A Brief Survey of Classification of PSO (Particle Swarm Optimization) Variants

Abstract- Particle swarm optimization (PSO) is a population based stochastic optimization technique proposed by Kennedy and Eberhart. Particle swarm optimization (PSO) is an artificial intelligence (AI) technique that can be used to find approximate solutions to extremely difficult or impossible numeric maximization and minimization problems. This paper presents the various available variants of PSO, which enhances the performance of original PSO and improves the limitations which are there in original PSO algorithms such as: the solution gets stuck in local optima, premature convergence. These different variants have distinct features appropriate for particular kind of environments.

Keywords- Particle swarm optimization, premature convergence, local optima, search space.

1. Introduction

A. Particle Swarm Optimization Algorithm

PSO algorithm has been evolved based on the behavior of individuals of a swarm[1]. It is based on the movement of individuals (e.g., fishes, birds, or insects) within a group. It has been noticed that members within a group seem to share the information among them. It leads to an increased efficiency of the group.

The algorithm of PSO emulates from behavior of animals societies that don’t have any leader in their swarm. Individual birds are only able to communicate with other individuals. PSO was originally adopted for balancing weights in neural networks [2] Particle Swarm Optimization (PSO) is a biologically inspired computational search and optimization method based on the social behaviors of birds flocking or fish schooling. [3]

The PSO algorithm PSO is an iterative process. During each iteration of algorithm, current velocity of every particle is updated according to equations:

\[
\begin{align*}
    \mathbf{v}(t+1) &= \mathbf{v}(t) + (c_1 \cdot \mathbf{r}_1 \cdot (\mathbf{p}(t) - \mathbf{x}(t))) + (c_2 \cdot \mathbf{r}_2 \cdot (\mathbf{g}(t) - \mathbf{x}(t))) \\
    \mathbf{x}(t+1) &= \mathbf{x}(t) + \mathbf{v}(t+1)
\end{align*}
\]

Here, \( \mathbf{v}(t+1) \) represents the updated velocity vector and \( \mathbf{x}(t+1) \) is the updated position vector \( c_1, c_2 \) are constant terms to weigh the performance of PSO. \( \mathbf{r}_1, \mathbf{r}_2 \) are random variables to weigh the performance of PSO.

\( p(t) \)- pbest solution

\( g(t) \)- gbest solution

2. Variants of Particle Swarm Optimization

Some of these variants have been proposed to incorporate the capabilities of other evolutionary computation techniques, such as hybrid versions of PSO or the adaptation of PSO parameters for a better performance (adaptive PSO). In other cases, the nature of the problem to be solved requires the PSO to work under complex environments as in the case of the multi-objective or constrained optimization problems or tracking dynamic systems. PSO variants are continually being devised [5] [6] [7] [8] [9] [10][11] [12] [13] to overcome some deficiencies in original PSO. Variants of PSO enhance the performance of original PSO and improves the limitations which are there in original PSO algorithms such as: the solution gets stuck in local optima, premature convergence. Variations to the original formulation can be included to improve its performance.
A. Basic Variants of PSO
Variations in PSO reduces the lacks of PSO. Many variations of PSO have been developed which can improve the convergence speed and solution’s quality. Basic variations in PSO include:

1. **Velocity clamping** - Velocity clamping controls the global exploration of particle. If the velocity of particle exceeds the maximum speed limit set then it is set back to the limit allowed.

2. **Inertia Weight** - Inertia weight controls the exploration-exploitation ratio of swarm and it eliminates the need of velocity clamping. Inertia weight weighs the contribution of previous velocity and thus, controls the momentum of particle.

3. **Constriction PSO** - The Constriction variant balances the exploration and exploitation, and improves the solution found by PSO. It is helpful in achieving convergence.

4. **Synchronous Versus Asynchronous Updates** - Synchronous updates are better for gbest (global best) solutions. These updates are done for the position updates, gives one feedback per iteration update. Asynchronous updates are better for lbest (local best) solutions. They give immediate feedback as opposite to synchronous updates, in which there is slow feedback.[14]

B. Extended PSO variants

1. **Attractive Repulsive PSO** - In this variant, problem of premature convergence is limited by switching between attraction and repulsion phase. Attraction phase is similar to basic PSO in which particles converge to a single point while in repulsive phase, particles are repelled by global best particle.

2. **Mutation based variants** - In order to bring the PSO solution out of local optima and enhancing the global search, a mutation operator is added that is conceptually equal to mutation in Genetic Algorithm.

3. **GLBest PSO** - The GLBest PSO comprises of PSO parameters (inertia weight and acceleration coefficient), in terms of local best and global best positions of the particles.

4. **PSO with Time Varying Acceleration Coefficients** - Time Varying Acceleration Coefficients are introduced to efficiently control local search and convergence to global optimum solution.

C. Hybridized PSO
A natural evolution of the particle swarm algorithm can be achieved by incorporating methods that have already been tested in other evolutionary computation techniques. Many authors have considered incorporating selection, mutation and crossover, as well as the differential evolution (DE), into the PSO algorithm. The main goal is to increase the diversity of the population by: either preventing the particles to move too close to each other and collide to self-adapt parameters such as the constriction factor, acceleration constants or inertia weight. As a result, hybrid versions of PSO have been created and tested in different applications. The most common ones include hybrid of genetic algorithm and PSO (GA-PSO), evolutionary PSO (EPSO) and differential evolution PSO (DEPSO and C-PSO).
Hybridized PSO | Features
--- | ---
*Hybrid of Genetic Algorithm and PSO (GA-PSO)* | 1. combines the advantages of swarm intelligence (PSO) and a natural selection mechanism (GA).
2. increase the number of highly evaluated agents.
3. decreases the number of lowly evaluated agents at each iteration step.

*Hybrid of Differential Evolution and PSO (DEPSO and C-PSO)* | 1. eliminate the particles falling into local minima (DEPSO).
2. find the optimal parameters (inertia and acceleration constants) for the canonical PSO (composite PSO).

*Hybrid of Evolutionary Programming and PSO (EPSO)* | 1. at each iteration step, half of the individuals are moved to positions of the search space that are closer to the optimal solution.
2. help the optimum to be found more consistently than the original particle swarm.
3. Self adaptation capabilities to the swarm by modifying the concept of a particle.

Table: PSO algorithm hybridized with certain other algorithms

3. Classification of PSO Variants
Various PSO variants have been proposed to improve the performance of PSO for global optimization. The existing PSO variants can be mainly classified into the following four categories.
1. By adjusting parameter configurations
   a. Introduction of New parameter - Inertia weight $w$ has been introduced to influence the convergence in PSO algorithm. Larger inertia weight is better for global search, while small inertia weight enhance the ability of the local refinement. The velocity update rule after introduction to PSO, changes to:
   $$v(t+1) = (w \cdot v(t)) + (c_1 \cdot r_1 \cdot (p(t) - x(t))) + (c_2 \cdot r_2 \cdot (g(t) - x(t)))$$

   b. Variants of PSO - Tuning parameters can significantly improve the performance of PSO [4]. Some new variants of PSO has been introduced:
   - *Time varying acceleration coefficients (TVAC)*
   - *Hierarchical PSO with TVAC (HPSO-TVAC)*

2. Defining neighbourhood topologies- This category of variants try to increase the diversity by defining the neighbourhood topologies. Topological structures like: *ring topology* and *von-Neumann topology* have been studied. A large neighbourhood is suitable for the simple optimization problems and a small neighbourhood is sufficient for the complex optimization multimodal problems.
   - The ring topology denotes the RPSO version of PSO
   - The von Neumann topology denotes the VPSO version of PSO

3. PSO hybridized with auxiliary search techniques- PSO is hybridized with some auxiliary search techniques. These can be categorized as follows:
   a. Search operators- Some search operators have been used to improve the performance of PSO algorithm. The genetic operators like: selection, crossover, mutation can be used.
b. Evolutionary computation paradigms- Some Evolutionary computation paradigms such as: ant colony optimization, genetic algorithm and differential evolution can be used.

c. Biology inspired operations- Some biology inspired operations such as species conserving, speciation can be used.

d. Conventional local search technique- This proposed technique resulted in small- population based PSO (SPPSO).

4. Introduce multiswarm techniques- The original works of PSO are quite simple and they work better in low dimensional space. Considering this, the search space has been decomposed and it used the multiswarm to optimize the components of a solution vector cooperatively.

   - Dynamic multiswarm PSO (DMS_PSO) : DMS-PSO has been introduced which has a small swarm size and it dynamically changes the swarm topologies. It manages to perform better on the complex problems [4].

4. **Modified PSO variants**
   There are three categories in which PSO can be modified. These include:
   - adjustment of parameters
   - extension of search space.
   - hybrid with another techniques

   1. Single solution PSO- PSO variations have been found to locate a single solution to continuous-valued and single-objective problems.

   2. Niching with PSO- Algorithms that locate multiple solutions are niching algorithms. These algorithms are based on natural process like: large number of individuals compete for the use of limited resources.

   3. Constrained Optimization using PSO- The constraint factor reduces search space for finding the solution to problem.

   4. Multi-objective Optimization with PSO- These algorithms are used in multi-objective environments, which find a set of solution that can optimally balance trade-offs among the objective of Multiple Objective PSO.

   5. Dynamic Environment with PSO- Several solutions to dynamic environment include:

      a) Environment detection
      b) Response to environment change
      c) Change in inertia weight update
      d) Re-initialize Particle Solution
      e) Limit memory
      f) Local search
      g) Split adaptive PSO
      h) Fine-grained
      i) Charged swarm

   6. Discrete PSO- Though PSO was developed for continuous-valued spaces, but can be defined for discrete valued spaces.

5. **Other Variants of PSO**
   This section presents discrete variants of PSO and other variations to the original formulation that can be included to improve its performance, such as dissipative PSO, which introduces negative entropy to prevent premature stagnation.; adaptive PSO that adjusts the parameters to improve its performance etc.
<table>
<thead>
<tr>
<th>PSO Variants</th>
<th>Features</th>
</tr>
</thead>
</table>
| **Dissipative PSO**                  | 1. Introduces negative entropy and Prevents premature stagnation.  
2. Stretching and passive congregation techniques and Prevents particles form being trapped in local minima.                                              |
| **Species-based PSO (SPSO).**        | 1. Each species is grouped around a dominating particle called the species seed.  
2. At each iteration step, the species seeds are identified and adopted as neighborhood bets for the species groups.  
3. find multiple local optima after multiple iterations, from which the global optimum can be identified.                                                  |
| **Adaptive PSO**                     | 1. adding a random component to the inertia weight applying Fuzzy logic.  
2. The adjustment of the number of particles.  
3. Adjust the swarm size.  
4. Adjust the inertia weight.  
5. Adjust the neighborhood size.                                                                                                     |
| **PSO Variants**                     |                                                                                                                                                                                                          |
| **Multi-objective PSO (MOPSO)**      | 1. Aggregate the multiple objectives into one objective function considering weights that can be fixed or dynamically changing during the optimization process.  
2. it is not always possible to find the appropriate weighted function.                                                                          |
| **Gaussian PSO (GPSO)**              | 1. introduce Gaussian functions for guiding the movements of the particles.  
2. the inertia constant is no longer needed  
3. the acceleration constant is replaced by random numbers with Gaussian distributions.                                                              |
| **PSO With Passive Congregation (PSOPC)** | 1. a mechanism that allows animals to aggregate into groups  
2. prevent the PSO algorithm from being trapped in local optima  
3. to improve its accuracy and convergence speed.                                                                                       |
| **Cooperative PSO (CPSO)**           | 1. employs cooperative behavior in order to significantly improve the performance of the original PSO algorithm.  
2. It uses multiple swarms to optimize different components of the solution vector cooperatively.                                              |
| **Comprehensive Learning PSO (CLPSO)** | 1. For each dimension of particle, a random number is generated  
2. if this random number is greater than a certain value, then the particle will follow its...                                                                 |
own, otherwise it will learn from another particle’s

**Binary PSO (BPSO)**
- Moving in the spaces means a change in the probability of the fact that the value of position coordinate is zero or one.

**Constrained optimization via PSO (COPSO)**
- Applied to constrained single-objective problems.
- A technique is employed to investigate the constraints.

**Evolutionary Programming and PSO (EPPSO)**
- Combination of the PSO and EP algorithms.
- Maintains a balance between local and global search
  Faster convergence of the EP algorithm.

2. **Genetic binary PSO model (GBPSO)**
- The dynamic conditions and discovery power are increased in the swarm.
- Some of the child particles are added to the swarm and some others die and are separated from the swarm.

**Self-organization PSO (SOPSO)**
- A feedback agent is used to improve the particle performance.
- It sets and improves its behavior in next iteration.
- It avoids premature convergence of the total algorithm.

3. **PSO with area extension (AEPSO)**
- Designed for movement of several robots in an area
- It has in fact some in regard to information increasing from an extended area

<table>
<thead>
<tr>
<th>Table 3: Other variants of PSO</th>
</tr>
</thead>
</table>

### 3. Conclusion and Future Scope
Many variants of PSO have been surveyed and presented here. These Variants perform better than the original PSO algorithms. Every variant has different feature, which can be used for different kinds of environments. A sound knowledge of these variants can help the researchers to extend their researches in a particular field. Many of the variants have been presented and many more can be found and developed in future.

### 4. References


Author
Er. Avneet Kaur