A Simple Strategy to Maintain Diversity and Reduce Crowding in Particle Swarm Optimization

Stephen Chen
School of Information Technology
York University
4700 Keele Street, Toronto, ON
(416) 736-2100 x30526
sychen@yorku.ca

James Montgomery
Faculty of ICT
Swinburne University
of Technology, Hawthorn, VIC
+61 3 9214 5735
jmontgomery@swin.edu.au

ABSTRACT
Each particle of a swarm maintains its current location and its personal best location. It is useful to think of these personal best locations as a population of attractors. When this population of attractors converges, the explorative capacity of the swarm is reduced. The convergence of attractors can occur quickly since the personal best of a particle is broadcast to its neighbours. If a neighbouring particle comes close to this broadcasting attractor, it may update its own personal best to be near the broadcasting attractor. This convergence of attractors can be reduced by having particles update the broadcasting attractor rather than their own attractor/personal best. Through this simple change which incurs minimal computational costs, large performance improvements can be achieved in multi-modal search spaces.

Categories and Subject Descriptors
I.2.8 [**Artificial Intelligence**]: Problem Solving, Control Methods, and Search – *Heuristic methods*

General Terms
Algorithms

Keywords
Particle swarm optimization, crowding

1. INTRODUCTION
To improve the balance between exploration and exploitation, standard PSO [1] recommends a ring topology – each particle communicates with only two neighbours. In a ring topology, a single good location will not immediately attract all of the other particles in the swarm. Specifically, several different locations can each act as the attractor for a small subset of particles, and the overall swarm can subsequently explore many regions of the search space. This increased exploration generally improves PSO performance in multi-modal search spaces [1].

In population search, one means to increase exploration is to reduce crowding (e.g. [2]). Crowding occurs when two or more population members are too close. As crowds gather, population diversity is reduced and the explorative capacity of the search process is similarly reduced. To counteract crowding, a new solution can be compared to the most similar existing population member. By replacing the nearest existing member in the population, the new solution cannot form a crowd with it.

Viewing the personal best locations as a population of attractors, it is easy to see how the standard operation of PSO promotes crowding. As an attractor draws another particle towards it, the purpose is to indeed have that other particle search the area around the broadcasting attractor. If the attracted particle subsequently finds a new personal best location near the broadcasting attractor, it will update its own personal best attractor to be near the broadcasting attractor – i.e. crowding.

Crowding is avoided in population search by having the new solution replace the existing population member with which it would crowd. The transfer of this idea to PSO requires a significant conceptual change to the meaning of the "personal best" (*pbest*) position. In the modified PSO, *pbest* represents only an attractor to a promising area in the search space – the best location ever visited by the particle may now be stored in the *pbest* of another particle.

The above strategy to reduce crowding has been implemented. Starting with standard PSO [1] (and its ring topology), the procedure to update *pbest* locations is changed to first check if the new location is close to its local best (*lbest*) attractor. If the new location is within a "threshold" distance to its *lbest* attractor, it is compared with and potentially updates this attractor. Outside this threshold distance, standard PSO comparisons and updates occur. The effectiveness of this strategy is tested across a broad range of benchmark functions.

2. CROWDING IN PSO
The basic technique of crowding is to compare each new solution with its most similar individual in a subset of the overall population. The fittest of these two solutions survives as a member of the population. The size of the subset to find a neighbour for comparison can be small, which can cause "replacement errors", or it can be large, which can cause significant increases to the required computational effort [2]. The new strategy to reduce the rate of convergence in particle swarm optimization is most similar to crowding with a subset of size two: the personal best and the local best for each particle.

Figure 1 shows an example of how the new strategy might behave. The attraction to *lbest* has helped to draw the particle to a position x_i that is close to *lbest*. Assuming that $f(x_i) < f(pbest)$, the standard update procedure would update *pbest* to become x_i. Since *lbest* is the *pbest* for another particle, the standard update

GECCO'11, July 12-16, 2011, Dublin, Ireland.
ACM 978-1-4503-0690-4/11/07.

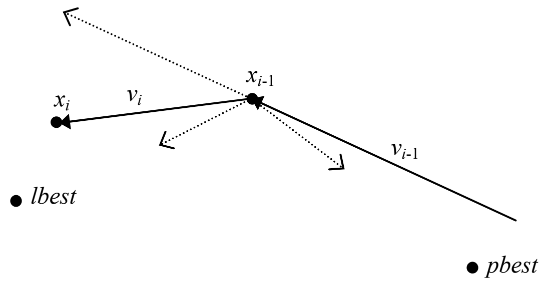

Figure 1. Particle update in PSO

procedure encourages the formation of a crowd between these two *pbest* attractors.

The crowding of attractors can have a cascading effect in PSO (even with a ring topology). For example, assume that the *pbest* for particle 1 is the *lbest* attractor for particle 2. After an update like the one shown in Figure 1, consider the case that the new *pbest* for particle 2 becomes the *lbest* attractor for particle 3. This third particle will now be drawn towards this area with a high concentration of *pbest* attractors. If it also finds a new *pbest* in this area, the population of *pbest* attractors can converge very quickly.

Focusing on this population of *pbest* attractors, the key concept from crowding is that a new solution should replace the most similar member in the existing population. Therefore, instead of replacing *pbest* in Figure 1, the new location *x* should replace *lbest*. To update *lbest* means that the *pbest* for a neighbouring particle is updated.

Essentially, the proposed approach separates the two roles of *pbest*: store the best known location and act as an attractor in the search space. The swarm as a whole still remembers the best known location (which is stored in *lbest*), but now *pbest* and *lbest* are still far apart in the search space. This reduction in crowding maintains diversity in the population of attractors which encourages the particles to explore a larger area of the search space.

3. RESULTS

The following experiments compare the performance of standard PSO [1] with a modified version that replaces the normal *pbest* update procedure with the new strategy. Preliminary experiments with this modified PSO determined that a "threshold" parameter was required to properly manage the new balance between exploration and exploitation. As shown in Figure 2, the new update procedure makes comparisons with *lbest* within the threshold distance and it performs standard PSO updates beyond the threshold distance.

```
if |xi - lbest| < threshold then
  if f(xi) < f(lbest) then
    lbest = xi
else if f(xi) < f(pbest) then
  pbest = xi
```

Figure 2. New update procedure

The parameter tuning experiments revealed that the *threshold* should decay over time (to allow the swarm to converge), and that the threshold should only be applied to *lbest*. If the new update strategy is applied to all of the particles/*pbests* in the population, the computational effort is much larger and the performance is much worse – a result presumably caused by a complete lack of convergence. The *threshold* used in the following experiments starts with an initial value of 10% of the search space diagonal (i.e. *spacing* = 0.1), and it decays with a cubic function – in (1), *n* is the total number of iterations and *i* is the current iteration.

$$threshold = (spacing * diagonal) * ([n - i] / n)^3 \quad (1)$$

The multi-modal functions used in the experiments are Fletcher-Powell, Langerman (with m = 7), Rastrigin, Schwefel, and Shubert, and all functions are in D = 20 dimensions. For 50 independent runs of 100,000 function evaluations each (i.e. 5000 * D), the mean and standard deviation (std dev) are presented in Table 1. The t-tests show that the current results are on the edge of strong statistical significance. Future research will include additional parameter analysis.

Table 1. Improvements with the new update strategy

Function	PSO	mean	std dev	t-test
Fletcher-Powell	Std	10,460	8,173	1.8%
	New	7,258	4,387	
Langerman m = 7	Std	-0.399	0.118	5.2%
	New	-0.440	0.086	
Rastrigin	Std	28.77	8.01	3.0%
	New	23.75	8.23	
Schwefel	Std	1,605	347	0.0%
	New	1,139	272	
Shubert	Std	-3.77e+21	5.16e+21	5.0%
	New	-7.34e+21	1.15e+21	

4. DISCUSSION

The proposed modification is simple and computationally efficient. The changes to standard PSO add a distance calculation that is only required between two specific points – the position of a particle and the position of its *lbest* attractor. In the current experiments, the total increase in running time is around 1%.

In general, diversification strategies based on crowding are either computationally expensive (as distances between a new solution and all existing population members must be calculated) or prone to "replacement errors" (if only a subset of the population is compared against) [2]. In PSO, it is possible to identify the most likely population member that a new candidate solution might form a crowd with – its *lbest* attractor. This insight allows the proposed modification to achieve many of the benefits of crowding at a fraction of the computational cost.

REFERENCES

[1] Bratton, D., and Kennedy, J. Defining a Standard for Particle Swarm Optimization. In *Proceedings of the 2007 IEEE Swarm Intelligence Symposium*. IEEE Press. Piscataway, NJ, 2007, 120–127.

[2] Thomsen, R. Multimodal Optimization using Crowding-based Differential Evolution. In *Proceedings of the 2004 IEEE Congress on Evolutionary Computation*. IEEE Press. Piscataway, NJ, 2004, 1382–1389.