Trabalho prático I:

Métodos de Pesquisa da Computação Natural

Elaborado por

Cátia Machado nº 64223

Felícia Madureira nº 65120

Inteligência Artificial

Paulo Moura Oliveira

# INTRODUÇÃO

No âmbito da unidade curricular de Inteligência Artificial, foi-nos proposto o trabalho prático nº 1 que visa a aplicação dos Métodos de Pesquisa de Computação Natural.

É nos dada uma função bidimensional para estudo e, através do algoritmo de Simulated Annealing (SA), iremos procurar encontrar o seu mínimo global.

Na parte inicial deste trabalho, serão introduzidos alguns conceitos relativos a este algoritmo, bem como algumas das suas aplicações práticas. Posteriormente, serão apresentados os resultados e gráficos obtidos que nos permite concluir qual o mínimo da função em estudo.

Em anexo, serão apresentados os códigos efetuados no contexto deste trabalho.

# SIMULATED aNNEALING

* 1. **O que é?**

O Simulated Annealing é um algoritmo motivado por uma analogia ao recozimento de metais. Simula esse processo através da diminuição gradual da temperatura do sistema até esta convergir a um estado estável, congelado.

* 1. **Quais as técnicas em que se baseia?**

Para resolver o problema de ficarmos presos num mínimo local, o SA faz uso de duas técnicas: Algoritmo Metropolis e diminuição da temperatura.

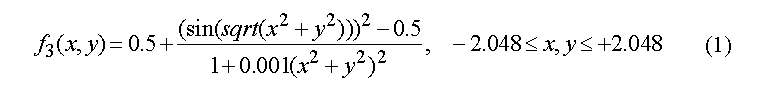
Na primeira técnica, algumas soluções que não diminuem o caminho percorrido, são aceites para explorar o espaço de soluções. A escolha de um estado pior depende de duas variáveis: a diferença entre o estado atual e o próximo estado e da Temperatura atual. Quanto maior a diferença entre o estado atual e o próximo estado, menor a possibilidade dele ser selecionado, e quanto maior a temperatura, maior a possibilidade de um estado pior ser escolhido. Esta estratégia consiste em permitir movimentos piores a altas temperaturas, que são geridos através de uma lei probabilística: p (t) = e^(-∆E(t) / T). Se a probabilidade for maior que um número aleatório entre 0 e 1 então a solução pior é aceite. À medida que a temperatura diminui, os movimentos piores são reduzidos.

Na segunda, após muitas trocas e observando que a função de custo diminui muito lentamente, diminui-se a temperatura, limitando o número de más trocas. Assim, se a temperatura for diminuída muitas vezes, pode-se optar por aceitar somente boas trocas e encontrar o mínimo local.

* 1. **Quais as suas aplicações?**

Utilizado para resolver problemas NP-completos, tais como o problema do caixeiro viajante. É uma técnica que muito dificilmente encontra a solução ótima, mas é frequentemente uma solução boa, mesmo na presença de algum ruído.

# Resolução do Problema com Simulated Annealing

* 1.  **Problema**: Encontrar o mínimo global da função (1):
  2. **Solução implementada** pelo Algoritmo Simulated Annealing, codificado na ferramenta MATLAB.
  3. **Resultados expectáveis**

A função apresenta 4 mínimos locais, localizados em (-2,-2),(2,2),(-2,2),(2,-2).

O mínimo global encontrado deverá estar localizado em (0,0).

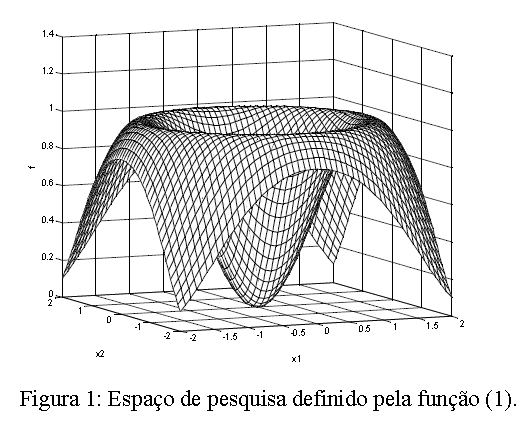
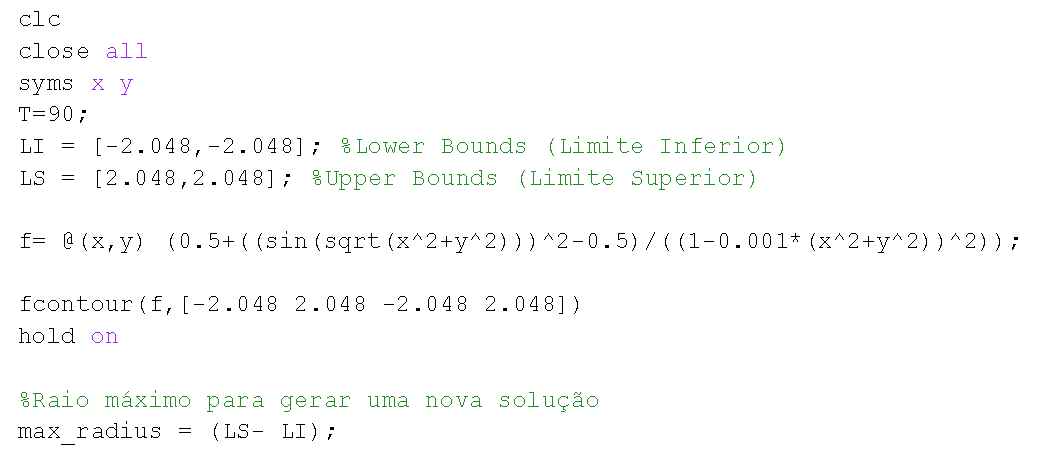
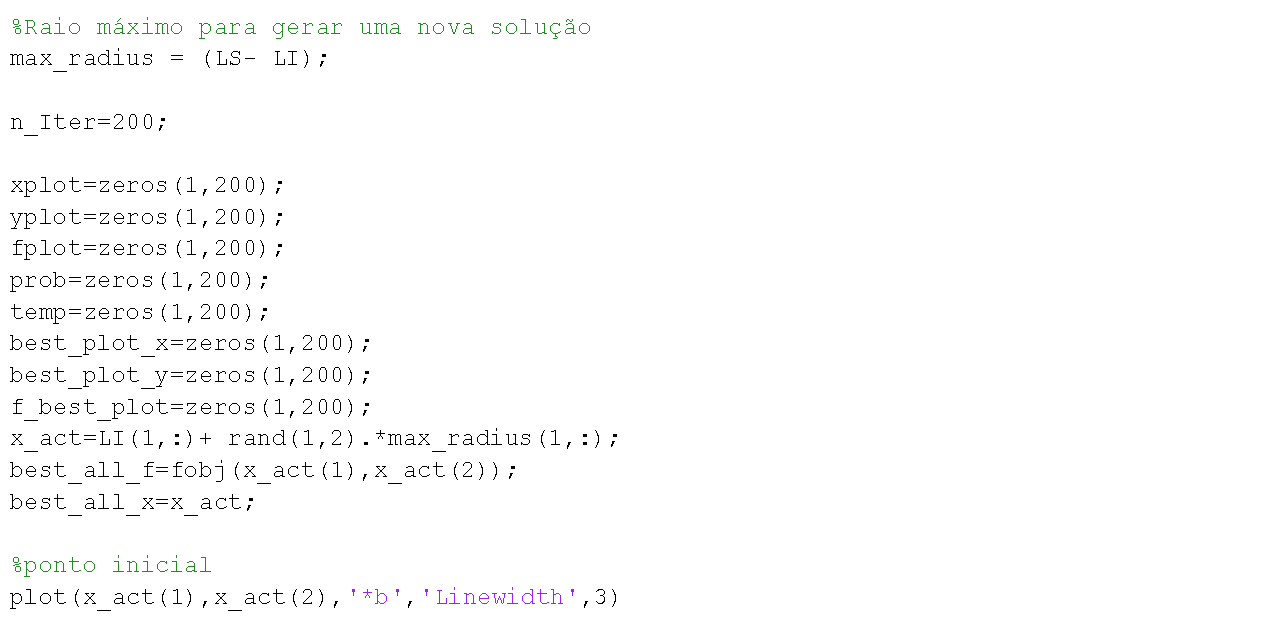


Figura 1 - Espaço de pesquisa definido pela função (1)

* 1. **Apresentação e análise do código utilizado**
     1. **Inicialização**



Nesta fase inicial, inicializa-se a temperatura a 90 graus (testaram-se vários valores iniciais de temperatura, sendo este o que produziu gráficos e resultados mais eficazes); definem-se os limites superiores e inferiores do gráfico onde se vai desenhar a função (estabelece-se a escala da figura), declara-se a função proposta; desenha-se a mesma com recurso à função *fcontour*, obtendo-se o gráfico da função.

Define-se um max\_radius, em que os pontos se podem movimentar, para que estes não saíam dos limites estabelecidos. Estabelece-se um n\_Iter=200, como critério de paragem para sair do ciclo ao atingir as 200 iterações.

De modo a que as variáveis que armazenam valores atuais da localização dos pontos não alterem o tamanho a cada iteração, inicializam-se como arrays de tamanho fixo (1 linha e 200 colunas – vetor linha), com as entradas todas a zero.

Inicializa-se aleatoriamente o x\_act (posição atual da partícula), obtendo-se um valor aleatório com x entre (-2,2) e y entre (-2,2). O x\_act(1) corresponde ao valor da coordenada x, e x\_act(2) corresponde ao valor da coordenada y. Neste momento, como só temos um único ponto, o seu valor corresponde ao melhor até agora encontrado, pelo que se atribui o valor da funçao objetivo e da sua imagem às variáveis best\_all\_x e best\_all\_f, respetivamente.

Seguidamente, desenha-se com o comando plot, a localização do ponto inicial no gráfico, a azul.

Em resumo, considerando os seguintes valores inicias para aplicação do SA:

* Critério de Paragem: 200 iterações;
* Número de repetições por valor de temperatura: a = 3;
* Expressão de horário de resfriamento: α=0.96
* Lei da probabilidade: p = e^-|∆T÷ T|
* Temperatura inicial: T=90;
* Soluções de teste geradas apenas permitidas no intervalo de amplitude [-2.048, 2.048]

Os resultados obtidos são apresentados na figura 2, na qual é visível a convergência dos pontos para o mínimo global.

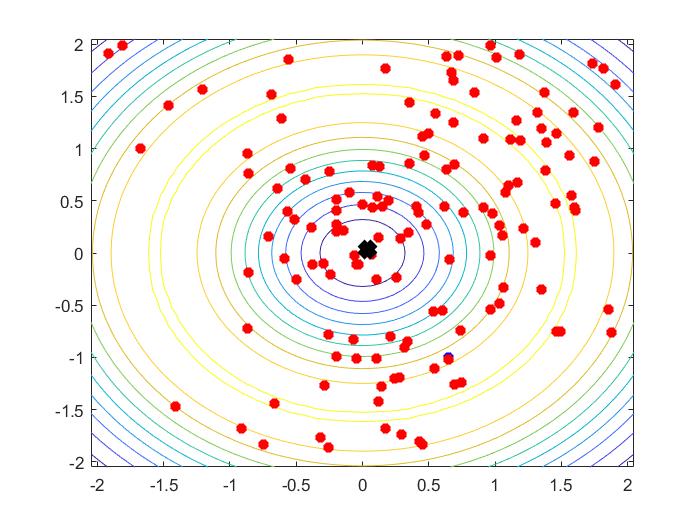
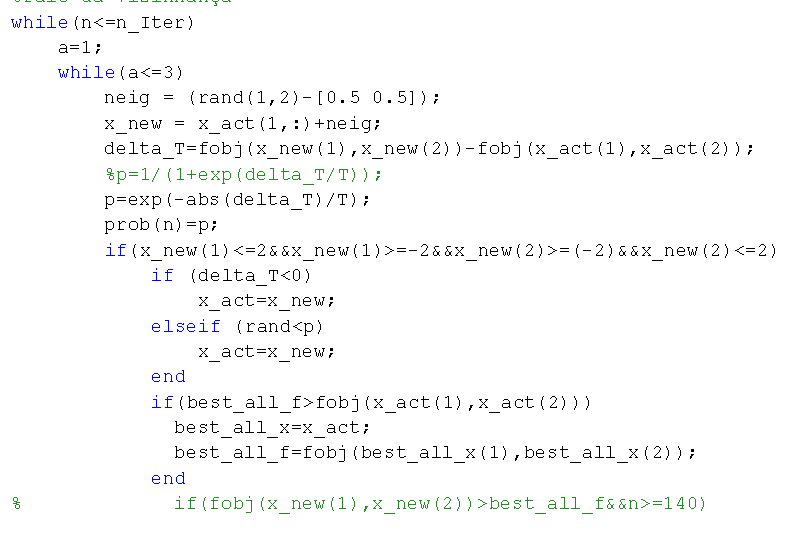


Figura 2 - Distribuição final da aplicação do SA

* Pontos vermelhos – Pontos atuais
* Cruz preta – Solução final encontrada

Como pretendido, os pontos percorrem a região, tentando alcançar o mínimo global; caso se encontrem num mínimo local, fazem ‘subidas na colina’ para escapar do mesmo. Reparamos que os pontos se ‘escapam’ dos extremos, onde se encontram os mínimos locais, atingindo o ponto (0,0) onde se encontra o mínimo global.

* + 1. **Ciclos interno e externo**





No ciclo interno, que é repetido 3 vezes para cada iteração do ciclo externo, define-se uma vizinhança, neig, atualiza-se o x atual, obtendo-se o valor do novo x. Atualiza-se a temperatura, que é a diferença da função objetivo de x\_new e x\_act.

Atualiza-se o valor da probabilidade com que se vai aceitar uma solução ‘pior’ e armazena-se esse valor de probabilidade para colocar no gráfico de variação da probabilidade.



Se estivermos dentro dos limites, e a diferença de temperatura for negativa, então aceita-se a nova solução – é uma solução de melhoria – aceita-se sempre. Caso contrário, se uma probabilidade aleatória for menor que a probabilidade de aceitação atual, então aceita-se a solução ‘pior’.

Até à iteração 135 (aproximadamente), são aceites muitas soluções de ‘pioria’, o que se comprova pela oscilação do gráfico que mostra F Atual ao longo das iterações. A partir dai, o valor de f atual vai apresentando menos amplitude de valores e aproxima-se bastante do zero, o que indica que estamos muitos próximos da solução desejada.

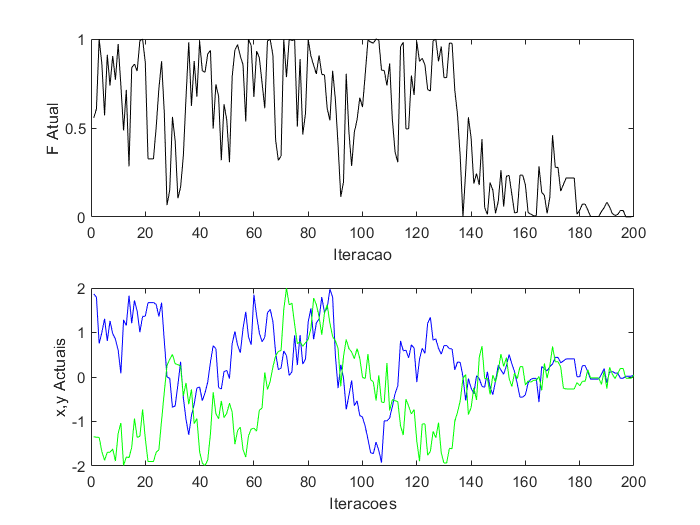


Figura 3 - Posições atuais da particula

A nível do gráfico de x e y atuais, verificamos que começamos perto de um dos extremos. x começa em cerca de 2 e y começa perto de -2. No entanto, na iteração 30 saem desse extremo, voltando durante as iterações seguintes a ir para extremos. No entanto, perto da iteração 135 (como acontece no gráfico do F atual), os valores da posição deixam de oscilar tanto, aproximando-se significativamente do valor 0, onde se localiza o mínimo global.

Os gráficos estão em consonância, verificando-se alterações de comportamento significativas em ambos os gráficos, nos mesmos instantes.

Atualiza-se a temperatura para uma diminuição de 0.04% a cada iteração.



No início, a probabilidade de aceitar soluções ‘piores’ é muito elevada (perto de 1), o que se verifica pela elevada condensação de bolas vermelhas (que correspondem a valores de temperatura atuais) até à iteração 80. A partir deste momento, são aceites valores de probabilidade cada vez mais próximos de zero, menos condensados e mais dispersos. Após crescente diminuição do seu valor ao longo das iterações, na iteração 180 o seu valor condensa em torno do 0. Ou seja, a probabilidade de aceitar soluções ‘piores’ é quase nula pois estamos imensamente perto da solução pretendida.

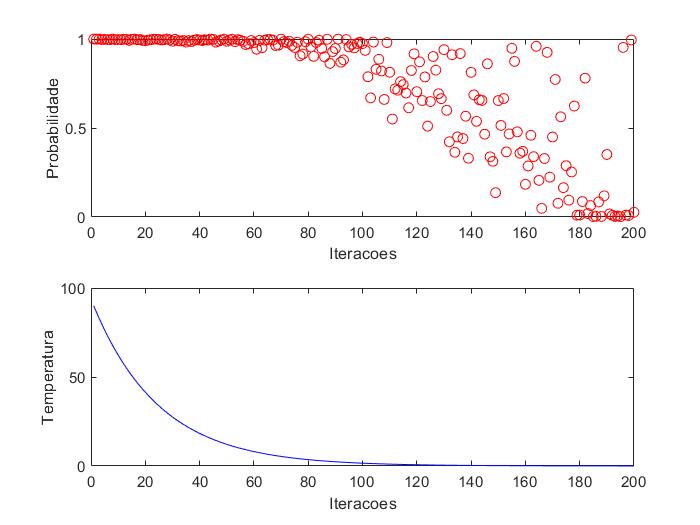


Figura 4 - Variação da probabilidade e temperatura

A temperatura, que definimos como começando em 90, vai diminuindo rapidamente e, na iteração 80 já apresenta um valor bastante próximo de 0. NA iteração 120 é quase nula. Com a diminuição da temperatura, vão sendo descartadas as soluções ‘piores’.

A análise dos gráficos permite constatar que estes estão em consonância um com o outro, pois as iterações decisivas são sensivelmente as mesmas em ambos os gráficos.

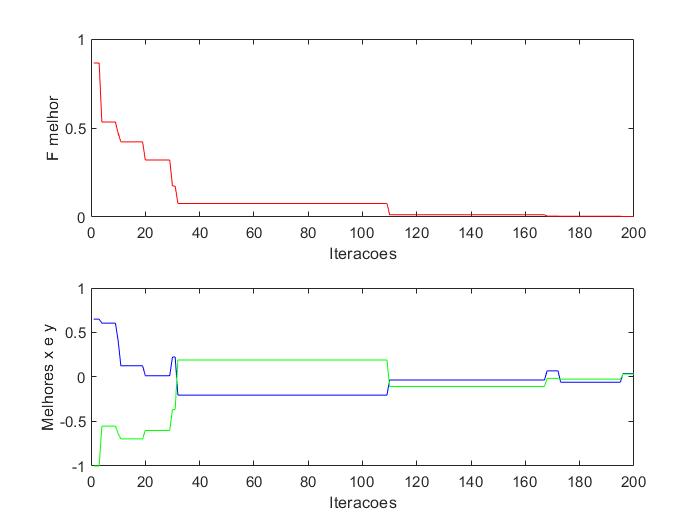


Figura 4 - Melhores posições alcançadas pela partícula

Nestes gráficos, apenas se analisam as soluções consideradas ‘melhorias’, pelo que não existem grandes oscilações de valores, só se registam as ‘descidas na colina’. Os valores de F melhor, que principiam perto de 1, vão descendo de forma monótona até 0, sendo que na iteração 30 F melhor já está muito próximo de 0 e na interação 110 atinge esse valor, o que indica a proximidade à solução desejada. De notar que por volta da iteração 2 o valor de F melhor já desceu drasticamente para perto de 0.5.

Resultados similares se obtêm para os valores melhores de x e y. Na iteração 30 x e y atingem praticamente o valor 0, afastam-se um pouco da solução ideal até a encontrarem por volta da iteração 110 e estabilizarem nesta posição, nesse momento.

# notas finais

Com base na sua lei probabilística, o SA possibilita a aceitação de soluções das quais não resultam uma diminuição da temperatura, permitindo-nos assim explorar todo o espaço de soluções e evitar que este fique bloqueado em mínimos globais. Desta forma, é-nos possível encontrar o valor exato para o mínimo local da função objetivo quando a temperatura converge para o estado estável.

### Bibliografia

1. Moura Oliveira P. B., Solteiro Pires E. J. e Novais P., (2016), “Revisiting the Simulated  
   Annealing Algorithm from a Teaching Perspective”, M. Graña et al. (eds.), InternationalJoint Conference SOCO’16-CISIS’16-ICEUTE’16, Advances in Intelligent Systems andComputing 527, DOI 10.1007/978-3-319-47364-2\_70
2. <http://www.inf.ufpr.br/aurora/disciplinas/topicosia2/downloads/trabalhos/SimulatedAnnealingTSP.pdf>