LAB - 2

24/10/24

Date
Page

Particle swarm optimization for function optimization

PSO is inspired by the social behavior of birds flocking or fish schooling. PSO is used to find optimal solutions by iteratively improving a candidate solution with regard to a given measure of quality. Implement the PSO algorithm using python to optimize a mathematical function

step 1 : Randomly initialize swarm population of N particles
x_i (i = 1, 2 ..... n)

step 2 : Select hyperparameter values
w, c1 and c2

step 3 : for iter in range(max_iter):
for i in range(n):
a. compute new velocity of ith particle
swarm[i].velocity = w*swarm[i].velocity +
r1*c1*(swarm[i].bestpos - swarm[i].position)+
r2*c2*(best_pos_swarm - swarm[i].position)
b. compute new position of ith particle using its new velocity. swarm[i].position += swarm[i].velocity
c. If position is not in range [minx, maxx] then clip
if swarm[i].position < minx :
swarm[i].position = minx
elif swarm[i].position > maxx :
swarm[i].position = maxx
d. update new best of this particle & new best of swarm
if swa insensitive to scaling of design variables
rm[i].fitness < swarm[i].bestFitness :
Swarm[i].bestfitness = swarm[i].fitness
Swarm[i].bestpos = swarm[i].position
End-for
End-for

step 4 : Return best particle of swarm.

Output

Best position : [-7.225174070e^{13} 2.07708164 e^{-13}]

Best Fitness : 5.651692632044056 e^{-25}