# Particles Prefer Walking Along the Axes: Experimental Insights into the Behavior of a Particle Swarm*

Manuel Schmitt    Rolf Wanka
Department of Computer Science, University of Erlangen-Nuremberg, Germany
{manuel.schmitt, rolf.wanka}@cs.fau.de

## ABSTRACT

We study the frequently observed phenomenon of *stagnation* in the context of particle swarm optimization (PSO). We show that in certain situations the particle swarm is likely to move almost parallel to one of the axes, which may cause stagnation. We provide an experimentally supported explanation in terms of a *potential* of the swarm and are therefore able to adapt the PSO algorithm slightly such that this weakness can be avoided.

## Categories and Subject Descriptors

I.2.8 [**Artificial Intelligence**]: Problem Solving, Control Methods, and Search—*heuristic methods*

## General Terms

Algorithms, Performance, Experimentation

## Keywords

Particle swarm optimization, stagnation, potential

## 1. INTRODUCTION

Particle swarm optimization (PSO) is a widely used nature-inspired meta-heuristic for solving continuous optimization problems. However, when running the PSO algorithm, one encounters the phenomenon of so-called stagnation, which in our context means, the whole swarm starts to converge to a solution that is not (even a local) optimum. The goal of this work is to point out possible *reasons* why the swarm stagnates at these non-optimal points. For this, we apply the newly defined *potential* of a swarm [2]. The total potential has a portion for every dimension of the search space, and it drops when the swarm approaches the point of convergence. As it turns out experimentally, the swarm is very likely to come into "unbalanced" states, i. e., almost all potential belongs to one axis. Therefore, the swarm becomes blind for improvements still possible in any other direction. Finally, we show how in the light of the potential and these observations, a slightly adapted PSO rebalances the potential.

*A full version of this paper is available as arXiv:1303.6145


*GECCO'13 Companion,* July 6–10, 2013, Amsterdam, The Netherlands.
ACM 978-1-4503-1964-5/13/07.

## 2. DEFINITIONS

**Definition 1 (Classical PSO Algorithm)** *A* swarm $\mathcal{S}$ *of* $N$ *particles moves through the* $D$*-dimensional search space* $\mathbb{R}^D$*. Each particle* $n$ *consists of a* position $X^n \in \mathbb{R}^D$*, a* velocity $V^n \in \mathbb{R}^D$ *and a* local attractor $L^n \in \mathbb{R}^D$*, storing the best position particle* $n$ *has visited so far. Additionally, the swarm shares information via the* global attractor $G \in \mathbb{R}^D$*, describing the best point* any *particle has visited so far.*

*The actual movement of the swarm is governed by the following update equations where* $\chi$*,* $c_1$ *and* $c_2$ *are some positive constants to be fixed later and* $r$ *and* $s$ *are drawn u. a. r. from* $[0,1]^D$*.*

$$V^n := \chi \cdot V^n + c_1 \cdot r \odot (L^n - X^n) + c_2 \cdot s \odot (G - X^n) \quad (1)$$
$$X^n := X^n + V^n \quad (2)$$

Now we define a swarm's potential measuring how close it is to convergence, i. e., we describe a measure for its movement. A swarm with high potential should be more likely to reach search points far away from the current global attractor, while the potential of a converging swarm approaches 0. These considerations lead to the following definition [2]:

**Definition 2 (Potential)** *For* $d \in \{1, \ldots, D\}$*, the potential of swarm* $\mathcal{S}$ *in dimension* $d$ *is* $\Phi_d$ *with* $\Phi_d := \sum_{n=1}^{N}(|V_d^n| + |G_d - X_d^n|)$ *the total potential of* $\mathcal{S}$ *is* $\Phi = (\Phi_1, \ldots, \Phi_D)$*.*

The current total potential of a swarm has a portion in every dimension. Between two different dimensions, the potential may differ much, and "moving" potential from one dimension to another is not possible. On the other hand, along the same dimension the particles influence each other and can transfer potential from one to the other. This is the reason why there is no potential of individual particles.

## 3. STAGNATION

Assume that the fitness function is (on some area) monotone in every dimension. One of our main observations is that in such a situation the swarm tends to pick one dimension and to favor it over all the others. As a consequence, the movement of the swarm becomes more and more parallel to one of the axes. Similarly, Spears et al. [3] point out that particles tend to gather close to the axes in the case of rotation invariant fitness functions.

We use the fitness-function $f(\vec{x}) = -\sum_{i=1}^{D} x_i$ which is monotonically decreasing in every dimension and set $D$ to 10. Initially, we distribute the particles randomly over $[-100;$

$100]^D$ and the velocities over $[-50;50]^D$, and let the swarm make 500 iterations. We set $\chi = 0.729$, $c_1 = c_2 = 1.49$ (as recommended in [1]) and $N = 10$. After each iteration, we calculate the potential for each dimension. We make 1000 runs and after each run, the dimensions are renamed according to the final value of $\Phi$, i. e., we switch the numbers of the dimensions such that after the last iteration dimension 1 always has the highest potential, dimension 2 the second highest and so on.

We calculate the mean of the potentials over the 1000 runs for each of the sorted dimensions. The results are stated in Fig. 1. One can see that the dimension with the greatest potential has a value far higher than the others, while the other dimensions do not show such a significant difference among each other. An explanation for this behavior is the following: Assume that at some time, one dimension $d_0$ has more potential than the others. Further assume that the difference is great enough such that for some number of steps the particle with the largest value in dimension $d_0$ is the one that determines the global attractor. In [2], a swarm in this situation is called "running". Since randomness is involved and this situation has a positive probability to occur, it will actually occur after sufficiently many iterations. Then, each update of the global attractor increases the potential in $d_0$ considerably because it increases the distance of every single particle to the global attractor except for the one particle that updated it. In any other dimension $d \neq d_0$, the situation is different. Here, the decision which particle updates the global attractor is stochastically independent of the value $x_d$ in dimension $d$. In other words: If one considers only dimension $d$, the global attractor is chosen uniformly at random from the set of all particles. As a consequence, after some iterations, the $d_0$'th coordinate of the velocity becomes positive for every particle, so the attraction towards the global attractor always goes into the same direction as the velocity, while in the remaining dimensions, the velocities may as well point away from the global attractor, meaning that the particles will be slowed down by the force of attraction. This situation is prototypically depicted in Fig. 2. That implies that the balanced situation is not stable in a sense that after the imbalance has reached a certain critical value, it will grow boundlessly.

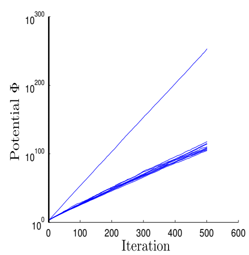


Figure 1: Growth of potential when processing $f(\vec{x}) = -\sum_{i=1}^{D} x_i$

If at some point no more improvements can be made in dimension $d_0$, this dimension still has the far highest potential and an improvement of the global attractor is still possible, but it is very unlikely and between two updates are many steps without an update. The reason is that any improvement in some of the remaining dimensions is voided by the much larger worsening in dimension $d_0$. Hence, the attractors stay constant for long times between two updates and so the swarm tends to converge and therefore looses potential. As long as the global attractor stays constant, the situation is symmetric in every dimension. That means after the same time the potential of every dimension is decreased by approximately the same factor, so dimension $d_0$ has still far more potential than any other dimension and the swarm stays blind for possible improvements in dimensions other than $d_0$.

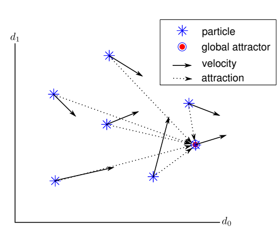


Figure 2: Particles running in direction $d_0$.

# 4. MODIFIED PSO

A small and simple modification of the PSO algorithm avoids the problem described in the previous section by enabling the swarm to rebalance the potentials of the different dimensions. When the swarm tends to converge, we replace the usual velocity update by a random choice of the new velocity out of some small but constant-sized interval.

**Definition 3 (Modified PSO)** *For some arbitrarily small but fixed $\delta > 0$, we define the modified PSO via the same equations as the classic PSO in Def. 1, only modifying the velocity update in (1) to*

$$V_d^n := \begin{cases} (2r-1)\cdot\delta, \\ \quad \textit{if } \forall\, d' \in \{1,...,D\} : |V_{d'}^n| + |G_{d'} - X_{d'}^n| < \delta, \\ \\ \chi \cdot V_d^n + c_1 \cdot r \cdot (L_d^n - X_d^n) + c_2 \cdot s \cdot (G_d - X_d^n), \\ \quad \textit{otherwise.} \end{cases}$$

*Whenever the first case applies, we call the step* forced.

To show that the modification does not fully take over, we plotted the forced points with $\delta = 10^{-7}$ and the 2-dimensional sphere function as objective function in Fig. 3. One sees, the particles get forced near $(-2\cdot 10^{-5}, 0)$ but their movement does not stay forced. Instead, the swarm becomes running again until the particles approach the optimum at $(0,0)$. This implies that for sufficiently smooth functions, the modification does not take over, replacing the PSO by some random search routine. Instead, the modification just helps to overcome "corners." As soon as there is a direction parallel to an axis with decreasing function value, the swarm becomes "running" again and the unmodified movement equations apply.

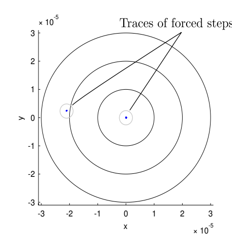


Figure 3: Behavior of the modified PSO on the sphere function

# 5. REFERENCES

[1] M. Clerc and J. Kennedy. The particle swarm – explosion, stability, and convergence in a multidimensional complex space. *IEEE Transactions on Evolutionary Computation*, 6:58–73, 2002.

[2] M. Schmitt and R. Wanka. Particle swarm optimization almost surely finds local optima. In *Proc. Genetic and Evolutionary Computation Conference (GECCO)*, 2013.

[3] W. M. Spears, D. Green, and D. F. Spears. Biases in particle swarm optimization. *International Journal of Swarm Intelligence Research*, 1(2):34–57, 2010.