Understanding Particle Swarm Optimization: A Component-Decomposition Perspective

Daqing Yi, Kevin D. Seppi and Michael A. Goodrich
Computer Science Department, Brigham Young University, Provo, UT USA 84604
Email: yi@byu.edu, kseppi@byu.edu, mike@cs.byu.edu

***Abstract*—Particle Swarm Optimization is an effective algorithm because of the combination of the stochastic behavior of particles and the swarm structure. Unfortunately these features also make it difficult to understand the dynamics of PSO. Common methods of analyzing PSO rely on simplifying the algorithm, e.g., assuming stagnation (a state where the swarm ceases finding better solutions) or treating the stochastic factors as constants. In this paper, we expand on earlier work to understand the dynamics of PSO which used input-to-state stability analysis. In particular, we decompose PSO more completely and use the properties of combinations of input-to-state stable components to model convergence at all levels up to and including the entire swarm. This approach allows us to conceptualize the swam as a leader-follower structure and analyze the swarm under a variety of conditions including various fitness functions.**

I. Introduction

Particle swarm optimization (PSO) is a popular and efficient optimization algorithm. It is inspired by the social behavior in flocks of birds and schools of fish. In PSO a particle is driven by a personal best and a global best. The personal best (x_i^P) is the location of the best solution particle i has found, while the global best (x^G) is the best solution that any of the particles in a swarm have found. The movement of each particle is the stochastic attractions toward these "best" locations. Mathematical models of the algorithm have been used to evaluate and understand the behavior of the algorithm but better models of the algorithm can enable better understanding. This understanding includes issues such as: why PSO can effectively converge to an optimal solution in some cases, and fail to find an optimal solution in other case?

Stagnation is an important phenomenon frequently used to understand the search process of evolutionary optimization algorithms. It is commonly defined as a condition in which an algorithm stops finding better solutions. In the PSO, this means that none of the particles in the swarm can find a better solution than the current global best. In this state, the global best, x^G, and the personal best, x^P, are constant for all particles [1], [2]. Understanding the dynamics of the particles in stagnation is helpful in analyzing the behavior and the optimization (search) capability of the particle swarm optimization. Plenty of work has focused on the behavior of PSO when this happens, including the investigation of the convergence of particles in the swarm [3], [4].

In this work we consider both the behavior at stagnation and the behavior during the time that the global best is unchanged, but in which the personal best of each particle is still changing. We refer to this state as *global-best stagnation*. If we partition the search process of PSO based on the times when the global-best is updated, we can model the execution of PSO as a sequence of global-best stagnations. A global-best update triggers a period of transition followed eventually by a new global-best stagnation. Similarly, a global-best stagnation can be decomposed into a sequence of stagnation states. In that case a personal-best update triggers a period of transition followed by a new stagnation. This way of decomposition the behavior of the algorithm will help us analyzing the process.

Since particles only share information through updates to the global best, during global-best stagnation, particles have no interaction and they act only as a set of independent agents. Furthermore, the particles are in a competitive relationship, a particle that finds a better solution than the current global best updates the global best and leads the swarm out of the previous global-best stagnation.

In Section III, we decompose the particle into two components, which form a feedback cascaded structure for the dynamics. We introduce the system decomposition of a particle and a swarm and review input-to-state stability analysis [5] in Section III-B. This enables the new investigation of particle movement found in Section IV. Then we extend the scope of our investigation from the level of a single particle to the level of the whole swarm, and show how input-to-state analysis explains the search capabilities of PSO in Section V.

II. Related work

The dynamics of PSO is hard to evaluate in general, due to the combination of the stochastic nature of the particle's path and the social interaction represented by the swarm topology. Early analysis began by replacing stochastic terms with constant terms in order to predict the convergence of a particle at stagnation [6] [7]. Based on such convergence analyses, parameters can be set for the best effect [8]. Another approach for handling the stochastic factors is based on the stochastic analysis. By taking the mean of the stochastic variables, the stochastic terms are naturally converted into constant terms. A convergence analysis of the mean and variance of a particle at stagnation can also be obtained using the characteristic equation in a discrete-time model [9]. In a similar way, other moments can be computed [10], [11], [4]. Furthermore, equilibria can be found using a discrete-time system model of different moments. The stability requirements can be obtained from the norm by setting the root values of the characteristic equation to all be less than 1 [10], [11], [4].

There are also several approaches involving the analysis of the stochastic behavior of particle swarm optimization. All of the stochastic terms can be modeled as bounded nonlinear feedback[12]. The global best and the personal best can be integrated into the nonlinear feedback. This depends on the assumption that the personal best and global best are both constant. The analysis tool of $l-2$ stability has also been applied and used to understand the impact of the stochastic terms on the convergence [13]. Since the global best and personal best are plugged into the feedback term, it becomes hard to evaluate the impact of the updates of the global best and personal best.

There is also some work that addresses the dynamics of a particle when it is not in stagnation. In one such approach the discrete-time dynamics of PSO, that is, the dynamics of particle trajectory, can be approximated using a continuous-time model [14]. In this way the global best and the personal best can be modeled as time-variant variables. Furthermore, the probability of convergence in time can be analyzed by viewing the update process as a random search process [15]. In addition to this approach to model the behavior of a particle in a swarm, the process of particles reaching a local optimum has also been analyzed [3].

Most of these prior efforts still seek to analyze the algorithm at the particle level. The influence from a swarm on the behavior of a particle has not been considered enough. The analysis of the particles impact each other in a swarm and in a fitness space seems critical to understanding the swarm behaviour during optimization.

III. System

In this paper we use the formulas from Kennedy's most recent definition of PSO [16], which can often be extended to many versions of PSO. This version of PSO includes a constricted position update rule, a personal best update and a star topology formed by a global best update. The constricted position update rule is

$$\begin{aligned} v_{ij}(k+1) = & \chi[v_{ij}(k) + \phi^P u_{ij}^P(k)(x_{ij}^P(k) - x_{ij}(k)) \\ & + \phi^G u_{ij}^G(k)(x_j^G(k) - x_{ij}(k))], \end{aligned} \quad (1a)$$

$$x_{ij}(k+1) = x_{ij}(k) + v_{ij}(k+1). \quad (1b)$$

$x_{ij}(k)$ represents the position of particle i in dimension j at time k. $v_{ij}(k)$ similarly represents the velocity of particle i in dimension j also at time k. $x_j^G(k)$ and $x_{ij}^P(k)$ are global (actually topology) and personal best positions observed by the swarm and the particle respectively. $u_{ij}^G(k)$ and $u_{ij}^P(k)$ are independent random values drawn from $[0,1]$. $\chi \in (0,1)$, ϕ^P and ϕ^G are algorithm parameters. The personal best update and the global best update are

$$x_i^P(k) = \arg\ max_{x \in \{x_i(k), x_i^P(k-1)\}} f(x). \quad (2a)$$

$$x^G(k) = \arg\ max_{x \in \{x_i(k), x^G(k-1)\}} f(x). \quad (2b)$$

A star topology is modeled by simply sharing the global best. When particle i finds a position that is better than the current global best, it updates its global best and its personal best. The swarm moves to a new global-best stagnation, which means that $x_i(k) = x_i^P(k) = x^G(k)$. Equation (1a) becomes

$$v_{ij}(k+1) = \chi[v_{ij}(k) + (\phi^P u_{ij}^P(k) + \phi^G u_{ij}^G(k))(x_j^G(k) - x_{ij}(k))]. \quad (3)$$

As the inertia of the previous velocity $v_{ij}(k)$ will decay to zero, the particle is attracted to $x_j^G(k)$ as $\phi^P u_{ij}^P(k) + \phi^G u_j^G(k) \geq 0$. This particle can be viewed as a leader of the swarm, which forms the star topology in Figure 1.

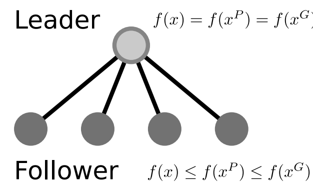


Fig. 1: A leader-follower relationship.

The star topology implies a leader competition in a swarm. A particle that finds a new global best becomes the leader of the swarm. A *leader particle* holds the same global best and personal best $x^G(k) = x^P(k)$, The other particles are *follower particles*, which follow the leader by virtue of the attraction to the global best. By Property 1, we know that a follower particle will never stop moving if the personal best and the global best are different. Thus, a follower particle wanders randomly in the search space till it reaches a position that is the same with the global best. The position of the global best is the only input from the rest of the swarm.

In PSO the movement of a particle can be seen as being driven by a force toward the personal best and a force toward the global best. Thus when the global best and the personal best are not the same, the two forces will never exactly balance and the particle will continue to move. Because of the stochastic terms in the PSO update formula a particle will continue to move toward some random position influenced by the global best and the personal best unless the global and personal best positions are equal.

In this paper, the precision cut-off in implementation is ignored, that is, that pso is running on an infinite precision machine. Thus we can assume that the probability that any two randomly selected points is zero. A point we depend on in several of our proofs including the following property.

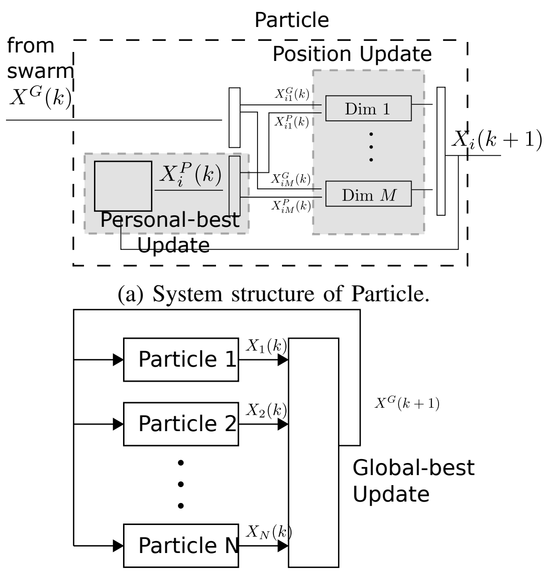
Property 1. *A particle will not stop moving unless its personal best and global best are equal, that is:* $\lim_{k\to\infty} v(k) \neq 0$*, if* $x^P \neq x^G$.

Proof. Assume that when $x^P \neq x^G$, $\lim_{k\to\infty} v(k) = 0$. It means that $\phi^P u_{ij}^P(k)(x_{ij}^P(k) - x_{ij}(k)) + \phi^G u_j^G(k)(x_j^G(k) - x_{ij}(k)) = 0$ for any $u_{ij}^P(k), u_{ij}^G(k) \in (0,1)$, which cannot be true. □

This indicates that the stochastic factors prevent the existence of an equilibrium position for a particle before the global best and the personal best reach a consensus. Furthermore, this implies that a conventional analyses of convergence that ignore the stochastic factors is not enough to understand the dynamics of the optimization process in PSO.

A. A Feedback Cascade Model in a Particle

With the global best as the input, we model the behavior of a particle as a *feedback cascade system* in the fashion used by Yi et. al.[5]. As shown in Figure 2a, this system is comprised of two components that form a feedback system structure. These two components are the *input-update component* for the personal best ($x_i^P(k)$) and the global best ($x^G(k)$), and the *position-update component* for particle position ($x_i(k+1)$), which depends on the inputs $x^G(k)$ and $x_i^P(k)$ as well as the last position $x_i(k)$.



(a) System structure of Particle.

(b) System structure of Swarm.

Fig. 2: System structure of PSO.

In the position-update component, the position at each dimension is updated by using $x^G(k)$ and the $x_i^P(k)$ in the corresponding dimension. By Equation (1), we can decompose the position-update component into subcomponent in each dimension. As shown in Figure 2a, the subcomponents in the position-update component form a parallel connection. From (1), we can write a linear form of the position update component in one dimension.

$$X(k+1) = A(k)X(k) + B(k)U(k) \tag{4}$$

with

$$A(k) = \begin{bmatrix} \chi & -\chi\phi^G u^G(k) - \chi\phi^P u^P(k) \\ \chi & 1 - \chi\phi^G u^G(k) - \chi\phi^P u^P(k) \end{bmatrix} \tag{5}$$

and

$$B(k) = \begin{bmatrix} \chi\phi^G u^G(k) & \chi\phi^P u^P(k) \\ \chi\phi^G u^G(k) & \chi\phi^P u^P(k) \end{bmatrix}. \tag{6}$$

The system state is $X(k) = [v(k), x(k) - x^R]^T$, and the system input is $U(k) = [x^G(k) - x^R, x^P(k) - x^R]^T$. [1] With a new $x(k+1)$, the personal best update component will update $x^P(k+1)$ that are fed into the position update component.

The input-update component consists of a *global-best input* and a *personal-best update*. The personal best and global best are the input of the position-update component. As in Figure 2a, the global-best input only receives the input from the swarm. Thus the state to the reference point x^R is $x^G(k) - x^R$. The personal-best update compares the current $x(k)$ with the $x^P(k-1)$. If $x_i(k)$ is better, the personal best is updated with it. We can write the personal-best update as

$$U = g^{PU}(V) \tag{7}$$

with $U = x^P(k) - x^R$ and $V = x(k) - x^R$ from (2a).

In Figure 2a, the position of a particle is modeled into a system with the input $x^G(k)$ and the output $x(k)$. By using this model, we can have the system structure of a swarm in Figure 2b. There is a *global update* that reads the states of all the particles and determines whether the global best should be updated. The global best is fed back to all the particles for the next optimization iteration.

B. Input-to-State Stability

It can be shown that PSO satisfies this definition when the parameters of PSO are set in the requisite range[5]. The bounds implied by the ISS property can also be derived, which can be applied to find bounds on particle motion.

Input-to-state stability analysis has been a useful tool in understanding the convergence dynamics of a system with interconnected components. As the decompositions of a swarm and a particle in Figures 2a and 2b, input-to-state stability to the components can be used to evaluate particle movement, e.g. how the bounds on $x_i^G(k)$ and $x_i^P(k)$ determine the position of particle $x_i(k)$ [17].

The definition of input-to-state stability depends on several types of functions [17].

- K-function $\mathbb{K}$: a function $\alpha() : [0, a) \to [0, \infty)$ is continuous, strictly increasing and $\alpha(0) = 0$; it is a K_∞-function, if $\alpha(s) \to \infty$ as $s \to \infty$;
- KL-function $\mathbb{KL}$: a function $\beta() : [0, a) \times [0, \infty) \to [0, \infty)$ satisfies:

1) $\forall t \geq 0$, $\beta(\cdot, t)$ is a K-function;
2) $\forall s \geq 0$, $\beta(s, \cdot)$ is decreasing and $\beta(s, t) \to 0$ as $t \to \infty$.

Definition 1 (Input-to-state stable [17])**.** *For x, a discrete-time system defined as follows:*

$$x(k+1) = f(x(k), u(k)), \tag{8}$$

with $f(0, 0) = 0$ [2]*, the system is* (globally) input-to-state stable *if there exist a* KL*-function* β *and a* K*-function* γ *such that,*

[1] x^R means a reference point to the system. When applying to the PSO, it can be a local optimum, a global optimum, or an estimated optimum. We use it as a reference point to check the bounds.

[2] This means that $x = 0$ is an equilibrium of the 0-input system.

for each input $u \in l_\infty^m$ and each $\xi \in \mathbb{R}^n$, it holds that $\forall k \in \mathbb{Z}^+$,

$$|x(k,\xi,u)| \leq \beta(|\xi|,k) + \gamma(\|u\|). \quad (9)$$

The $\beta()$ term in equation (9) defines an initial bound with a decaying property. The $\gamma()$ term in equation (9) defines a bound determined by the input. This means that the influence of the $\beta()$ term gradually decreases to zero and the position is bounded by a range determined by the bound on the input [5].

In the case of PSO and as shown in Figure 2a, if each component (representing a single dimension) is input-to-state stable, the position-update component which combines all the dimensions is also input-to-state stable. We have Property 2.

Property 2. *The position-update component is input-to-state stable if the update in each dimension is input-to-state stable.*

This simplifies the analysis of the system since it allows us to consider each dimension separately. In our analysis of the PSO algorithm, we seek to understand how the particles converge to some position x^R, which is intended (not guaranteed) by the algorithm to be the optimal position. For this analysis we use a one-dimension particle and extract the linear form of the position-update component. As noted above, the one dimensional case can be extended to many dimensions.

The conditions of ISS [5] helps explain what happens to a particle given different conditions on personal best and global best. When the position update component is input-to-state stable, the input-to-state stability of the particle is determined by the input-to-state stability of the personal best update component. However, the input-to-state stability of the input update component cannot be guaranteed because it depends on the fitness distribution. In section IV, we will analyze the behavior of a particle when the position update component is input-to-state stable. We will later extend the analysis to a swarm in section V.

IV. Particle analysis

In a global-best stagnation, the global best is constant. For convenience, we denote the global best as a constant x^G in a global-best stagnation. A particle is isolated from the impact of other particles in a global-best stagnation. A leader particle will gradually converge to $x^G(k)$ by Equation (3). Follower particles are hard to evaluate, because the dynamics are driven by both the personal best and the global best. In global-best stagnation, we are interested with

- whether a particle converges to the global best;
- and the probability that a particle finds a new global best.

To measure how the particle converges to the global best, we let x^G be the reference position x^R and get Equation (10).

$$\begin{bmatrix} v(k+1) \\ x(k+1) - x^G \end{bmatrix} = A(k) \begin{bmatrix} v(k) \\ x(k) - x^G \end{bmatrix} + B(k) \begin{bmatrix} 0 \\ x^P(k) - x^G \end{bmatrix} \quad (10)$$

It is obvious that if the movement bounded region of a particle does not cover the optimum, the particle will not converge to the optimal solution. Whether a particle can reach the optimum is impacted by the boundary of its movement.

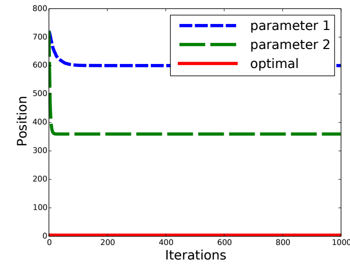


Fig. 3: The convergence to the optimum of particles with different parameters. Parameter 1 is $\chi = 0.3, \phi^P = 0.2, \phi^G = 0.2$ and parameter 2 is $\chi = 0.72984, \phi^P = 2.05, \phi^G = 2.05$.

By Corollary 1 in [5], we know that the boundary range is determined by both $|x^P - x^G|$ and the swarm parameters. Figure 3 gives an example on the boundary impacting the moving toward the optimum. As illustrated in Figure 3, parameter 1 leads a smaller movement boundary, in which case the particle never has a chance of reaching the optimum.

A. Stagnation

We denote the personal best and the global best as constant x^P and x^G in stagnation, because they are not updated. By Corollary 1 in [5], we know that $\|x(k) - x^G\|$ indicates the distance between position $x(k)$ and the global best x^G, which is determined by $\|x^P - x^G\|$. Thus we have Theorem 1.

Theorem 1. *If a particle is input-to-state stable, there exists a boundary function $\gamma()$ such that*

$$|x(k) - x^G| \leq \gamma(|x^P(k) - x^G|), \quad (11)$$

And thus when x^P is constant, $|x(k) - x^G| \leq \gamma(|x^P - x^G|)$.

Theorem 1 indicates how the movement of a particle is bounded by the distance between global best and personal best. We look at two categories of fitness distributions, unimodal and multi-modal respectively in following two subsections.

B. Unimodal Fitness Distribution

Unimodal functions are a common type of fitness distribution and which provides a partial monotonic shape. Let's define a unimodal function as a function $f(x)$ that (1) there exists only single optimum; and (2) the function is convex. Any shape of fitness distribution can be approximated by a combination of unimodalities. In most cases, particles gradually converge towards one unimodal region of a fitness distribution. In a unimodal fitness distribution, we can categorize the behavior of the swarm into $x^G = x^*$ and $x^G \neq x^*$.

Case one $x^G = x^*$: $x^G = x^*$ indicates that a swarm has already found the optimal solution. No particle will find a better solution. By Theorem 3, we know that all particles in the swarm should gradually converge to the global best if the position-update component is input-to-state stable. The input-to-state stability of the personal-best update component is determined by the fitness distribution. We have the condition of the fitness distribution that guarantees the personal best update component input-to-state stable.

Lemma 1. *If there exist $\alpha_1()$ and $\alpha_2()$ that are K_∞-functions and if $f(x^*) - \alpha_2(|x|) < f(x) < f(x^*) - \alpha_1(|x|)$ when $x^G = x^*$, then the personal-best update component is input-to-state stable.*

Proof. Let $V(x^P(k)) = f(x^G) - f(x^P(k))$. Because we have $\alpha_1(|x^P(k)|) \leq V(x^P(k)) \leq \alpha_2(|x^P(k)|)$. We have $V(x^P(k))$ satisfying condition 1 of the ISS-Lyapunov function definition [17].

We also have

$$\begin{aligned}
& V(x^P(k+1)) - V(x^P(k)) \\
=& f(x^P(k)) - f(x^P(k+1)) \\
=& f(x^P(k)) - f(\arg\max_{\{x^P(k), x(k+1)\}} f(x)) \\
=& \begin{cases} 0 & \text{if } f(x(k+1)) \leq f(x^P(k)) \\ f(x^P(k)) - f(x(k+1)) & \text{if } f(x(k+1)) > f(x^P(k)) \end{cases} \\
\leq& \begin{cases} f(x^P(k)) - f(x(k+1)) & \text{if } f(x(k+1)) \leq f(x^P(k)) \\ f(x^P(k)) - f(x(k+1)) & \text{if } f(x(k+1)) > f(x^P(k)) \end{cases} \\
\leq& f(x^P(k)) - f(x(k+1)) \\
\leq& - V(x^P(k)) + V(x(k+1)) \\
\leq& - \alpha_1(|x^P(k)|) + \alpha_2(|x(k+1)|).
\end{aligned} \tag{12}$$

We have that $\alpha_1()$ is K_∞-function and $\alpha_2()$ is K-function. Thus the condition 2 of the ISS-Lyapunov function definition is also satisfied. $V(x)$ is an ISS-Lyapunov function. By Lemma 3.5 in [17], in this case, the personal best update component is input-to-state stable. □

There usually exists no K_∞-functions $\alpha_1()$ and $\alpha_2()$ for Lemma 1 when there is "plateau" in $f()$, particularly when $x*$ is on the plateau. Thus for the rest of this paper we assume that $f()$ contains no "plateaus".

When both the personal-best update component and the position-update component of a particle are input-to-state stable, we have the condition that the particle is input-to-state stable in Lemma 2.

Lemma 2. *When the position update component is input-to-state stable, $|x(k)| \leq \max\{\beta_1(|x(0)|, k), \gamma_1^P(|x^P(k)|), \gamma_1^G(|x^G|)\}$, and the personal best update component is input-to-state stable, $|x^P(k)| \leq \max\{\beta_2(|x^P(0)|, k), \gamma_2(|x^P(k)|)$, if $\gamma_1^P \circ \gamma_2(s) < s$, the feedback model of a particle is input-to-state stable.*

Proof. In Figure 2a, consider the global best as a constant, the personal best update component and the position update component forms a feedback structure, which enables us to apply Theorem 2 in [17] to provide input-to-state stability analysis. □

We have the condition that a particle converges to the optimum x^* in Theorem 2. When the fitness distribution is unimodal, $f(x^*) - \alpha_2(|x|) \leq f(x) \leq f(x^*) - \alpha_1(|x|)$ is satisfied. We have Theorem 2.

Theorem 2. *When $x^G = x^*$, if $f(x^*) - \alpha_2(|x|) \leq f(x) \leq f(x^*) - \alpha_1(|x|)$ and $\gamma_1 \circ \gamma_2(s) < s$, a particle will converge to x^* if the position-update component of the particle is input-to-state stable. $\gamma_1()$ is the gain of the position update component and $\gamma_2()$ is the gain of the input update component.*

Proof. By Lemma 2, we have the feedback model of a particle is input-to-state stable. Because $x^G = x^*$, the input to the cascade model of the particle is zero, $|x^G - x^*| = 0$. By the property of the input-to-state stability, $|x(k) - x^*|$ will converge to zero, which means that the particle will converge to x^*. □

Theorem 2 shows the condition that convergence could be guaranteed. Because $\phi^P u^P(k)$ and $\phi^G u^G(k)$ are randomly sampled at each iteration, $\gamma_1 \circ \gamma_2(s) < s$ is not easy to guarantee; however, the stochastic terms contribute to almost surely convergence. Thus, we can have Theorem 3.

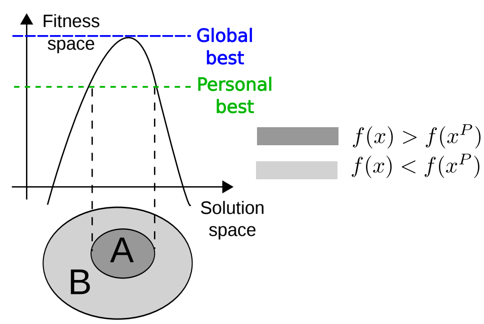


Fig. 4: How sets A and B are defined.

Theorem 3. *In a unimodal case, when $x^G = x^*$, a particle will almost surely converge to x^* if the position-update component of the particle is input-to-state stable.*

Proof. The convergence of a particle depends on the the personal best. In this case, given the personal best x^P, the solution space $\Omega = A \cup B \wedge A \cap B = \phi$, in which $A = \{x|f(x) > f(x^P)\}$ and $B = \{x|f(x) \leq f(x^P)\}$. As long as the particle gets into the set A, the volume of the set A decreases. When $A = \phi$, $x^P = x^G = x^*$. By the nature of the update of personal best, x^P will only move into set A or be unchanged. Assume that it is possible that when $x^P \neq x^G = x^*$, x^P will be unchanged. It only happens that when the particle never moves into set A. However, because X^G is in set A, by the property of input-to-state stability, we know that there exists a set near around x^G that the particle will move into and is a subset of set A, which contradict with the assumption. Thus the probability that the particle moves into set A is also larger than zero. Once the particle moves into set A, the personal best shall be updated, and the volume of A will decrease. Thus, the volume of A will almost surely converge to zero. It means that the personal best will almost surely converge to the global best, which leads to that the particle almost surely converges to $x^G = x^*$. □

Case two $x^G \neq x^*$: When a swarm has not yet found the optimal solution, the convergence of a particle is harder to evaluate. If we have $f(x^*) - \alpha_2(|x|) < f(x) < f(x^*) - \alpha_1(|x|)$, we can estimate the dynamics by Theorem 3. Otherwise, if a particle accidentally reaches into a region where the fitness is better than the current global best, the global best will be updated and a new global-best stagnation is triggered. If the particle at least finds a solution that is better than the current personal best but worse than the current global best, the personal best will be updated. Another possibility is that the particle wanders stochastically in a region where the fitness is worse than both the personal best and the global best.

In order to analyze what happens when $x^G \neq x^*$, we re-emphasize that a particle should never stop at the current global best x^G, which is stated in Lemma 3.

Lemma 3. *In a unimodal case, if $x^G \neq x^*$, a particle will never stop at x^G.*

Proof. Assume that a particle will stop at x^G. Consider three possible cases:

- $f(x^*) < f(x^G) < f(x^P)$: By Property 1, we know that the particle will never stop.
- $f(x^*) < f(x^G) = f(x^P)$ and $x^G \neq x^P$: Because x^P will not be changed, by Property 1, we know that the particle will never stop.
- $f(x^*) < f(x^G) = f(x^P)$ and $x^G = x^P$: There exists a set A that includes all the states that $f(x) < f(x^G)$. And there exist a set $\neg A$ that includes all the states that $f(x) \leq f(x^G)$ Assume that for any current state x_k, a particle will stop at x^G. By the nature of global-best update and personal-best update, it means that it will never moves into set A. When a particle is input-to-state stable, there exists a bound near x^G. The convexity of a unimodal function guarantees that the intersection with set A is always greater than zero. So it contradicts with the assumption.

Thus, theoretically the particle will never stop at x^G when $x^G \neq x^*$. □

By Lemma 3, we can prove that a particle will finally get to a position that is at least better than the current global best if given enough run time in a unimodal case.

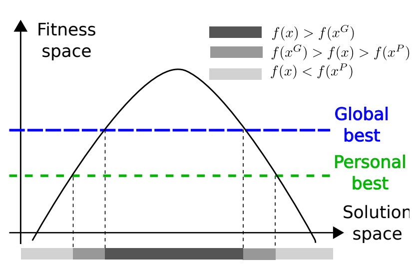


Fig. 5: How global best and personal best divide the solution space.

Theorem 4. *In a unimodal case, when $f(x^G) < f(x^*)$, a particle will almost surely find a $\hat{x^*}$ that $f(\hat{x^*}) > f(x^G)$.*

Proof. Because a particle cannot stop at $x(k) = x^G = x^P$. We will show it will finally arrives into a region that $f(x) > f(x^G)$.

As in Figure 5, the solution space can be divided into three types of regions by the global best and the personal best. Any global-best or personal-best update triggers a re-division.

- $f(x) > f(x^G)$ Once a particle gets into the region, it updates the global best and the personal best, and becomes a leader particle.
- $f(x^G) > f(x) > f(x^P)$ Once a particle gets into the region, it updates only the personal best. The solution space is re-divided.
- $f(x) < f(x^P)$ When a particle is in the region, it only moves as a random walk.

The movement of a particle is determined by the region of the solution space it moves in. The region transition forms a Markov process. The states of the Markov process can be defined as

- **A** $[f(x) \leq f(x^P) \leq f(x^G)]$
- **B** $[f(x) = f(x^P) \leq f(x^G)]$
- **C** $[f(x) = f(x^P) \leq f(x^G) \wedge v > 0]$
- **D** $[f(x) = f(x^P) \leq f(x^G) \wedge v = 0]$
- **E** $[f(x) > f(x^P) = f(x^G)]$

The Figure 6 shows the state transitions. When the particle is in state A, it follows a random walk with attractions to the global best and personal best. In state B, it means that the particle finds a better personal best. In state C, the particle moves into the current global best, but the velocity is not zero. The only chance that the particle cannot find a better global best happens when it gets into state D. In Figure 6, the state will almost surely move into state E. It means that the particle will almost surely find a better solution. □

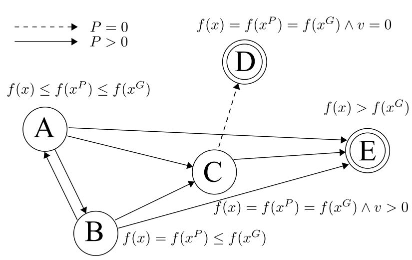


Fig. 6: The state transition of the movement of a particle.

C. Multi-Modal Fitness Distribution

When the movement of a particle is bounded, there are multiple cases where the particle cannot reach the region that contains a better solution. The exploration range is determined by a bound on a particle's movement. We are interested in the way in which a particle can get closer to the optimal position x^*. We have Lemma 4.

Lemma 4. *The bound of a particle's movement can be either*

$$|x(k) - x^*| < \max(\beta^*(x(0) - x^*, k), \gamma^*(\max(|x^G - x^*|, |x^P(k) - x^*|))), \tag{13}$$

or

$$|x(k) - x^0| < \gamma^0(\max(|x^G - x^0|, |x^P(k) - x^0|)), \quad (14)$$

in which β^* is KL-function , γ^* and $\gamma^0()$ are K_∞-functions.

Proof. By applying x^* and x^0 to Corollary 1 in [5], we can have the boundaries of the particle movement. □

Lemma 4 shows that the boundary of a particle's movement is determined by where the global best x^G and the personal best $x^P(k)$ locate.

As we are interested in the probability that a particle gets into a position that is better than the current global best, we hope the distance between current position and the optimum is smaller than that from the global best to the optimum.

If the bound on the particle's movement does not contain a solution that is better than the personal best, the personal best and the global best have no chance to be updated and the particle will not be able to find a better solution.

Theorem 5. *If $\forall s \in \{s| |s - x(0)| < \gamma^0(\max(|x^G - x^0|, |x^P(k) - x^0|))\}, f(s) < f(x^P)$, the probability that a particle finds a better solution is zero.*

Proof. Because a particle has no chance of getting into any position that has better solution than the global best and the personal best. Thus the boundary will also not be changed. □

In order to measure how likely a particle can move to a position that $f(x) > f(x^G)$, we can indirectly measure the probability that $|x(k) - x^*| < |x^G - x^*|$. We then have Corollary 1.

Corollary 1. *If there exists a boundary function $\gamma()$ that*

$$\begin{aligned} &P(| - x^*| < |x^G - x^*|) \\ &> 1 - \frac{\gamma(\max(E(|x^G - x^*|), E(|x^P(k) - x^*|)))}{|x^G - x^*|}. \end{aligned} \quad (15)$$

Proof. By Markov's inequality, we have

$$P(|x(k) - x^*| \geq |x^G - x^*|) \leq \frac{E(|x(k) - x^*|)}{|x^G - x^*|}. \quad (16)$$

By the mean of the position update component (details in Section 4.2 in [5]) we can have the boundary of the mean

$$E(|x(k) - x^*|) \leq \gamma(\max(E(|x^G - x^*|), E(|x^P(k) - x^*|))), \quad (17)$$

in which $\gamma()$ is the boundary function.

$$\begin{aligned} &P(|x(k) - x^*| < |x^G - x^*|) \\ =&1 - P(|x(k) - x^*| \geq |x^G - x^*|) \\ >&1 - \frac{E(|x(k) - x^*|)}{|x^G - x^*|} \\ >&1 - \frac{\gamma(\max(E(|x^G - x^*|), E(|x^P(k) - x^*|)))}{|x^G - x^*|}. \end{aligned} \quad (18)$$

□

If there exits a monotonic function that bounds a fitness function, we can measure the probability that a particle finds a better solution, which is given in Theorem 6.

Theorem 6. *Given a region R of $|x - x^*| < \epsilon$ is monotonic, $x(k) \in R$ and $x^G \in R$, the probability that a particle finds a better solution is:*

$$P > 1 - \frac{\gamma(\max(E(|x^G - x^*|), E(|x^P(k) - x^*|)))}{|x^G - x^*|} \quad (19)$$

V. Swarm analysis

The movement of a particle is equivalent to a random walk in an attractive potential field defined by its personal best and the global best. There are a few factors that prevent a particle from reaching the optimum. Organizing particles into a swarm enhances the capability of search. A swarm means that particles are randomly initialized in a search space, and search simultaneously in a beam-search style. An interaction topology is also formed that adds information exchange to a beam-search style optimization, which upgrades the search capability. In this section, our analysis is based on the star topology of global-best update. Such a star topology shapes a competition among the particles. A particle that finds a new global best becomes a leader particle, as in Figure 1.

As we can see in Figure 2b, the system of a swarm can be viewed as that the global best update component provides a global best x^G and feedbacks into individual particle. The rule of the global best update component is the same with that of the personal best update component, as defined in Equation (2b). Several properties of the personal best update component can be inherited here. We have Lemma 5. The proof is the same with the proof of Lemma 1.

Lemma 5. *If there exist K_∞-functions $\alpha_1()$ and $\alpha_2()$ that $f(x^*) - \alpha_2(|x|) \leq f(x) \leq f(x^*) - \alpha_1(|x|)$, the global best update component is input-to-state stable.*

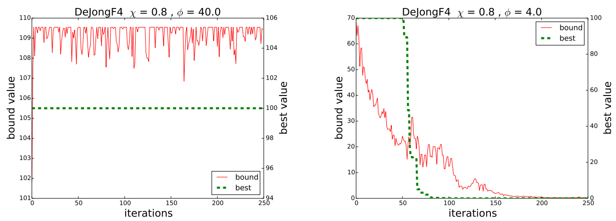
By Lemma 5, we can derive Theorem 7.

Theorem 7. *If $f(x^*) - \alpha_2(|x|) \leq f(x) \leq f(x^*) - \alpha_1(|x|)$, and the position-update component of a particle is input-to-state stable, and there exist a boundary function $\gamma_s()$ for the global best update component and a boundary function $\gamma_p()$ for the particle component that $\gamma_s \circ \gamma_p(s) < s$, , the particles in the swarm will converge to x^*.*

Proof. If both the particles and the global best update are input-to-state stable, it forms a cascade connection of input-to-state stable components as in Figure 2b. We can have the particles all gradually converge to x^*, by Corollary 4.2 in [17]. □

Figure 7 shows an example that contains a stable case and a unstable case on a multi-modal fitness distribution. In Figure 7a, the swarm is unstable and fails to converge to the optimum.

As we know, when $x^G(k) = x^*$, a swarm falls into only a set of particles without interactions. Then each particle becomes randomly wandering to fix the inconsistency between the personal best and the global best, which leads to a



(a) Unstable component (b) Stable component

Fig. 7: Two swarms on a DeJong F4 function [18].

convergence to the optimum x^*. The analysis of a swarm's behavior depends on how the global best x^G of the swarm converges to x^*. In a unimodal fitness distribution, as long as a better solution is guaranteed to be found, the optimal solution will be found by the *monotone convergence theorem* [3]. For a single particle $x^G = x^P$, however in a swarm there will be at least one particle with $x^G(k) \neq x^P$ with probability 1. Thus, over time, there will be refinements that will not stop till the optimum is reached.

Theorem 8. *In the unimodal case, given a swarm of more than two particles, x^G converges to x^* if all the particles have input-to-state stable position update components.*

Proof. There are two cases, (a) $x^G = x^*$ and (b) $x^G \neq x^*$. (a) When the global best is x^*, the particles will converge to x^* by Theorem 3. (b) When the global best is not x^*, x^G converges to x^*. This can be proved using contradiction. Assume that x^G will not converge to x^*. It means that at least more than one particle stop finding better solution. This contradicts with Theorem 4. □

For multi-modal fitness distributions it is more difficult prove properties of convergence, but Theorem 7 provides a condition where a swarm can converge to the optimum and which applies to a multi-modal fitness distribution as well as the unimodal case. In addition, Theorem 6 provides the probability that a particle can find a better solution in the unimodal case. Applying it to a particle in the optimal modal region (a "hill" within a multi-modal landscape) yields a lower bound of finding the optimum. Because of the competitive leader-follower behavior of PSO, whether a swarm can find the optimum depends on whether particles in the optimal modal region can win the competition. If particles in a suboptimal modal region dominate the competition, it is possible that particles in the optimal modal region will be diverted into suboptimal modal regions by Lemma 4. Similarly, particles in a suboptimal modal region might move into the optimal modal region. Since either of these are possible in the multimodal case, it is hard to guarantee behavior.

VI. Conclusion

In this paper, we model particle swarm optimization using a network structure. The particles in the swarm form a parallel connection with a global-best update as feedback. The dynamics of each particle has also been decomposed into an input-update component and a position-update component. This decomposition enables us to apply input-to-state stability analysis to PSO. We provide the conditions that guarantee various convergence properties. Our analysis starts from the particle level and then extends to the swarm level.

[3]https://en.wikipedia.org/wiki/Monotone_convergence_theorem

References

[1] M. Clerc and E. R. Poli, "Stagnation analysis in particle swarm optimisation or what happens when nothing happens," Tech. Rep., 2006.

[2] M. Jiang, Y. Luo, and S. Yang, "Stagnation analysis in particle swarm optimization," in *2007 IEEE Swarm Intelligence Symposium.*, April 2007, pp. 92–99.

[3] M. Schmitt and R. Wanka, "Particle swarm optimization almost surely finds local optima," in *Proceeding of the Fifteenth Annual Conference on Genetic and Evolutionary Computation Conference*, ser. GECCO '13. New York, NY, USA: ACM, 2013, pp. 1629–1636.

[4] R. Poli, "Dynamics and stability of the sampling distribution of particle swarm optimisers via moment analysis," *J. Artif. Evol. App.*, vol. 2008, pp. 15:1–15:10, Jan. 2008.

[5] D. Yi, K. D. Seppi, and M. A. Goodrich, "Input-to-state stability analysis on particle swarm optimization," in *Proceedings of the 2015 Annual Conference on Genetic and Evolutionary Computation.* ACM, 2015, pp. 81–88.

[6] M. Clerc and J. Kennedy, "The particle swarm - explosion, stability, and convergence in a multidimensional complex space," *Evolutionary Computation, IEEE Transactions on*, vol. 6, no. 1, pp. 58–73, Feb 2002.

[7] N. Samal, A. Konar, S. Das, and A. Abraham, "A closed loop stability analysis and parameter selection of the particle swarm optimization dynamics for faster convergence," in *Evolutionary Computation, 2007. CEC 2007. IEEE Congress on*, Sept 2007, pp. 1769–1776.

[8] I. C. Trelea, "The particle swarm optimization algorithm: convergence analysis and parameter selection," *Information Processing Letters*, vol. 85, no. 6, pp. 317 – 325, 2003.

[9] M. Jiang, Y. Luo, and S. Yang, "Stochastic convergence analysis and parameter selection of the standard particle swarm optimization algorithm," *Information Processing Letters*, vol. 102, no. 1, pp. 8 – 16, 2007.

[10] R. Poli, "Mean and variance of the sampling distribution of particle swarm optimizers during stagnation," *Evolutionary Computation, IEEE Transactions on*, vol. 13, no. 4, pp. 712–721, Aug 2009.

[11] R. Poli and D. Broomhead, "Exact analysis of the sampling distribution for the canonical particle swarm optimiser and its convergence during stagnation," in *Proceedings of the 9th Annual Conference on Genetic and Evolutionary Computation*, ser. GECCO '07. New York, NY, USA: ACM, 2007, pp. 134–141.

[12] V. Kadirkamanathan, K. Selvarajah, and P. Fleming, "Stability analysis of the particle dynamics in particle swarm optimizer," *Evolutionary Computation, IEEE Transactions on*, vol. 10, no. 3, pp. 245–255, June 2006.

[13] Y. Wakasa, K. Tanaka, and T. Akashi, "Stability and l2 gain analysis for the particle swarm optimization algorithm," in *American Control Conference, 2009. ACC '09.*, June 2009, pp. 1748–1753.

[14] J. Fernandez-Martinez and E. Garcia-Gonzalo, "Stochastic stability analysis of the linear continuous and discrete pso models," *Evolutionary Computation, IEEE Transactions on*, vol. 15, no. 3, pp. 405–423, June 2011.

[15] F. van den Bergh and A. P. Engelbrecht, "A convergence proof for the particle swarm optimiser," *Fundam. Inf.*, vol. 105, no. 4, pp. 341–374, Dec. 2010.

[16] D. Bratton and J. Kennedy, "Defining a standard for particle swarm optimization," in *2007 IEEE Swarm Intelligence Symposium.*, April 2007, pp. 120–127.

[17] Z.-P. Jiang and Y. Wang, "Input-to-state stability for discrete-time nonlinear systems," *Automatica*, vol. 37, no. 6, pp. 857 – 869, 2001.

[18] D. Yuret, "From genetic algorithms to efficient optimization," Ph.D. dissertation, Massachusetts Institute of Technology, 1994.