

SESION 1 DEL PROPE, INGRESO A LA MAESTRIA EN CIENCIAS DE LA COMPUTACION.

ENVIAR A: artha@cimat.mx

Archivo en ZIP

Agregar su nombre completo !!!

PREGUNTAS

- 1) COMO SE AJUSTA EL TAMAÑO DEL PASO ? (CONSIDERAR PASOS GRANDES Y PASOS CORTOS)
 - 2) COMO SE ORIENTA LA MATRIZ DE COVARIANZAS?
 - 3) COMO LA ELIPSE SE ALARGA/ACORTA EN CADA DIRECCIÓN ?
 - 4) COMO DA DELTA_X_Y EN CIERTA DIRECCIÓN? (renglón 3 del pseudocódigo)
- NOTA: ES MAS IMPORTANTE CONTESTAR LAS PREGUNTAS QUE HACER EL ALGORITMO DIRECTO SIN CONTESTAR "NADA".

```
clc;
clear all;
close all;

lambda=60;
mu=lambda/4;

dimension=2;
sigma_inicial=2;

C=eye(dimension);
Ngen=1000;
epsilon=0.0001;
boolGraphics=true;

auxGraficar=15;
aux_inc_Graficar=0.1;
a=20; % de por aqui va a comenzar la b $\sqrt{}$  s queda
b=20;
y_inicial=a + (b-a).*rand(1,dimension);
minimo_conocido=zeros(dimension,1);
fmin_convergencia=rastrigin(minimo_conocido);

% @rastrigin
% @esfera
% @ackley
```

```
% @rosenbrock
% @griewank
```

```
[vmin,fmin]=cmsa(mu, lambda, dimension, C, Ngen,
@esfera, sigma_inicial,y_inicial,
fmin_convergencia, epsilon,
boolGraphics,auxGraficar, aux_inc_Graficar);
```

```
%AQUI INICIA
```

```
function [vmin,fmin]=cmsa(mu, lambda, dimension, C,
Ngen, funcion,sigma_inicial,y_inicial,
fmin_convergencia, epsilon,
boolGraphics,auxGraficar, aux_inc_Graficar)
```

```
tc=1+(dimension*(dimension+1))/(2*mu);
tau=(1.0/sqrt(2*dimension))/4;
```

```
y_l=zeros(lambda,dimension); %vectores
fitness=zeros(lambda,1);
y=y_inicial;
sigma_l=ones(lambda,1)*sigma_inicial;
s_l=zeros(lambda,dimension);
z_l=zeros(lambda,dimension);
index=1:mu;
```

```
%ESTO ES PARA GRAFICAR
```

```
%-----mostrar los contornos-----%
```

```
if boolGraphics && dimension ==2
```

```
wtext= strcat('mu= ', num2str(mu), ' lambda= ', num2str(lambda), '
Ngen= ', num2str(Ngen), ' sigma= ', num2str(sigma_l(1)), ' tau= ',
num2str(tau), ' tau_c= ', num2str(tc));
[X_m, Y_m]=meshgrid(-auxGraficar:aux_inc_Graficar:auxGraficar, -
auxGraficar:aux_inc_Graficar:auxGraficar);
```

```
%Z=X_m.^2+Y_m.^2;
aux1=size(X_m,1);
aux2=size(X_m,2);
Z=zeros(aux1,aux2);
for i=1:aux1
    for j=1:aux2
        Z(i,j)=funcion([X_m(i,j),Y_m(i,j)]);
    end
end
```

```
contour(X_m,Y_m,Z);
title(wtext);
hold on;
end
```

```
%FIN GRAFICAR
```

```
%-----ALGORITMO CMA-SA-ES -----%
```

```
display('iniciando el algoritmo..');
```

```
for i=1:Ngen
```

```
    %actualizar parametros
```

```
    %[B,C]=eig(C);
```

```
C_2= chol(C);
```

```
    sigma_mean_best=mean(sigma_l(index(1:mu))));
```

```
sigma_l=sigma_mean_best*exp(tau*lognrnd(0,1,lambda,
1));%mutaciones de cada individuo
```

```
%-----Generar poblacion-----%
```

```

    for j=1:lambda
        s_l(j, :)=C_2*randn(dimension,1);
        z_l(j, :)=sigma_l(j)*s_l(j,:);
        y_l(j, :)=y+z_l(j,:);
        fitness(j)=funcion(y_l(j,:));
    end
    [ fitness_sorted, index]=sort(fitness);

    %actualizar el centro con los mejores.
    y=y+ mean(z_l(index(1:mu),:));

    %actualizar la matriz de covarianza
    sst=zeros(dimension,dimension);
    for j=1:mu
        sst=sst+s_l(index(j),:)'*s_l(index(j),:);
    end
    sst=sst./mu;

    C=(1-1/tc).*C+(1/tc).*sst;

    %%%-----RESULTADOS-----%%%
    if abs(fitness(index(1))-
    fmin_convergencia)<=epsilon
    %convergencia
    fmin=fitness(index(1));
    vmin=y_l(index(1),:);
    display(strcat('convergencia  -- ',num2str(i)));
    return
    end

    % MAS CODIGO SOPORTE PARA GRAFICAR CUANDO DIMENSION 2

    if boolGraphics && dimension ==2
    %-----%%
    %mostrar poblacion inicial
    if i==1
        plot(y_l(:,1), y_l(:,2), 'g+');
    end
    %mostrar el mejor de cada generacion

        plot(y_l(index(1),1), y_l(index(1),2), 'r.');
```

```

    plot(y(:,1), y(:,2), 'r+');

%mostrar la elipse
[B,CC]=eig(C);
    plotellipse(y(:,1), y(:,2), CC(1,1), CC(2,2), -
atan(B(2,2)/B(1,2)), 'b-');
    %puse en signo menos a la funcion atan
end
%-----%%
%%-----%%
end

display('Generaciones terminadas');
fmin=fitness(index(1))
vmin=y_l(index(1),:)
end

```

$$N = DIM$$

$$\mu = \frac{\lambda}{4}$$

$$\tau = \frac{1}{\sqrt{2 * N}}$$

$$\tau_c = 1 + \frac{N(N+1)}{2\mu}$$

$$best = 1$$

$$generation = 0$$

while ($best > 10^{-7}$ & $generation < 10000$)

$generation + 1$

for $l = 1$ **to** λ

$$\sigma_l = \langle \sigma \rangle e^{\tau N_l(0,1)}$$

$$s_l = chol(C)' N_l(0, I)$$

$$z_l = \sigma_l s_l$$

$$Y_l = Y + z_l$$

$$f_l = f(Y_l)$$

$$Population(l) = \{\sigma_l, s_l, z_l, Y_l, f_l\}$$

end(for)

$$Kids = select\{Population, \mu\}$$

$$Y = Y + \langle z \rangle$$

$$C = \left(1 - \frac{1}{\tau_c}\right) C + \frac{1}{\tau_c} \langle ss^T \rangle$$

$$best = select_best(Kids)$$

end(while)

$(\mu/\mu_I, \lambda)$ -CMA- σ -SA-ES (one generation cycle)

For $l = 1$ **To** λ

$$\sigma_l \leftarrow \langle \sigma \rangle e^{\tau \mathcal{N}_l(0,1)} \quad (\text{R1})$$

$$\mathbf{s}_l \leftarrow \sqrt{\mathbf{C}} \mathbf{N}_l(\mathbf{0}, \mathbf{I}) \quad (\text{R2})$$

$$\mathbf{z}_l \leftarrow \sigma_l \mathbf{s}_l \quad (\text{R3})$$

$$\mathbf{y}_l \leftarrow \mathbf{y} + \mathbf{z}_l \quad (\text{R4})$$

$$f_l \leftarrow f(\mathbf{y}_l) \quad (\text{R5})$$

End

$$\mathbf{y} \leftarrow \mathbf{y} + \langle \mathbf{z} \rangle \quad (\text{R6})$$

$$\mathbf{C} \leftarrow \left(1 - \frac{1}{\tau_c}\right) \mathbf{C} + \frac{1}{\tau_c} \langle \mathbf{s} \mathbf{s}^T \rangle \quad (\text{R7})$$

