5K-2

Introducing the Evolutionary Dynamics of Prisoner's Dilemma into Particle Swarm Optimization with Population Structure

Haoli Sun, Reiji Suzuki, Takaya Arita

Abstract

This paper proposes the introduction of the evolutionary dynamics of Prisoner's Dilemma (PD) into a Particle Swarm Optimization (PSO) algorithm with population structure. The dynamics of PD affect the range of information transmission of each particle, and the distribution of particles within the search space affects the payoff matrix for each particle. Our goal is to use this interaction as a feedback mechanism, letting it automatically adjust the balance between exploration and exploitation in response to the state of search process. Our preliminary evaluation shows that the introduction of PD into PSO can help PSO search avoid premature convergence.

I. INTRODUCTION

PSO is a swarm intelligence based optimization method originally proposed by Kennedy and Everhart [1]. In PSO, a solution space is searched using a population of particles, each of whose movement within the search space (function) is typically derived from the best points found so far by the particle itself and some subset of the other particles. The uniqueness of the method lies in the dynamic information exchange between the particles [2]. Therefore, the rules that determine which particles will exchange information are an important aspect of PSO algorithms, and many methods have been investigated. The simplest rule is known as *gbest*, in which the best point in the entire population is used. However, adopting *gbest* tends to lead to premature convergence, owing to a lack of diversity in particle movement.

Some researchers have proposed using complex networks to represent a relationship structure between particles, with particles sharing information only with their direct neighbors on the network. Based on the results of a comparative study, Kennedy recommended using the von Neumann configuration, as it performed more consistently than other topologies [2]. Some other researchers have proposed methods in which the network topology changes dynamically [3][4]. However, as far as we know, there are few or no PSO methods that gradually adjust the topology of the particle network in response to the current state of the search process.

G1: The interaction between PSO and the dynamics of the prisoner's dilemma game.

In this paper we propose a novel algorithm in which the balance between exploration and exploitation is automatically adjusted to the state of the search process, by introducing the prisoner's dilemma (PD) game into the PSO algorithm, and allowing these two elements to interact dynamically, as illustrated in G1. This paper explains our method and presents a preliminary evaluation.

II. THE METHOD

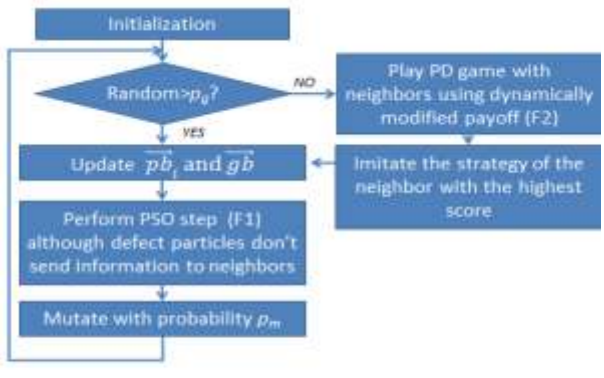
The proposed algorithm is shown in G2. With probability p_g, each particle plays a PD game with its direct neighbors while executing the usual PSO update steps (F1.1-3) looking for the optimal solution. A particle's *score* is obtained by multiplying its PD game payoff with its relative fitness (which in turn derives from its function value) (F2). Fitness normalization is performed each iteration, so that the maximum and minimum fitness in the population are always 1 and 0, respectively. The particles then adopt the PD strategy (Cooperate or Defect) of the particle with the highest score among their neighbors. The PSO update steps are performed as usual, except that the particles currently using the Defection strategy do not inform their neighbors of their personal best point, although they do receive information from their cooperating neighbors.

$$\vec{v}_i = c0\left(\vec{v}_i + rand1 * c1(\overrightarrow{pb}_i - \vec{x}_i) + rand2 * c2(\overrightarrow{gb} - \vec{x}_i)\right) \quad \text{(F1.1)}$$

$$\vec{x}_i = \vec{x}_i + \vec{v}_i \quad \text{(F1.2)}$$

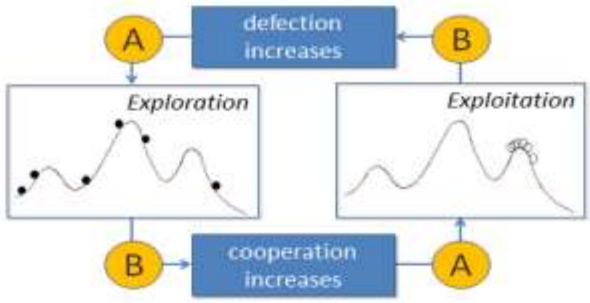
$$c0 = 0.729844 \text{ and } c1 = c2 = 2.01 \quad \text{(F1.3)}$$

Our intention with this design is as follows (G3). Excessive exploration will lead to high variation in the function values of particles. This enlarges the effect of payoff modification, which should drive up the number of cooperators. Cooperators share their information with others, which reduces locality. This accelerates convergence, shifting the system's global behavior toward exploitation. Conversely, excessive exploitation will reduce variation among particles' function values. This reduces the effect of payoff modification, which should increase the number of defectors. Defectors promote locality by hiding their information, thus accelerating exploration. We expect this feedback mechanism to adaptively adjust the balance between exploration and exploitation.



G2: The proposed algorithm.

$$\text{score} = \text{average\_payoff (in PD game with neighbors)}$$
$$* \ relative\_fitness^{\alpha} \qquad \text{(F2)}$$



G3: Dynamic adjustment of the balance between exploration and exploitation.

III. PRELIMINARY EVALUATION

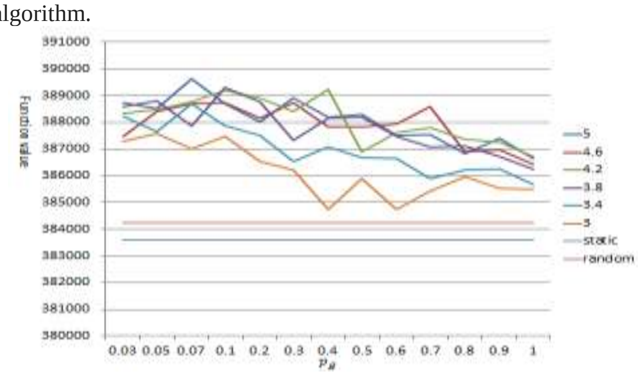
We performed a preliminary evaluation of the proposed method using F3 as a benchmark.

$$f(x, y) = x * \sin\left(\frac{\sqrt{|x|}}{2\pi} * 360\right) + y * \sin\left(\frac{\sqrt{|y|}}{2\pi} * 360\right)$$

$$x \in (-200000, 200000) \quad y \in (-200000, 200000) \qquad \text{(F3)}$$

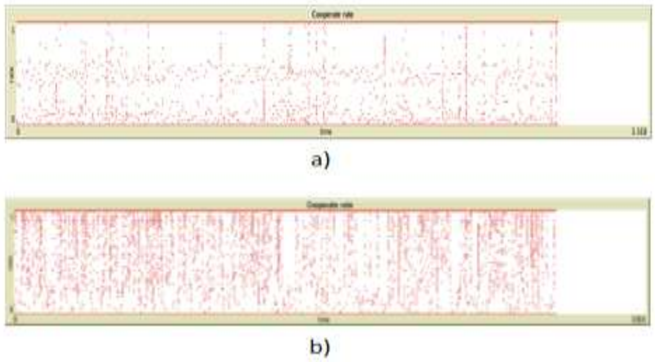
Following Kennedy's recommendation [2], we adopted a toroid von Neumann neighborhood topology, without dynamic modification. The payoff values for each strategy pair (self, opponent) were set as follows: (C, C), (C, D), (D, C) and (D, D) yield scores of 3, 0, T and 1, respectively. T was varied over 3.0~5.0 in intervals of 0.2. The other parameters were set as follows: number of agents = 48, c0 = 0.729844, c1 = c2 = 2.01, α = 1.0, cooperation probability for each particle in the initial population = 0.5, mutation probability p_m = 0.007, probability for playing the PD game p_q: {0.03, 0.05, 0.07} and 0.1~1.0 in intervals of 0.1. For every possible value combination for T and p_q we ran 300 trials of 10000 steps each. The average of the best fitness for each (T, p_q) pair was calculated in order to evaluate the performance of the method.

Results are shown in G4. The horizontal axis represents p_q (the probability for playing the PD game) and the vertical axis represents the fitness (function value) in the optimization. This figure additionally includes the results without strategy imitation (and hence without the regulatory feedback mechanism) for comparison: "static" shows the results obtained with populations composed entirely of cooperators, and in "random" each particle choose to cooperate or defect randomly every step. We see from the graph that all experiments with strategy imitation outperformed the two experiments without, although the difference is 1-2%. This result supports our hypothesis that the interaction dynamics we introduced can improve PSO performance as intended, likely by avoiding premature convergence. We observe that performance tended to be better when the value of T was comparatively small. We also found that there was a mild performance peak around p_g = 0.1. These findings suggest that the dilemma properties of the game play an important role in our algorithm.



G4: Average of best function value found under various settings of parameters p_g and T.

G5 a) and b) show the transitions of the proportion of cooperative particles in the best case (T = 5 and p_g = 0.07) and the worst case (T = 3.2 and p_g = 0.9), respectively. We observe that in the best case, cooperative particles dominated the population less frequently than in the worst case, which made the particles explore more, so that the particles might have avoided the premature convergence better. We also observe that, compared to the best case, the worst case has the dominant strategy jumping back and forth between cooperation and defection very frequently, which may preclude the particles from exploring and exploiting effectively.



G5: Portion of cooperative particles in different situations. Transitions of the portion of cooperative particles measured under our best (a: T = 5 and p_g $= 0.07$) and worst (b: T = 3.2 and p_g $= 0.9$) settings.

IV. CONCLUSION

We have proposed a new particle swarm optimization (PSO) method which automatically adjusts the balance between exploration and exploitation depending on the state of the search, by introducing interaction between the dynamics of prisoner's dilemma game and PSO search. We conducted a preliminary evaluation, which showed that this technique can improve PSO performance. We are presently investigating in further detail how this feedback mechanism is controlling the balance between exploration and exploitation.

REFERENCES

[1] Kennedy, J. and Everhart, R. 1995. Particle Swarm Optimization. *Proceedings of IEEE.*

[2] Kennedy, J. and Mendes, R. 2002. Population Structure and Particle Swarm Performance. *Proceedings of Congress on Evolutionary Computation.*

[3] Matsushita, H. 2012. Swarm-Based Optimization with Dynamically-Changing Topology. *IEEE Workshop on Nonlinear Circuit Networks.*

[4] Richards, M. and Ventura, D. 2003. Dynamic Sociometry in Particle Swarm Optimization. *Proceedings of the Sixth International Conference on Computational Intelligence and Natural Computing, 1557-1560, Cary, North Carolina, September.*