leftarrow k + 1$
**fim**

## A.5 Gonzaga e Karas (2013)

**Algoritmo 8:** Método de Gonzaga e Karas de 2013 [4]

**Dados:** $x^0 \in \mathbb{R}^n$, $\gamma_0 > 0$, $\beta > 1$, $\epsilon > 0$ $k_{\max} > 0$, $k = 0$

**se** $\mu$ *é dado* **então**
  $\mu^* \leftarrow \mu$
**senão**
  $\mu^* \leftarrow 0$
**fim**
$z^0 \leftarrow x^0$
**enquanto** $k < k_{max}$ **faça**
  $d^k = z^k - x^k$
  Escolha de $\theta_k$ pelo Algoritmo 9
  $y^k \leftarrow x^k + \theta_k d^k$
  **se** $\|\nabla f(y^k)\|_2 < \epsilon$ **então**
    $x^{k+1} \leftarrow y^k$
    **interrompa**
  **fim**
  **se** $L$ *é dado* **então**
    $x^{k+1} \leftarrow y^k - \frac{1}{L}\nabla f(y^k)$
  **senão**
    Calcule $x^{k+1}$ por um passo do Algoritmo 2 a partir de $y^k$
  **fim**
  **se** $\gamma_k - \mu^* < \beta(\mu_+ - \mu^*)$ **então**
    $\mu_+ \leftarrow \max\{\mu^*, 0.1\gamma_k\}$
  **fim**
  $\tilde{\mu} \leftarrow \frac{\|\nabla f(y^k)\|_2^2}{2(f(y^k) - f(x^{k+1}))}$
  **se** $\mu_+ \geq \tilde{\mu}$ **então**
    $\mu_+ \leftarrow \max\{\mu^*, 0.1\tilde{\mu}\}$
  **fim**
  Calcule $\alpha_k$ como a maior raiz da equação (2.19) de [4] com $\mu \leftarrow \mu_+$
  $\gamma_{k+1} = (1 - \alpha_k)\gamma_k + \alpha_k\mu_+$
  $z^{k+1} = \frac{1}{\gamma_{k+1}}\left((1 - \alpha_k)\gamma_k z^k + \alpha_k(\mu_+ y^k - \nabla f(y^k))\right)$
  $k \leftarrow k + 1$
**fim**

**Algoritmo 9:** Determinação de $\theta_k$ para o Algoritmo 8

**se** $\mu$ *e* $L$ *não são dados* **então**
- **se** $f(x^k + d^k) \leq f(x^k)$ **então**
  - $\theta_k \leftarrow 1$
- **senão se** $\nabla f(x^k)^T d^k \geq 0$ **então**
  - $\theta_k \leftarrow 0$
- **senão**
  - $\theta_k \leftarrow 1$
  - **enquanto** $f(x^k + \theta_k d^k) > f(x^k)$ **faça**
    - $\theta_k \leftarrow \theta_k / 2$
  - **fim**
- **fim**

**fim**

**se** $L$ *é dado* **e** $\mu$ *não é dado* **então**
- $\alpha_{\mathcal{N}} \leftarrow \frac{1}{4L}\left(\mu_+ - \gamma_k + \sqrt{(\mu_+ - \gamma_k)^2 + 8L\gamma_k}\right)$
- $\theta_{\mathcal{N}} \leftarrow \frac{\alpha_{\mathcal{N}}\gamma_k}{\gamma_k + \alpha_{\mathcal{N}}\mu_+}$
- $\theta_k \leftarrow \theta_{\mathcal{N}}$
- **se** $f(x^k + \theta_k d^k) < f(x^k)$ **então**
  - **enquanto** $2\theta_k \leq 1$ **e** $f(x^k + 2\theta_k d^k) < f(x^k)$ **faça**
    - $\theta_k \leftarrow 2\theta_k$
  - **fim**
- **senão**
  - **enquanto** $f(x^k + \theta_k d^k) > f(x^k)$ **faça**
    - $\theta_k \leftarrow \theta_k / 2$
  - **fim**
- **fim**

**fim**

**se** $\mu$ *e* $L$ *são dados* **então**
- $\bar{\theta} \leftarrow \frac{\mu^*}{2L}$
- **se** $f(x^k + \bar{\theta} d^k) \geq f(x^k) - \frac{(\mu^*)^2}{8L}\|d^k\|_2^2$ **então**
  - $\theta_k \leftarrow 0$
- **senão se** $f(x^k + d^k) \leq f(x^k)$ **então**
  - $\theta_k \leftarrow 1$
- **senão**
  - $\theta_k \leftarrow 1$
  - **enquanto** $f(x^k + \theta_k d^k) > f(x^k)$ **faça**
    - $\theta_k \leftarrow \theta_k / 2$
  - **fim**
- **fim**

**fim**