International Journal of Advanced Multidisciplinary Research (IJAMR)
ISSN: 2393-8870
www.ijarm.com
Volume 3, Issue 4 -2016

Research Article

SOI: http://s-o-i.org/1.15/ijarm-2016-3-4-10

Mathematical Modelling of Particle Swarm Optimization Algorithm

Maithem Hassen Kareem[1*], Jawad Mahmoud Jassim[2], Nadeem Khargan Al-Hareeb[3]

[1]Electrical Engineering Department, ALmustansiriyah University, Baghdad, Iraq
maythem.hk@gmail.com
[2]Mathematical Department
j.jawad51@yahoo.com
[3]Communication Engineering Department
nadeemkhargan@ymail.com

Keywords

Convergence,
Multi-objective,
Optimization,
swarm intelligence,
Particle swarm,
Social network,
Stagnation.

Abstract

In this manuscript we have introduced mathematical technique that concerns the finding of minimum or maximum of functions in specific possible area. In fact there is no business or industry which is not involved in solving optimization problems. A diversity of optimization techniques compete for the greatest solution. Particle Swarm Optimization technique is a relatively novel, powerful, and modern technique of optimizations that have been empirically shown to achieve well on several of these optimization problems. It is extensively used to find the worldwide optimum explanation and solution in a complex search space. This manuscript introduces a theoretical idea and explanation of the PSO algorithm, the judicious and effects selection of the numerous parameters. Furthermore, this manuscript debates a study of boundary conditions with the invisible wall technique, controlling the convergence behaviors of PSO, and applications of PSO. The method presented fulfills all requirements algorithm and shows better results as compared to previous work.

Abbreviations

PSO: Particle Swarm Optimization.
f : The Function Being Minimized or Maximized. It Takes A Vector Input And Returns A Scalar Value.
v_{ij}^{t} :particle velocity vector *i* in dimension *j* at time *t*.
$x_{i,j}^{t}$:particle position vector *i* in dimension *j* at time *t*.
G_{best} :particle global best position *i* in dimension *j* found from initialization through time *t*.
$P_{best,i}^{t}$: Particle Personal Best Position *i* In Dimension *j* Found from Initialization Through Time *t*.
c_1, c_2: Positive acceleration coefficients which are used to level the contribution of social components and the cognitive respectively.
r_{1j}^{t}, r_{2j}^{t} :Unsystematic quantities from uniform distribution U (0,1) at time *t*.
n: The swarm size or number of particles.
t: Denotes time or time steps.
x:The constriction coefficient.

Introduction

Managers, economists, engineers and scientists all the time have to take several managerial and technological choices at many times for manufacture and maintenance of any system. Gradually the world has been more competitive and complex hence the assessment making should be taken in an best way. For that reason optimization is the main performance of obtaining the greatest outcome under given circumstances.

Optimization invented in the 1940s, when the British military challenged the problem of allocating limited resources (such as fighter, submarines, airplanes and so on) to several activities [1].In the last two decades, a vast amount of research works have been done in the area of generated diverse solutions to non-liner and linear optimization problems. Mathematically an optimization problem has a capability function, describing the problem under a set of constrictions which shown the solution space for the problem. However, most of the traditional optimization technique has determined the first derivatives to localize the optima on a given constricted surface. Due to the problems in evaluation the first derivative for several discontinuous and rough optimization spaces, a number of derivatives free optimization method has been constructed in current time [2].Indeed there is no well-known single optimization method available for solving all optimization problems. Some of optimization methods have been established for solving diverse varieties of optimization problems in latest years. The recent optimization methods are very popular and powerful techniques for solving difficult manufacturing problems. These techniques are particle swarm optimization algorithm, genetic algorithms, neural networks, fuzzy optimization, ant colony optimization, and artificial immune systems [1] [3].

The Particle Swarm Optimization algorithm is a unique population-based stochastic exploration algorithm and an alternate solution to the difficult non-linear optimization problem [4]. The PSO algorithm was first presented by Eberhart and Kennedy in 1995 [5] and its elementary idea was formerly inspired by simulation of the social behavior of animals such as fish schooling, bird flocking and so on. It is based on the ordinary process of group communication to share single knowledge when a group of birds or insects search food or migrate and so forth in a searching space, although all insects or birds do not know where the greatest location [6]. However from the nature of the social behavior, if any participant can discover a desired path to go, the other of the participant will follow rapidly.

The PSO algorithm essentially learned from animal's behavior or activity to solve optimization problems [7]. In PSO, every member of the population is named a particle and the population is named a swarm. Starting with a arbitrarily initialized moving and population in arbitrarily chosen directions, each particle goes through the searching space and remembers the best previous locations of itself and its neighbors. Particles of a swarm communicate good positions to each other than dynamically adjust their own velocity derived and position from the best position of all particles [8]. The next step begins when all particles have been moved. Lastly, all particles incline to fly towards better and better positions over the searching process pending the swarm move to close to an optimum of the fitness function $f : R^n \rightarrow R.$

The PSO technique is becoming very widespread because of its simplicity of execution position ore over capability to swiftly converge to suitable solution [9]. It does not need any gradient information of the function to be optimized and uses only primitive mathematical operators.

As compared with other optimization techniques, it is cheaper, more efficient and faster. Furthermore, there are a few parameters to adjust in PSO. That's why PSO is an ideal optimization problem solver in optimization problems. PSO is well suited to solve the non-linear, non-convex, continuous, discrete, integer variable type problems

The basic model of PSO algorithm

Kennedy and Eberhart first established a solution to the complex non-linear optimization problem by imitating the behavior of bird flocks [10]. They generated the concept of function-optimization by means of a particle swarm [2]. Consider the global optimum of an *n*-dimensional function defined by:

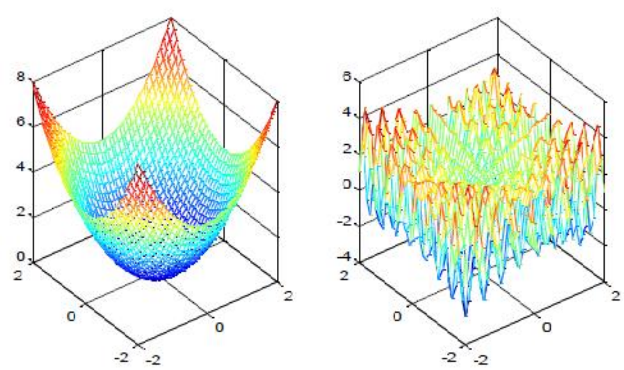
$$f(x_1, x_2, x_3, ...x_n) = f(x) \qquad (1)$$

Where x_i is the search variable, which represents the set of free variables of the given function. The aim is to find a value x^* such that the function f(x^*) is either a maximum or a minimum in the search space. Consider the functions given by:

$$f_1 = x_1^2 + x_2^2 \, (2)$$

and

$$f_2 = x_1 \sin(4\text{®} x_2) - x_2 \sin(4\text{®} x_1 + \text{®}) + 1 \, (3)$$



a- unimcde:　　b-Multi-model

Figure (1) Plot of the functions f1 and f2.

From figure (1a), it is clear that the global minimum of the function f_1 is at $(x_1, x_2) = (0, 0)$,i.e. at the origin of function in the search space. That means it is a *unimodel* function, which has only one minimum. However, to find the global optimum is not so easy for *multi-model* functions, which have multiple local minima. Figure (1b) shows the function f_2 which has a rough search space with multiple peaks, so many agents have to start from different initial locations and continue exploring the search space until at least one agent reach the global optimal position. During this process all agents can communicate and share their information among themselves [4].This manuscript discusses how to solve the *multi-model* function problems. The Particle Swarm Optimization (PSO) algorithms are a multi-agent parallel search method which preserves a swarm of particles and every particle denotes a possible solution in the swarm. All particles fly during a multi-dimensional search space where every particle is modifying its position agree to its own experience and the neighbors. Suppose x_i^t represent the position vector of particle i in the multi-dimensional search space $(i.e. R^n)$ at time step then the position of every particle is updated in the search space by:

$$x_i^{t+1} = x_i^t + v_i^{t+1} \text{ with } x_i^0 \quad U(x\min, x\max) \quad (4)$$

where,

v_t^i is the velocity vector of particle that drives the optimization process and reflects both the own experience knowledge and the social experience knowledge from the all particles;

$U(x\min, x\max)$ is the uniform distribution

where $x_{\min}$ and $x_{\max}$ are its minimum and maximum values respectively.

Hence, in a Particle Swarm Optimization technique, all particles are originated unsystematically and estimated to calculate suitability together with finding the best value of each particle and best value of particle in the entire swarm. After that a loop starts to find an optimum solution. In the loop, first the particles' velocity is updated by the personal and global bests, and then each particle's positionsare updated by the current velocity. The loop is ended with a stopping criterion predetermined in advance.

Problem formulation of PSO algorithm

Problem: Find the maximum of the function

$f(x) = -x^2 + 5x + 20$ with $-10 \le x \le 10$ using the PSO algorithm. Use 9 particles with the initial positions

$$x_1 = 9.6, x_2 = -6, x_3 = -2.6, x_4 = -1.1, x_5 = 0.6$$

$$x_6 = 2.3, x_7 = 2.8, x_8 = 8.3, x_9 = 10.$$

Show the detailed computations for iterations 1, 2 and 3.

Solution:

Step 1: Choose the number of particles

$$x_1 = 9.6, x_2 = -6, x_3 = -2.6, x_4 = -1.1, x_5 = 0.6, x_6 = 2.3$$

$$x_7 = 2.8, x_8 = 8.3, x_9 = 10.$$

The initial population (i.e. the iteration number $t = 0$) can be represented as x_i^0 , where

$i = 1, 2, 3, 4, 5, 6, 7, 8, 9$

$$x_1^0 = -9.6, x_2^0 = -6, x_3^0 = -2.6,$$

$$x_4^0 = -1.1, x_5^0 = 0.6, x_6^0 = 2.3,$$

$$x_7^0 = 2.8, x_8^0 = 8.3, x_9^0 = 10.$$

Evaluate the objective function values as

$$f_1^0 = -120.16, \ f_2^0 = -46, \ f_3^0 = 0.24$$

$$f_4^0 = 13.29, \ f_5^0 = 22.64, \ f_6^0 = 26.21$$

$$f_7^0 = 26.16, \ f_8^0 = -7.39, \ f_9^0 = -30$$

Let $c_1 = c_2 = 1$

Set the initial velocities of each particle to zero:

$$v_i^0 = 0, i.e. v_1^0 = v_2^0 = v_3^0 = v_4^0 = v_5^0 = v_6^0$$

$$= v_7^0 = v_8^0 = v_9^0 = 0.$$

Step 2: Set the iteration number as $t = 0 + 1 = 1$ and go to step 3.

Step 3: Find the personal best for each particle by

$$P_{best,i}^{t+1} = \begin{cases} P_{best,i}^t \rightarrow if \rightarrow f_i^{t+1} > P_{best,i}^t \\ x_i^{t+1} \rightarrow if \rightarrow f_i^{t+1} \le P_{best,i}^t \end{cases}$$

So

$$P_{best,1}^1 = -9.6, P_{best,2}^1 = -6, P_{best,3}^1 = -2.6,$$

$$P_{best,4}^1 = -1.1, P_{best,5}^1 = 0.6, P_{best,6}^1 = 2.3,$$

$$P_{best,7}^1 = 2.8, P_{best,8}^1 = 8.3, P_{best,9}^1 = 10.$$

Step 4: Find the global best by

$$G_{best} = \min\left\{P_{best,i}^t \text{ where } i = 1,2,3,4,5,6,7,8,9.\right.$$

Since, the maximum personal best is:

$P^1_{best,6} = 2.3$ thus $G_{best} = 2.3$

Step 5: Considering the random numbers in the range

(0,1)

As $r_1^1 = 0.213$ and $r_2^1 = 0.876$ and find the velocities of the particles by

$v_i^{t+1} = v_i^t + c_1 r_1^t \left[P^t_{best,i} - x_i^t\right] + c_2 r_2^t \left[G^t_{best,i} - x_i^t\right]$

Where i =1,…..,9.

So

$v_1^3 = 4.4052, v_2^3 = 3.0862, v_3^3 = 1.8405,$

$v_4^3 = 1.2909, v_5^3 = 0.6681, v_6^3 = 0.053,$

$v_7^3 = -0.1380, v_8^3 = -2.1531, v_9^3 = -2.7759$

Step 6: Find the new values of $x_i^1, i = 1,.....,9$ by

$x_i^{t+1} = x_i^t + v_i^{t+1}$ So:

$x_1^1 = 0.8244, x_2^1 = 1.2708, x_3^1 = 1.6924,$

$x_4^1 = 1.8784, x_5^1 = 2.0892, x_6^1 = 2.3,$

$x_7^1 = 2.362, x_8^1 = 3.044, x_9^1 = 3.2548.$

Step 7: Find the objective function values

of $x_i^1 = 1,....,9:$

$f_1^1 = 23.4424,\ f_2^1 = 24.7391,\ f_3^1 = 25.5978,$

$f_4^1 = 25.8636,\ f_5^1 = 26.0812,\ f_6^1 = 26.21,$

$f_7^1 = 26.231,\ f_8^1 = 25.9541,\ f_9^1 = 25.6803.$

Step 8: Stopping criterion:

If the terminal rule is satisfied, go to step 2, Otherwise stop the iteration and output the results.

Step 2: Set the iteration number as $t = 1+1 = 2$, and go to step 3.

Step 3: Find the personal best for each particle.

$P^3_{best,1} = 0.8244, P^3_{best,2} = 1.2708, P^3_{best,3} = 1.6924,$

$P^3_{best,4} = 1.8784, P^3_{best,5} = 2.0892, P^3_{best,6} = 2.3438,$

$P^3_{best,7} = 2.362, P^3_{best,8} = 3.044, P^3_{best,9} = 3.2548.$

Step 4: Find the global best.

$G_{best} = 2.362$

Step 5: By considering the random numbers in the range (0, 1) as

$r_1^3 = 0.178$ and $r_2^3 = 0.507$, find the velocities of the particles by

$v_i^{t+1} = v_i^t + c_1 r_1^t [P^t_{best,i} - x_i^t] + c_2 r_2^t [G_{best} - x_i^t]$

Where i =1,….9. so:

$v_1^2 = 11.5099, v_2^2 8.0412, v_3^2 = 4.7651, v_4^2 = 5.1982,$

$v_5^2 = 1.6818, v_6^2 = 0.0438, v_7^2 = -0.04380,$

$v_8^2 = -5.7375, v_9^2 = -7.3755.$

Step 6: Find the new values of x_i^2 , where i= 1,….9 by

$x_i^{t+1} = x_i^t + v_i^{t+1}$, so

$x_1^2 = 12.3343, x_2^2 = 9.312, x_3^2 = 6.4575,$

$x_4^2 = 5.1982, x_5^2 = 3.7710, x_6^2 = 2.3438,$

$x_7^2 = 1.9240, x_8^2 = -2.6935, x_9^2 = -4.1207.$

Step 7: Find the objective function values of f_i^2

Where i = 1,….,9;

$f_1^2 = -70.4644,\ f_2^2 = -20.1532,\ f_3^2 = 10.5882$

$f_4^2 = 18.9696,\ f_5^2 = 24.6346,\ f_6^2 = 26.2256,$

$f_7^2 = 25.9182,\ f_8^2 = -0.7224,\ f_9^2 = -17.5839.$

Step 8: Stopping criterion:

If the terminal rule is fulfilled, go to step 2, Else stop the iteration and output the results.

Step 2: Set the iteration number as$t$=1+1 =3, and go to step

Step 3: Find the personal best for each particle.

$P^3_{best,1} = 0.8244, P^3_{best,2} = 1.2708, P^3_{best,3} = 1.;6924,$

$P^3_{best,4} = 1.8784, P^3_{best,5} = 2.0892, P^3_{best,6} = 2.3438,$

$P^3_{best,7} = 2.362, P^3_{best,8} = 3.044, P^3_{best,9} = 3.2548.$

Step 4: Find the global best.

$G_{best} = 2.362$.

Step 5: since the random numbers in the range (0,1) as $r_1^3 = 0.178$ and $r_2^3 = 0.507$ find the velocities of the particles by :

$$v_i^{t+1} = v_i^t + c_1 r_1^t [P_{best,i}^t - x_i^t] + c_2 r_2^t [G_{best,i}^t - x_i^t]$$

Where i=1,….,9. So

$v_1^3 = 4.4052, v_2^3 = 3.0862, v_3^3 = 1.8405,$

$v_4^3 = 1.2909, v_5^3 = 0.6681, v_6^3 = 0.053,$

$v_7^3 = -0.1380, v_8^3 = -2.1531, v_9^3 = -2.7759.$

Step 6: Find the new values of x_i^3, i=1,…,9 by

$x_i^{t+1} = x_i^t + v_i^{t+1}$. So

$x_1^3 = 16.7395, x_2^3 = 12.3982, x_3^3 = 8.298,$

$x_4^3 = 6.4892, x_5^3 = 4.4391, x_6^3 = 2.3968,$

$x_7^3 = 1.786, x_8^3 = -4.8466, x_9^3 = -6.8967.$

Step 7: Find the objective function values of f_i^3 Where i=1,….9:

$f_1^3 = -176.5145,\ f_2^3 = -71.7244,\ f_3^3 = -7.3673,$

$f_4^3 = 10.3367,\ f_5^3 = -22.49,\ f_6^3 = 26.2393,$

$f_7^3 = 25.7402,\ f_8^3 = -27.7222,\ f_9^3 = -62.0471.$

Step 8: Stopping criterion:

If the terminal rule is fulfilled, go to step 2, Else stop the iteration and output the results.

Lastly, the values of x_i^2, i=1,2,3,4,5,6,7,8,9.Did not converge, so we increment the iteration number as$t$=4 and go to step2When the positions of all particles converge to similar values, then the method has converged and the corresponding value of x_i^t is the optimum solution. Therefore the iterative process is continued until all particles meet a single value.

Conclusion

This paper discussed the basic Particle Swarm Optimization algorithm, mathematical explanation and geometrical of PSO, particles' the velocity and movement update in the search space, the acceleration coefficients and particles' neighborhood topologies. Particle swarm optimization show convergence and speed, while the power behind genetic programming is the concept. The PSO ran with three parameter options: swarm size, randomization, and topology. The swarm size parameter yielded the best data to offer a conclusion. The obtained result show very good agreement and superior compared to previously work.

References

[1] D. Bratton and J. Kennedy, "Defining a Standard for Particle Swarm Optimization ," in *IEEE Swarm Intelligence Symposium*, 2007, pp. 120-127.

[2] X. Li and K. Deb, "Comparing lbest PSO Niching algorithms Using Different Position Update Rules," in *In Proceedings of the IEEE World Congress on Computational Intelligence*, Spain, 2010, pp. 1564-1571.

[3] Riccardo Poli, "Review Article-Analysis of the Publications on the Applications of Particle Swarm Optimisation," *Journal of Artificial Evolution and Applications*, p. 10, 2008.

[4] Nanbo Jin and Yahya Rahmat-Samii, "Advances in Particle Swarm Optimization for Antenna Designs: Real-Number, Binary, Single-Objective and Multiobjective Implementations," *IEEE Transactions on Antennas and Propagation*, vol. 55, no. 3, pp. 556-567, March 2007.

[5] V.Selvi and R.Umarani, "Comparative Analysis of Ant Colony and Particle Swarm Optimization techniques," *International Journal of Computer Applications*, , vol. 5, no. 4, pp. 1-6, 2010.

[6] Y. Rahamat-Samii and Shenheng Xu, "Boundary Conditions in Particle Swarm Optimization Revisited," *IEEE Transactions on Antennas and Propagation*, vol. 55, no. 3, pp. 760-765, March 2007.

[9] James Kennedy and Tim Blackwell Riccardo Poli, "Particle swarm optimization An overview," *Swarm Intelligence*, vol. 1, no. 1, pp. 33–57, 2007.

[10] Lihua Gong and Zhengyuan Jia, "Multi-criteria Human Resource Allocation for Optimization Problems Using Multi-objective particle Swarm Optimization Algorithm," *International Conference on Computer Science and Software Engineering*, vol. 1, pp. 1187-1190, 2008.

Author Profile:

Maithem Hassen Kareem was born in Nasiriyiah city. Thi-Qar province, Iraq. On November 1981. Received his B.S. degrees in Electrical Engineering from ALmustansiriyah University, Baghdad, Iraq in 2007. Received his MS.C degree in electrical engineer (industrial power) 2014. He is currently working toward the PH.D degree in the department of Electrical Engineering. His research interests include Transient stability of power system using catastrophe theory.

Access this Article in Online	
	Website: **www.ijarm.com**
Quick Response Code	**Subject:** **Electrical Engineering**

How to cite this article:
Maithem Hassen Kareem, Jawad Mahmoud jassim , Nadeem Khargan Al-Hareeb. (2016). Mathematical Modelling of Particle Swarm Optimization Algorithm. Int. J. Adv. Multidiscip. Res. 3(4): 54-59.