6

Particle Swarm Optimization

6.1 The PSO Method

Inspired by animal behavior, Eberhart and Kennedy [49, 22] proposed in 1995 an optimization method called *Particle Swarm Optimization* (PSO). In this approach, a swarm of particles simultaneously explore a problem's search space with the goal of finding the global optimum configuration.

6.2 Principles of the Method

In PSO the *position* $\mathbf{x}_i$ of each particle i corresponds to a possible solution to the problem, with fitness $f(\mathbf{x}_i)$. In each iteration of the search algorithm the particles move as a function of their *velocity* $\mathbf{v}_i$. It is thus necessary that the structure of the search space allows such movement. For example, searching for the optimum of a continuous function in $\mathbb{R}^n$ offers such a possibility.

The particles' movement is similar to a flock of birds or a school of fish, or to a swarm of insects. In these examples, it is assumed that the animals move by following the individual in the group that knows the path to the optimum, perhaps a source of food. In addition, however, the individuals also follow their instinct and integrate the knowledge they have about the optimum into their movements.

In the PSO method two quantities $\mathbf{x}_i^{best}(t)$ and $\mathbf{B}(t)$ have to be defined and updated in each iteration. The first one, $\mathbf{x}_i^{best}(t)$, which is often called *particle-best*, corresponds to the best fitness point visited by particle i since the beginning of the search. The second quantity, $\mathbf{B}(t)$, called *global-best*, is the best fitness point reached by the population as a whole up to time step t:

$$\mathbf{B}(t) = \text{argmax}_{\mathbf{x}_i^{best}} f(\mathbf{x}_i^{best}(t))$$

In certain variants of PSO the *global-best* position $\mathbf{B}(t)$ is defined with respect to a sub-population to which a given individual belongs. The subgroup can be defined

B. Chopard, M. Tomassini, *An Introduction to Metaheuristics for Optimization*,
Natural Computing Series, https://doi.org/10.1007/978-3-319-93073-2_6

by a neighborhood relationship, either geographical or social. In this case, $\mathbf{B}$ will depend on i.

Therefore, as illustrated in Figure 6.1, the particles' movement in PSO is determined by three contributions. In the first place, there is a term accounting for the "inertia" of the particles: this term tends to keep them on their present trajectory. Second, they are attracted towards $\mathbf{B}(t)$, the global best. And third, they are also attracted towards their best fitness point $\mathbf{x}_i^{best}(t)$.

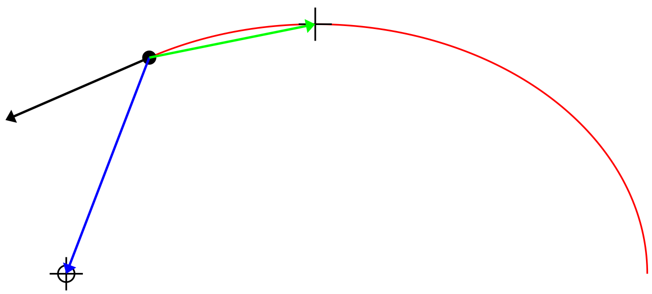

Fig. 6.1. The three forces acting on a PSO particle. In red, the particle's trajectory; in black, its present direction of movement; in blue, the attraction toward the *global-best*, and in green, the attraction towards the *particle-best*

Mathematically, the movement of a particle from one iteration to the next is described by the following formulas:

$$\begin{aligned}\mathbf{v}_i(t+1) &= \omega \mathbf{v}_i(t) + c_1 r_1(t+1)[\mathbf{x}_i^{best}(t) - \mathbf{x}_i(t)] \\ &\quad + c_2 r_2(t+1)[\mathbf{B}(t) - \mathbf{x}_i(t)] \\ \mathbf{x}_i(t+1) &= \mathbf{x}_i(t) + \mathbf{v}_i(t+1)\end{aligned} \tag{6.1}$$

where ω, c_1 and c_2 are constants to be specified, and r_1 and r_2 are pseudo-random numbers uniformly distributed in the interval $[0, 1]$. We remark that a different random number is used for each velocity component.

The c_1 parameter is called the *cognitive coefficient* since it reflects the individual's own "perception," and c_2 is called the *social coefficient*, since it takes into account the group's behavior. For example, $c_1 \approx c_2 \approx 2$ can be chosen. The ω parameter is the *inertia* constant, whose value is in general chosen as being slightly less than one.

Besides formulas (6.1), one must also impose the constraints that each velocity component must not be allowed to become arbitrarily large in absolute value. To this end, a $\mathbf{v}_{max}$ cutoff is prescribed. In the same way, the positions $\mathbf{x}_i$ are constrained to lie in a finite domain having diameter x_{max}.

In the initialization phase of the algorithm the particles are distributed in a uniform manner in the search domain and are given zero initial velocity. The relations above make it clear that it is necessary to work in a search space in which the arithmetic operations of sum and product make sense. If the problem variables are Boolean it is possible to temporarily work in real numbers and then round the results. The method can also be extended to combinatorial problems [56], although this is not the natural frame for this approach, which is clearly geared towards mathematical optimization.

Similarly to the ant colony method, PSO is a population-based metaheuristic. In each iteration, n candidate solutions are generated, one per particle, and the set of solutions is used to construct the next generation. PSO is characterized by rapid convergence speed. Its problem-solving capabilities are comparable to those of other metaheuristics such as ant colonies and evolutionary algorithms, with the advantage of simpler implementation and tuning. There have been several applications of the method [68, 75, 1], and it has proved very competitive in the field of optimization of difficult continuous functions.

6.3 How Does It Work?

In order to intuitively understand how and why PSO can find an optimum in a given fitness landscape, perhaps the global one, we shall consider a toy example. Figure 6.2 illustrates a PSO with two particles in a one-dimensional space ($x \in [-1, 8]$) with a simple fitness function $f(x)$ that is to be maximized. We find that, after a sufficient number of iterations, the two particules have traveled towards the maximum of f, as they should.

According to the general PSO equations (6.1), here the following system must be solved for $i = 1, 2$:

$$v_i(t+1) = 0.9v_i(t) + [b_i(t) - x_i(t)] + [B(t) - x_i(t)]$$
$$x_i(t+1) = x_i(t) + 0.2v_i(t+1)$$

where t is the iteration number of the process, $b_i(t)$ is the *particle-best*, and $B(t)$ is the *global-best*.

Initially the two particles are at rest, randomly placed in the search space. Their $b_i(0)$ are thus their respective positions $x_i(0)$. The *global-best* $B(0)$ corresponds to the position of the "best" particle, represented here by the black one.

- The grey particle is attracted towards the black particle but the latter, being already the *global-best*, doesn't move.
- Since the grey particle is increasing its fitness, its *local-best* continues to be its current position, which doesn't modify its attraction towards the black particle.
- Thanks to its momentum, the grey particle will overtake the black one and will reach a better fitness value.
- In this way, the grey particle becomes the new *global-best*, slows down progressively and stops.

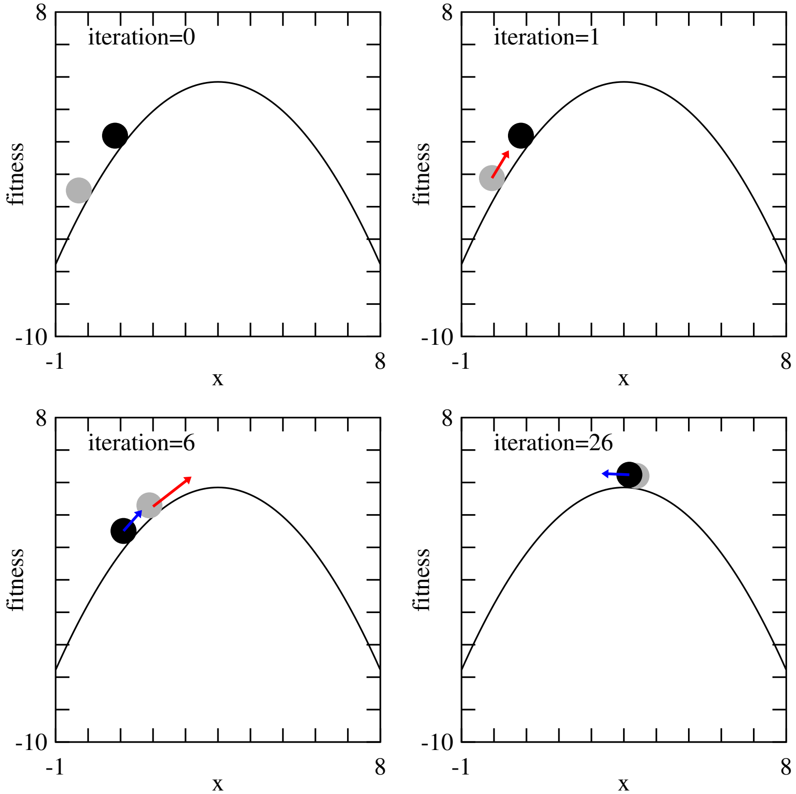

Fig. 6.2. PSO example with two particles in the one-dimensional space $x \in [-1, 8]$ with a parabolic fitness function of which the maximum is sought. To better illustrate the evolution, particles are shown here moving on the fitness curve; actually, they only move along the x axis

- Once the grey particle has passed it, the black particle starts moving towards the grey particle.

6.4 Two-Dimensional Examples

In this section we look at two examples of PSO in which several particles explore a relatively complex subspace of $\mathbb{R}^2$. For the sake of the numerical simulation, the continuous space has been discretized as a grid of 80×60 points. It is interesting to observe the trajectory of the moving particles and their approach to the global maximum. The problem is simple enough for an exhaustive search to be applied since there are only $80 \times 60 = 4{,}800$ points in the search space.

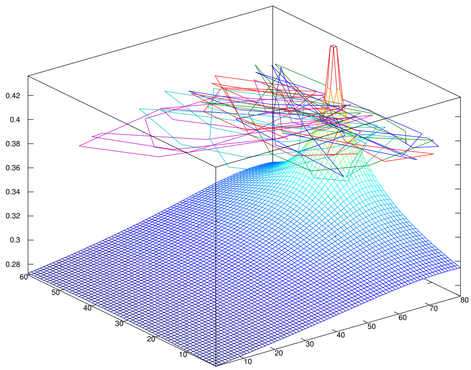

Fig. 6.3. An example of PSO in 2D on a single-maximum fitness landscape

The example, illustrated in Figure 6.3, has the following properties:

- Global optimum at: (75, 36); fitness value at the global optimum: 0.436,
- With five particles, 50 iterations, the best solution found in a single run (Figure 6.3) was

$$\mathbf{B} = (75, 39) \qquad f(\mathbf{B}) = 0.384$$

- With 20 particles, 100 iterations, the optimal solution was found in each run.
- We note that the particles are grouped around the maximum at the end.
- In this example $r_1 = r_2 = 1$ (see eq. 6.1), and the particles are reflected by the domain borders.

Figure 6.4 gives an example of a more difficult search space with several maxima. The global optimum is at $(22, 7)$, with a fitness value of 0.754. With 10 particles and 200 iterations the best solution found by PSO in one run was

$$\mathbf{B} = (23, 7) \qquad f(\mathbf{B}) = 0.74$$

This is very close to the global optimum. The computational effort can be estimated as the product of the number of particles times the number of iterations, that is $10 \times 200 = 2{,}000$, which is less than half the effort needed for an exhaustive search of the 4,800 points in the space.

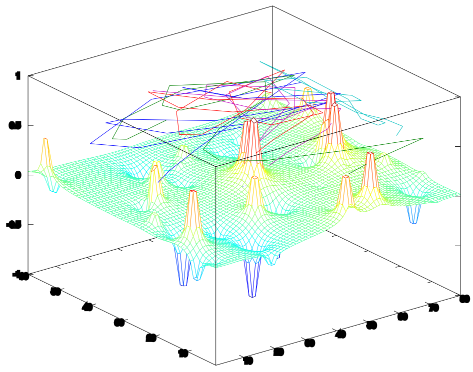

Fig. 6.4. An example of PSO in 2D with a multimodal fitness landscape