UNIVERSIDADE DO ESTADO DO RIO DE JANEIRO
INSTITUTO POLITÉCNICO
CAMPUS REGIONAL DE NOVA FRIBURGO
PÓS-GRADUAÇÃO EM MODELAGEM COMPUTACIONAL

Problemas Inversos

Josiele da Silva Teixeira

Nova Friburgo, 01 de Agosto de 2023

Sumário

- Problemas Diretos e Métodos de Solução:
  - Definição e exemplos de Problemas Diretos;
  - Métodos de Solução de Problemas Diretos: Método das Diferenças Finitas.
- Problemas de Otimização e Problemas Inversos:
  - Problemas Inversos e Exemplos;
  - Problemas de Otimização e Exemplos;
  - Problemas Inversos formulados como problemas de Otimização;
- Métodos de Solução de Problemas Inversos:
  - Métodos Estocáticos;
  - Métodos Baseados em Inferência Bayesiana.

Sumário

- Inteligência Artificial (IA):

  - O objetivo geral da IA é a concepção e implementação de sistemas inteligentes computacionais baseados em teorias e modelos criados para expressar a capacidade cognitiva do ser humano, imitando aspectos do comportamento, tais como: Raciocínio, percepção, aprendizado e adaptação, entre outros.

- Inteligência Computacional (IC):

  - A IC é subconjunto da IA, sendo esta, um conjunto de técnicas, tais como redes neurais, lógica nebulosa e computação evolucionária, que levou a contribuições significativas, permitindo a percepção e a utilização do comportamento inteligente de sistemas de elevado grau de complexidade.

Problema Direto e Inverso

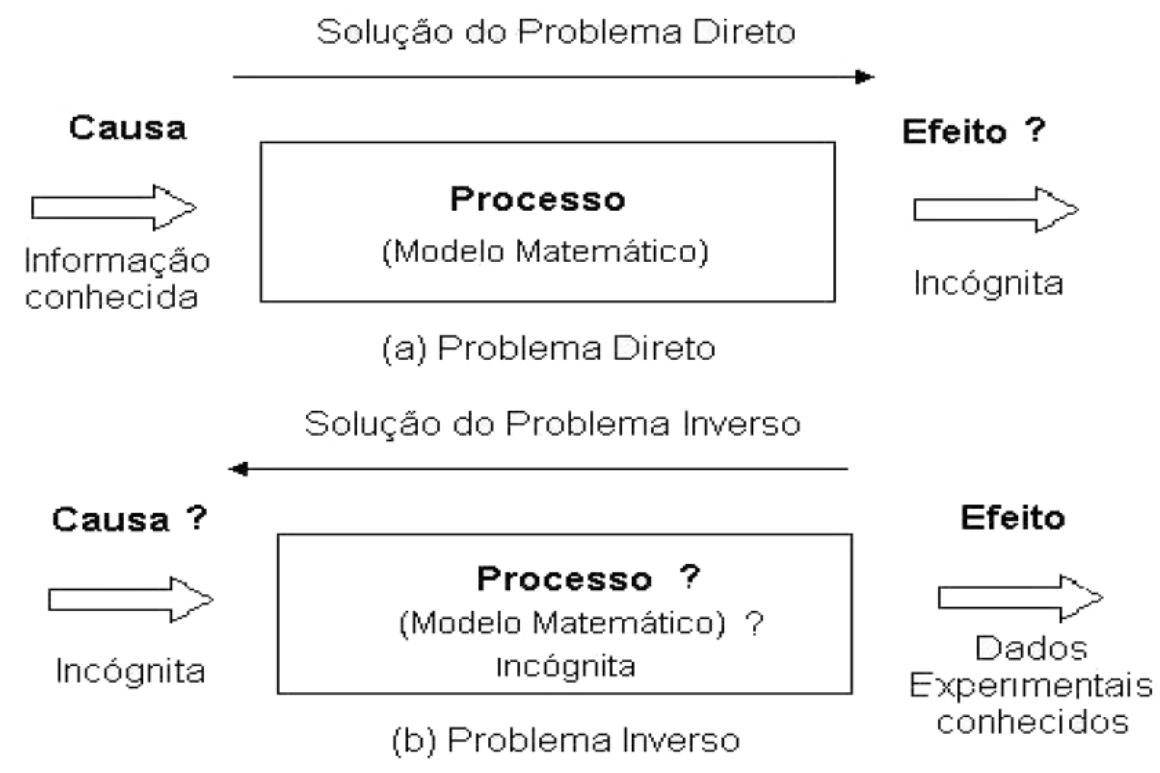


Figura 1: representação esquemática do Problema Direto (a) e do Problema Inverso (b) .

Problemas Inversos: Exemplos

A modelagem do problema direto segue a teoria elementar de flexão de vigas, na qual o deslocamento transversal $y(x,t)$ de um ponto com coordenada x, no eixo central de uma viga de comprimento L, é descrito pela equação diferencial parcial

$$\rho_L \frac{\partial^2 y(x,t)}{\partial t^2} + \frac{\partial^2}{\partial x^2}\left(E(x)I(x)\frac{\partial^2 y(x,t)}{\partial x^2}\right) = f(x,t) \tag{1}$$

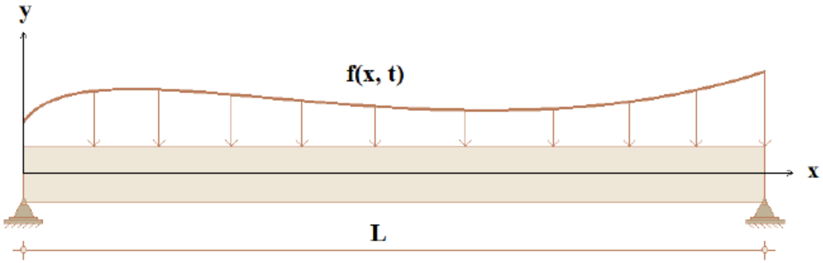
onde ρ_L é a massa específica por unidade de comprimento, $E(x)$ é o módulo de Young, $I(x)$ é o momento de inércia de área da seção transversal da viga e $f(x,t)$ é o carregamento distribuído externo por unidade de comprimento. In the case of a simply supported beam, the boundary conditions are: the transverse deflections and bending moments acting at the edges of the beam are zero, viz.

$$y(0,t) = 0, \qquad y(L,t) = 0 \tag{2}$$

$$E(x)I(x)\frac{\partial^2 y(x,t)}{\partial x^2}\bigg|_{x=0} = 0, \qquad E(x)I(x)\frac{\partial^2 y(x,t)}{\partial x^2}\bigg|_{x=L} = 0 \tag{3}$$

The initial conditions are given by

$$y(x,0) = y_0(x)$$
$$\dot{y}(x,0) = v_0(x)$$



Problemas Inversos: Exemplos

Therefore, the bending stiffness of the beam is defined as

$$E(x)I(x) = \beta(x)E_0I_0 \tag{6}$$

where E_0 and I_0 are, respectively, the nominal Young modulus and the nominal area moment of inertia of the cross section.

A discretização espacial do campo de coesão é realizada pelo Método de Elementos Finitos, onde o vetor de parâmetros de coesão nodal é definido como

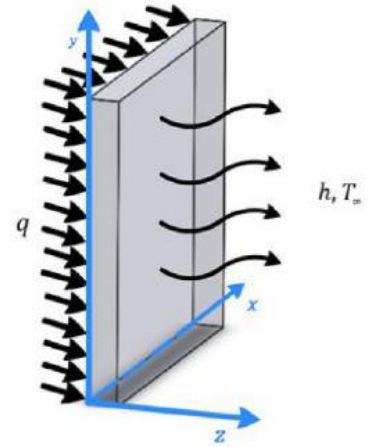
$$\boldsymbol{\beta} = [\beta_1, \beta_2, \ldots, \beta_{n_p}]^T \tag{7}$$

Considerando um sistema com n graus de liberdade, a equação de movimento obtida pelo método de elementos finitos é dada

$$\mathbf{M}\ddot{\mathbf{u}}(t) + \mathbf{D}\dot{\mathbf{u}}(t) + \mathbf{K}(\boldsymbol{\beta})\mathbf{u}(t) = \mathbf{f}(t) \tag{12}$$

where $\mathbf{u}(t)$ is the generalized nodal displacement vector, $\mathbf{M}$, $\mathbf{D}$ and $\mathbf{K}(\boldsymbol{\beta})$ are, respectively, the mass, damping, and stiffness matrices of the structure, and $\mathbf{f}$ is the vector containing the external generalized nodal loads. It is worth emphasizing that in the

Problemas Inversos: Exemplos



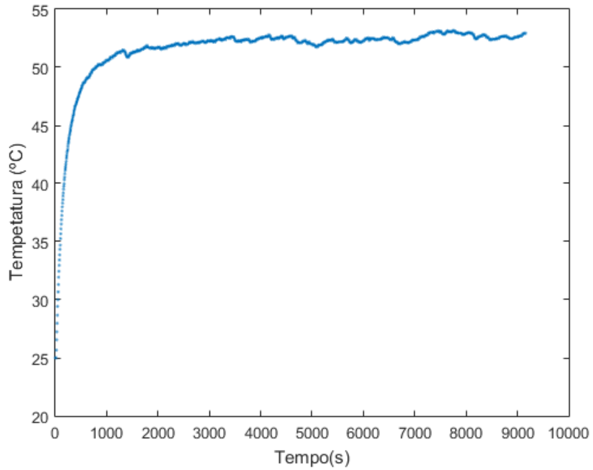


Figura 1 – Representação esquemática do problema de condução de calor

A temperatura média da placa em função do tempo durante o aquecimento é dada por:

$$\begin{cases} T_m(\mathrm{t}) = T_\infty + \dfrac{q'' A_1}{h A_2}\left(1 - \exp\left(\dfrac{-h A_2 t}{\rho_{al} V c_{p_{al}}}\right)\right) & (1.\mathrm{a}) \\ T_m(0) = T_{mi} & (1.\mathrm{b}) \end{cases}$$

onde

$A_1 = 0.0016\ m^2$, $A_2 = 0.00192\ m^2$, $T_\infty = 25^oC$, $h = 22.491\ W/m^2\,K$, $T_{mi} = 25^oC$.

Problemas de Otimização

- O problema de encontrar um conjunto de parâmetros considerados ótimos, ou seja, com o melhor custo/benefício, buscando determinar o mínimo ou máximo de uma Função Objetivo (Função custo), que está sujeita a um conjunto de restrições, é conhecido com Problema de Otimização;

- Como encontrar esses parâmetros?
  - Através de Métodos Determinísticos.
  - Através de Métodos Estocásticos.

Problemas Inversos formulados como Problemas de Otimização

- Como formular uma Função Objetivo?

1. Defina seu vetor de Incógnitas;

$$\boldsymbol{\beta} = [\beta_1, \beta_2, \ldots, \beta_{n_p}]^T \qquad \boldsymbol{Z} = [q'', \rho, c_p]^T$$

2. Parametrize seu problema Direto com relação ao seu vetor de incógnitas;

$$\mathbf{M}\ddot{\mathbf{u}}(t) + \mathbf{D}\dot{\mathbf{u}}(t) + \mathbf{K}(\boldsymbol{\beta})\mathbf{u}(t) = \mathbf{f}(t) \qquad T_m(\boldsymbol{Z}, \mathrm{t}) = T_\infty + \frac{q'' A_1}{h A_2}\left(1 - \exp\left(\frac{-h A_2 t}{\rho_{al} V c_{p_{al}}}\right)\right)$$

Problemas Inversos formulados como Problemas de Otimização

3. Escolha uma métrica de comparação entre os dados medidos experimentalmente e os dados calculados pelo modelo, por exemplo:

$$\sum_{i=1}^{N_d} [I_i(\boldsymbol{Z}) - Y_i]; \qquad \sum_{i=1}^{N_d} [I_i(\boldsymbol{Z}) - Y_i]^2; \qquad \sum_{i=1}^{N_d} \frac{[I_i(\boldsymbol{Z}) - Y_i]^2}{Y_i}$$

4. Defina a sua Função Objetivo a partir da métrica escolhida

$$\boldsymbol{F}_{\mathbf{obj}}(\boldsymbol{Z}) = \sum_{i=1}^{N_d} \frac{[I_i(\boldsymbol{Z}) - Y_i]^2}{Y_i}$$

Problemas Inversos formulados como Problemas de Otimização

- Formulação do Problema Inverso como um Problema de Otimização:

$$Z^* = \min_{Z} F_{\mathbf{obj}}(Z)$$

Métodos Estocásticos

Luus-Jaakola (LJ)

Luus-Jaakola

- Proposto por R. Luus e T. H. I. Jaakola em 1973 para resolver problemas de programação não-linear, dentre eles problemas de engenharia química;

- **Objetivo:** Desenvolver um procedimento simples de otimização aplicável para resolver problemas de programação não-linear, de modo que qualquer pessoa possa utilizá-lo;

Luus-Jaakola

- Ao longo dos anos, modificações no algoritmo de Luus-Jaakoa foram sugeridas por diversos autores, tais como Wang e Luus (1978), Luus *et al.* (1995), Wang e Luus (1997), Luus (1998), Michinev (2000), Jezowski e Bochenek (2002).

- Atualmente, o algoritmo comumente utilizado para o referido método é dado por:

Luus-Jaakola

Pseudocódigo – Luus-Jaakola
Escolha um espaço de busca inicial $r^{(0)}$, um número de loops externos n_{out} e um número de loops internos n_{int}.
Escolha um coeficiente de contração ε.
Gere aleatoriamente uma solução inicial X^*;
Para i=1 até n_{out}
Para i=1 até n_{int}
$X^{(j)} = X^* + R^{(j)}\, r^{(i-1)}$, onde $R^{(j)}$ é uma matriz diagonal de números aleatórios entre -0.5 e 0.5.
Se $Fitness(X^{(j)}) < Fitness(X^*)$, **então**
$X^* = X^{(j)}$
Fim Se
Fim Para
$r^{(i)} = (1-\varepsilon)\, r^{(i-1)}$
Fim Para

Luus-Jaakola

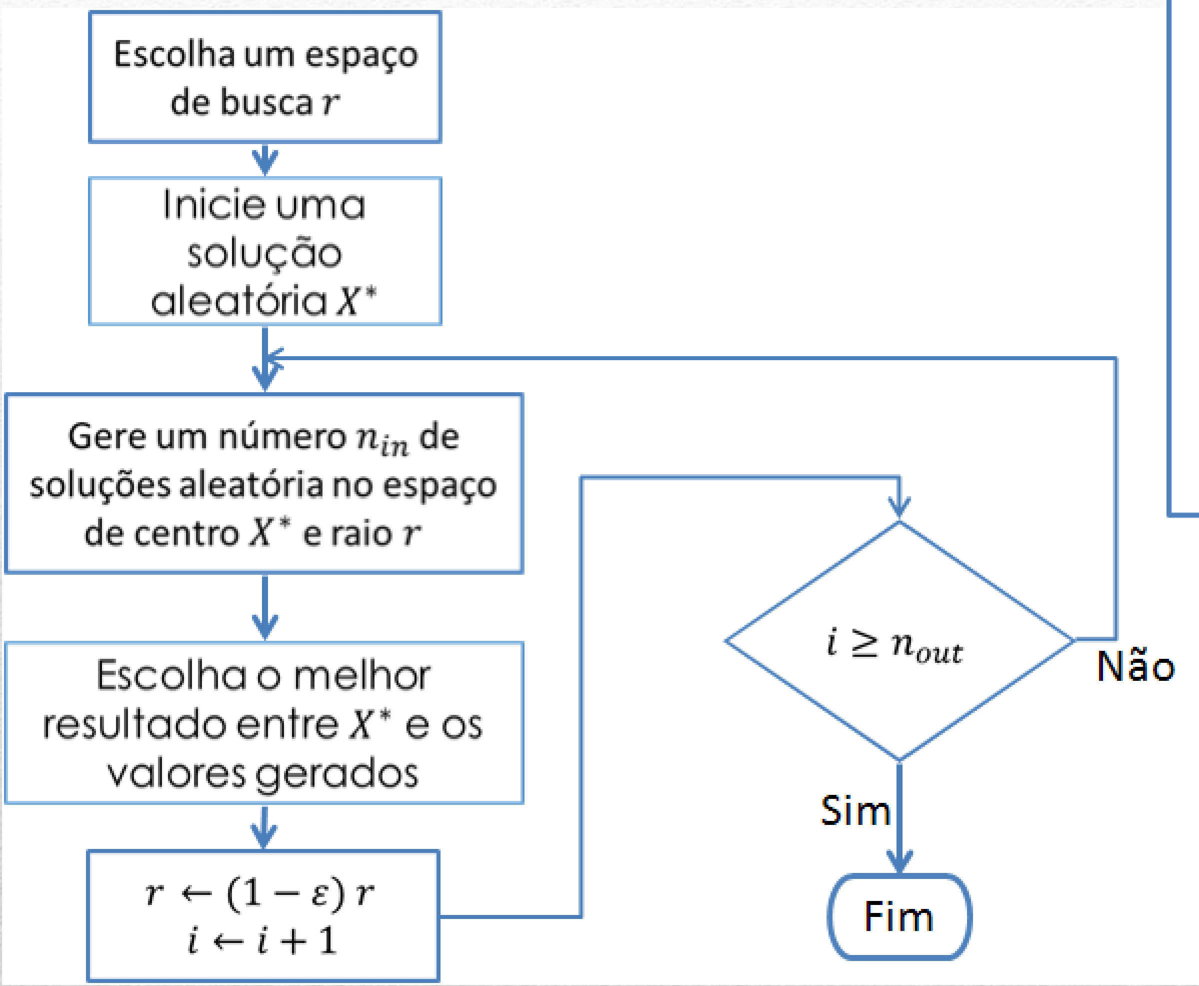
- **Considerações:**

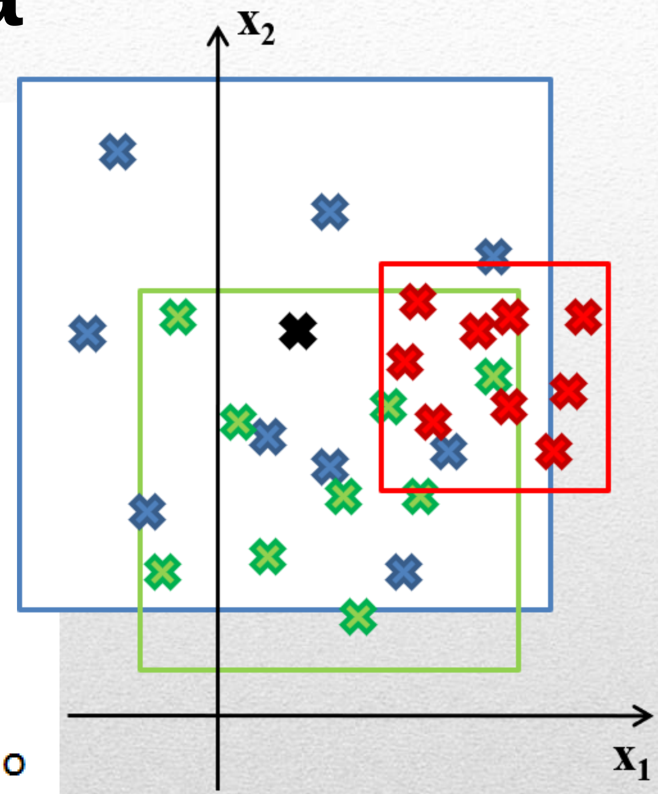
- Podemos definir o valor de r_i como sendo a amplitude do intervalo de busca de x_i (informado nas restrições);

- Após gerarmos uma estimativa $X^{(j)}$ verificamos se a mesma atende as restrições do problema;

- Caso a estimativa esteja fora do intervalo de busca original, podemos defini-la como o limite inferior ou superior deste, por exemplo.

Luus-Jaakola





Métodos Estocásticos

Algoritmos Genéticos
(GA)

Algoritmos Genéticos

- Os Algoritmos Genéticos foram desenvolvidos inicialmente por John Holland (1975) a partir da Teoria da Evolução das espécies inicializadas por Charles Darwin (1859), onde os indivíduos mais aptos sobrevivem;

- Em computação, os indivíduos mais aptos são aqueles que produzem resultados mais próximos da solução do problema proposto.

Algoritmos Genéticos

Os GA se baseiam em três operações básicas:

1. **Mutação**: é responsável por inserir uma alteração aleatória nos membros da população.

2. **Recombinação (Crossover)**: é a operação de troca aleatória de subsequências entre dois cromossomos (membros) da população, para criar dois filhos;

3. **Seleção**: é a operação responsável por escolher os cromossomos mais aptos, que serão utilizados na próxima geração;

Algoritmos Genéticos

Implementação:

- **Operadores Genéticos:**
  - Cruzamento
  - Mutação

- **Parâmetros:**
  - Tamanho da População
  - Taxa de Cruzamento
  - Taxa de Mutação
  - Cálculo da Aptidão (*Fitness*)
  - Tamanho do Cromossomo
  - Número Máximo de Gerações

Algoritmos Genéticos

Pseudocódigo – Algoritmos Genéticos
t=0;
Crie a População Inicial *P(t)*;
Enquanto o critério de parada não for satisfeito, **faça**
Avalie a população *P(t)*;
Selecione os indivíduos reprodutores de *P(t)*;
Gere uma nova população *P(t+1)*;
t=*t*+*1*;
Fim Enquanto

Métodos Estocásticos

Evolução Diferencial
(ED)

Evolução Diferencial

Journal of Global Optimization **11**: 341–359, 1997. 341

Differential Evolution – A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces

RAINER STORN
Siemens AG, ZFE T SN2, Otto-Hahn Ring 6, D-81739 Muenchen, Germany.
(e-mail:rainer.storn@mchp.siemens.de)

KENNETH PRICE
836 Owl Circle, Vacaville, CA 95687, U.S.A. (email: kprice@solano.community.net)

(Received: 20 March 1996; accepted: 19 November 1996)

Abstract. A new heuristic approach for minimizing possibly nonlinear and non-differentiable continuous space functions is presented. By means of an extensive testbed it is demonstrated that the new method converges faster and with more certainty than many other acclaimed global optimization methods. The new method requires few control variables, is robust, easy to use, and lends itself very well to parallel computation.

Key words: Stochastic optimization, nonlinear optimization, global optimization, genetic algorithm, evolution strategy.

Evolução Diferencial

O DE se baseia em três operações básicas:

1. **Mutação**: é responsável por inserir uma alteração aleatória nos membros da população.

2. **Cruzamento (Crossover)**: nessa operação escolhe-se aleatoriamente as informações dos pais que serão transmitidos aos filhos;

3. **Seleção**: é a operação responsável por escolher os cromossomos mais aptos, que serão utilizados na próxima geração;

Evolução Diferencial

INICIALIZAÇÃO:

$$\{\boldsymbol{\beta}^{1,0}, \cdots, \boldsymbol{\beta}^{N_{pop},0}\} \quad \text{População inicial}$$

$$\boldsymbol{\beta}^{i,0} = \begin{bmatrix} \beta_1^{i,0} \\ \beta_2^{i,0} \\ \vdots \\ \beta_{n_p}^{i,0} \end{bmatrix}$$

$$\{\boldsymbol{\beta}^{i,k}; i = 1, 2, ..., N_{pop}; k = 0, 1, 2, ..., N_G\}$$

$$\beta_j^{i,k} = \beta_L^{i,k} + rand\,(\beta_U^{i,k} - \beta_L^{i,k})$$

Evolução Diferencial

1. **Mutação**: é responsável por inserir uma alteração aleatória nos membros da população.

população mutante

$$\mathbf{v}_G = \{\mathbf{v}^{i,k}\}$$

$$\mathbf{v}^{i,k} = \boldsymbol{\beta}^{r_1,k} + F(\boldsymbol{\beta}^{r_2,k} - \boldsymbol{\beta}^{r_3,k})$$

r_1, r_2, r_3 e i diferentes entre si

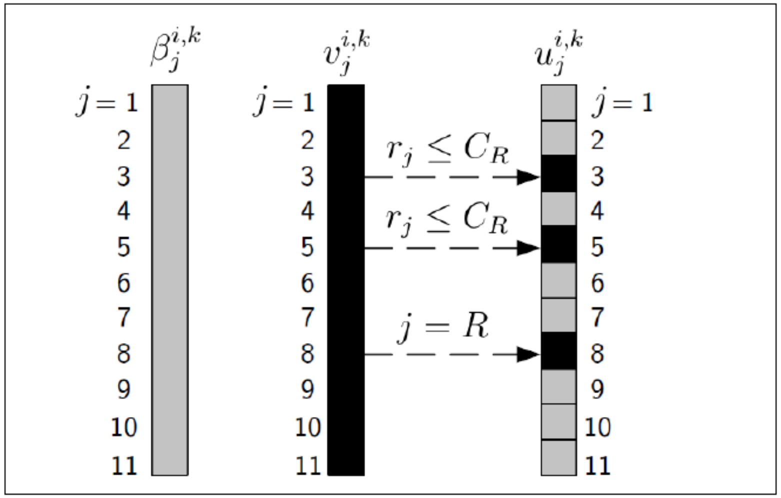
Evolução Diferencial

2. Cruzamento (Crossover): escolhe-se aleatoriamente as informações dos pais que serão transmitidos aos filhos;

O vetor tentativa (trial) é obtido pelas regras a seguir

$$u_j^{i,k} = \begin{cases} v_j^{i,k}, \text{se} \quad r_j \leq C_R \text{ ou } j = R, \\ \beta_j^{i,k}, \text{caso contrário.} \end{cases}$$

Exemplo do processo do Crossover



Evolução Diferencial

3. Seleção: é a operação responsável por escolher os cromossomos mais aptos, que serão utilizados na próxima geração;

$$\beta^{i,k+1} = \begin{cases} u^{i,k} & \text{se } f_{obj}(u^{i,k}) \leq f_{obj}(\beta^{i,k}) \\ \beta^{i,k} & \text{caso contrário} \end{cases}$$

Métodos Estocásticos

Algoritmo de Vaga-lumes
(FireFly)

FireFly

O Firefly, desenvolvido por Yang (2008), Yang (2009), foi baseado no comportamento dos vaga-lumes:

1. Todos os vaga-lumes são unissex, de modo que um vaga-lume é atraído por outros vaga-lumes, independentemente do sexo.

2. A atratividade é proporcional ao brilho, portanto, para quaisquer dois vaga-lumes "piscantes", o menos brilhante seguirá o mais brilhante.

- A atratividade é proporcional ao brilho e ambos diminuem à medida que a distância aumenta. Se ninguém é mais brilhante que um vaga-lume em particular, ele se move aleatoriamente;

FireFly

Como a atratividade de um vaga-lume é proporcional à intensidade da luz observada pelos vaga-lumes adjacentes, pode-se definir a atratividade de um vaga-lume por

$$\beta = \beta_0 e^{-\gamma r^2}$$

β is the attractiveness

β_0 is the attractiveness at $r = 0$

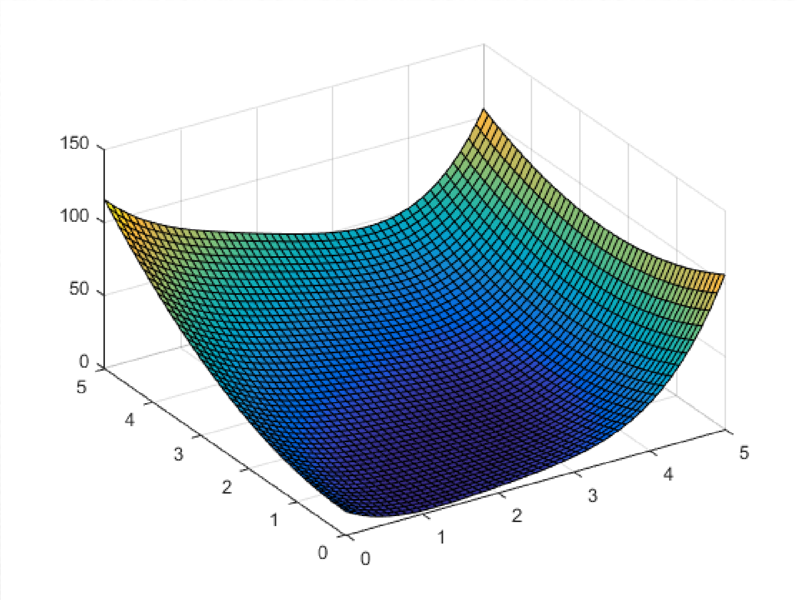
γ is the light absorption coefficient

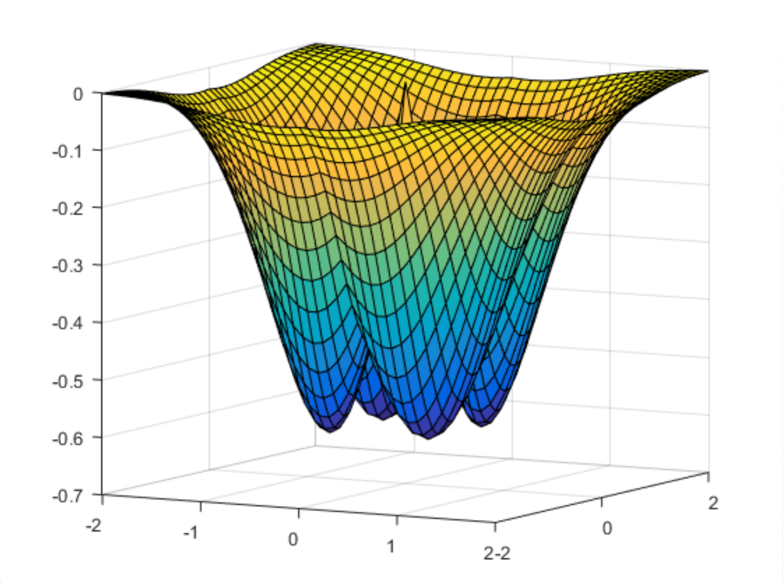
FireFly

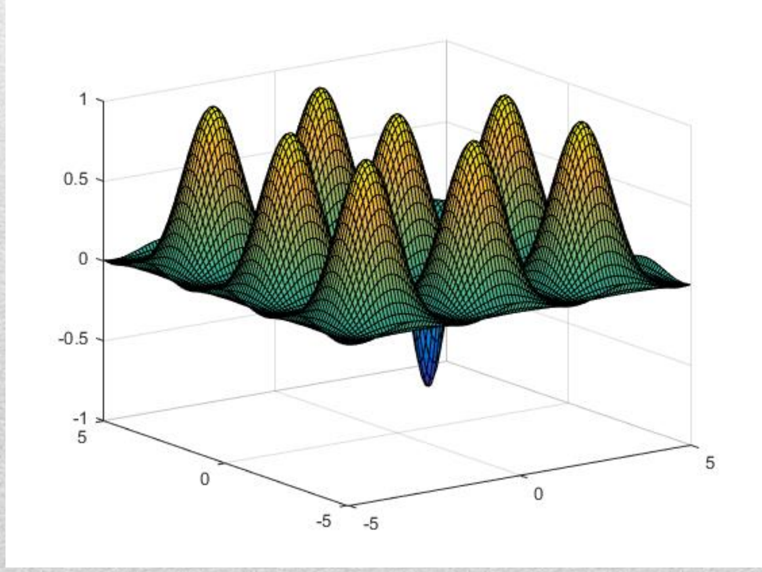
Firefly Algorithm

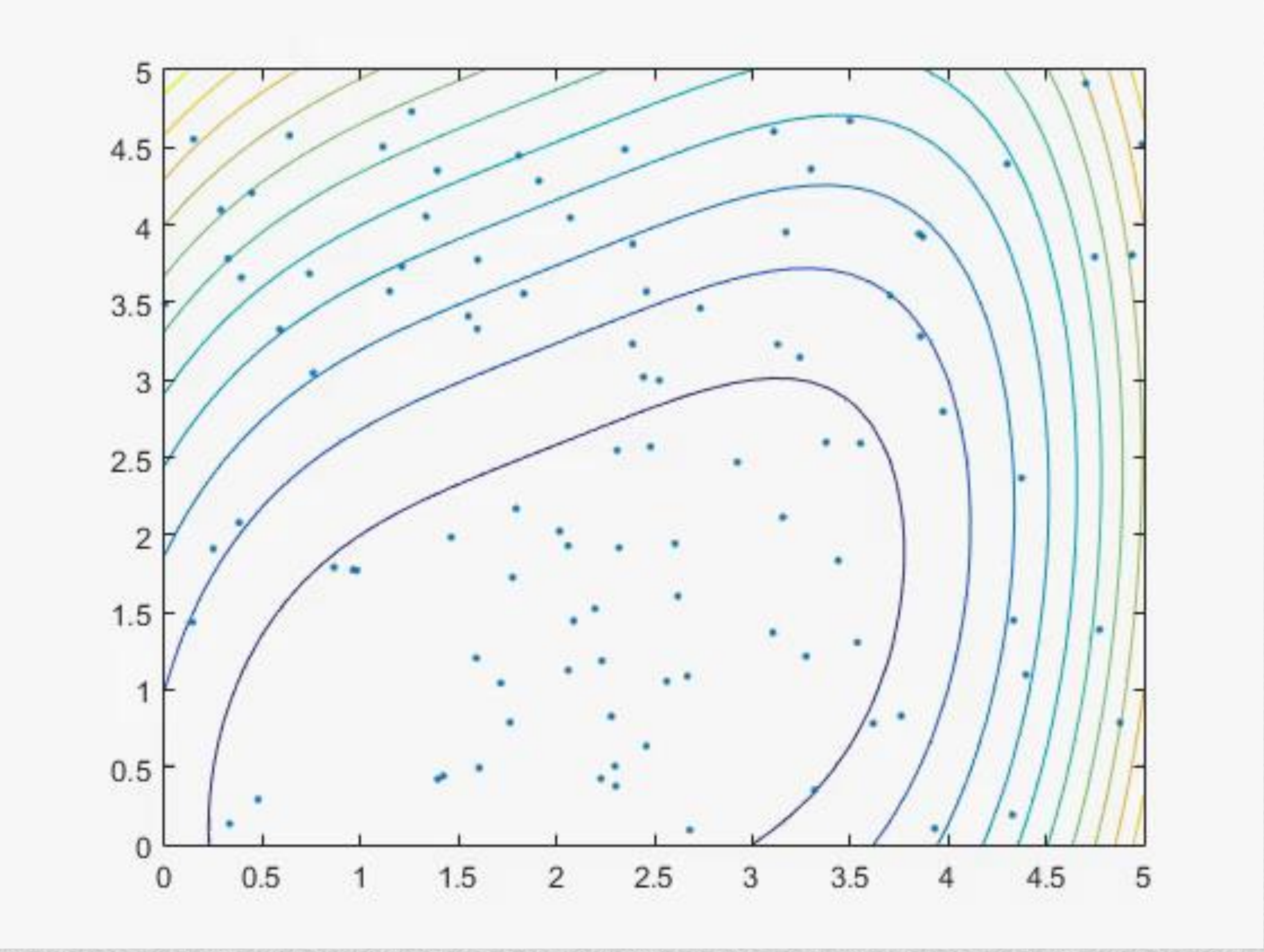
Objective function $f(\mathbf{x})$, $\quad \mathbf{x} = (x_1, ..., x_d)^T$
Initialize a population of fireflies $\mathbf{x}_i$ $(i = 1, 2, ..., n)$
Define light absorption coefficient γ
while ($t <$MaxGeneration)
for $i = 1 : n$ all n fireflies
  for $j = 1 : i$ all n fireflies
    Light intensity I_i at $\mathbf{x}_i$ is determined by $f(\mathbf{x}_i)$
    if $(I_j > I_i)$
    Move firefly i towards j in all d dimensions
    end if
    Attractiveness varies with distance r via $\exp[-\gamma r]$
    Evaluate new solutions and update light intensity
  end for j
end for i
Rank the fireflies and find the current best
end while
Postprocess results and visualization

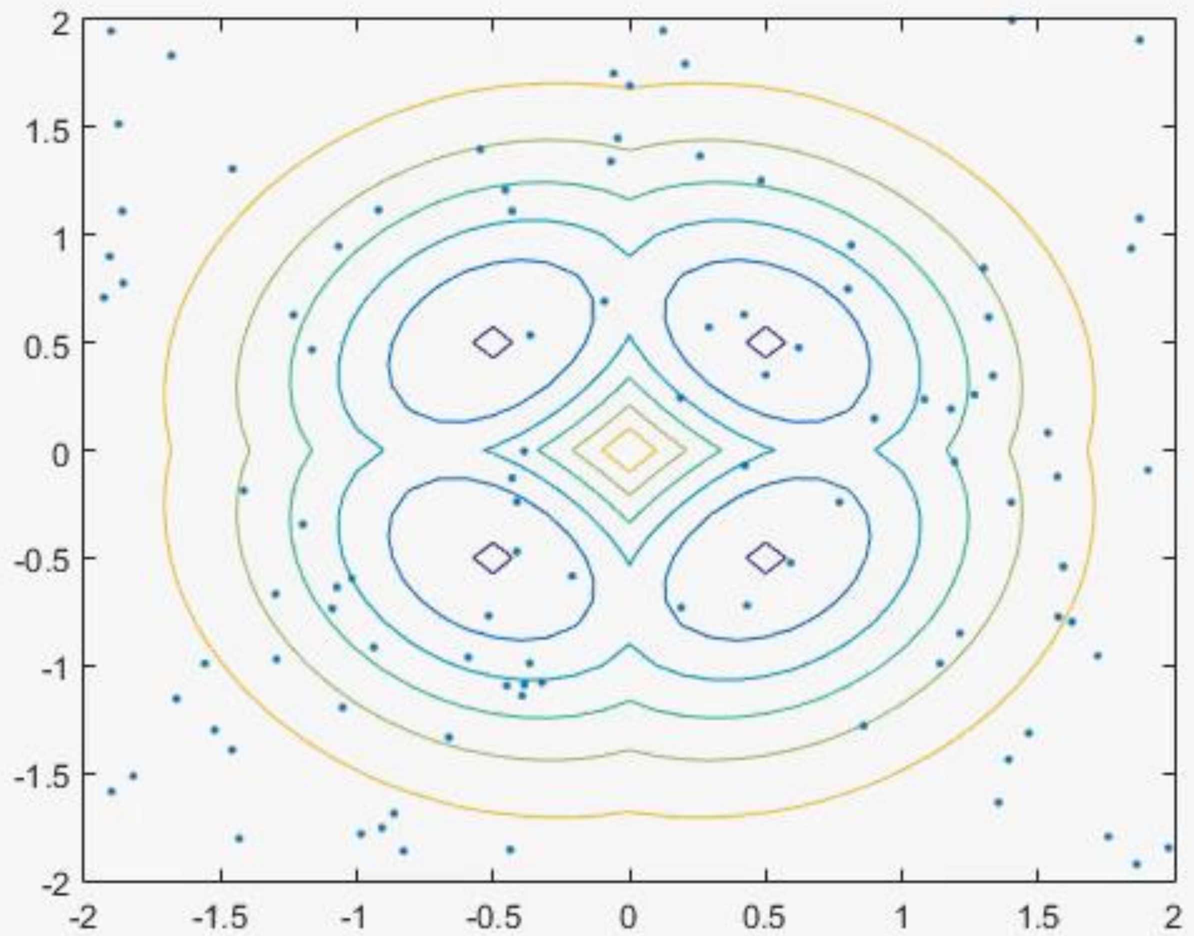
$$r_{ij} = ||\mathbf{x}_i - \mathbf{x}_j||_2 \qquad \mathbf{x}_i = \mathbf{x}_i + \beta_0 e^{-\gamma r_{ij}^2} (\mathbf{x}_j - \mathbf{x}_i) + \alpha \boldsymbol{\epsilon}_i$$

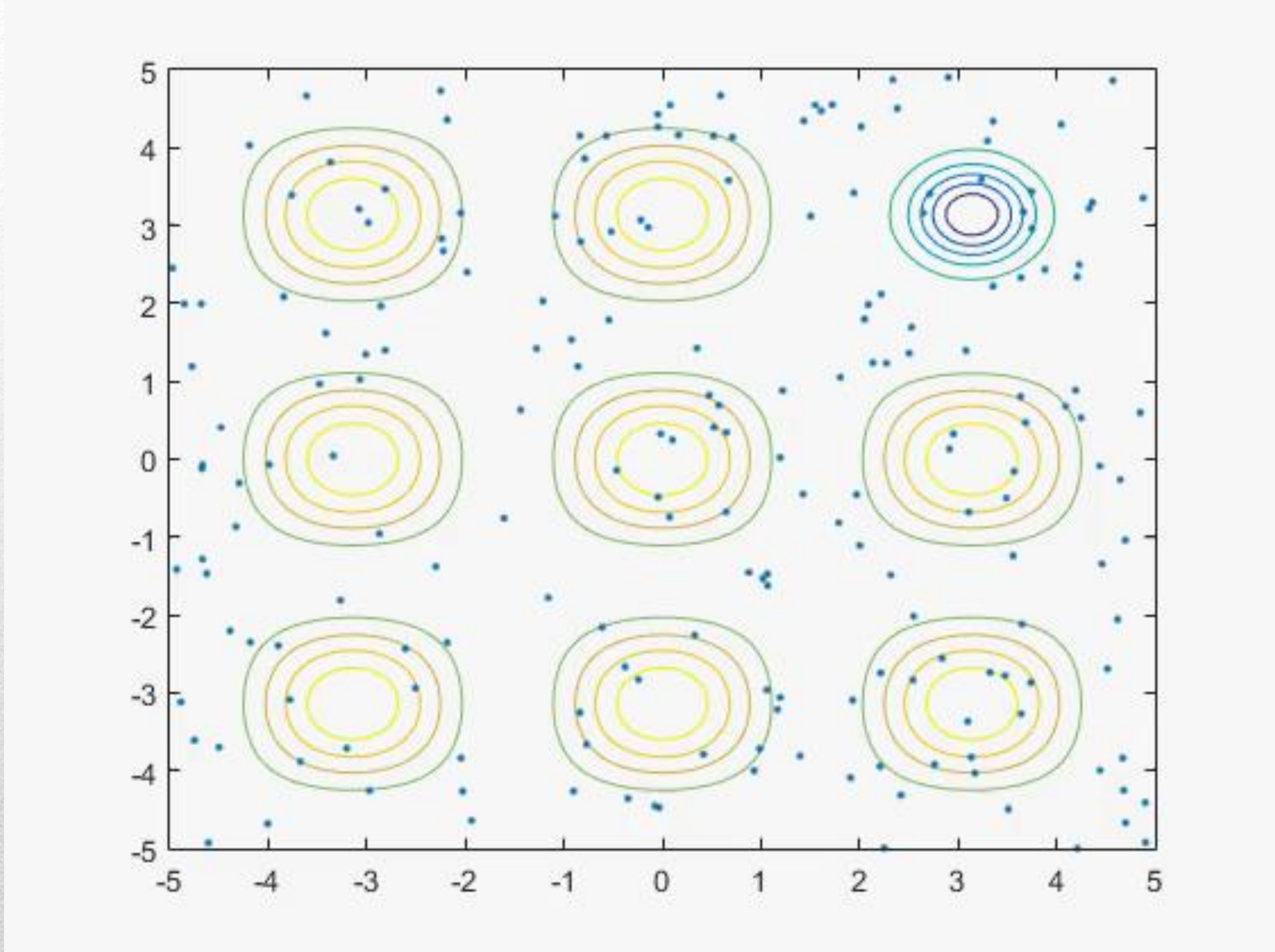

150
100
50
0
5
4
3
2
1
0
0
1
2
3
4
5


0
-0.1
-0.2
-0.3
-0.4
-0.5
-0.6
-0.7
-2
-1
0
1
2-2
0
2


1
0.5
0
-0.5
-1
5
0
-5
-5
0
5


5
4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5


2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-2
-1.5
-1
-0.5
0
0.5
1
1.5
2



5
4
3
2
1
0
-1
-2
-3
-4
-5
-5
-4
-3
-2
-1
0
1
2
3
4
5