1.10 Otimização

Prof. Dr. Sidney Bruce Shiki

e-mail: bruce@ufscar.br

Prof. Dr. Vitor Ramos Franco

e-mail: vrfranco@ufscar.br

UFSCar – Universidade Federal de São Carlos

DEMec - Departamento de Engenharia Mecânica

Conteúdo

- Introdução
- Classificação
- Função objetivo/Função de custo
- Otimização linear
- Otimização não linear
- Ferramenta optimtool
- Noções de otimização global
- Exercícios

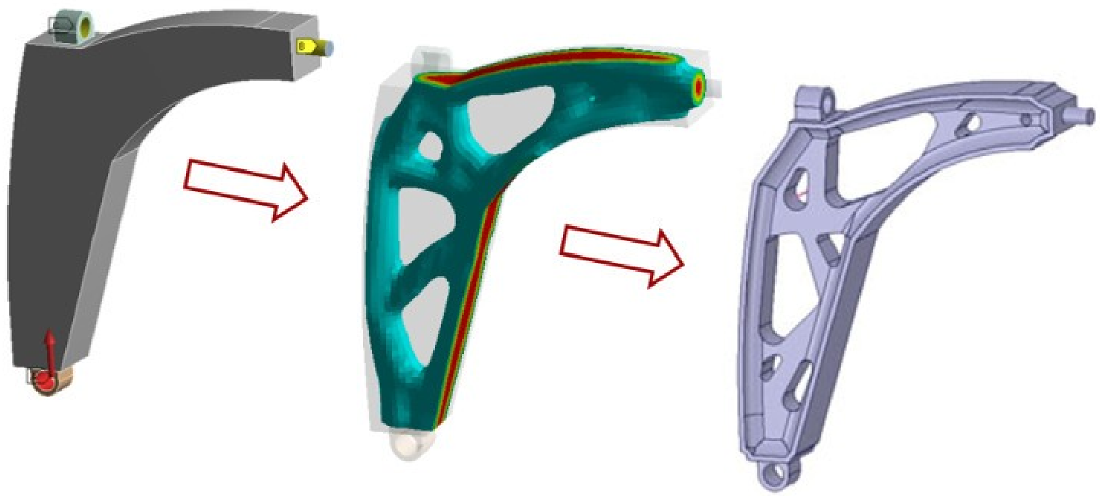
Introdução

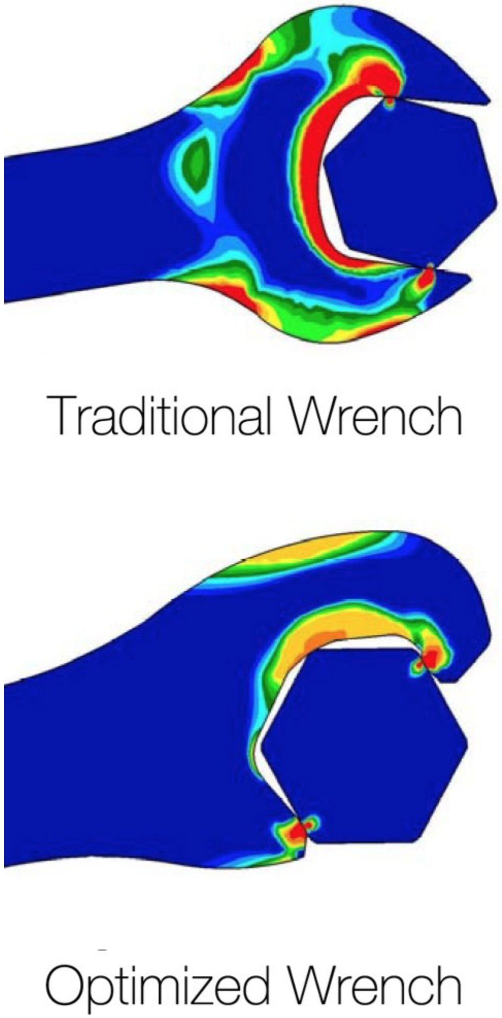
- Técnicas de otimização visam encontrar configurações ótimas para um determinado problema;
- Esses problemas devem ser formulados como problemas de:
  - Maximização: achar ponto de máximo;
  - Minimização: achar ponto de mínimo.
- O problema que será resolvido deve ser representado em uma função objetivo (ou função de custo).

Introdução

- Aplicações:
  - Minimização de custos de um processo/maximização do lucro;
  - Maximização da capacidade de carga de uma estrutura mecânica;
  - Maximização da vida à fadiga;
  - Ajuste de modelos matemáticos à dados experimentais (e.g. ajuste de curvas).

Introdução





Introdução

- Será abordado nesse curso:
  - Formulação de problemas na forma de funções objetivo;
  - Otimização local com e sem restrições em MATLAB;
  - Noções de otimização global.

Classificação

- Otimização:
  - Mono-objetivo: apenas uma função (e.g. minimizar custo de produção)
  - Multi-objetivo: múltiplas funções (e.g. maximizar resistência mecânica minimizando massa total)

Classificação

- Otimização:
  - Linear: função objetivo do problema é linear.

$$\min f(\boldsymbol{x}) = 20x_1 + 64x_2 \quad \begin{cases} 25x_1 + 70x_2 \geq 2100 \\ 0 \leq x_1 \leq 70 \\ 0 \leq x_2 \leq 50 \end{cases}$$

  - Não linear: função objetivo é não linear.

$$\max f(\boldsymbol{x}) = 20{x_1}^2 + 64\sin\left({x_2}^3\right)$$

Classificação

- Otimização:

  Restrita: parâmetros tem restrição de valores

$$\min f(\boldsymbol{x}) = 20{x_1}^2 + 64{x_2}^3 \quad \begin{cases} -70 \le x_1 \le 70 \\ -100 \le x_2 \le 50 \end{cases}$$

  – Irrestrita: parâmetros não tem restrição de valores

$$\min f(\boldsymbol{x}) = 20{x_1}^2 + 64{x_2}^3$$

Função objetivo/Função de custo

- Função objetivo ou função de custo são as funções matemáticas que descrevem o problema a ser otimizado em função de certos parâmetros;

$$\min f(\boldsymbol{x}) = \underbrace{20x_1 + 64x_2}_{\text{2 variáveis}}$$

$$\underbrace{\left\{\begin{array}{l} 25x_1 + 70x_2 \geq 2100 \\ 0 \leq x_1 \leq 70 \\ 0 \leq x_2 \leq 50 \end{array}\right.}_{\text{Restrições do problema}}$$

Problema de minimização

Função objetivo/Função de custo

- Funções objetivos serão declaradas no MATLAB na forma de funções ou matrizes e vetores.

$$\min f(\boldsymbol{x}) = \underbrace{20x_1 + 64x_2}_{\text{2 variáveis}} \qquad \underbrace{\begin{cases} 25x_1 + 70x_2 \geq 2100 \\ 0 \leq x_1 \leq 70 \\ 0 \leq x_2 \leq 50 \end{cases}}_{\text{Restrições do problema}}$$

$\min$ → Problema de minimização

Otimização linear

- Exploraremos para otimização linear a função `linprog()`

```
X = linprog(f,A,b,Aeq,beq,LB,UB);
```

`f` – vetor descrevendo função objetivo
`A` e `b` – inequalidades tal que A*b<=b
`Aeq` e `beq` -igualdades tal que Aeq*beq=beq
`LB` e `UB` – limite inferior (LB) e superior (UB) das variáveis

Otimização linear

- Exemplo de criação – resolver o problema:

$$\min f(\boldsymbol{x}) = 20x_1 + 64x_2 \quad \begin{cases} 25x_1 + 70x_2 \geq 2100 \\ 0 \leq x_1 \leq 70 \\ 0 \leq x_2 \leq 50 \end{cases}$$

```
% Função objetivo
f = [20 64];
A = [-25 -70];b = -2100;          % inequalidades
Aeq = [];beq = [];        % igualdades
LB = [0;0];       % limite inferior
UB = [70;50];     % limite superior
X = linprog(f,A,b,Aeq,beq,LB,UB);
% X contém as soluções x1 e x2
```

Otimização linear

- Exercício 1 –Resolver o problema abaixo:

$$\min f(\boldsymbol{x}) = 0.5x_1 + 2x_2 + 2x_3 \quad \begin{cases} 5x_1 + 10x_2 + 7x_3 \leq 230 \\ x_1 + x_2 + x_3 = 40 \\ -10 \leq x_1 \leq 90 \\ -10 \leq x_2 \leq 90 \\ -10 \leq x_1 \leq 90 \end{cases}$$

- Exercício 2 – Uma viga de concreto armado tem sua rigidez estipulada segundo a seguinte expressão: $f(\boldsymbol{v}) = 7v_1 + 8v_2$

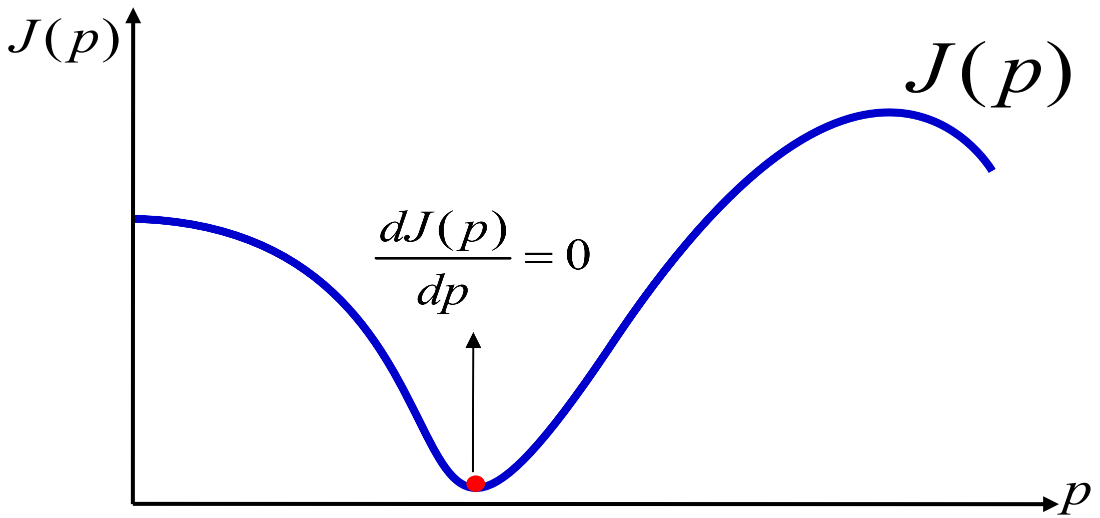
  v_1 – volume do componente 1 [L]

  v_2 – volume do componente 2 [L]

- O litro do componente 1 custa R$ 10,00 reais e do componente 2 custa R$ 80,00 reais, o custo total desses componentes deve ser no máximo de R$ 5000,00 reais para seu projeto;
- O fornecedor dos componentes fornece esses componentes apenas de modo que a soma de seus volumes seja igual à 90 litros;
- O fornecedor pode fornecer no mínimo 5 litros, e no máximo 90 litros de cada componente;
- Calcular os volumes dos componentes 1 e 2 para maximizar a rigidez da viga de concreto dadas as restrições acima.

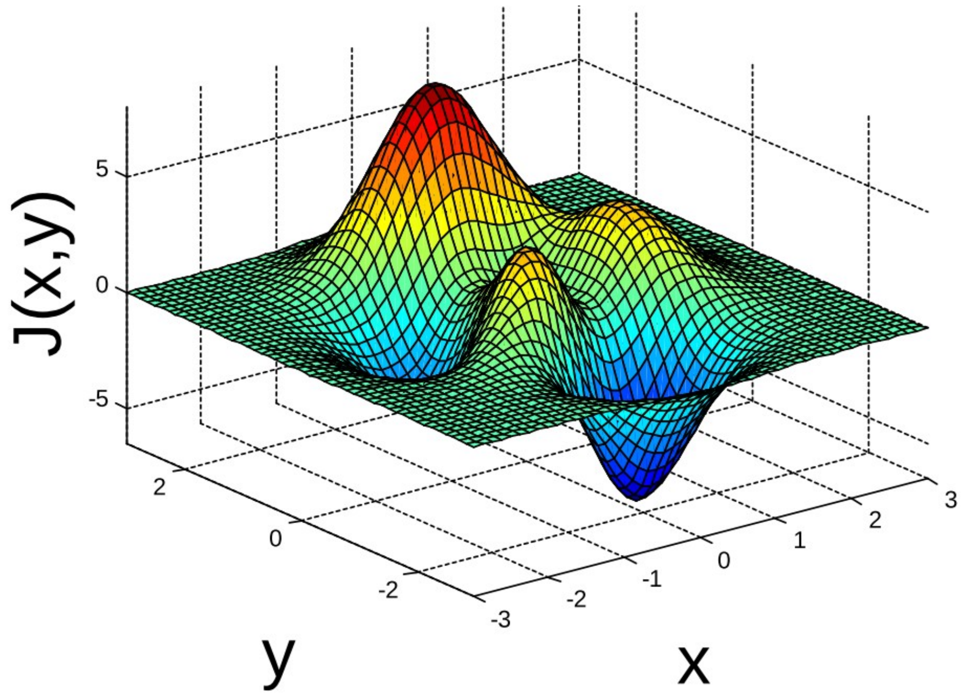
Otimização não linear

- Técnicas de otimização não linear são usadas para se minimizar/maximizar funções objetivo mais gerais;
- Boa parte desses métodos procura um ponto de gradiente (derivada) quase nulo na superfície da mesma (as vezes isso pode ser feito de forma analítica).



Otimização não linear

- Em funções simples é possível achar esses pontos facilmente, mas com múltiplas variáveis ou em funções mais elaboradas isso passa a ser mais complexo.



- Otimização irrestrita (`fminsearch`) – sintaxe:

```
X = fminsearch(fun,x0,options);
```

`fun` – função objetivo
`x0` – estimativa inicial
`options` – opções do algoritmo de otimização

- As opções podem ser configuradas com:

```
options = optimset();
```

- Exemplo de criação – resolver o problema:

$$\min f(\boldsymbol{x}) = 100\left(x_2 - x_1{}^2\right)^2 + \left(1 - x_1\right)^2$$

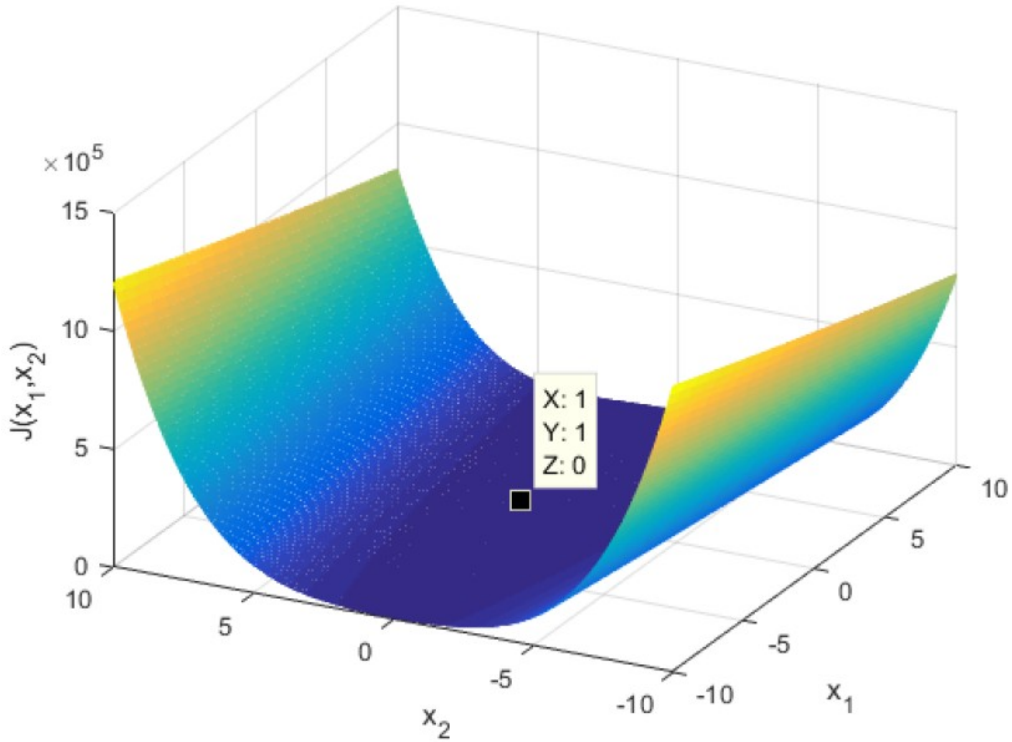
```
% Função objetivo
fun = @(x)100*(x(2) - x(1)^2)^2 + (1 - x(1))^2;
% Opções - display por iteração e plot da FObj
options =
optimset('Display','iter','PlotFcns',@optimplotfval);
x0 = [100,100];        % Chute inicial
x = fminsearch(fun,x0,options);
```

Como garantir que a solução achada é a ótima?

- Exemplo de criação – resolver o problema:

$$\min f(\boldsymbol{x}) = 100\left(x_2 - x_1^{\,2}\right)^2 + \left(1 - x_1\right)^2$$

Como garantir que a solução achada é a ótima?



Mapeie as soluções possíveis!
(busca exaustiva)

- Otimização restrita (`fmincon`) – sintaxe:

```
X = fmincon(fun,x0,A,B,Aeq,Beq,LB,UB,NLCon,options);
```

`fun` – função objetivo
`x0` – estimativa inicial
`A` e `b` – inequalidades tal que A*b<=b
`Aeq` e `beq` -igualdades tal que Aeq*beq=beq
`LB` e `UB` – limite inferior (LB) e superior (UB) das variáveis
`NLCon` – restrições não lineares
`options` – opções do algoritmo de otimização

Otimização não linear

- Exercício 3 – resolver o problema:

$$\min f(\boldsymbol{x}) = x_1^{\,2} + 10x_2^{\,2} - 3x_1x_2 \qquad \begin{cases} 2x_1 + x_2 \geq 2100 \\ x_1 + x_2 \geq -5 \\ -5 \leq x_1 \leq 5 \\ -5 \leq x_2 \leq 5 \end{cases}$$

```
X = fmincon(fun,x0,A,b,Aeq,beq,LB,UB,NLCon,options);
```

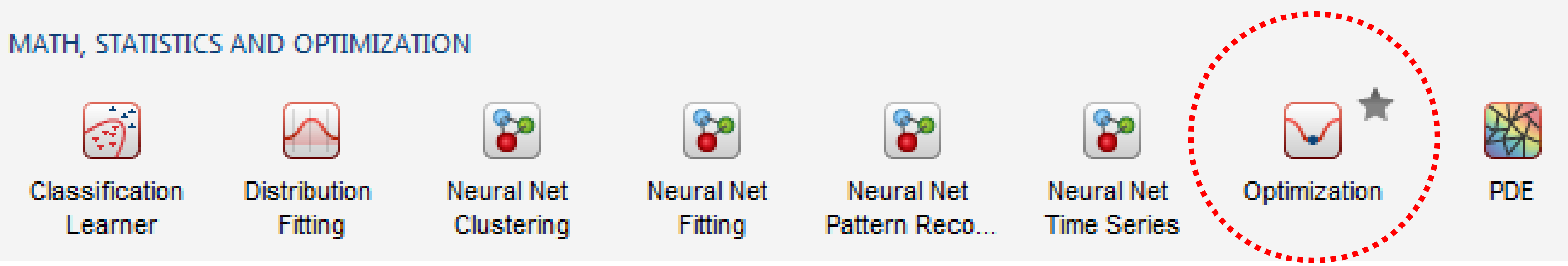
`fun` – função objetivo
`x0` – estimativa inicial
`A` e `b` – inequalidades tal que A*b<=b
`Aeq` e `beq` -igualdades tal que Aeq*beq=beq
`LB` e `UB` – limite inferior (LB) e superior (UB) das variáveis
`NLCon` – restrições não lineares
`options` – opções do algoritmo de otimização

Ferramenta optimtool

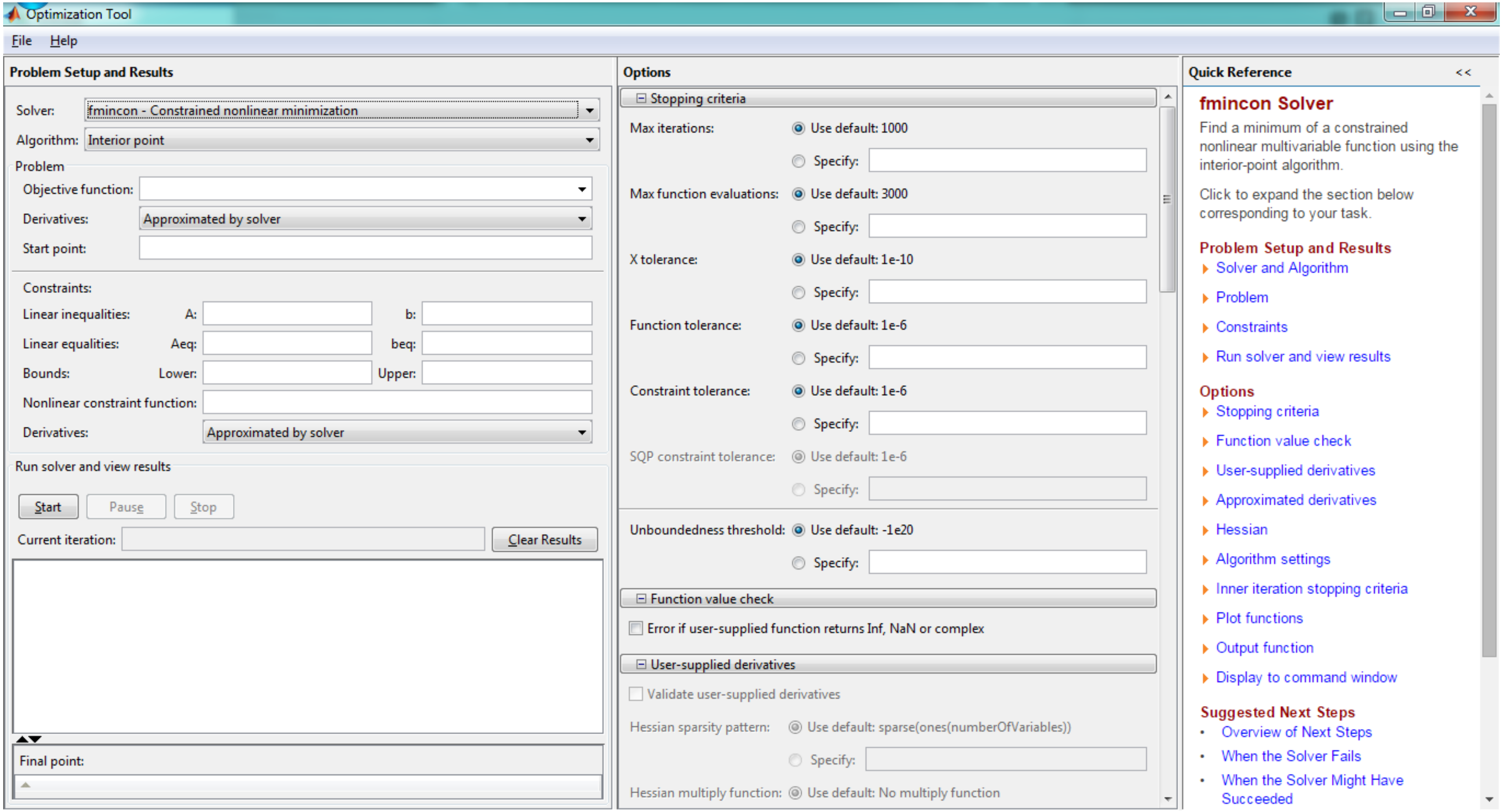
- Assim como para outras toolboxes do MATLAB, existe uma interface gráfica para lidar com problemas de otimização:
- A mesma pode ser chamada por:

```
optimtool;
```

- Ou ainda na aba APPS:

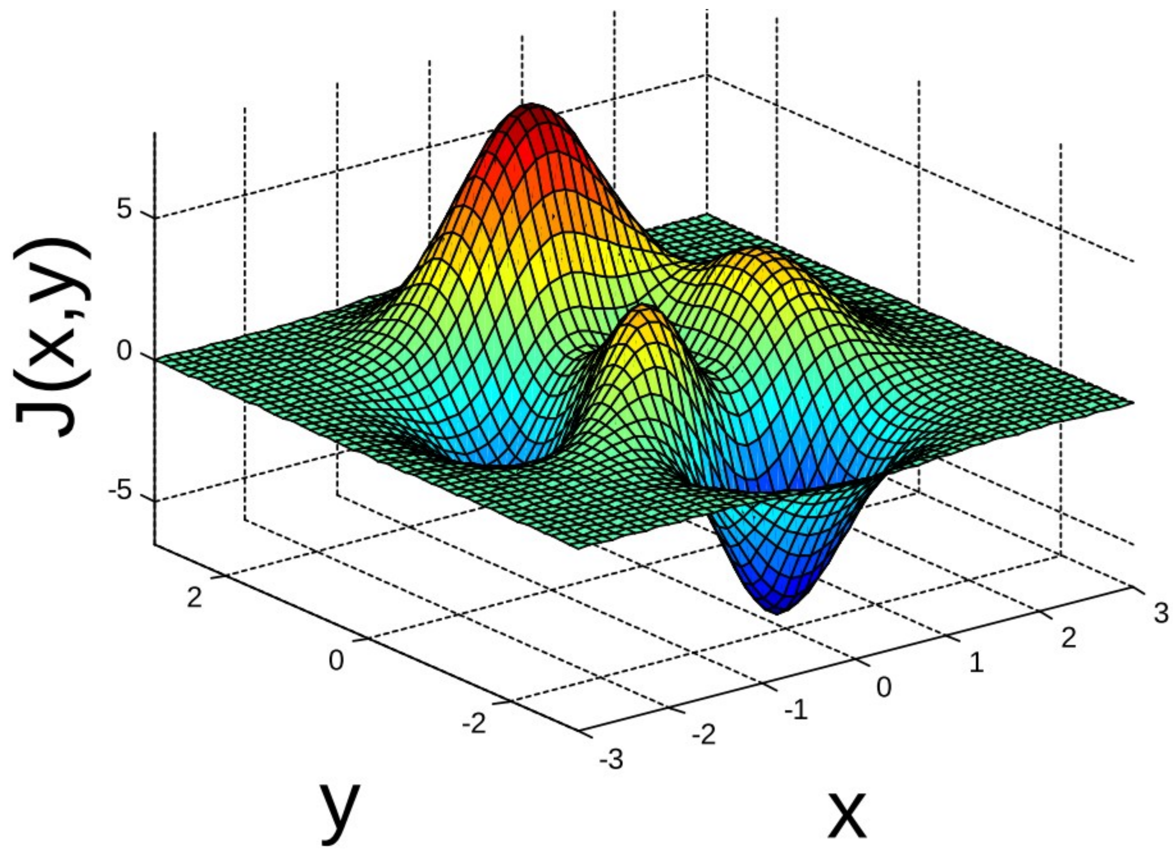


Ferramenta optimtool



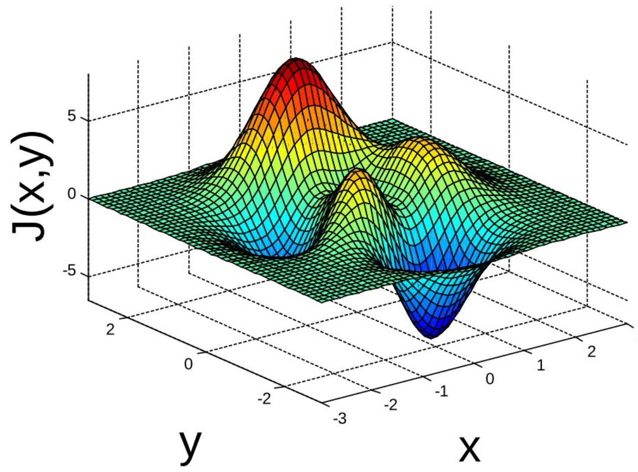
Noções de otimização global

- O que fazer com um problema com vários mínimos locais?



Noções de otimização global

- O que fazer com um problema com vários mínimos locais?
  - Fazer uma busca exaustiva (elevado custo computacional);
  - Usar vários chutes iniciais diferentes para tentar chegar na melhor resposta;
  - Utilizar algoritmos heurísticos (algoritmos genéticos, recozimento simulado, enxame de abelhas, colônia de formigas,...).



Exercícios

- Exercício 4 -

Perguntas ?