

Study and Analysis of Particle Swarm Optimization and Its Implementation

Swati Sharma
Student of M.tech CSE
N.C.C.E , Israna , Panipat
Jind , Haryana , India

Dr. Sukhvir Singh
H.O.D in C.S.E dept.
N.C.C.E , Israna
Jind , Haryana , India

Abstract— This paper is basically about Particle Swarm optimization technique which is basically an optimization technique .My paper focuses on the aspect that how PSO is a better optimization technique and the reason behind the increasing usage of PSO nowadays. There will also be a brief discussion on the algorithm and its implementation. Particle Swarm optimization is a heuristic global optimization method .This works on a particular search space .Optimization is a mechanism of finding minimum and maximum values from a given set of values. This is done by taking some particular parameters and calculating results according to it. This technique was first discovered by James Kennedy and Russell C. Eberhart in 1995 .^[1] This technique is widely used because of its easy and fast implementation in comparison to other techniques. Idea of PSO is originated from two separate concepts - the first one is swarm intelligence and second one is evolutionary computation.

Keywords— Particle swarm optimization , swarm intelligence , particle , heuristic .

I. INTRODUCTION

Particle Swarm optimization is a heuristic global optimization technique . It was firstly discovered and described by James Kennedy and Russell C. Eberhart in 1995.^[1] This technique was proposed from the study of swarm intelligence . Swarm or a group of flocks when search for food , the type of intelligence they use in interacting with their friend swarms is the main principle behind the origin of this technique. When a group of swarms go for searching food , either they go together or in group till they find a place where food can be found. One of the bird among them searches the best food called the optimum search . Now every bird is moving with some velocity to search the food. Now the method which this bird adopts to convey the message of best food to all other birds and the flock comes to that place is used in PSO . It was thought that this interaction among birds can be efficiently utilized in finding the optimum solution. These birds or swarms are said to be particles in particle swarm optimization. Now in PSO every particle is moving with some velocity and when the optimum solution is found by one particle , there is a memory which helps in conveying the

message to all other particles.

II. THE BASIC IDEA OF ALGORITHM

The Basic idea used in Particle Swarm Optimization is as follows :

- There are several particles .
- Each particle is searching for an optimum solution.
- Each individual particle has some particular velocity with which it is moving.
- Each particle has a memory . It remembers its best location or best result found so far . It is called its p-best or personal best.
- Now particles in a swarm co-operate with each other and they continuously keep on telling each other which which locations they have visited.
- Now how they co-operate?
- This is very simple. Firstly they find their neighbourhood particles.
- They get to know the fitnesses of the neighbourhood particles and uses the one with the best fitness value to adjust its velocity.
- In each timestamp each particle has to change its position by adjusting its velocity.
- Calculate new velocity .
- New position = Previous location + New velocity

III. PSO OPTIMIZATION

Optimization means to find maximum or minimum values of a function.

A minimization task can be defined mathematically as :

Given $f : R^n \rightarrow R$

Find $x_0 \in R^n$ such that $f(x_0) \leq f(x), \forall x \in R^n$

Similarly, a maximization task is defined as:

Given $f : R^n \rightarrow R$

Find $x_0 \in R^n$ such that $f(x_0) \geq f(x), \forall x \in R^n$

Maximization of a task is opposite of minimization of a task. If maximization is f then minimization is $-f$.

The domain R^n is referred to as search space. Each element of R^n is referred to as candidate solution in the search space. x_0 is called the optimal solution. n denotes no. of parameters

used in optimization. The function f is called the objective function .

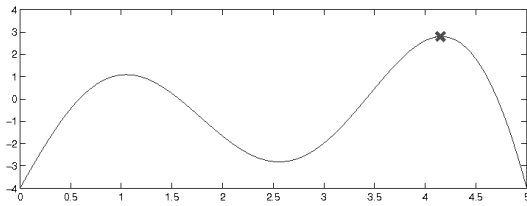


Figure 1: Function Maximum^[1]

This figure shows local maximum in addition to global maximum.

A local maximum is that candidate solution which has the highest value of objective function than any other candidate solution in that given search space.

For example, in the given figure in the interval $[0, 2.5]$ the local maxima is at point $x = 1.05$ and global maxima is at point $x = 4.2$.

IV. PSO ALGORITHM EXPLANATION

In simple terms PSO algorithm consists of just 3 steps :

1. Evaluate the fitness of each particle
2. Update individual and global best fitnesses and positions
3. Update velocity and position of each particle

Explained algorithm :

1. For each particle , Initialize particle .
2. END
3. Do
4. For each particle
Calculate fitness value
5. If the fitness value is better than its best personal fitness value in history , set current value as a new best personal fitness.
6. End
7. Choose the particle with the best fitness value of all the particles, and if that fitness value is better than current global best, set as a global best fitness value
8. For each particle calculate particle velocity according to the velocity change equation.
9. Update particle position according position change equation.
10. End
11. While maximum iterations or minimum error criteria is not attained.
12. Set current value as the new **pBest**.
13. End
14. Choose the particle with the best fitness value of all as **gBest**.
15. For each particle calculate particle velocity according to equation (a).
16. Update particle position according to equation (b).
17. End

18. While maximum iterations or minimum error criteria is not attained.

Mathematically , the two equations used above are:

Equation (a)

$$v[] = c0 * v[] + c1 * \text{random1}() * (\text{pbest}[] - \text{present}[]) + c2 * \text{random2}() * (\text{gbest}[] - \text{present}[])$$

($c0$ value is different for many researchers now but in the original method, $c0 = 1$)

Equation (b)

$$\text{present}[] = \text{present}[] + v[]$$

(New particle position = old particle position + new velocity)

Here $v[]$ is the particle velocity and **present[]** is the particle's position. **pbest[]** is particle's personal best value and **gbest[]** is particle's global best value.

Here **random1** and **random2** are random values generated for velocity update and the values lie between.

($0 \leq \text{random1} \leq 1$ and $0 \leq \text{random2} \leq 1$) .

The values of $c1$ and $c2$ ranges between

$0 \leq c1 \leq 2$ and $0 \leq c2 \leq 2$.

- 1 .The second term $c1 * \text{random1}() * (\text{pbest}[] - \text{present}[])$ is called the cognitive component and acts as the particle's memory, causing it to tend to return to the regions of the search space in which it has experienced high individual fitness.
- 2 .The **cognitive coefficient** $c1$ is usually close to 2, and affects the size of the step the particle takes toward its individual best candidate solution x_0 .
- 3 .The third term $c2 * \text{random2}() * (\text{gbest}[] - \text{present}[])$, called the **social component**, causes the particle to move to the best region the swarm has found so far. This represents the size of the step the particle takes toward the global best candidate solution $g(x)$ the swarm has found up until that point.

V. NEIGHBOURHOOD SELECTION

Neighbourhoods are of 3 types :-

1. Pbest type
2. Gbest type
3. Lbest type

Gbest Type : In this type a particle includes all other particles in its neighbourhood . It is said to have the strongest communication . As each particle can communicate with every other particle so all information is passed among all. So the Gbest or global best value is searched out . Every particle P_i sends information to every other p_i and the best value among them is said to be the Global best value.

Pbest Type : In this type particle are in total isolation . It does not pass information to any particle . It just calculate its own Pbest value or personal best value. It is worst in case of efficiency and performance.

Lbest Type : In this type information is transferred to just a subset of particles called $P_i \in P$ where P are the total number

of particles . In this gbest or global best value is communicated among these Pi or subset of particles.

Particle 1's 3 neighbours

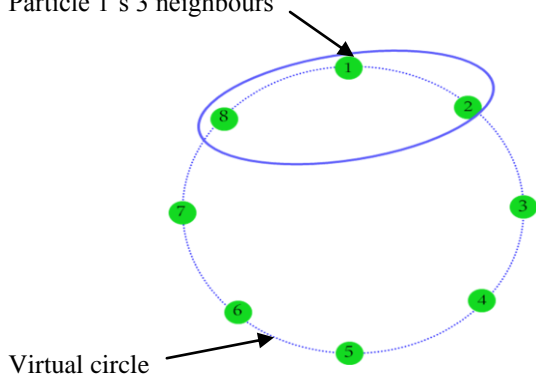


Figure 2: Neighbours of a point

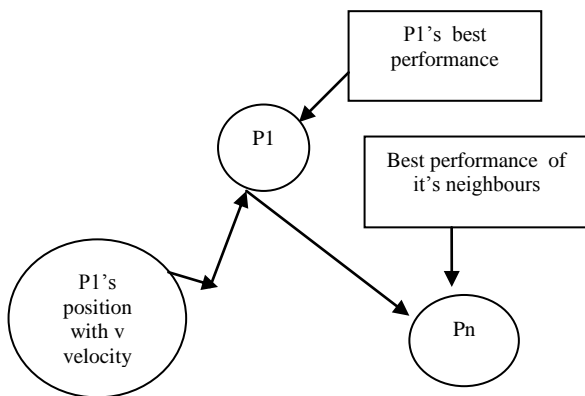


Figure 3 : How particles communicate with neighbours.

- Particle can compute its position or the objective function at the place it is . The particle always remember the best position it ever found.
- Now this particle communicates with its neighbours .
- Neighbours communicate with each other and find the best value of them all.
- Now whichever the value is greater is said to be the value of P1 and its position and velocity are updated accordingly.

VI. PSO IMPLEMENTATION

Pso implementation is described below through a simple example given below :-
The given illustration is a simple example in which algorithm has three numbers that add up and provide a target value.

In this example we take example of particles which are fully connected to each other , so all particles can be compared to each other.

Simplicity is that characteristic of this example which makes it very easy to experiment with. This example's simplicity makes it very easy to experiment with.

TARGET – This value represents the answer the algorithm is looking for.

MAX_INPUTS - This shows the number of operands in the expression.

MAX_PARTICLES – This shows the number of particles employed in the test.

V_MAX – It shows the maximum velocity change allowed in the algorithm.

MAX_EPOCHS – It shows the number of iterations the algorithm will have.

START_RANGE_MIN – It is the minimum start range means the smallest random number to start operands at.

START_RANGE_MAX – It is the maximum starting range or the largest random number to start operands at.

EXAMPLE RESULTS 1

It is much easy. Find three operands that add up to 50. Ten particles used V_MAX = 10.

- 47 + -3 + 8 = 52
- 27 + 3 + 10 = 40
- 41 + 6 + 31 = 78
- 47 + 40 + -4 = 83
- 41 + -3 + 30 = 68
- 6 + 14 + 35 = 55
- 3 + 11 + 36 = 50
- 47 + 2 + 9 = 58
- 40 + -3 + 29 = 66
- 1 + 8 + 11 = 20

epoch number: 21

Particle 6 has achieved target.

$$3 + 11 + 36 = 50$$

EXAMPLE RESULTS 2

Ten operands and 20 particles. V_MAX = 10. The algorithm typically has worked a little harder to find the solution.

- 5 + -3 + 18 + 12 + 9 + 15 + 16 + 27 + -17 + -1 = 81
- 11 + 29 + -5 + 15 + -11 + 15 + -9 + 12 + -1 + 6 = 62
- 10 + -2 + 5 + 12 + 14 + 21 + -23 + 10 + 5 + -5 = 27
- 23 + -10 + -3 + 12 + -9 + 1 + 7 + 8 + -42 + -15 = -28
- 23 + 1 + -9 + 12 + -12 + 23 + -9 + 12 + -1 + 10 = 50
- 22 + 19 + 36 + 8 + -18 + 3 + 36 + -17 + -1 + 0 = 88
- 10 + -3 + 7 + 12 + 15 + 11 + 24 + -7 + 15 + -18 = 46
- 0 + -16 + -9 + -3 + 9 + 15 + 15 + 22 + 5 + 10 = 48
- 10 + 25 + 30 + 12 + -18 + -3 + -9 + -7 + -1 + 10 = 29
- 0 + 2 + -9 + 12 + -1 + 6 + 26 + 7 + -36 + -17 = -10
- 9 + -11 + -4 + 12 + -10 + 15 + -30 + 17 + 14 + -20 = -26
- 18 + -5 + 21 + -22 + -1 + 5 + 26 + 17 + 23 + -21 = 61
- 20 + -17 + -10 + -8 + -1 + 6 + 13 + 14 + 21 + -8 = -10
- 23 + 1 + -1 + -18 + 7 + 13 + 9 + 18 + 25 + -6 = 71
- 23 + 25 + -9 + 12 + 26 + -19 + -9 + -7 + -1 + 10 = 51
- 13 + 25 + 1 + 7 + 26 + -19 + -9 + 7 + -35 + -16 = -26
- 11 + -4 + 21 + 3 + 9 + 20 + 36 + 9 + -15 + -5 = 63
- 20 + 2 + -14 + 3 + 18 + 21 + -9 + 25 + -1 + 10 = 75
- 0 + -12 + -9 + 12 + -7 + 4 + 7 + 14 + -1 + 10 = 18
- 4 + 6 + 34 + -20 + 8 + 23 + -9 + -25 + -18 + 10 = 13

epoch number: 103

Particle 4 has achieved target.

$$23 + 1 + -9 + 12 + -12 + 23 + -9 + 12 + -1 + 10 = 50$$

Example Results 3

Ten operands and only three particles. V_MAX = 10. It doesn't always find the solution quickly.

$$20 + 22 + -10 + -3 + 19 + 7 + 13 + 19 + -21 + -19 = 47$$

$$-8 + -11 + -10 + -3 + 4 + 7 + 22 + 7 + 14 + 24 = 46$$

$$20 + 22 + 29 + -19 + -12 + -5 + 13 + 3 + 14 + -19 = 46$$

epoch number: 196

$$20 + 22 + -10 + -3 + 19 + 7 + 13 + 19 + -21 + -19 = 47$$

$$-16 + -19 + -10 + -3 + -4 + 7 + 14 + -1 + 6 + 16 = -10$$

$$20 + 22 + 19 + -29 + -22 + -15 + 13 + -7 + 4 + -19 = -14$$

epoch number: 197

$$20 + 22 + -10 + -3 + 19 + 7 + 13 + 19 + -21 + -19 = 47$$

$$-6 + -9 + -10 + -3 + 6 + 7 + 24 + 9 + 16 + 26 = 60$$

$$20 + 22 + 29 + -19 + -12 + -5 + 13 + 3 + 14 + -19 = 46$$

epoch number: 198

$$20 + 22 + -10 + -3 + 19 + 7 + 13 + 19 + -21 + -19 = 47$$

$$-11 + -14 + -10 + -3 + 1 + 7 + 19 + 4 + 11 + 21 = 25$$

$$20 + 22 + 19 + -29 + -22 + -15 + 13 + -7 + 4 + -19 = -14$$

epoch number: 199

No solution found.

VII. ADVANTAGES OF PSO OVER OTHER ALGORITHMS

There are several advantages of PSO over other algorithms . Some are described below:-

- The results with PSO are much accurate as compared to other algorithms.
- PSO takes much less time as compared with other algorithms due to its interaction with other particles.
- It is based on intelligence.
- In PSO the real number code is adopted and it is decided directly by the solution .
- With the help of PSO researching is much fast and it occupies better optimization ability.

VIII. CONCLUSION

PSO rather a new concept or algorithm than other algorithms of optimization like Genetic algorithm is gaining much popularity because of its several advantages.

PSO research ability is very fast . It completes very fastly and needs very fewer parameters. The examples or results shown above show that PSO works much better for less number of operands but for large number of operands it does not show much accurate results. PSO came into existence in year 1995 but in a very short time it has developed a lot. Much research has been done on this topic . Thousands of research papers has been published on this topic but still it has to progress a lot. In it's future scope it's shortcomings can be worked out.

IX. REFERENCES

- [1] James Blondin , Particle Swarm Optimization : A tutorial , September 4 , 2009.
- [2] Analysis of particle swarm optimization algorithm , Qinghai Bai , College of Computer Science and Technology , Inner Mongolia University for Nationalities , Tongliao 028043, China
- [3] Particle swarm optimization – A Survey , Keisuke Kemeya
- [4] J Kennedy and R.C Eberhart “ Particle Swarm Optimization , IEEE International conference on “ Neural Networks , 1995
- [5] Particle Swarm Model Selection , Hugo Jair Escalante , 2009