



ENG1786 - Otimização de Projetos

## Trabalho 3

**Alunos:**

Manoel Feliciano - 1310824

Rayanne Gonçalves de Souza - 1221778

# Sumário

<b>1</b>	<b>Introdução</b>	<b>5</b>
<b>2</b>	<b>Otimização de uma treliça de 10 barras</b>	<b>6</b>
2.1	Função objetivo . . . . .	6
2.2	Restrições . . . . .	6
2.3	Métodos indiretos . . . . .	7
2.3.1	Método da penalidade . . . . .	8
2.3.2	Método da barreira . . . . .	9
2.4	Resultados . . . . .	10
2.4.1	Método da penalidade . . . . .	10
2.4.2	Método da barreira . . . . .	12
<b>3</b>	<b>Conclusão</b>	<b>14</b>

## Lista de Figuras

1	Treliça de 10 barras com carregamento aplicado . . . . .	5
---	--	---

## Lista de Tabelas

1	Parâmetros do problema . . . . .	5
2	Parâmetros de ajuste - método da penalidade . . . . .	10
3	Diâmetros obtidos a cada iteração . . . . .	11
4	Peso e $r_p$ obtidos a cada iteração . . . . .	12
5	Parâmetros de ajuste - método da barreira . . . . .	12
6	Diâmetros obtidos a cada iteração . . . . .	13
7	Valor da função peso e do fator de barreira a cada iteração . . . . .	14

# 1 Introdução

No presente trabalho, o objetivo é minimizar a função correspondente ao peso da treliça plana de 10 barras mostrada na Figura 1. O parâmetro de projeto a ser minimizado será o diâmetro individual de cada barra presente na estrutura. Nesse caso, será importante considerar a presença de restrições, visto que as tensões nas barras não devem superar as tensões críticas de escoamento e de flambagem do material.

Para isso, utilizando o MATLAB como ambiente de programação, serão implementados os métodos de penalidade e barreira, que são métodos de otimização com restrição. Eles serão responsáveis por gerar uma função onde, a cada iteração, o algoritmo de otimização sem restrições, desenvolvido no Trabalho 1, poderá ser adaptado e utilizado. Adicionalmente, vale ressaltar que as tensões nas barras serão calculadas iterativamente através de um programa de elementos finitos fornecido.

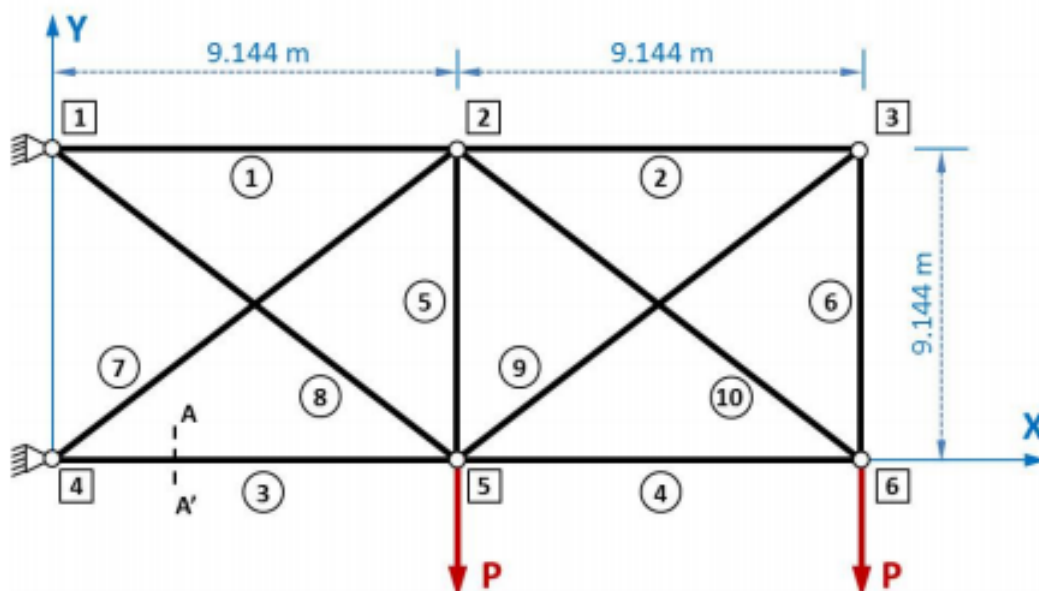


Figura 1: Treliça de 10 barras com carregamento aplicado

Como dados disponíveis para a estrutura acima temos: o peso específico ( $\rho$ ), o módulo de elasticidade ( $E$ ) do material das barras, a tensão crítica de escoamento, as dimensões da treliça (conforme mostrado na imagem) e o valor das cargas  $P$  aplicadas na treliça. A Tabela 7 abaixo condensa as informações já não presentes na Figura 1:

Parâmetro	Símbolo	Valor
Peso específico	$\rho$	27,143 $kN/m^3$
Módulo de elasticidade	$E$	68.947.600,00 $kN/m^2$
Tensão crítica de escoamento	$\sigma_y$	172.369,00 $kN/m^2$
Carga aplicada	$P$	444,822 $kN$

Tabela 1: Parâmetros do problema

## 2 Otimização de uma treliça de 10 barras

### 2.1 Função objetivo

Uma vez que são conhecidos os comprimentos e o peso específico de todas as barras, a expressão de minimização do peso das treliças é obtida em função desses dados.

Para cada barra o peso é dado pela expressão abaixo.

$$P_i = \rho_i V_i \quad (1)$$

Sendo,

$$V_i = A_i L_i \quad (2)$$

Então o peso total da treliça é fornecido pela seguinte função objetivo.

$$f = \sum_{i=1}^{10} \rho_i A_i L_i \quad (3)$$

Onde:

$\rho_i$  - Peso específico da barra i.

$L_i$  - Comprimento da barra i.

$A_i$  - Área da seção transversal da barra i.

### 2.2 Restrições

As tensões nas barras da treliça não devem superar os valores da tensão de escoamento do material  $\sigma_y$  e da tensão crítica de Euler  $\sigma_{cr}$ , para que as barras não sofram escoamento e flambagem.

Abaixo temos as expressões para a tensão axial e a tensão crítica de Euler.

$$\sigma_i = E_i \frac{L_{final} - L_{inicial}}{L_{inicial}} \quad (4)$$

$$\sigma_{cr} = \frac{\pi^2 E_i I_i}{L_{final}^2 A_i} \quad (5)$$

Sendo  $I_i$  o momento de inércia da barra i necessário para o cálculo da tensão crítica de Euler.

$$I_i = \frac{\pi d_i^4}{L_{final}^2 A_i} \quad (6)$$

Nas expressões acima tem-se:

$E_i$  - Módulo de elasticidade da barra  $i$ .

$A_i$  - Área da seção transversal da barra  $i$ .

$d_i$  - Diâmetro da barra  $i$ .

Para que não haja flambagem, a tensão axial deve ser menor que a tensão crítica de Euler, ou seja,  $\sigma_i - \sigma_{cr} < 0$ . Contudo, apenas quando as barras estão submetidas a compressão axial que pode haver o fenômeno de flambagem, então para que na tração a restrição seja satisfeita  $\sigma_i - \sigma_{cr} < 0$  é adaptada para seguinte restrição.

$$-\frac{\sigma_i}{\sigma_{cr}} - 1 < 0 \quad (7)$$

Onde:

$\sigma_i$  - Tensão axial.

$\sigma_{cr}$  - Tensão crítica de Euler.

A restrição de não haver escoamento deve ser respeitada, ou seja,  $\sigma_i - \sigma_y < 0$  deve ser satisfeita tanto para a compressão, quanto para a tração das barras, então a segunda restrição deste problema de minimização do peso de uma treliça é a seguinte.

$$\frac{|\sigma_i|}{\sigma_y} - 1 < 0 \quad (8)$$

Onde:

$\sigma_i$  - Tensão axial.

$\sigma_y$  - Tensão de escoamento.

## 2.3 Métodos indiretos

Os métodos indiretos foram uma das primeiras estratégias adotadas para solução de problemas de otimização com restrições. Uma de suas maiores vantagens reside na simplicidade, visto que iterativamente solucionam-se vários problemas de otimização sem restrição. Por outro lado, sua desvantagem está na sua baixa eficiência computacional, o que fez com que ao longo do tempo fossem substituídos pelos métodos diretos.

Existem basicamente dois diferentes métodos nesse grupo: o de penalidade e o de barreira. São eles que vão gerar a chamada pseudo-função objetivo, onde será possível reaproveitar o que foi desenvolvido no Trabalho 1 para sua minimização. Ela é construída, basicamente, através da adição de fatores que representam uma penalidade baseada na violação ou não das restrições de projeto, conforme abaixo:

$$\phi(\mathbf{x}, r) = f(\mathbf{x}) + r \cdot p(\mathbf{x}) \quad (9)$$

onde:

$\phi$  - pseudo-função objetivo

$f$  - função objetivo

$\mathbf{x}$  - variável de projeto

$r$  - fator de penalidade

$p$  - função de penalidade

### 2.3.1 Método da penalidade

Esse método, também conhecido como "penalidade exterior", deve ter seu ponto inicial, preferencialmente, fora da região viável, como próprio nome já sugere. Dessa forma, a função objetivo é penalizada até que todas as restrições sejam satisfeitas e a função de penalidade seja nula, isto é, enquanto o ponto  $\mathbf{x}^k$  estiver no exterior da região viável. Quanto isso acontecer, com uma certa tolerância, significa que a pseudo-função objetivo é equivalente a função objetivo do problema, conforme pode ser observado na equação (9). Por fim, para esse método a função de penalidade é definida da seguinte forma:

$$p(\mathbf{x}) = \sum_{k=1}^m [h_k(\mathbf{x})]^2 + \sum_{l=1}^p \max[0, c_l(\mathbf{x})]^2 \quad (10)$$

onde:

$h_k$  - restrições de igualdade

$c_l$  - restrições de desigualdade

Pode-se observar pela expressão (10) acima que as restrições de igualdade sempre penalizam a função objetivo. Por outro lado, para as restrições de desigualdade essa penalização é dependente da violação ou não das mesmas. Isso acontece caso as mesmas sejam maiores que zero, caso contrário não penalizadas, conforme pode ser observado ao utilizar a função de máximo entre zero e o valor da restrição.

Vale lembrar que, conforme exposto na seção anterior, no caso que estamos analisando não temos restrições de igualdade. Dessa forma, a expressão final da função de penalidade que será utilizada na implementação será conforme abaixo:

$$p(\mathbf{x}) = \sum_{l=1}^p \max[0, c_l(\mathbf{x})]^2 \quad (11)$$

Por fim, nesse caso temos que o algoritmo para a implementação desse método é da seguinte forma:

- (1)  $k = 0$ ;  $\mathbf{x}^0$  (de preferência que viole alguma restrição);  $r_p^0 > 0$ ;  $\beta > 1$
- (2) Definir a função penalidade para o ponto  $\mathbf{x}^k$  conforme a equação (11)
- (3) Criar a pseudo-função objetivo, conforme abaixo:

$$\phi(\mathbf{x}, r_p^k) = f(\mathbf{x}) + \frac{1}{2} \cdot r_p^k \cdot p(\mathbf{x})$$

- (4) Minimizar  $\phi(\mathbf{x}, r_p^k)$  por meio dos métodos de otimização sem restrição implementados no Trabalho 1, obtendo  $\mathbf{x}^{k+1}$ .
- (5) Verifica a convergência:

$$\frac{1}{2} \cdot r_p^k \cdot p(\mathbf{x}) < TOL \Rightarrow \mathbf{x}^* = \mathbf{x}^{k+1} \Rightarrow \mathbf{FIM}$$

- (6)  $r_p^{k+1} \leftarrow r_p^k \cdot \beta$ ;  $k \leftarrow k + 1 \Rightarrow$  volta para o passo (2)



### 2.3.2 Método da barreira

De maneira análoga ao método anterior, o método da barreira também é conhecido como "penalidade interior", ou seja, nesse caso seu ponto inicial deve estar sempre dentro da região viável. Nesse caso, temos uma característica mais vantajosa em relação ao método anterior: em caso de parada do processo iterativo por qualquer razão, têm-se que a solução encontrada será uma solução viável do problema. Assim como no método anterior, também deseja-se que a função de penalidade tenda a zero com uma certa tolerância. Entretanto, nesse caso a penalização acontece a partir do interior da região viável: quanto mais próximo o ponto  $\mathbf{x}^k$  da fronteira, maior será a penalização. As funções de barreira mais usadas são a de Barreira Inversa e a de Barreira Logarítmica, definidas conforme abaixo:

Barreira Inversa:

$$p(\mathbf{x}) = \frac{1}{2} \cdot r_p \cdot \sum_{k=1}^m [h_k(\mathbf{x})]^2 - r_b^k \cdot \sum_{l=1}^p \frac{1}{c_l(\mathbf{x})} \quad (12)$$

Barreira Logarítmica:

$$p(\mathbf{x}) = \frac{1}{2} \cdot r_p \cdot \sum_{k=1}^m [h_k(\mathbf{x})]^2 - r_b^k \cdot \sum_{l=1}^p -\log[-c_l(\mathbf{x})]^2 \quad (13)$$

onde:

$h_k$  - restrições de igualdade

$c_l$  - restrições de desigualdade

Nesse caso, diferentemente do método da penalidade as restrições de desigualdade sempre penalizam a função objetivo, conforme pode ser observado nas equações (12) e (13).

Como dito nas seções anteriores, não temos restrições de igualdade para o caso analisado. Dessa forma, a expressão final da função de penalidade que será utilizada na implementação desse método será conforme abaixo:

$$p(\mathbf{x}) = -r_b^k \cdot \sum_{l=1}^p \frac{1}{c_l(\mathbf{x})} \quad (14)$$

Por fim, nesse caso temos que o algoritmo para a implementação desse método, muito semelhante ao do método da penalidade, é da seguinte forma:

- (1)  $k = 0$ ;  $\mathbf{x}^0$  (necessariamente viável);  $r_b^0 > 0$ ;  $\beta \in (0, 1)$
- (2) Definir a função penalidade para o ponto  $\mathbf{x}^k$  conforme a equação (14)
- (3) Criar a pseudo-função objetivo, conforme abaixo:

$$\phi(\mathbf{x}, r_b^k) = f(\mathbf{x}) - r_b^k \cdot p(\mathbf{x})$$

- (4) Minimizar  $\phi(\mathbf{x}, r_b^k)$  por meio dos métodos de otimização sem restrição implementados no Trabalho 1, obtendo  $\mathbf{x}^{k+1}$ .

- (5) Verifica a convergência:

$$r_b^k \cdot p(\mathbf{x}^{k+1}) < TOL \Rightarrow \mathbf{x}^* = \mathbf{x}^{k+1} \Rightarrow \mathbf{FIM}$$

- (6)  $r_b^{k+1} \leftarrow r_b^k \cdot \beta$ ;  $k \leftarrow k + 1 \Rightarrow$  volta para o passo (2)

## 2.4 Resultados

### 2.4.1 Método da penalidade

Para o método da penalidade, os seguintes parâmetros de tolerância, incremento do passo constante, fator de penalidade, beta e diâmetros iniciais foram utilizados:

<b>Parâmetro</b>	<b>Valor</b>
Diâmetros iniciais	$1m$ para todos
Incremento $\Delta\alpha$	$2 \cdot 10^{-5}$
Tolerância	$10^{-5}$
Fator de penalidade $r_b$	0.01
Fator $\beta_b$	10

Tabela 2: Parâmetros de ajuste - método da penalidade

Com esses parâmetros, os resultados de diâmetros para cada iteração foram obtidos na página seguinte, utilizando o método univariante para a otimização sem restrição de  $\phi$ :

Iteração	d1	d2	d3	d4	d5	d6	d7	d8	d9	d10
1	0.0157923	-0.0000038	0.0796161	0.0540961	0.0665961	0.0000038	0.0333914	0.0407514	0.0342361	0.0000114
2	0.0158161	-0.0000038	0.0796161	0.0540961	0.0665961	0.0000038	0.0333914	0.0407514	0.0342361	0.0000114
3	0.0448397	-0.0000038	0.1599611	0.1052579	0.0980656	0.0000038	0.0882022	0.0630914	0.0621000	0.0000114
4	0.0453436	-0.0000038	0.1599611	0.1052579	0.0980656	0.0000038	0.0882022	0.0630914	0.0621000	0.0000114
5	0.0454321	-0.0000038	0.1599611	0.1052579	0.0980656	0.0000038	0.0882022	0.0630914	0.0621000	0.0000114
6	0.0808627	-0.0000038	0.1768209	0.1211445	0.0311074	0.0000038	0.0893975	0.1474741	0.0683781	-0.0000123
7	0.0809465	-0.0000038	0.1768209	0.1211445	0.0311074	0.0000038	0.0893975	0.1474741	0.0683781	-0.0000123
8	0.0809465	-0.0000038	0.1777409	0.1210015	0.0315541	0.0000038	0.0893975	0.1474741	0.0684220	-0.0000123
9	0.0809465	-0.0000038	0.1777647	0.1210015	0.0315056	0.0000038	0.0893975	0.1474741	0.0684220	-0.0000123
10	0.0809465	-0.0000038	0.1777399	0.1210015	0.0316779	0.0000038	0.0893975	0.1474741	0.0684220	-0.0000123
11	<b>0.0809465</b>	<b>-0.0000038</b>	<b>0.1777399</b>	<b>0.1210015</b>	<b>0.0317018</b>	<b>0.0000038</b>	<b>0.0893975</b>	<b>0.1474741</b>	<b>0.0684220</b>	<b>-0.0000123</b>

Tabela 3: Diâmetros obtidos a cada iteração

Iteração	Função Peso	Fator de penalidade $r_p$
1	8.1614752	0.0100000
2	47.3461347	0.100000
3	15.6013953	1
...	...	...
9	19.9746920	1000000
10	19.974567	10000000
11	19.974809	100000000

Tabela 4: Peso e  $r_p$  obtidos a cada iteração

#### 2.4.2 Método da barreira

Para o método da barreira, foi verificada a necessidade de ajustar o parâmetro de incremento do passo constante apenas quando necessário. Em outras palavras, diminuí-lo apenas quando alguma restrição fosse violada até o ponto que isso não mais acontecesse.

Os parâmetros de tolerância, incremento padrão do passo constante, fator de barreira, beta e diâmetros iniciais foram ajustados conforme abaixo:

Parâmetro	Valor
Diâmetros iniciais	$0.1m$ para todos
Incremento $\Delta\alpha$	$10^{-3}$
Tolerância	$10^{-5}$
Fator de barreira $r_b$	100
Fator $\beta_b$	0.5

Tabela 5: Parâmetros de ajuste - método da barreira

Com os parâmetros acima, obtêm-se os resultados para os diâmetros a cada iteração do programa principal na página seguinte, utilizando o método univariante na otimização sem restrição de  $\phi$ :

Iteração	d1	d2	d3	d4	d5	d6	d7	d8	d9	d10
1	0.1963607	-0.0000041	0.3638024	0.2487649	0.3377254	-0.0000041	0.4029959	0.3209959	0.2598288	0.0000729
2	0.1620090	-0.0000041	0.3638024	0.2487648	0.3377254	-0.0000041	0.4029959	0.3209959	0.2598288	0.0000729
3	0.1300050	-0.0000041	0.3638024	0.2487649	0.3377254	-0.0000041	0.4029959	0.3209959	0.2598288	0.0000730
...	...	...	...	...	...	...	...	...	...	...
14	0.0875388	0.0002259	0.1742038	0.1347496	0.0131683	0.0002209	0.0712325	0.1523827	0.0821845	0.0016900
15	0.0852430	0.0002259	0.1742038	0.1347496	0.0131683	0.0002209	0.0712325	0.1523827	0.0821845	0.0016900
16	0.0840028	0.0002259	0.1742038	0.1347496	0.0131683	0.0002209	0.0712325	0.1523827	0.0821845	0.0016900
...	...	...	...	...	...	...	...	...	...	...
27	0.0817225	0.0000659	0.1769813	0.1246115	-0.0000007	0.0000709	0.0685577	0.1501910	0.0687440	0.0002126
28	0.0817225	0.0000659	0.1769813	0.1246115	-0.0000007	0.0000709	0.0685577	0.1501910	0.0687440	0.0002126
29	0.0817326	0.0000659	0.1769814	0.1246116	-0.0000008	0.0000709	0.0685577	0.1501911	0.0687441	0.0002127
...	...	...	...	...	...	...	...	...	...	...
37	0.0817126	0.0000659	0.1769814	0.1246116	-0.0000008	0.0000709	0.0685577	0.1501911	0.0687441	0.0002127
38	0.0819280	0.0000184	0.1766069	0.1244949	0.0010323	0.0000386	0.0682252	0.1500889	0.0684181	0.0000890
<b>39</b>	<b>0.08193797</b>	<b>0.0000185</b>	<b>0.1766069</b>	<b>0.1244949</b>	<b>0.0010323</b>	<b>0.0000386</b>	<b>0.0682252</b>	<b>0.1500890</b>	<b>0.0684181</b>	<b>0.0000890</b>

Tabela 6: Diâmetros obtidos a cada iteração

Iteração	Função Peso	Fator de barreira $r_b$
1	2.2075145e+03	100
2	1.1827618e+03	50
3	6.6935367e+02	25
...	...	...
14	21.6035808	0.0122070
15	21.1009219	0.0061035
16	20.8222121	0.0030518
...	...	...
27	19.2524245	1.4901161e-06
28	19.2521573	7.4505806e-07
29	19.2521842	3.7252903e-07
...	...	...
37	19.1932450	1.4551915e-09
38	19.1935439	7.2759576e-10
<b>39</b>	<b>19.1938539</b>	<b>3.6379788e-10</b>

Tabela 7: Valor da função peso e do fator de barreira a cada iteração

### 3 Conclusão

O objetivo desse trabalho foi otimizar uma treliça de 10 barras, onde o diâmetro das mesmas era o parâmetro de projeto. No Trabalho 1 foi desenvolvido um algoritmo que implementa diversos métodos de otimização sem restrição. Ele foi readaptado e utilizado durante o presente trabalho, visto que o procedimento de otimização com restrições por métodos indiretos necessitam de uma etapa de otimização sem restrições. Por simplicidade, apenas os métodos Univariante e de Powell foram implementados, visto que os mesmos não utilizam o gradiente da função objetivo para determinar a direção de busca.

Para o primeiro método indireto, o método de penalidade, obtiveram-se diâmetros negativos muito pequenos para as barras 2 e 10 e um peso final da treliça de  $19.97kN$ . Esse resultado pode ser explicado, visto que esse método parte da região não viável do problema em busca do mínimo da função. Dessa forma, a solução do problema não estará na região viável, mas sim muito próxima da mesma.

Para o segundo método indireto, o método da barreira, obtiveram-se diâmetros positivos e peso final da treliça muito próximo ao encontrado no método anterior:  $19.19kN$ . Como esse método parte da região viável e, ao longo de todas as iterações, nunca devemos violar as restrições, é natural que a solução encontrada esteja dentro da região viável.