PSO with Control of Velocity Change for Feature Selection

S.A. Subbotin, Al.A. Oleynik

Department of program tools, Zaporozhye National Technical University,
69063, Zaporozhye, Zhukovskiy str., 64. Ukraine

subbotin@zntu.edu.ua, lex1@mail.zp.ua

Abstract. *Particle Swarm Optimization (PSO) is analyzed. Using PSO for feature selection problem solving is considered. PSO with control of velocity change for feature selection problem solving is proposed.*

Keywords

Feature Selection, Particle Swarm Optimization, Sample,
Swarm Intelligence

1 Introduction

For synthesis of complex objects and systems models it is necessary to carry out processing significant samples of data that is very much labour-intensive process in case of application of traditional optimization methods. That's why the methods based on probabilistic approach are applied. One of new directions of the methods based on probabilistic approach, are multiagent methods based on modelling of swarm intelligence of social animals, insects and other (Swarm Intelligence) [1]

Multiagent methods of swarm intelligence are: ant colony optimization (ACO) [2, 3], bee colony optimization (BCO) [4,5], particle swarm optimization (PSO) [6], bacteria foraging optimization (BFO) [7] and others [8]. This methods are used widely for more optimization problems solving [2–8]. In this work particle swarm optimization is considered. PSO traditionally is used for optimization of multivariable function [6, 9] and her application for other optimization problems solving is not very good developed. That's why development of new modifications of PSO for different optimization problems solving is actually.

Purpose of this work is development of PSO modification for feature selection problem solving, which is very important problem for synthesis of complex objects and systems models.

2 Particle Swarm Optimization

The PSO approach utilizes a cooperative swarm of particles, where each particle represents a candidate solution, to explore the space of possible solutions to an optimization problem [9–11]. Each particle is randomly or heuristically initialized and then allowed to 'fly'. At each step of the optimization, each particle is allowed to evaluate its own fitness and the fitness of its neighboring particles. Each particle can keep track of its own solution, which resulted in the best fitness, as well as see the candidate solution for the best performing particle in its neighborhood.

Base steps of PSO for optimization fitness function $f(x_1, x_2, \ldots x_{nx})$ are next.

Step 1. Set parameters of method: n_s – agents count, which are modeled behaviour of particle swarm; n_x – variables count in fitness function.

Step 2. If conditions of termination are executed then halt, otherwise – go to step 3.

Step 3. Create and initialize n_x-dimensional swarm.

Step 4. Set: $i = 1$.

Step 5. Local best position is defined. If condition $f(x_i)<f(y_i)$ is true then set $y_i=x_i$, where x_i – the position of agent i; y_i – local best position of agent i.

Step 6. Global best position is defined. Set: $y^*=y_i$, where y* – global best position, which was chosen from solutions of all agents.

Step 7. Set: $i = i + 1$.

Step 8. If $i<n_s$, then go to step 3, otherwise – go to setp 9.

Step 9. Set: $i = 1$.

Step 10. Update velocities of swarms:

$$v_{ij}(t+1) = v_{ij}(t) + c_1 r_{1j}(t)[y_{ij}(t) - x_{ij}(t)] + c_2 r_{2j}(t)[y^*_j(t) - x_{ij}(t)],$$

where $v_{ij}(t)$ – velocity of particle i in dimension $j = 1, ..., n$ at time t; $x_{ij}(t)$ – position of swarm i in dimension j; c_1 and c_2 – positive constants of acceleration; $r_{1,j}$, $r_{2,j} = rand(0,1)$ are random numbers from range [0,1]. These random variables introduce a stochastic element to work of method.

Step 11. Update swarm positions:

$$x_i(t + 1) = x_i(t) + v_i(t + 1).$$

Step 12. Set: $i = i + 1$.

Step 13. If $i<n_s$, then go to step 10, otherwise – go to step 2.

3 PSO with control of velocity change for feature selection

Modeling of particles moving by PSO in search space $X = \{x_1, x_2, \dots x_M\}$, $x_i = \overline{1, N}$ (N – count of all features in sample, M – a feature count, which should be left in the reduced set) for feature selection problem solving is offered. Error of model constructed on position of particle is proposed to use as fitness-function.

Good optimization method must have two important properties: he must entirely investigate a search space of problem and fix search near a potential optimum. In other words, the method should balance between two these conditions. In PSO-method the balance is determined by the formula of particle velocity.

In early realization of PSO it was revealed, the velocities can sharply rise, especially velocities of those particles which are far from an own optimum or the general optimum of neighbors. In result, such particles can leave a search space, that extremely negatively influences work of the method as a whole. For prevention of similar situations it is necessary to control a range of particle velocities change. If a particle velocity exceeds as much as possible allowable it should be lowered to the allowed level. Let $V_{\max,j}$ is a maximum allowed velocity of particle in dimension j. Then particle velocity is offered to be changed as follows:

$$v_{ij}(t+1) = \begin{cases} v_{ij}^{`}(t+1), \text{ if } v_{ij}^{`}(t+1) < V_{\max,j}; \\ V_{\max,j}, \text{if } v_{ij}^{`}(t+1) \ge V_{\max,j}. \end{cases}$$

where $v_{ij}^{`}$ is calculated by formula for v_{ij}.

Variable $V_{\max,j}$ is very important beacuse it contol common scale of the search. If value of this variable is big then PSO investigate a search space more careful. If value of $V_{\max,j}$ is small then PSO can not leave local areas, and also count of the iterations necessary for achievement of optimum increases.

So, it is clear, that $V_{\max,j}$ it is necessary to select so that to balance between fast and slow movement of particles in a search space, and between fixing of search and the common investigate of space. It is offered to calculate $V_{\max,j}$ as follows:

$$V_{\max,j} = \delta \cdot (N–M),$$

where $\delta \in (0,1]$ – experimental coefficient.

There are two important properties in proposed modification of PSO.

1. Variable $V_{\max,j}$ constrains not search space inside which particles move but only velocities of particles (is more exact the range of their change for one iteration).

2. Maximum value of particle velocity is defined for each measurement separately and it is determined by its dimension.

In spite of the fact that in the offered modification velocities of particles are controlled by variable $V_{\max,j}$ the proposed method has also lacks. Restriction of velocity of particle also can cause change of a direction of its movement. On the other hand, such "intervention" can direct particles in the direction of an optimum.

The second undesirable effect can arise in case when velocities of all particles become equal $V_{\max,j}$. For his prevention it is possible to suggest to reduce value $V_{\max,j}$ with increase of iterations. Then the modified PSO will consistently narrow scales of research of search space that makes optimization process more effective.

Proposed PSO with control of velocity change for feature selection has been implemented in environment of packet Matlab 7.0. The choice of Matlab is connected by that it contains powerful libraries of mathematical functions for construction of numerical models of dependences. The experiments have shown that using of PSO with control of velocity change for feature selection provide enough results at small expenses of time resources [12].

4 Acknowledgments

In work actual problem of feature selection based on PSO with control of velocity change has been solved.

Scientific novelty of work consists that has received the further development the PSO: on its basis the PSO with control of velocity change for feature selection is developed. Proposed modification takes into account the lacks of the base PSO connected to an opportunity of sharp rise of separate particles velocities. Control of velocity change allows to avoid the give undesirable effect.

References

[1] Beni G., Wang J. Swarm Intelligence // Annual Meeting of the Robotics Society: Proceedings of Seventh International Conference. – Tokyo: RSJ Press, 1989. – P. 425-428.

[2] Dorigo M., Maniezzo V., Colorni A. The Ant System: Optimization by a colony of cooperating agents // IEEE Transactions on Systems, Man, and Cybernetics. – 1996. – Part B, №26(1). – P. 29–41.

[3] Subbotin S.A., Oleynik A.A., Yatzenko V.K. Feature Selection Based on the Modification of Ant Colony Optimization Method // Radioelectronics and Informatics. – 2006. – №1. – P. 65–69.

[4] Camazine S., Sneyd J. A Model of Collective Nectar Source by Honey Bees: Self- organization Through Simple Rules // Journal of Theoretical Biology. – 1991. – №149. – P. 547-571.

[5] Sumpter D.J.T., Broomhead D.S. Formalising the Link between Worker and Society in Honey Bee Colonies // Lecture Notes In Computer Science: Proceedings of the First International Workshop on Multi-Agent Systems and Agent-Based Simulation (MABS '98). – Berlin: Springer, 1998. – P. 95-110.

[6] Kennedy J., Eberhart R.C. Particle swarm optimization // Proceedings of the 1995 IEEE International Conference on Neural Networks, Vol. 4. – NJ: IEEE Press, 1995. – P. 1942–1948.

[7] Liu Y., Passino K.M. Biomimicry of Social Foraging Bacteria for Distributed Optimization: Models, Principles, and Emergent Behaviors // Journal of Optimization Theory and Applications. – 2002. – №3 (115). – P. 603–628.

[8] De Castro L.N., Von Zuben F.J. Artificial Immune Systems. Part I. Basic Theory And Applications. – Technical Report No. Rt Dca 01/99. – Brazil: Feec/Unicamp, 1999. – 95 p.

[9] Eberhart R., Kennedy J. A New Optimizer using Particle Swarm Theory // Symposium on Micro Machine and Human Science: Proceedings of the Sixth International Symposium on Micro machine and Human Science. – Nagoya: IEEE Service Center, 1995. – P. 39-43.

[10] Abraham A., Grosan G. Swarm Intelligence in Data Mining. – Berlin: Springer, 2006. – 267 p.

[11] Peer E., Engelbrecht A. Using Neighborhoods with the Guaranteed Convergence PSO // Swarm Intelligence Symposium: Proceedings of the IEEE SIS-2003. – Indianapolis: IEEE Press, 2003. – P. 235-242.

[12] Boguslave V.A., Yatzenko V.K., Pritchenko V.F. Technological supporting and prediction of suffer ability of gas turbine engine details. – Kiev: Manuscript Pub., 1993. – 333 p.