Uma implementação do método de Penalidades para problemas de otimização restrita

Ramon de Oliveira Rocha ramonoliveira98@hotmail.com
Instituto Federal do Sudeste de Minas Gerais, Rio Pomba, MG, Brazil

Hernando José Rocha Franco hernando.franco@ifsudestemg.edu.br
Instituto Federal do Sudeste de Minas Gerais, Rio Pomba, MG, Brazil

Resumo

A otimização é um ramo da Matemática Aplicada que busca minimizar ou maximizar funções de determinados problemas a fim de obter suas melhores soluções possíveis. No campo da chamada otimização restrita, uma possível abordagem é a de transformar um problema restrito em um subproblema irrestrito, geralmente mais fácil e que pode ser resolvido e usado como base de um processo iterativo. Nesse contexto, este trabalho apresenta e discute a implementação de um método de otimização com restrições: o método de Penalidades. Para isso, o algoritmo foi programado e testado no *software* MatLab. Em seguida, o método foi aplicado na resolução de dez problemas testes da literatura matemática. Ressalta-se que, para a resolução de cada subproblema de otimização gerado, utilizou-se a função de otimização irrestrita *fminsearch* disponível no MatLab. Por último, coletou-se os resultados obtidos pelo algoritmo nessa aplicação e os comparou com suas respectivas soluções de referência. Assim, constatou-se que o algoritmo obteve os resultados de acordo com o previsto na literatura. Além disso, enfatiza-se que, quanto maior o número de restrições, maior a complexidade envolvida na implementação do problema, podendo aumentar a dificuldade de utilização do método.

Palavras-chave

Minimização restrita, Algoritmos, Programação matemática.

1 Introdução

Otimização é uma área da Matemática Aplicada que se refere ao estudo de problemas que consistem em minimizar ou maximizar funções de uma ou várias variáveis.

Os problemas reais de otimização estão presentes em diversos campos científicos, por exemplo, nas Engenharias, Economia, Mecânica, Logística, Administração e outros, em que se busca obter soluções ótimas, ou as melhores possíveis para esses problemas a partir dos modelos matemáticos que os representam [2].

No campo da intitulada otimização restrita, uma possível abordagem é a de transformar um problema em um subproblema mais fácil que pode, então, ser resolvido e usado como base de um processo iterativo.

Nesse sentido, uma característica de uma grande classe de métodos iniciais é a transformação de um problema restrito para um problema básico irrestrito, utilizando uma função de penalidade para restrições que estão perto ou além do conjunto de restrições. Dessa forma, o problema restrito é resolvido gerando-se uma sequência de otimizações irrestritas parametrizadas que, em seu limite, converge para a solução ótima do problema restrito.

Em face do acima exposto, este trabalho tem por objetivo apresentar e discutir a implementação de um método de otimização com restrições: o método de Penalidades. Os principais critérios avaliados foram a precisão numérica dos resultados obtidos e a implementação computacional dos problemas.

2 Referencial Teórico

Um problema geral de otimização é dado da seguinte forma:

$$\min f(\mathbf{x}) \tag{1}$$
$$\text{sujeito à } \mathbf{x} \in M \subset \mathbb{R}^n,$$

onde $f : \mathbb{R}^n \to \mathbb{R}$ e $M = \{\mathbf{x} \in \mathbb{R}^n \mid g_i(\mathbf{x}) \leq 0,\ i = 1, \ldots, m;\ h_j(\mathbf{x}) = 0,\ j = 1, \ldots, l\ \}$. Assim, queremos obter um ponto $\mathbf{x}^* \in \mathbb{R}^n$ que minimize a função f, satisfazendo as restrições g_i e h_j do conjunto M.

Chamamos f de função objetivo e M de conjunto viável (ou factível). Dizemos ainda que $\mathbf{x}^* \in M$ é um ponto ótimo de f se

$$f(\mathbf{x}^*) \leq f(\mathbf{x});\ \forall \mathbf{x} \in M,$$

sendo $f(\mathbf{x}^*)$ um valor ótimo.

2.1 Método de Penalidades

No chamado método de Penalidades Exterior, ou simplesmente método de Penalidades, qualquer violação das restrições g_i ou h_j é penalizada no valor da função objetivo f.

Para mostrarmos a versão mais básica desse método, vamos primeiro definir a função indicadora do conjunto M como sendo:

$$I_M(\mathbf{x}) = \begin{cases} 0 & \text{para } \mathbf{x} \in M, \\ \infty & \text{para } \mathbf{x} \notin M. \end{cases}$$

Com isso, podemos pensar em substituir o problema inicial (1) pelo seguinte problema irrestrito:

$$\min f(\mathbf{x}) + I_M(\mathbf{x}) \tag{2}$$

$$\text{sujeito à } \mathbf{x} \in \mathbb{R}^n.$$

Nesse caso, note que se $\mathbf{x} \in M$, estaremos minimizando a própria função objetivo f. Por outro lado, quando $\mathbf{x} \notin M$, a função I_M se torna uma “penalidade infinita" para pontos não viáveis, isto é, que estão fora do conjunto M.

Na prática, como não lidamos diretamente com o “infinito", aproximamos a função I_M por um termo de penalidade p que envolve as restrições do problema e que, normalmente, é dado por:

$$p(\mathbf{x}) = \sum_{i=1}^{m} [g_i^+(\mathbf{x})]^\beta + \sum_{j=1}^{l} [h_j(\mathbf{x})]^\beta, \tag{3}$$

tal que g_i^+ é denotada como a parte positiva da restrição g_i e definida por:

$$g_i^+(\mathbf{x}) = \begin{cases} g_i(\mathbf{x}) & \text{se } g_i(\mathbf{x}) > 0, \\ 0 & \text{se } g_i(\mathbf{x}) \leq 0. \end{cases}$$

Vale destacar aqui que o expoente $\beta \in \mathbb{Z}_+^*$ em (3) está relacionado à diferenciabilidade de p e, em aplicações gerais, assume os valores 2 ou 4 [7]. Em face disso, só precisamos que $f, g_1, \ldots, g_m, h_1, \ldots, h_l$ sejam funções contínuas em $\mathbb{R}^n$.

Uma vez que estamos considerando $I_M(\mathbf{x}) \approx p(\mathbf{x})$, pela caracterização (3) vem que $p(\mathbf{x}) = 0$, para $\mathbf{x} \in M$, e $p(\mathbf{x}) > 0$, para $\mathbf{x} \notin M$. Nesse contexto, podemos adotar um parâmetro $r > 0$ de modo que $rp(\mathbf{x}) = I_M(\mathbf{x})$, quando $\mathbf{x} \in M$ e, se $\mathbf{x} \notin M$, o termo $rp(\mathbf{x}) \to I_M(\mathbf{x})$ quanto mais r cresce [7].

De posse disso, podemos reformular o problema irrestrito (2) da seguinte maneira:

$$\min_{\mathbf{x} \in \mathbb{R}^n} f(\mathbf{x}) + rp(\mathbf{x})$$

Para um processo iterativo, podemos escolher um ponto de partida $\mathbf{x}_0$ e fazer $\mathbf{x} = \mathbf{x}_{k+1}$; $r = r_k > 0$, com $k = 0, 1, 2, \ldots$, sabendo que $\lim\limits_{k \to \infty} r_k = \infty$ e presumindo que $\mathbf{x}_{k+1} \to \mathbf{x}^*$. Queremos dizer com isso que r é um parâmetro que pode ser atualizado até que o ponto $\mathbf{x}$ se torne ótimo.

Sintetizando, podemos obter a solução de (1) através de:

$$\min_{\mathbf{x}\in\mathbb{R}^n} q(\mathbf{x}_{k+1}, r_k) \tag{P}$$

onde $q(\mathbf{x}_{k+1}, r_k) = f(\mathbf{x}_{k+1}) + r_k p(\mathbf{x}_{k+1})$ é denominada função auxiliar.

A fim de que $q(\mathbf{x}_{k+1}, r_k) \approx f(\mathbf{x}_{k+1})$, então, na prática, o procedimento iterativo se encerra quando $r_k p(\mathbf{x}_{k+1}) \leq \varepsilon$, onde ε trata-se de um valor suficientemente pequeno escolhido como critério de parada.

Em suma, o que buscamos é gerar uma sequência de soluções do problema transformado (P) que converge para a solução do problema geral (1).

Algoritmo de Penalidades

Algoritmo 1: O algoritmo de Penalidades para minimizar funções multidimensionais com restrições.

1. Escolha um escalar $\varepsilon > 0$, um inteiro $\beta > 0$, um ponto inicial $\mathbf{x}_0 \in \mathbb{R}^n$, e um parâmetro $r_0 > 0$. Faça $k = 0$.
2. A partir de $\mathbf{x}_k$, determine uma solução $\mathbf{x}_{k+1}$ do subproblema irrestrito (P).
3. Se $r_k p(\mathbf{x}_{k+1}) \leq \varepsilon$, pare. Caso contrário, continue com a etapa (4).
4. Faça $r_{k+1} = \alpha r_k$, com $\alpha > 1$; $k = k + 1$ e retorne à etapa (2).

É válido ressaltar que, se r é um número muito grande, então o termo $p(\mathbf{x})$ deve ser bem pequeno, pois, senão, o problema fica sujeito a uma penalidade considerável, como foi dito antes. Esse argumento justifica intuitivamente a etapa (3) e, de fato, prova-se que $rp(\mathbf{x}) \to 0$ quando $r \to \infty$ [1].

Vamos ver um exemplo de como esse método funciona efetivamente.

Exemplo 3.1: Resolva o problema de otimização abaixo:

$$\min f(x) = 3x^2 - 2x$$

$$\text{sujeito à } g(x) = 2 - x \leq 0,$$

considerando $\varepsilon = 10^{-3}$, $\beta = 2$, $x_0 = 0.5$, $r_0 = 1$ e $\alpha = 10$.

1) Dado que $x_0 = 0.5$ e $r_0 = 1$, faça $k = 0$.

2) Para obtermos x_1, devemos montar e avaliar a função auxiliar $q(x_1, r_0)$ tendo em vista o ponto inicial x_0 tomado. Para isso, dado que $\beta = 2$, teremos que a parte positiva da restrição g será:

$$[g^+(x_{k+1})]^2 = \begin{cases} (2-x_{k+1})^2 & \text{se } 2 > x_k, \\ 0 & \text{se } 2 \leq x_k. \end{cases}$$

Nesse contexto, sendo $q(x_1, r_0) = f(x_1) + r_0[g(x_1)]^2$, então, temos:

$$q(x_1, r_0) = \begin{cases} 3x_1^2 - 2x_1 + r_0(2-x_1)^2 & \text{se } 2 > x_0, \\ 3x_1^2 - 2x_1 & \text{se } 2 \leq x_0. \end{cases}$$

Assim, como $x_0 = 0.5 < 2$ e uma vez que $r_0 = 1$, para encontrarmos x_1 teremos de resolver o seguinte problema irrestrito:

$$\min_{x\in\mathbb{R}} 3x_1^2 - 2x_1 + (2-x_1)^2$$

Poderíamos aqui usar vários algoritmos da otimização irrestrita (método de Newton, método do Gradiente, dentre outros) para determinarmos x_1. No entanto, com o intuito de reduzirmos os cálculos, podemos obtê-lo analiticamente aplicando o Teorema de Fermat [6], pois, neste caso, q é derivável em todos os pontos do domínio e admite ainda um valor de mínimo absoluto.

Logo, vem que:

$$\begin{aligned} \frac{d}{dx_1}\left[3x_1^2 - 2x_1 + (2-x_1)^2\right] = 0 \quad &\Leftrightarrow \quad 8x_1 - 6 = 0 \\ &\Leftrightarrow \quad x_1 = 0.75 \end{aligned}$$

é o ponto de mínimo dessa função.

3) Perceba agora que:

$$\begin{aligned} r_0 p(x_1) &= r_0(2-x_1)^2 \\ &= 1(2-0.75)^2 \\ &= 1.5625 \end{aligned}$$

Como $1.5625 > 10^{-3} = \varepsilon$, seguimos para a etapa (4).

4) Fazemos então $r_1 = 10r_0 = 10$, atualizamos $k = 0$ para $k = 1$ e retornamos ao passo (2).

Refazendo todo o procedimento até que $r_k p(x_{k+1}) \leq \varepsilon$, vamos chegar nos valores dispostos na Tabela 1 adiante:

Iteração k	r_k	x_{k+1}	$r_k p(x_{k+1})$	$q(x_{k+1}, r_k)$	$f(x_{k+1})$
0	1	0.75	1.5625	1.75	0.1875
1	10	1.61538	1.47932	6.07692	4.59759
2	100	1.95145	0.23571	7.75728	7.52215
3	1000	1.99501	0.0249	7.97507	7.95017
4	10000	1.9995	0.0025	7.9975	7.995
5	100000	1.99995	0.00025	7.99975	7.9995

Tabela 1: Método de Nelder-Mead após uma contração externa, uma contração interna e uma redução.

Fonte: Elaboração própria.

De modo geral, perceba que $x \to 2$ à medida que r cresce e $rp(x)$ se aproxima de zero, como já era esperado. Por conseguinte, se fizéssemos mais algumas iterações, poderíamos chegar em $x^* = 2$ tal que $f(x^*) = 8$ é o valor ótimo procurado.

Para a realização da etapa (2) do algoritmo de Penalidades, isto é, a resolução do subproblema irrestrito (P), utilizamos neste trabalho a função de otimização *fminsearch* do Matlab. Essa se baseia no método de Nelder-Mead, um dos métodos de busca direta mais utilizados em problemas de minimização sem restrições de uma função de n variáveis.

2.1.1 *Fminsearch*

Apresentaremos, nesta seção, o algoritmo de Nelder-Mead e suas principais características geométricas.

O método de Nelder-Mead [4], assim como a maior parte dos métodos de busca direta, utiliza os conceitos de um simplex no $\mathbb{R}^n$. No caso de duas variáveis, por exemplo, um simplex é um triângulo, e o método compara os valores da função nos três vértices do triângulo.

O algoritmo usa 4 operações possíveis, cujos coeficientes escalares são:

- coeficiente de reflexão: $\rho > 0$,
- coeficiente de expansão: $\chi > 1$ com $\chi > \rho$,
- coeficiente de contração: $0 < \gamma < 1$,
- coeficiente de redução (shrink): $0 < \sigma < 1$.

Uma escolha, praticamente universal, é definir $\rho = 1$, $\chi = 2$, $\gamma = 1/2$ e $\sigma = 1/2$.

Sejam $f : \mathbb{R}^n \to \mathbb{R}$ e um simplex com $n + 1$ vértices. Em uma dada iteração do método, os $n+1$ vértices do simplex, $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{n+1}$ pertencentes a $\mathbb{R}^n$, são ordenados de acordo com o crescimento dos valores de f, ou seja,

$$f(\mathbf{x}_1) \leq f(\mathbf{x}_2) \leq \ldots \leq f(\mathbf{x}_{n+1}),$$

onde $\mathbf{x}_1$ é definido como o melhor vértice e $\mathbf{x}_{n+1}$ o pior, isto é, o vértice com o maior valor da função objetivo. O método busca, então, substituir esse pior vértice do simplex por outro com melhor valor. O novo vértice é obtido através da reflexão, expansão ou contração do pior vértice ao longo da reta que passa por este vértice e o centroide dos n melhores vértices ($\bar{\mathbf{x}}$), que é dado por:

$$\bar{\mathbf{x}} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{x}_i$$

A cada iteração o pior vértice é substituído por um novo vértice. Um novo simplex é formado e a pesquisa continua, gerando uma sequência de simplex na qual os valores da função nos vértices ficam cada vez menores.

Assim, as principais etapas do método podem ser observadas nas Figuras 1 e 2:

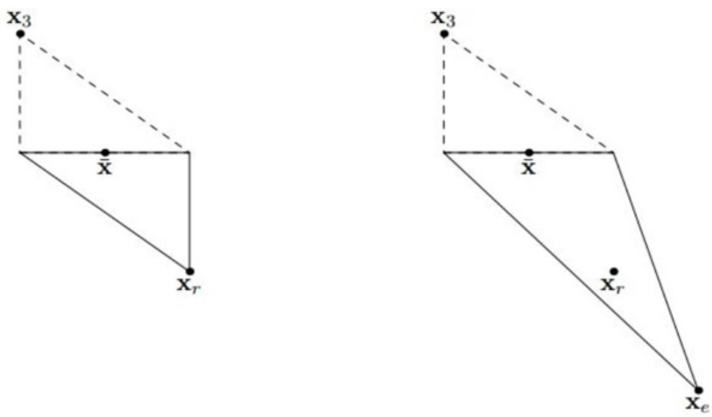

Figura 1: Método de Nelder-Mead após uma etapa de reflexão e expansão.
Fonte: LAGARIAS et al, 1998, p. 117.

Na Figura 1, observa-se que, se o ponto refletido $\mathbf{x}_r$ é melhor que o vértice, então o simplex foi refletido numa direção que minimiza a função objetivo. É importante, neste caso, fazer a expansão $\mathbf{x}_e$ na mesma direção. Caso contrário, se o ponto refletido não possuir um valor melhor da função objetivo, então é provável que o simplex esteja próximo do ponto de mínimo da função, sendo testados, assim, os pontos de contração externa $\mathbf{x}_e$ e interna $\mathbf{x}_i$. Se nenhuma das operações resultou em um novo vértice com valor da função objetivo pelo menos melhor do que aquele correspondente ao vértice a

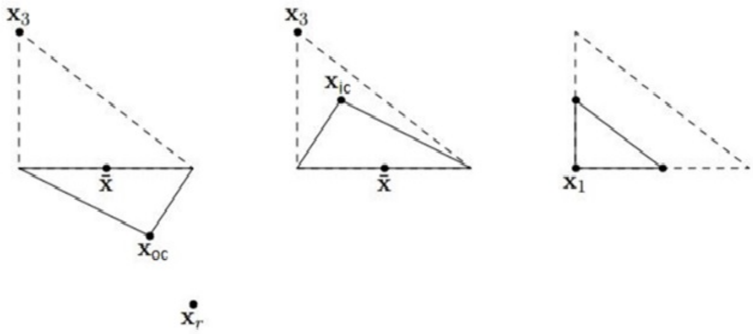

Figura 2: Método de Nelder-Mead após uma contração externa, uma contração interna e uma redução.

Fonte: LAGARIAS et al, 1998, p. 117.

ser rejeitado, deve-se reduzir o simplex, pois o ponto de mínimo está em seu interior. Esta operação de redução é feita preservando o vértice $\mathbf{x}_1$ e aproximando os demais vértices na direção de $\mathbf{x}_1$, conforme a Figura 2.

Algoritmo de Nelder-Mead

Algoritmo 2: Método Simplex de Nelder-Mead

Dados:

(1) Função $f(\mathbf{x})$ com $\mathbf{x} \in \mathbb{R}^n$

(2) Conjunto de vértices iniciais: $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{n+1} \in \mathbb{R}^n$

(3) Coeficiente de reflexão $\rho > 0$

(4) Coeficiente de expansão $\chi > 1$

(5) Coeficiente de contração $0 < \gamma < 1$

(6) Coeficiente de redução $0 < \sigma < 1$

Passo 1: Calcule o valor da função objetivo para os $n + 1$ vértices e ordene-os de modo que $f(\mathbf{x}_1) \leq f(\mathbf{x}_2) \leq \ldots \leq f(\mathbf{x}_{n+1})$

Passo 2: Calcule o centroide $\bar{\mathbf{x}}$ dos n melhores vértices

Passo 3: Enquanto algum critério de parada não for atingido:

(3a) Calcule o ponto de reflexão $\mathbf{x}_r = \bar{\mathbf{x}} + \rho(\bar{\mathbf{x}} - \mathbf{x}_{n+1})$

(3b) Calcule $f(\mathbf{x}_r)$

(3c) Se $f(\mathbf{x}_1) \leq f(\mathbf{x}_r) < f(\mathbf{x}_n)$, então $\mathbf{x}_{n+1} = \mathbf{x}_r$

(3d) Se $f(\mathbf{x}_r) < f(\mathbf{x}_1)$, então calcule o ponto de expansão $\mathbf{x}_e = \bar{\mathbf{x}} + \chi(\mathbf{x}_r - \bar{\mathbf{x}})$

Calcule $f(\mathbf{x}_e)$

Se $f(\mathbf{x}_e) \leq f(\mathbf{x}_r)$, então $\mathbf{x}_{n+1} = \mathbf{x}_e$

Senão, $\mathbf{x}_{n+1} = \mathbf{x}_r$

(3e) Se $f(\mathbf{x}_n) \leq f(\mathbf{x}_r) < f(\mathbf{x}_{n+1})$, então calcule o ponto de contração externa $\mathbf{x}_{oc} = \bar{\mathbf{x}} + \gamma(\mathbf{x}_r - \bar{\mathbf{x}})$

Calcule $f(\mathbf{x}_{oc})$

Se $f(\mathbf{x}_{oc}) \leq f(\mathbf{x}_r)$, então $\mathbf{x}_{n+1} = \mathbf{x}_{oc}$

Senão, vá para o passo (3g)

(3f) Se $f(\mathbf{x}_r) \geq f(\mathbf{x}_{n+1})$, então calcule o ponto de contração

interna $\mathbf{x}_{ic} = \bar{\mathbf{x}} + \gamma(\mathbf{x}_r - \bar{\mathbf{x}})$

Calcule $f(\mathbf{x}_{ic})$

Se $f(\mathbf{x}_{ic}) \leq f(\mathbf{x}_{n+1})$, então $\mathbf{x}_{n+1} = \mathbf{x}_{ic}$

Senão, vá para o passo (3g)

(3g) Redução: para $2 \leq i \leq n + 1$, faça $\mathbf{x}_i = \mathbf{x}_1 + \sigma(\mathbf{x}_i - \mathbf{x}_1)$

(3h) Volte ao passo 2

Passo 4: Retorna $\mathbf{x}$

3 Metodologia

Realizou-se uma pesquisa bibliográfica e exploratória do seguinte modo: a primeira etapa consistiu em estudar e programar o algoritmo. Para isso, foi utilizada a versão 2018(b) do *software* MatLab em um microcomputador core i5, 6GB de RAM.

Assim, avaliou-se a implementação a partir de alguns testes, além de realizar os devidos ajustes para melhoria de sua convergência e programação.

Ressalta-se que, para a resolução de cada subproblema de otimização gerado, utilizou-se a função de otimização irrestrita *fminsearch* disponível no MatLab.

Na segunda etapa, o método foi aplicado na resolução de um conjunto de dez problemas testes da literatura matemática, descritos na seção 3.1 e retirados de [3], [5] e [7].

Para fins de aplicação, a tolerância do critério de parada foi de $\varepsilon = 10^{-4}$, o parâmetro inicial estabelecido foi $r_0 = 1$, o expoente de cada parcela do termo de penalidade foi definido como $\beta = 2$, e o fator de atualização de r foi de $\alpha = 10$. Quanto ao ponto inicial $\mathbf{x}_0$, este foi adotado de acordo com a literatura consultada.

Por último, foram coletados os resultados obtidos pelo algoritmo nessa aplicação, e elaborada uma tabela a fim de analisar e comparar com as respectivas soluções de referência.

3.1 Descrição dos problemas numéricos

Problema 01
Função objetivo: $f(x) = 2x^2 - x$
Restrições:
$x \geq 1$
Ponto inicial: $x_0 = 12$
Valor ótimo: $f(x^*) = 1$ em $x^* = 1$

Problema 02
Função objetivo: $f(x_1, x_2) = (x_1 - 1)^2 + x_2^2$
Restrições:
$x_2 - x_1 \leq 1$
Ponto inicial: $\mathbf{x}_0 = (15, 11)$
Valor ótimo: $f(\mathbf{x}^*) = 2$ em $\mathbf{x}^* = (0, 1)$

Problema 03
Função objetivo: $f(x_1, x_2) = x_1 + x_2$

Restrições:

$$x_2 - x_1^2 = 0$$

Ponto inicial: $\mathbf{x}_0 = (5, 17)$

Valor ótimo: $f(\mathbf{x}^*) = -1/4$ em $\mathbf{x}^* = (-1/2, 1/4)$

Problema 04

Função objetivo: $f(x_1, x_2) = (x_1 - 2)^2 + (x_2 - 1)^2$

Restrições:

$$\begin{array}{rcl} x_1 + x_2 & \leq & 2 \\ x_1^2 - x_2 & \leq & 0 \end{array}$$

Ponto inicial: $\mathbf{x}_0 = (0, 5)$

Valor ótimo: $f(\mathbf{x}^*) = 1$ em $\mathbf{x}^* = (1, 1)$

Problema 05

Função objetivo: $f(x_1, x_2) = x_1$

Restrições:

$$\begin{array}{rcl} x_1^2 - 2x_1 - x_2 & \leq & 0 \\ x_1^2 - 2x_1 + x_2 & \leq & 0 \\ -x_1 & \leq & 0 \end{array}$$

Ponto inicial: $\mathbf{x}_0 = (0, 1)$

Valor ótimo: $f(\mathbf{x}^*) = 0$ em $\mathbf{x}^* = (0, 0)$

Problema 06

Função objetivo: $f(x_1, x_2, x_3) = x_1 + x_2^2 + 3x_2x_3 + x_3^2$

Restrições:

$$\begin{array}{rcl} -x_1 & \leq & 0 \\ -x_2 & \leq & 0 \\ -x_3 & \leq & 0 \end{array}$$

Ponto inicial: $\mathbf{x}_0 = (2, 5, 10)$

Valor ótimo: $f(\mathbf{x}^*) = 0$ em $\mathbf{x}^* = (0, 0, 0)$

Problema 07

Função objetivo: $f(x_1, x_2, x_3, x_4, x_5) = (x_1 - x_2)^2 + (x_3 - 1)^2 + (x_4 - 1)^4 + (x_5 - 1)^6$

Restrições:

$$\begin{array}{rcl} x_1 + x_2 + x_3 + 4x_4 - 7 & = & 0 \\ x_3 + 5x_5 - 6 & = & 0 \end{array}$$

Ponto inicial: $\mathbf{x}_0 = (-15, 2, 0, -45, 2)$

Valor ótimo: $f(\mathbf{x}^*) = 0$ em $\mathbf{x}^* = (1, 1, 1, 1, 1)$

Problema 08

Função objetivo: $f(x_1, x_2, x_3, x_4) = -x_1x_2x_3x_4$

Restrições:

$$\begin{aligned} x_1^3 + x_2^2 - 1 &= 0 \\ x_1^2x_4 - x_3 &= 0 \\ x_4^2 - x_2 &= 0 \end{aligned}$$

Ponto inicial: $\mathbf{x}_0 = (0.8, 0.8, 0.8, 0.8)$

Valor ótimo: $f(\mathbf{x}^*) = -1/4$ em $\mathbf{x}^* = (2^{-1/3}, 2^{-1/2}, (-1)^i 2^{-11/12}, (-1)^i 2^{-1/4})$, com $i = 1, 2$.

Problema 09

Função objetivo: $f(x_1, x_2, x_3, x_4, x_5) = (x_1 - x_2)^2 + (x_2 - x_3)^3 + (x_3 - x_4)^4 + (x_4 - x_5)^4$

Restrições:

$$\begin{aligned} x_1 + x_2^2 + x_3^3 - 3 &= 0 \\ x_2 - x_3^2 + x_4 - 1 &= 0 \\ x_1x_5 - 1 &= 0 \end{aligned}$$

Ponto inicial: $\mathbf{x}_0 = (-20, \sqrt{2}, -16, 7 - \sqrt{2}, -10)$

Valor ótimo: $f(\mathbf{x}^*) = 0$ em $\mathbf{x}^* = (1, 1, 1, 1, 1)$

Problema 10

Função objetivo: $f(x_1, x_2) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2$

Restrições:

$$\begin{aligned} x_1x_2 - 1 &\geq 0 \\ x_1 + x_2^2 &\geq 0 \\ x_1 &\leq 0.5 \end{aligned}$$

Ponto inicial: $\mathbf{x}_0 = (-2, 1)$

Valor ótimo: $f(\mathbf{x}^*) = 306.5$ em $\mathbf{x}^* = (0.5, 2)$

4 Resultados e discussões

Na Tabela 2 a seguir, apresentamos os resultados numéricos obtidos pelo método implementado.

Problemas	**Algoritmo de Penalidades**		Soluções de referência	
	$\mathbf{x}$	$f(\mathbf{x})$	$\mathbf{x}^*$	$f(\mathbf{x}^*)$
01	1	1	1	1
02	$(0.0050, 1.0051)$	2.0002	$(0, 1)$	2
03	$(-0.5, 0.25)$	-0.25	$(-0.5, 0.25)$	-0.25
04	$(0.9997, 1.0003)$	1.0006	$(1, 1)$	1
05	$(0, 0)$	0	$(0, 0)$	0
06	$(0, 0, 0)$	0	$(0, 0, 0)$	0
07	$(0.9992, 0.9992, 1, 1.0004, 1)$	0	$(1, 1, 1, 1, 1)$	0
08	$(0.7940, 0.7068, 0.53, 0.8407)$	-0.25	$(0.7937, 0.7071, 0.5297, 0.8408)$	-0.25
09	$(1, 1, 1, 1, 1)$	0	$(1, 1, 1, 1, 1)$	0
10	$(0.5, 2.0001)$	306.5184	$(0.5, 2)$	306.5

Tabela 2: Resultados coletados pelo método de Penalidades.
Fonte: Dados da pesquisa.

A partir da Tabela 2, constata-se que o algoritmo resolveu todos os dez problemas testes em consonância com suas soluções de referência $\mathbf{x}^*$ e $f(\mathbf{x}^*)$.

Além disso, observou-se que o valor de $\mathbf{x}_0$ atribuído não afetou a convergência final do método nesse conjunto de problemas avaliados.

No que se refere às variações obtidas nas soluções de alguns problemas, verifica-se que essas foram bem sutis, sendo em geral da ordem de 10^{-2} à 10^{-4}.

É válido frisar que, em relação a implementação computacional de cada problema, quanto maior o número de restrições, maior a complexidade envolvida na implementação do problema, podendo aumentar a dificuldade de utilização do método.

Isso se justifica uma vez que, na prática, para um problema com n restrições de desigualdade, deve-se considerar 2^n sentenças a fim de formular a função auxiliar $f(\mathbf{x})+rp(\mathbf{x})$ e, em particular, formular o termo $p(\mathbf{x})$. Além disso, se houver restrições de igualdade, deve-se lembrar que essas serão sempre acopladas a cada uma das sentenças avaliadas.

5 Considerações finais

Apresentamos neste trabalho um estudo sobre problemas de otimização com restrições, a partir da implementação do chamado método de Penalidades.

A utilização da função *fminsearch* para a resolução do subproblema irrestrito trouxe robustez ao algoritmo de Penalidades, no que se refere a não necessidade do cálculo de derivadas, como ocorre em outros métodos de otimização irrestrita.

Para trabalhos futuros, destacamos a possibilidade de utilização de outras outras estratégias para a minimização do subproblema irrestrito.

Referências

[1] Mokhtar S. Bazaraa, Hanif D. Sherali, and Chitharanjan M. Shetty. *Nonlinear programming: theory and algorithms*. John Wiley & Sons, 2006.

[2] Frederick S. Hillier and Gerald J. Lieberman. *Introdução à pesquisa operacional*. McGraw Hill Brasil, 2013.

[3] Willi Hock and Klaus Schittkowski. Test examples for nonlinear programming codes. *Journal of optimization theory and applications*, 30(1):127–129, 1980.

[4] Jeffrey C. Lagarias, James A. Reeds, Margaret H. Wright, and Paul E. Wright. Convergence properties of the Nelder–Mead simplex method in low dimensions. *SIAM Journal on optimization*, 9(1):112–147, 1998.

[5] Ademir Alves Ribeiro and Elizabeth Wegner Karas. Otimização contínua: Aspectos teóricos e computacionais. *São Paulo: Cengage Learning*, 2013.

[6] James Stewart. Cálculo, volume I. 7. ed. *São Paulo: Cengage Learning*, 2013.

[7] Peter Zörnig. Introdução à programação não linear. *Brasília: UNB*, 2011.