## 課程名稱：最佳化演算法

## 題目：CDO演算法專題實現報告

## 指導老師：蕭瑛東　教授

## 報告人姓名：林翡

## 學號：110916015

## 班級：資四甲

## 日期：2024/05/05

1. 題目:使用matlab實現Chernobyl disaster optimizer (CDO)演算法，以CEC 2017的23個測試函數進行測試。
2. CDO演算法（以流程圖或虛擬碼表示，並加以說明）

* 虛擬碼:

Step 1: 初始化Alpha、Beta和Gamma的位置、分數、搜索代理的位置、收斂曲線

Step 2:while (the end iteration is not achieved) do，開始迭代

Step 3:計算α、β、γ粒子的目標函數值

If fitness<gammaScore

gammaScore = fitness

更新γ粒子的位置

End

If fitness<betaScore

betaScore = fitness

更新β粒子的位置

End

If fitness<alphaScore

alphaScore = fitness

更新α粒子的位置

End

Step 4:更新搜索代理的位置

計算α、β、γ粒子的梯度下降因子

Step 5:更新總位置的平均

* 流程圖:

輸入搜索代理數量、最大迭代次數、搜索空間下界、搜索空間上界、解的維度、目標函數

開始

連接

Alpha\_pos=zeros(1,dim); Alpha\_score=inf;

Beta\_pos=zeros(1,dim); Beta\_score=inf; Gamma\_pos=zeros(1,dim); Gamma\_score=inf;

Positions=initialization(SearchAgents\_no,dim,ub,lb)

初始化Alpha、Beta和Gamma的位置和分數、搜索代理的位置

**-----**

WSh=3-l\*((3)/Max\_iter);

Sa = ((log10((160001)\*rand(1,1)+16000)));

Sb = ((log10((270000-1)\*rand(1,1)+270000)));

Sr = ((log10((300000-1)\*rand(1,1)+300000)));

更新α、β、γ的score、位置

yes

fitness<Alpha\_score、Beta\_score、Gamma\_score

計算每個搜索代理的目標函數值

fitness=fobj(Positions(i,:));

for i=1:size(Positions,1)

while l<Max\_iter

更新收斂曲線

Convergence\_curve(l)=Alpha\_score;

計算α、β、γ粒子的梯度下降因子，並更新位置

for j=1:size(Positions,2)

i=1:size(Positions,1)

連接

1. 決定群體規模 (如何決定族群數量)

* 群體規模也就是程式中的SearchAgents\_no變數，我一開始先設定成30，是根據網路參考的程式設定的，後面會在設定成100比較看看跟30的差異或是優劣性，這邊先看F1函數的結果。

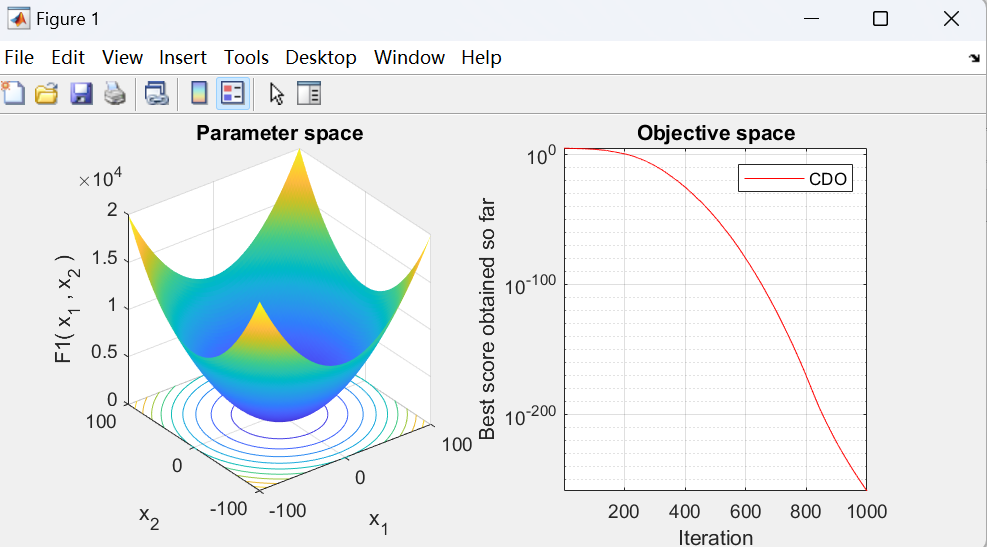
(The best optimal value of the objective funciton found by CDO is : 1.9015e-259)

圖1. F1函數設定SearchAgents\_no=30

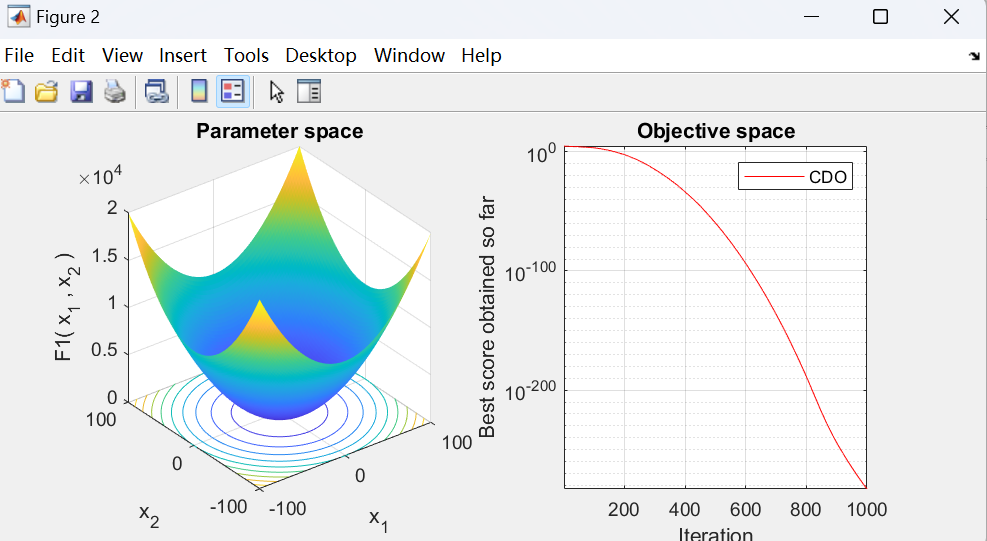
(The best optimal value of the objective funciton found by CDO is : 1.0554e-283)

圖2. F1函數設定SearchAgents\_no=100

* 單就F1函數看的話，SearchAgents\_no設定30及100所獲得的最佳解好像是設定成100的會比較接近0(該函數的期望最佳值)，但看圖的話好像沒什麼太大的差異。

1. 設計適應函數 (決定個體適應度的評估標準)

* 這23個測試函數都是求最小化問題，而它們的期望最佳值如圖3，因此我會以最靠近函數的期望最佳值去做評估，運行次數都只運行1次。

圖3. 23個測試函數期望的最佳值

1. 決定終止條件

* 利用設定最大迭代次數的上限(1000次)，當程式執行了指定的迭代次數後停止。

1. 結果(答案)與討論(含繪出函數f之圖形與收斂過程圖)

* 群體大小設為30的結果

|  |  |  |  |
| --- | --- | --- | --- |
| 函數 | F1 | F2 | F3 |
| 群體大小 | 30 | | |
| 最大迭代次數 | 1000 | | |
| 維度 | 30 | | |
| 上下限 | [-100,100] | [-10,10] | [-100,100] |
| 最佳目標函數值 | 2.5004e-262 | 1.0344e-131 | 4.0505e-218 |
| 函數f之圖形與收斂過程圖 |  |  |  |

|  |  |  |
| --- | --- | --- |
| F4 | F5 | F6 |
| 30 | | |
| 1000 | | |
| 30 | | |
| [-100,100] | [-30,30] | [-100,100] |
| 6.9256e-126 | 27.2374 | 7.5 |
|  |  | C:\Users\meimei\AppData\Local\Packages\Microsoft.Windows.Photos_8wekyb3d8bbwe\TempState\ShareServiceTempFolder\螢幕擷取畫面 2024-05-02 234731.jpeg |

|  |  |  |
| --- | --- | --- |
| F7 | F8 | F9 |
| 30 | | |
| 1000 | | |
| 30 | | |
| [-1.28,1.28] | [-500,500] | [-5.12,5.12] |
| 0.00015932 | -5264.8685 | 213.49 |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| F10 | F11 | F12 |
| 30 | | |
| 1000 | | |
| 30 | | |
| [-32,32] | [-600,600] | [-50,50] |
| 3.9968e-15 | 0 | 1.1061 |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| F13 | F14 | F15 |
| 30 | | |
| 1000 | | |
| 30 | 2 | 4 |
| [-50,50] | [-65.536,65.536] | [-5,5] |
| -8.7014 | 12.7623 | 0.00034143 |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| F16 | F17 | F18 |
| 30 | | |
| 1000 | | |
| 2 | 2 | 2 |
| [-5,5] | [[-5,0], [10,15]] | [-2,2] |
| -1 | 0.39932 | 3 |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| F19 | F20 | F21 |
| 30 | | |
| 1000 | | |
| 3 | 6 | 4 |
| [0,1] | [0,1] | [0,10] |
| -3.8592 | -3.1941 | -5.2182 |
|  |  |  |

|  |  |
| --- | --- |
| F22 | F23 |
| 30 | |
| 1000 | |
| 4 | 4 |
| [0,10] | [0,10] |
| -5.3308 | -9.4543 |
|  |  |

* 群體大小設為100的結果

|  |  |  |  |
| --- | --- | --- | --- |
| 函數 | F1 | F2 | F3 |
| 群體大小 | 100 | | |
| 最大迭代次數 | 1000 | | |
| 維度 | 30 | | |
| 上下限 | [-100,100] | [-10,10] | [-100,100] |
| 最佳目標函數值 | 2.7962e-292 | 2.5607e-145 | 6.516e-229 |
| 函數f之圖形與收斂過程圖 |  |  |  |

|  |  |  |
| --- | --- | --- |
| F4 | F5 | F6 |
| 100 | | |
| 1000 | | |
| 30 | | |
| [-100,100] | [-30,30] | [-100,100] |
| 6.3461e-132 | 26.9137 | 7.5 |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| F7 | F8 | F9 |
| 100 | | |
| 1000 | | |
| 30 | | |
| [-1.28,1.28] | [-500,500] | [-5.12,5.12] |
| 9.4118e-07 | -4514.1656 | 173.4186 |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| F10 | F11 | F12 |
| 100 | | |
| 1000 | | |
| 30 | | |
| [-32,32] | [-600,600] | [-50,50] |
| 3.9968e-15 | 0.0078374 | 1.1061 |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| F13 | F14 | F15 |
| 100 | | |
| 1000 | | |
| 30 | 2 | 4 |
| [-50,50] | [-65.536,65.536] | [-5,5] |
| 0.28886 | 6.9046 | 0.00031151 |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| F16 | F17 | F18 |
| 100 | | |
| 1000 | | |
| 2 | 2 | 2 |
| [-5,5] | [[-5,0], [10,15]] | [-2,2] |
| -1 | 0.39794 | 3 |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| F19 | F20 | F21 |
| 100 | | |
| 1000 | | |
| 3 | 6 | 4 |
| [0,1] | [0,1] | [0,10] |
| -3.8626 | -3.1893 | -8.9711 |
|  |  |  |

|  |  |
| --- | --- |
| F22 | F23 |
| 100 | |
| 1000 | |
| 4 | 4 |
| [0,10] | [0,10] |
| -9.187 | -7.5566 |
|  |  |

* 比較最佳目標函數值

|  |  |  |
| --- | --- | --- |
| 搜索代理數量 | 30 | 100 |
| F1 | 2.5004e-262 | 2.7962e-292 |
| F2 | 1.0344e-131 | 2.5607e-145 |
| F3 | 4.0505e-218 | 6.516e-229 |
| F4 | 6.9256e-126 | 6.3461e-132 |
| F5 | 27.2374 | 26.9137 |
| F6 | 7.5 | 7.5 |
| F7 | 0.00015932 | 9.4118e-07 |

|  |  |  |
| --- | --- | --- |
| F8 | -5264.8685 | -4514.1656 |
| F9 | 213.49 | 173.4186 |
| F10 | 3.9968e-15 | 3.9968e-15 |
| F11 | 0 | 0.0078374 |
| F12 | 1.1061 | 1.1061 |
| F13 | -8.7014 | 0.28886 |
| F14 | 12.7623 | 6.9046 |
| F15 | 0.00034143 | 0.00031151 |
| F16 | -1 | -1 |
| F17 | 0.39932 | 0.39794 |
| F18 | 3 | 3 |
| F19 | -3.8592 | -3.8626 |
| F20 | -3.1941 | -3.1893 |
| F21 | -5.2182 | -8.9711 |
| F22 | -5.3308 | -9.187 |
| F23 | -9.4543 | -7.5566 |

* 紅色字為最靠近測試函數的最佳值
* 藍色字為兩者相等
* 討論:由上面的比較表格可以看出搜索代理數量為100的值大部分都比搜索代理數量為30還更接近函數的最佳值，但因為我只讓每個函數運行1次而已，所以結果可能也不太標準，畢竟論文中是運行了10次，其中的F11、F18的結果都是跟函數的最佳值相同。
* CDO與GA演算法比較

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | CDO | GA | |
| 最佳目標函數值 | F1 | 2.1986e-259 | 1.8628e-07 | |
| 函數f之圖形與收斂過程圖 |  | | |
|  | F2 | 9.432e-135 | 0.00061046 | |
|  |  | | |
|  | F3 | 5.9963e-225 | | 9.3138e-08 |
|  |  | | |
|  | F4 | 2.8694e-123 | | 0.00030519 |
|  |  | | |
|  | F5 | 27.2506 | | 0.0027504 |
|  |  | | |
|  | F6 | 7.5 | | 0.25031 |
|  |  | | |
|  | F7 | 4.4026e-05 | | 0.17655 |
|  |  | | |
|  | F8 | -4597.9299 | | -7.8675 |
|  |  | | |
|  | F9 | 3.8313 | | 1.6463 |
|  |  | | |
|  | F10 | 3.9968e-15 | | 0.0012257 |
|  |  | | |
|  | F11 | 0 | | 0.0074533 |
|  |  | | |
|  | F12 | 1.1061 | | 0.13871 |
|  |  | | |
|  | F13 | 0.52359 | | 0.012692 |
|  |  | | |
|  | F14 | 14.7692 | | 12.6705 |
|  |  | | |
|  | F15 | 0.00032188 | | 0.14841 |
|  |  | | |
|  | F16 | -1 | | -0.95164 |
|  |  | | |
|  | F17 | 0.39824 | | 0.40472 |
|  |  | | |
|  | F18 | 3.0001 | | 265.0071 |
|  |  | | |
|  | F19 | -3.862 | | -0.091861 |
|  |  | | |
|  | F20 | -3.3128 | | -0.10645 |
|  |  | | |
|  | F21 | -7.807 | | -0.037277 |
|  |  | | |
|  | F22 | -7.9726 | | -0.037643 |
|  |  | | |
|  | F23 | -8.3637 | | -0.070375 |
|  |  | | |

* 討論:由以上可以看出CDO比GA演算法能更有效率的收斂，且CDO也較接近函數的最佳值，在特殊函式F15、F19、F20、F21、F22、F23，CDO比GA來的好很多，GA完全沒在收斂的感覺，收斂圖超級震盪，但也有可能是因為GA的設計演算法有問題，Debug了好久。

1. 感想與心得

* 這算是研究所前第一次的論文報告，本來聽到要報告時還非常焦慮，覺得自己對於研讀論文還非常陌生，但藉由這堂課讓我提前適應研究所的例行公事我覺得超棒，學習怎麼有效率地看懂論文，怎麼上網查到有幫助的資料等等，雖然一開始還是一頭霧水，程式都要上網查人家怎麼實現的。而這次報告的好處還有可以同時聽其他同學介紹新的優化演算法，因為大部分都是去年才提出的，所以覺得格外新鮮。

1. 參考資料

* GA演算法: <https://github.com/alpsayin/genetic-algorithm-matlab>
* CDO演算法: https://github.com/sh7adeh1990/CDO