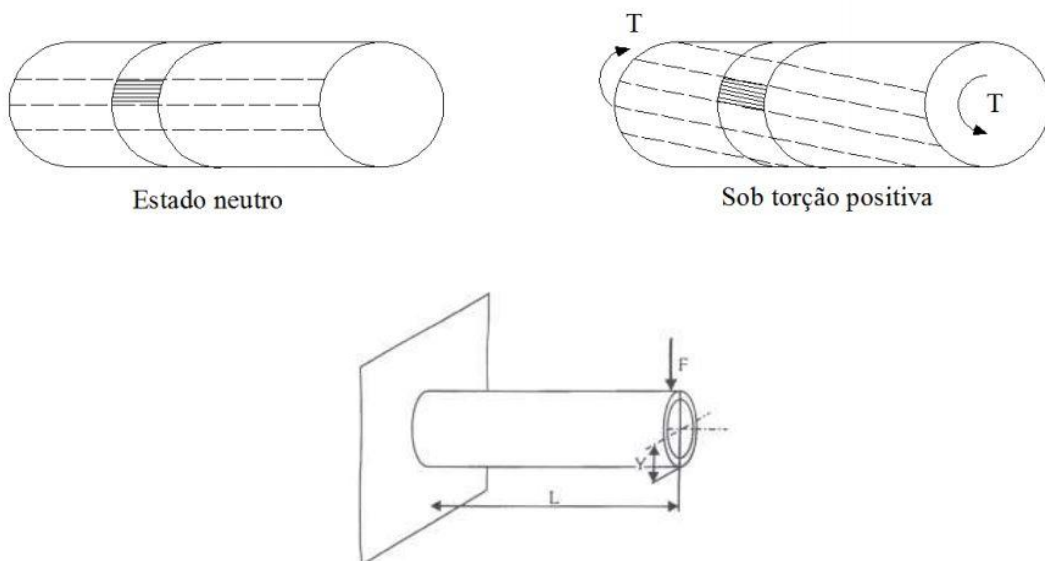


Busca e otimização de uma função na área de instrumentação industrial



Problema Resolvido

O fenômeno da flexão causa o encurvamento da estrutura, em estruturas unidimensionais e bidimensionais, isto ocorre quando o carregamento é aplicado perpendicularmente ao seu eixo principal ou plano principal.

Para ocorrer o encurvamento, algumas fibras devem sofrer alongamento e outras, encurtamento. Porém, existirá uma fibra que não sofrerá alongamento, nem encurtamento e é chamada de linha neutra. Um tipo de aplicação bastante comum é o da barra engastada de comprimento L e seção coroa-circular, onde atua uma força F (unidade em Newtons) na extremidade (Figura 1).

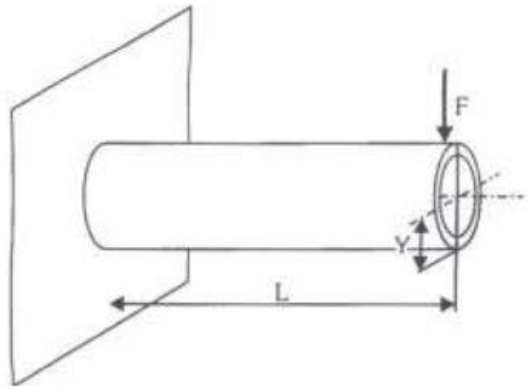


Figura 1- Barra coroa circular engastada

Objetivo e Descrição do funcionamento

Neste exemplo o algoritmo genético tenta encontrar a melhor solução para minimizar a deformação da barra, de modo que as fibras não venham a se romper mantendo-as na chamada linha neutra onde não ocorre mudança. Para calcular a deformação máxima é utilizado a seguinte formula:

$$\varepsilon = \frac{32 \times F \times L \times D}{E \times \pi \times (D^4 - d^4)}$$

Passos de funcionamento:

Passo 1: Inicializa os parâmetros que serão utilizados no algoritmo genético.

```
#Individuos e geracao
```

```
NumIndividuals = 50
```

```
IndividualSize = 22
```

```
MaxGeneration = 40
```

```
#Taxas
```

```
MutationRate = 0.008
```

```
CrossoverRate = 0.65
```

```
#Diametros
```

```
MinD = 51
```

```
MaxD = 99
```

```
Mind = 1
```

```
Maxd = 99
```

```
#Precisao e melhor escolha
```

```
Target = 0.00005
```

```
Elitism = True
```

Passo 2: Inicializa a população que será analisada.

```
self.population = self.problem.initPopulation(population_size)
```

Passo 3: Pega o melhor individuo da população.

```
self.best_fit,self.best_individual = self.__bestFitness()
```

Passo 4: Estrutura de repetição que realiza o processo genético completo. Nesta estrutura ocorre a avaliação do indivíduo.

```
while (np.abs(self.best_fit-last_best_fit) > target) and (generation <
max_generation):
```

Passo 4.1: Seleção, crossover e Mutação.

```
def __newPopulation(self,population_size):

    ##
    .....
    ##

    for i in range(offset_population_index,population_size,2):
        #Selection
        x = self.__selection()
        y = self.__selection()

        #Crossover
        new_individual_x,new_individual_y = self.__crossover3(x,y)

        #Mutation
        new_individual_x = self.__mutation(new_individual_x)
        new_individual_y = self.__mutation(new_individual_y)
```

```
self.best_fit,self.best_individual = self.__bestFitness()
```

Passo 4.3: Retorna ao passo 4.

Valores dos Parâmetros utilizados

Foi replicado um problema já apresentado por outro artigo [1], para se assemelhar melhor com os resultados obtidos pelo artigo alguns parâmetros foram alterados.

Parâmetros genéticos utilizados:

- Número de indivíduos: 50;
- Tamanho do indivíduo (Cromossomo): 22;
- Taxa de mutação: 0.008;
- Taxa de crossover: 0.65;
- Precisão do melhor indivíduo (Target): 0.00005;
- Máximo de geração: 40;

Parâmetros utilizados para cálculos da Função:

- Força (F): 2000 [Newtons];
- Comprimento da barra engastada (L): 300 [mm];
- **Diâmetro maior** (D): ...[mm]; ()
 - Mínimo: 51 [mm];
 - Máximo: 99 [mm];
- **Diâmetro menor** (d): ...[mm];
 - Mínimo: 1 [mm];
 - Máximo: 99 [mm];
- Módulo de Young (E): 2.065e5 [mm]

*** *Parâmetros que variam no algoritmo genético*

Análise dos resultados

Apesar do algoritmo genético ter padrões de funcionamento não foi possível chegar no resultado idêntico ao do artigo, porém foi obtido valores semelhantes.

```
C:\WINDOWS\system32\cmd.exe - python main.py

Generation: 35
Best fit: Individual[0] = 0.00236618

Generation: 36
Best fit: Individual[0] = 0.00236618

Generation: 37
Best fit: Individual[0] = 0.00236618

Generation: 38
Best fit: Individual[0] = 0.00236618

Generation: 39
Best fit: Individual[0] = 0.00236618

Generation: 40
Best fit: Individual[0] = 0.00236618
Solution found: [0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

Solution:
[0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
F=2000
L=300
D=51.2579
d=50.0239
f(F,L,D,d)=0.00236618
```

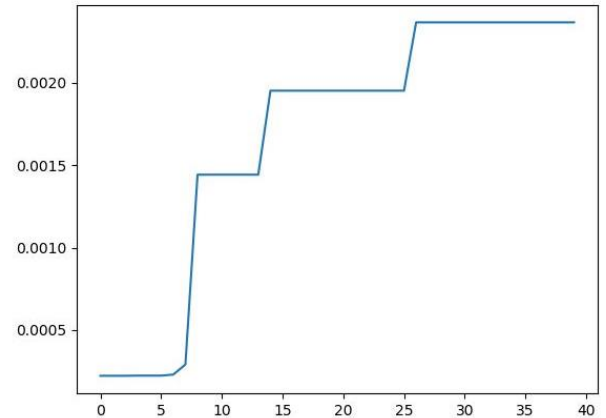


Figura 2 - Resultado 1

```
C:\WINDOWS\system32\cmd.exe - python main.py

Generation: 35
Best fit: Individual[0] = 0.00236618

Generation: 36
Best fit: Individual[0] = 0.00236618

Generation: 37
Best fit: Individual[0] = 0.00236618

Generation: 38
Best fit: Individual[0] = 0.00236618

Generation: 39
Best fit: Individual[0] = 0.00236618

Generation: 40
Best fit: Individual[0] = 0.00236618
Solution found: [0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

Solution:
[0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
F=2000
L=300
D=51.2579
d=50.0239
f(F,L,D,d)=0.00236618
```

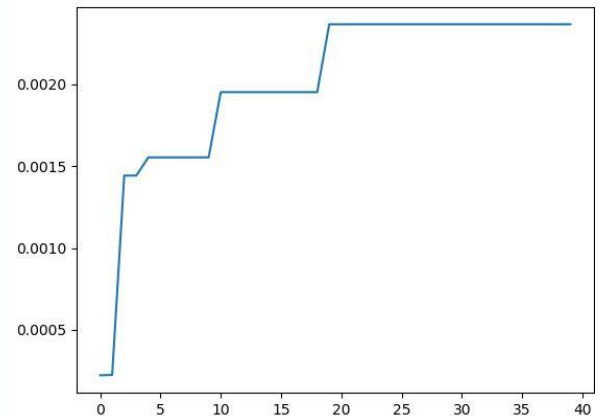


Figura 3 - Resultado 2

Referências

[1] - MONTEIRO, Luiz Fernando Ribas; RODRIGUES, Luiz Guilherme Ramos; DE SOUZA LIMA, Rodolfo. Aplicação de Algoritmos Genéticos em Implementações de Controles Industriais. Disponível em: <https://www.aedb.br/seget/arquivos/artigos15/30522361.pdf>, acessado em 23 de outubro de 2019.

[2] - MORA, Nora Díaz. Apostila de Resistência dos Materiais I. **Universidade Federal de Juiz de Fora. Juiz de Fora**, 2012. Disponível em: http://www.ufjf.br/mac/files/2012/11/Apostila_Res_Mat_outubro_2012-atualizada.pdf, acessado em 20 de outubro de 2019.