Optimal Solution by Application of PSO with Differential Algorithm

Manisha Rani

Department of Computer Science & Applications, Kurukshetra University, Kurukshetra 136119

Email: manishadcsa@gmail.com

Dr. Ramesh Kait

Department of Computer Science & Applications, Kurukshetra University, Kurukshetra 136119

Email: rameshkait@kuk.ac.in

---Abstract---

First scientist were interested only to calculate the shortest path but further they all are moved towards to find out the optimal with different aspects of network (traffic and congestion control). The elements of the network continuously changing their values, to tattle this property: Differential Evolution (DE/DA) was introduced. This category is potentially capable of getting compiled with other evolutionary algorithm and can produce a better result. This paper defines a brief introduction of DE with Practical Swarm Optimisation (PSO). Not every time the local optimal techniques and the discrete optimisation algorithm may solve the entire class of problems. DE algorithms come across to face these exceptions: to handle the situation of global and continuous optimisation. The motivation is to evaluate the DA for solving the continuous optimal path problem and framework of the Metaheuristic Algorithm to present a heuristic for finding the optimal solutions to the shortest path problem.

Keywords: ACO, Differential evolution, Meta-heuristic algorithm, PSO, TSP.

1. Introduction

Else than GA, nowadays, there are so many intelligent shortest path algorithms which are enabled enough with the property to consider the concept of optimal paths too and have used a heuristic method (mathematical functions/ heuristic approach) within the traffic network. Well if we take a glance over the results of all the path related issues. We can easily conclude that they all are producing the result related to the shortest path only; they are not providing any influence to the best and optimal. As there may be the case of having congestion to a single path and which is the shortest too. In those cases these algorithm are not actually fully sufficient to produce the output. So for that instance we have more effective and proficient methods as "*Metaheuristic Algorithm*". In this class of methods the framework is totally unaware of the problem and what the result is going to look like.

Rather than to be described as the primary field 'Metaheuristic algorithm' term is used as the subfield for the stochastic operation. So it is a general class of the algorithm and technologies to employ it to calculate the optimal path for continuous changing, huge, complex & heavy problems. These are the most general type of algorithm to find out the most optimal solution for wide range of situations. The algorithm works well where we don't know the principled manner to calculate the solution. We can use Brute Force search as the searching space is too large, and we have very little knowledge about the issue. But if there is a solution sample than we can test that according to the problem domain. The main problem in meta-heuristic approach is that the user is totally unknown with the nature of problem and its size; and don't attains any sequence to execute that algorithm sufficiently. These are designed to find, generate, or select the complete/ partial search algorithms which are collectively accurate for the production of solution; these algorithms are higher level procedures/ heuristics. These algorithms are very effective where we have incomplete/ imperfect or limited computational capacity (Marek Antosiewicz 2013). More overly as new requirements are arising day by day these high level heuristics are even not sufficient to answer the higher complexity of problem. In some situations these are not effective and need some extra hands to deliver the solution: here comes the first face of '*Differential Evolution / Algorithm*'. A new category of such hybrid algorithm has the property to manage the continuous changing of element as well as with the size and complexity of problem.

Paper Content: Section 2 gives the brief study of DA; the concept of continuous space global optimisation is given in section 3a brief study on DPSO in section 4. Section 5 provides the different class of differential algorithm. Finally in section 6 the conclusion for this paper is provided.

2. Differential algorithm:

But when we talk about the complete different concept of merging any two high levels meta-heuristic approaches then we came across the all new umbrella of *'Differential Evolution / Algorithm'*. This new class of algorithm allows the merging of different field technology under one roof. This enables the users to find out the best result for any problem by merging the properties of different algorithms and also has the feature of managing the continuous changing behaviour of elements (Yu Chen 2014). The simple and straight-forward evolving mechanisms of DE pulls it with the powerful capabilities of solving continuous optimization problems (CoOPs), however, hampers itself to get applied on discrete optimization problems (DOPs). Differential Evolution (DE) is a search heuristic which was introduced by Storn and Price (1997). It has a remarkable performance for global search optimisation with continuous slope of change. Such that it has became a powerful tool for many sophisticated applications. DE completely adhere the class of genetic algorithms (GAs) which use nature and biological operations such as crossover, mutation, and selection on a population set in order to minimize an objective function over the requirement of successive generations. As compared with other evolutionary algorithms (EA), DE solves optimization problems better by evolving a population set of candidate solutions by using alteration and selection operators. Here DE adopted floating-point in place of bit-string encoding of population members, and arithmetic operations are used instead of logical operations in the mutation process, in contradiction to classic GA approach (Differential Evolution with DEoptim 2011).

2.1 PSO:

This is another class of swarm optimisation algorithm which is mostly applied the continuous search space. Originally developed by Kennedly and Eberhart (J.Kennedly 1995). And has become a popular bio-inspired (nature inspired) optimisation principle. The typical PSO algorithm maintains the swarm of elements, and each element presents a solution to that problem in hand. Every element is allowed in the search space according to a particular velocity/ speed . The velocity of elements, in every iteration gets updated with the direction of its own. Best solution or its best individual to its neighbourhood. This algorithm determines the combined behaviour of elements in terms of cognitive and social effects of movement.

2.2 ACO:

This algorithm is specifically derived for the implementation of the biological feature of chemical called as pheromone into topology operation problem (Chun-Yin Wu 2009). When the problem comes under the case of mesh topology this algorithm works well, as it treats every structure to reach the destination and every element model was treated as possible path. The pheromone va.lue on every element gets altered after a single movement and that va.lue was re-evaluated to calculate the shortest and optimal path. ACO is best for mesh topology networks as it can produce optimal structural design at every movement.

2.3 ACO & PSO:

ACO algorithm is totally depends on the cooperation and updation of optimal solution depending upon the value of pheromones. And PSO relay on the population size and the movement of intelligent swarms. Two algorithms can be merged together to produce a hybrid ACO-PSO algorithm to calculate optimal solution for various situations as VANET (J.Amudhavel 2015). PSO was originally developed for social interaction of elements that move between the search spaces to find the best solution. The time complexity of ACO is more as compared to other. And PSO as individual is unable to get communicated with other cluster. The time complexity of both algorithms get reduced when worked together and also get capable enough to produce good optimal results from the searching space.

2.4 GA & PSO:

The very first evolutionary algorithm (GA) is capable enough to work out with large population set. But GA itself is not able to work sound in case of continuous changing elements of population (Muhammad Shahzad 2009). GA needs another algorithm to get merged to it which has the capability of pampering itself to tackle this issue of alteration. So here comes GA with PSO. PSO can work in situation of continuous changing of elements wrt there movements and GA has the property of production of generations with increased fitness value. This combination is also called as dynamic optimiser. PSO & GA are well known for providing efficient online solutions to time varying & dynamic optimisation problems. This result explains well the use of hybrid approach of evolutionary algorithm.

3. Global Optimization over Continuous Spaces:

The problem wakes up when the concept of optimisation comes to area of global space. In general, the task is to optimize some properties of the system by randomly choosing the system parameters. The system parameters are represented as vectors. To implement these, the system needs a model to design an objective function that can process the problem’s objectives while incorporating these and other constraints. Although these methods are often used to make the problem simple, they all are actually inferior to each other in terms of using the objective function (RAINER STORN 1997). In parallel we only concern with the optimisation of the objective function. In most cases the objective is to define the optimality to maximum by minimising the total cost and time requirement. In all these approaches the central strategy is to generate variations of the parameter vectors. Direct search method has the advantage of being applied easily to experimental minimization where the cost value is derived from a physical experiment rather than a computer simulation.

4. DISCRETE PARTICLE SWARM OPTIMIZATION

Another class of evolutionary optimization algorithm is Particle Swarm Optimization (PSO) which is inspired by nature. PSO has been used widely to optimize several continuous\dynamic functions as well as combinatorial problems. Many researchers were used PSO as a tool for solving combinatorial problems because of its simplicity in structure, easy to implement and performance robustness. Particle swarm optimization is inspired by the social common behaviour of bird flocking and fish schooling, and is a population based meta-heuristic. In PSO, every particle represents a solution to itself and the swarm of particles flies in the search space by carrying the motive of reaching the global optimum. Every member (particles\ elements) from the population is maintained throughout the opted search procedure and their information is collectively shared between the elements to direct the search towards the best position within the search space. Individuals' flies along the multidimensional problem space with a particular speed and follow the particles which are known as current best. During every flight the individuals manipulate their positions according to their own experiences and the experience of their neighbourhood particles (Shanthi Muthuswamya 2011). Discrete PSO (DPSO) is the modified application of PSO which emphasises of discrete distinction between the variables. Kennedy and Eberhart (1997) have developed the first DPSO algorithm with binary valued elements. Since then several modifications have been done to this algorithm. DPSO facilitates the solving of combinatorial optimization problems because of its easy implementation, its robustness and simple structure.

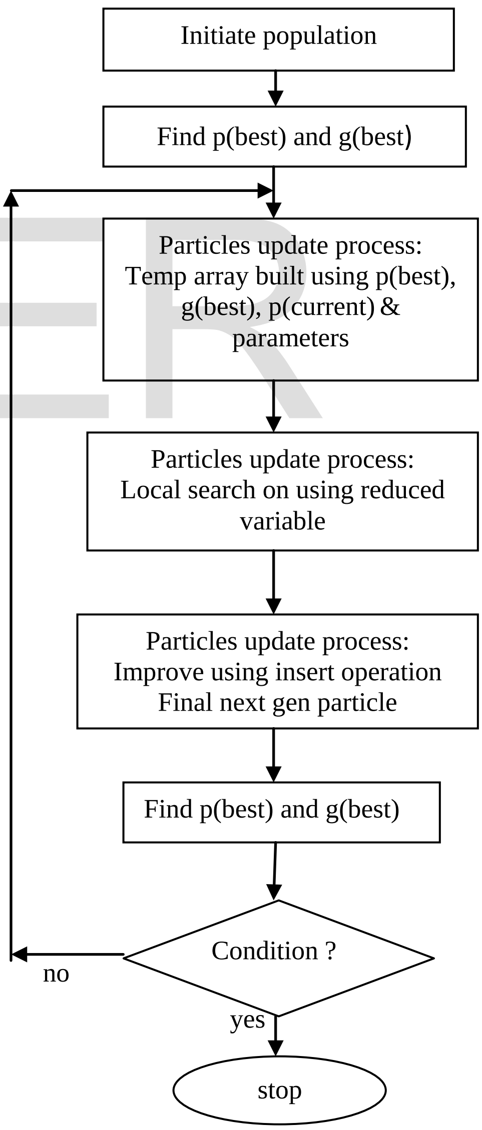

Fig.1 DPSO Flowchart

5. Types of Differential Evolution Algorithm

In comparison to other EAs the performance of DE is more sensitive towards the choice of its control parameters (R. Gämperle WSEAS Press, 2002). Hence, it has become one of the most attractive and appealing research fields in DE community to define appropriate parameter's setting for the DE algorithm to get applied and perform well on different optimization problems and on different evolutionary states. The parameter control methods can be categorized as three classes; deterministic, adaptive, and self-adaptive techniques.

5.1 Deterministic technique: Defines the specific parameter adjusting rules which adjusts the parameters along with the evolutionary process according to these conditions without making use of any information from the search space, e.g., a deterministic rule (S. Das 2005) to linearly make change to the parameter wrt the generation.

5.2 Adaptive technique: The control to the parameters along with the utilization of the search space information during the evolutionary process, e.g., adaptive control wrt the feedback from the search space (Lampinen 2005.).

5.3 Self Adaptive technique: a self-adaptive technique is allowed the parameters to get evolved with the search space during the evolution process.

5.4 Binary Differential Evolution Algorithm: To take the advantage of other operators of evolution algorithm, this type of DA mixes the different type of mutation operation to give the output as a string only. Some trigonometric generating functions are used here to transform the real-coded individuals of DE into binary strings.

5.5 Strategy adaptive technique: A more suitable and efficient generation strategy along with its parameter settings can be used adaptively to match different evolution phases. Especially, with each generation, a set of trial generation strategies with their associated parameters will be assigned differently to different individuals from the current population by viewing the selection probabilities taken from the previous generations (A. K. Qin 2009).

6. Conclusion:

The study focuses on the hybridizations of algorithms with other soft computing tools. It finally discussed the mutual corporation of PSO with DE which leads to a more powerful global\ continuous search algorithm. Applications of these algorithms can diverse the domains of engineering problems have been faced. This paper elaborates one such great application of PSO and DE to design new process. This presents that a significant progress in the field of swarm intelligence and evolutionary computing is needed for their better performance. Engineering search and optimization problems including pattern recognition, bioinformatics and machine intelligence will find new dimensions in the light of hybridization of swarm intelligence with other algorithms.

7. References:

1.A. K. Qin, V. L. Huang, and P. N. Suganthan. "Differential Evolution Algorithm With Strategy Adaptation for Global Numerical Optimization." *IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION* VOL. 13, no. NO. 2 (APRIL 2009): pp 398-417.

2.Chun-Yin Wu, Ching-Bin Zhang, Chi-Jer Wang. "Topology Optimization of Structures

Using Ant Colony Optimisation." *acm*, june 2009.

3.*Differential Evolution with DEoptim.* Vol. 3/1. 2011.

4.J.Amudhavel, K.Prem Kumar, B.Bhuvneshwari,S.Jaiganesh,A.Monica,S.Sampath Kumar. "A HYBRID ACO-PSO BASED CLUSTERING PROTOCOL IN VANET ." *acm* 4, no. 2 (JULY 2015): 124-135.

5.J.Kennedly, R.C.Eberhart. "pactical swarm optimisation." *proc. in IEEE Internationalconference on neural network*, no. IEEE-Press (1995): 1942-1048.

6.Lampinen, J. Liu and J. "“A fuzzy adaptive differential evolution algorithm,” Soft Comput. – A Fusion of Foundation, Methodologies." vol. 9, no. no.6 (Apr 2005.): pp. 448–462.

7.Marek Antosiewicz, Grzegorz Koloch, Bogumił Kamiński. "Choice of best possible metaheuristic algorithm for the." *Journal of Theoretical and Applied Computer Science* Vol. 7, no. NO.1 (2013): 10.

8.Muhammad Shahzad, Saira Zahid,. Muddassar Farooq. "A HYBRID GA PSO FUZZY SYSTEM FOR USER IDENTIFICATION ON SMART PHONES." *acm* 1, no. 1 (2009): 345-351.

9.R. Gämperle, S. D. Müller, and P. Koumoutsakos. "“A parameter study for differential evolution,” in Advances in Intelligent Systems, Fuzzy Systems, Evolutionary Computation." *A. Grmela and N. E. Mastorakis, Eds. Interlaken, Switzerland*, WSEAS Press, 2002: pp 293–298.

10.RAINER STORN, KENNETH PRICE. "Differential Evolution – A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces." Edited by Printed in the Netherlands. *Journal of Global Optimization 11* (Kluwer Academic Publishers), 1997: 341–359.

11.S. Das, A. Konar, and U. K. Chakraborty. ""Two improved differential evolution schemes for faster global search”." *in Proc. Genetic Evol. Comput. Conf.*, 2005: pp. 991-998.

12.Shanthi Muthuswamya, and Sarah Lamb. "DISCRETE PARTICLE SWARM OPTIMIZATION FOR THE ORIENTEERING PROBLEM." *International Journal of Industrial Engineering* 18(2) (2011): 92-102.

13.Yu Chen, Weicheng Xie, Xiufen Zou. "A binary differential evolution algorithm learning from explored solutions." *cs.NE* 2, no. 5 (2014): 16.