NEW APPROACH TO GLOBAL OPTIMIZATION BASED ON PSO

Meparishvili Badri, Qoroglishvili Ciuri
Georgian Technical University

Summary

This paper is devoted to one of the global optimization algorithms, which especially focuses on Evolutionary Computation by discussing Particle Swarm Optimization (PSO). Swarm intelligence is an exciting new research field still in its infancy compared to other paradigms in Artificial Intelligence. Particle swarm optimization algorithms have gained popularity in recent years. PSO is a population-based method, a variant of evolutionary algorithms with moving towards the target rather than evolution, through the search space. The movements of the particles around in the search-space are guided by their own best known position in the search-space as well as the entire swarm's best known position. The improvement of positions is necessary condition to guide the movements of the swarm. The gradient of fitness or cost function, which must be optimized, is not known. The goal is to find a solution in the search-space, which would mean is the global optimum. The process is repeated and by doing so it is hoped, but not guaranteed, that a satisfactory solution will eventually be discovered.

Keywords*:* Global Optimization. Particle Swarm Optimization. Financial Market Risk.

1. Introduction

Particle swarms are attractive to the user as they do not require gradient and derivative information, are intuitive to understand and can be parallelized [1]. They can be used to solve a wide variety of problems, including neural network training [2], static function optimization [3], dynamic function optimization [4], multimodal optimization [5] and data clustering [6].

The idea was originally derived from modeling social behavior, in particular modeling the flight of a flock of birds, the social outlook of this methodology being discussed in [7]. This population based approach is different from other population based evolutionary methods which use some form of evolutionary operators in order to move the population towards the global optimum. Here the "particles", which make up the population move in the search range with a velocity that is determined by a simple equation relating the experience of each individual particle and the population.

This paper proposes a dynamic clustering approach that can adaptively detect local regions and assign particles in different neighborhoods (i.e. clusters). We change the computing mechanism in standard PSO that each particle learns information from its own history best position and the history best position of its nearest neighbor other than the global best one. By using clustering approach, the whole swarm can be adaptively divided into a number of sub-swarms which cover different local regions.

2. Proposed approach

Based on this brief review of metrics to evaluate the quality of swarm behavior, we argue that entropy of swarm cluster, as degree of disorder, can be also calculated using relative positions or average angular velocity by collecting the heading value of the s.

We consider the entropy of dynamic system as an internal behavioral incompatibility or antagonism, certain contradiction between disoriented components behavioral vectors [8]. Hence, the particle swarm behavior metric, which consists in estimation of disoriented particle behavioral vectors can be derived as below.

This approach is based on vector algebraic addition of the velocity-vectors $\vec{v}_i(t)$ of mobile particles at time *t*. Metric of whole particle group in time *t* can be measured as:

$$H(t) = -\sum_{k=1}^{K} P_k(t)\log_2(P_k(t)) \ , \qquad (1)$$

Where:

$$p_k(t) = \frac{\sum_{i=1}^{N} \vec{v}_i(t)}{\sum_{i=1}^{N} | \ \vec{v}_i(t) \ |} , \qquad (2)$$

Each of particles represents a potential solution to an optimization problem, navigate through the search space. The goal of algorithm is to converge to the global (over the search space) or local (into the particular cluster) optimum of a target function [9].

3. Algorithm flow diagram. Assuming that the set of particles with their parameters are given initial part of algorithm proceeds as follow steps [10]:

I. Initialize:

Particle Description. Each particle has three features:

p_k^i – for simplifying the calculation, the value of target function in this position can be identified (this is the *i-th* particle at time or step *k*, notice vector notation) with the coordinates:

$$p_k^i = [x_k^i, y_k^i], \qquad i=1,2, \dots N.$$

The particles are assumed to move within the search space iteratively. This is possible by adjusting their *position* using a proper position shift, called *velocity* (similar to search direction, used to update the position) and denoted as: v_k^i

$f(p_k^i)$ – Fitness or objective (determines which particle has the best value in the swarm and also determines the best position of each particle over time.

For simplicity, we assume the following identify: $f(p_k^i) = p_k^i$.

The swarm is defined as a set:

$$P_k = \{p_k^i\}, \qquad i=1,2, \ldots N.$$

(a) Set parameters N, c_1, c_2, x_{min}, x_{max}, y_{min}, y_{min}, G, μ.

where: c_1, c_2 are weighting factors, called the *cognitive* and *social* parameter, respectively.

The parameters c_1 and c_2 are important control parameters that affect the PSO's convergence.

(b) Set $k \leftarrow 0$.

Initial Swarm.

(c) Generate N particles (in 2-D space) with random locations i.e. positions with their coordinates and "velocities" (*the steps) for each particle.*

$$p_0^i = p_{min} + rand(p_{max} - p_{min}). \qquad (3)$$

where: p_{min} and p_{max} are vectors of lower and upper limit values respectively.

Evaluate the fitness of each particle and store:

• particle best ever position (particle memory b^i here is same as p_0^i).

• Best position in current swarm.

Initial velocity is randomly generated.

$$v_0^i = \frac{p_{min} + rand(p_{max} - p_{min})}{\Delta t}. \qquad (4)$$

II. Classification:

(a) Fitness function p_k^i evaluation for each particle in given coordinates.

(b) After sorting the set of particles by decreasing of the values p_k^i we can sort out the leader particle (with the best position).

(c) Composition of new classified set of particles by the distance from the leader particle. This new set is dividing by the tangent of linear regression straight line.

(d) Election the leader (or leaders) as best position and the outsiders in the cluster (or clusters). Given a set of leaders with their positions $l_r = \{p_k^i\}$, r=1,2, …M.

(e) Clustering of swarm (part of outsider particles around of each leader) by K-Means algorithm.

III. Clustering:

K-means clustering aims to partition the *N* outsiders into *M* sets: $L = \{l_r\}$, r=1,2, …M, so as to minimize the within-cluster sum square:

$$\underset{L}{\operatorname{argmin}} = \sum_{l_c=1}^{M} \sum_{p_k^i \in L} \left\| p_k^i - p_k^j \right\|^2$$

IV. Updating:

(a) Velocity Update:

– Provides search directions.

– Includes deterministic and probabilistic parameters.

– Combines effect of current motion, particle own memory, and swarm influence.

$$v_{k+1}^{i} = wv_{k}^{i} + c_1 rand \frac{(p_k^l - p_k^i)}{\Delta t} + c_2 rand \frac{(p_k^g - p_k^i)}{\Delta t} \quad (5)$$

where:

w – inertia factor;

p_k^l – local best position;

p_k^g – global best position;

wv_k^i – current motion;

$\frac{(p_k^l - p_k^i)}{\Delta t}$ – particle memory influence;

$\frac{(p_k^g - p_k^i)}{\Delta t}$ – swarm influence.

This paper evaluates an adaptive approach to tune the c_1 and c_2 based on proportions:

$c_1 = p_k^l / p_k^g$

$c_2 = 1 - (p_k^l / p_k^g)$

(b) Position Update:

Position of each particle is updated by own velocity vector.

$$p_{k+1}^{i} = p_k^i + v_{k+1}^{i} \Delta t \quad (6)$$

Constraints: If a particle is infeasible, last search direction (velocity) was not feasible. Set current velocity to zero.

$$v_{k+1}^{i} = c_1 rand \frac{(p_k^l - p_k^i)}{\Delta t} + c_2 rand \frac{(p_k^g - p_k^i)}{\Delta t} \quad (7)$$

(c) Memory Update:

At each iteration, after the update and evaluation of particles, best positions are also updated. Thus, the new best position p_{k+1}^g of leader l_{k+1}^r at iteration k+1 is defined as follows:

$$l_{k+1}^r = p_{k+1}^g = [x_{k+1}^g, y_{k+1}^g], \qquad r = 1,2, \ldots M. \quad (8)$$

$$p_{k+1}^g = \begin{cases} p_{k+1}^i & if\ f(p_{k+1}^i) \le f(p_k^g), \\ p_k^g & Otherwise \end{cases} \quad (9)$$

(d) Set $k \leftarrow k+1$.

V. Cycling

At each iteration, an angle of linear regression tangent decreases in proportion to golden section. Consequently, the dimension of set of leaders decreases too.

VI. Stopping Criteria

If the angle of linear regression tangent is equal to zero or less than μ.

Stopping criteria satisfied?

If "Yes", then end. It means that one leader remains and global optimum achieved.

If "No", go to II (a).

(e) Output results.

Consequently, we use a conditionally fixed number of fitness evaluations or swarm iterations as a stopping criteria.

5. Conclusion

This work is motivated by the idea that the financial market risks management can be realized by continuously collecting particles from a financial market data warehouse. Relevance of problems is particularly pointed by the assessment and forecasting of fitness function, using Artificial Neural Network or Kohonen Network, which consists of an uncertainty and dynamism of financial processes. We have discussed different kind of metrics to particle swarm behavior as well. We defined some number of metrics, which will help us in evaluation of performance of the swarming behavior.

References:

1. Parsopoulos K. E., Vrahatis M. N.. Particle swarm optimization and intelligence: advances and applications. Published in the United States of America by Information Science Reference. Hershey, New YorkISBN 978-1-61520-666-7, 2009

2. Schutte J.F., Reinbolt J.A., Fregly B.J. Haftka R.T., George A.D., Parallel Global Optimisation with the Particle Swarm Algorithm. International Journal of Numerical Methods in Engineering, 61, 2004

3. Shi Y.H., Eberhart R.C. Empirical Study of Particle Swarm Optimisation. Proceedings of the Congress on Evolutionary Computation, (Washington D.C. USA), IEEE Service Centre, Piscataway, NJ, 1995, pp. 1945-1949

4. Shi, Y.H., Eberhart, R.C. Parameter selection in particle swarm optimisation. Annual Conference on Evolutionary Programming, San Diego, 1998

5. Blackwell T., Branke J. Multi-swarm Optimisation in Dynamic Environments. Applications of Evolutionary Computing (2004) pp. 489-500

6. Brits R., Engelbrecht A.P., F. van den Bergh. A Niching Particle Swarm Optimiser. Fourth Asia-Pacific Conf. on Simulated Evolution and Learning, 2002, pp 692-696.

7. Cohen S.C.M, de Castro L.N. Data Clustering with Particle Swarms. IEEE Congress on Evolutionary Computation, 2006, pp.1792-1798

8. Kervalishvili P., Meparishvili B., Janelidze G. Self-Organization Modelling Of Multi-Agent Systems". SynEnergy Forum (S.E.F.) -2 The conf. for International Synergy. 2009

9. Kennedy J., Eberhart, R.C. A discrete binary version of the particle swarm algorithm. Proc. 1997 Conf. on Systems, Man, and Cybernetics, 4104–4109. Piscataway, NJ: IEEE Service Center.

10. Kervalishvili P., Meparishvili B., Janelidze G. One Approach to Adaptive Control of Mobile System, NATO Science for Peace and Security Series "Philosophy and Synergy of Information: Sustainability and Security", ISSN 1874-6276. 2011, pp. 100-108.

НОВЫЙ ПОДХОД К ГЛОБАЛЬНОЙ ОПТИМИЗАЦИИ, ОСНОВАННЫЙ НА МЕТОД РОЯ ЧАСТИЦ

Мепаришвили Б., Короглишвили Ц.
Грузинский Технический Университет

Резюме

Эта статья посвящена одному из алгоритмов глобальной оптимизации, который основывается на Эволюционном Вычислении, в частности на оптимизационном методе роя частиц. Метод роя – эта новая область исследования все еще находиться в начальной стадии по сравнению с другими парадигмами в области искусственного интеллекта. Алгоритмы оптимизации роя частиц завоевали популярность в последние годы. Алгоритмы оптимизации роя частиц основанны на поведении роя, вариант эволюционных алгоритмов с движением цели в области поиска. Движения частиц вокруг области поиска управляются их собственным положением в области поиска, а также положением всего роя. Улучшение положений - необходимое условие передвижения роя. В статье предлогается новый подход динамической кластеризации. Процесс итерационно повторяется и таким образом отсеиваются множества лидеров до достижения в конечном счете глобального оптимума.

ნაწილაკთა გროვის მეთოდზე დაფუძნებული გლობალური ოპტიმიზაციისადმი ახალი მიდგომა

ბადრი მეფარიშვილი, ციური ქოროღლიშვილი
საქართველოს ტექნიკური უნივერსიტეტი

რეზიუმე

განიხილება გლობალური ოპტიმიზაციის ერთერთი ალგორითმი, რომელიც ეფუძნება ევოლუციურ გამოთვლებს, კერძოდ ნაწილაკთა გროვის ოპტიმიზაციურ მეთოდს. ნაწილაკთა გროვის მეთოდი - წარმოადგენს კვლევის ახალ მიმართულებას, რომელიც ჯერ კიდევ საწყის სტადიაში იმყოფება ხელოვნური ინტელექტის სფეროში არსებულ პარადიგმებთან შედარებით. ნაწილაკთა გროვის ოპტიმიზაციის ალგორითმებმა საკმაო პოპულარობა მოიპოვეს უკანასკნელ ხანებში. ნაწილაკთა გროვის ოპტიმიზაცია წარმოადგენს ხელოვნურ ნაწილაკთა პოპულაციაზე დაფუძნებულ სტოქასტიკურ-ევრისტიკულ მეთოდს, ევოლუციური ალგორითმების ერთერთ ვარიანტს, სადაც საძიებელ სივრცეში მიზნისაკენ გადაადგილება ხდება თანამიმდევრულად, ევოლუციურად. მისი მოდელი შეიცავს ნაწილაკთა გროვას, რომლის პოპულაციის ინიციალიზაცია ანუ ნაწილაკთა საძიებელ სივრცეში „გაბნევა“ ხდება რანდომიზებულად. ნაწილაკთა გროვის ოპტიმიზაციის ალგორითმში პრობლემის ამონახსნი მიიღება სწორედ ამ ინდივიდუალური აგენტების სიმრავლის ინტერაქციის შედეგად.