An Estimation of Distribution Particle Swarm Optimization Algorithm

Mudassar IQBAL[1] and Marco A. MONTES DE OCA[2]

[1] Computing Laboratory, University of Kent, Canterbury, United Kingdom
mi26@kent.ac.uk

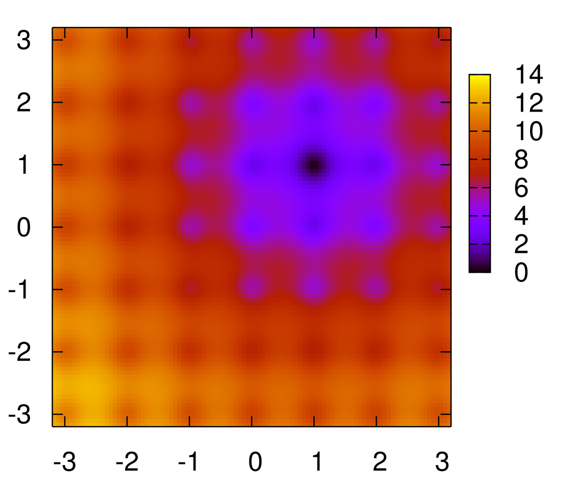
[2] IRIDIA, CoDE, Université Libre de Bruxelles, Brussels, Belgium
mmontes@ulb.ac.be

Summary

An estimation of distribution mechanism is added to the so-called canonical particle swarm optimizer. The resulting algorithm uses the information gathered throughout the optimization process, to probabilistically guide the particles' movement towards the estimated promising regions of the search space. Our experiments show that the new algorithm is able to find similar or better solutions than the canonical particle swarm optimizer but with fewer function evaluations.

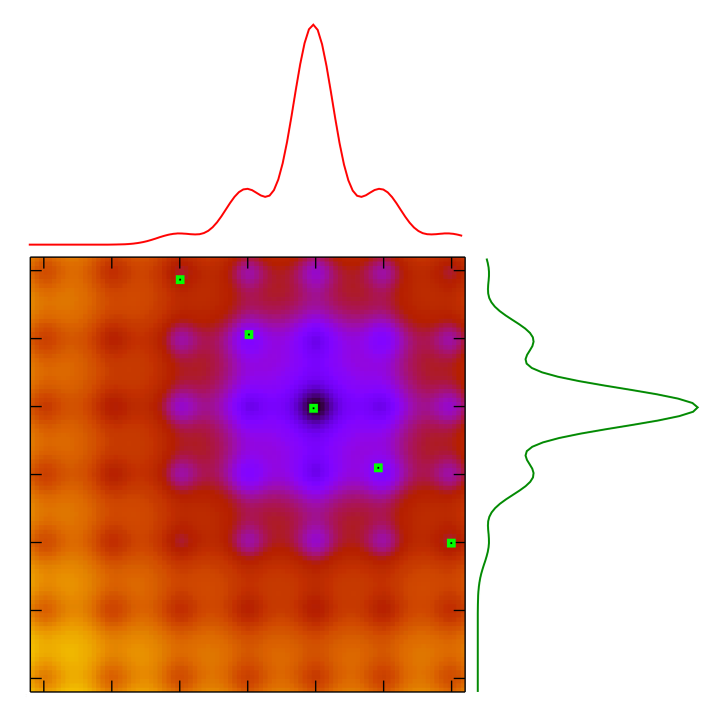
Motivation

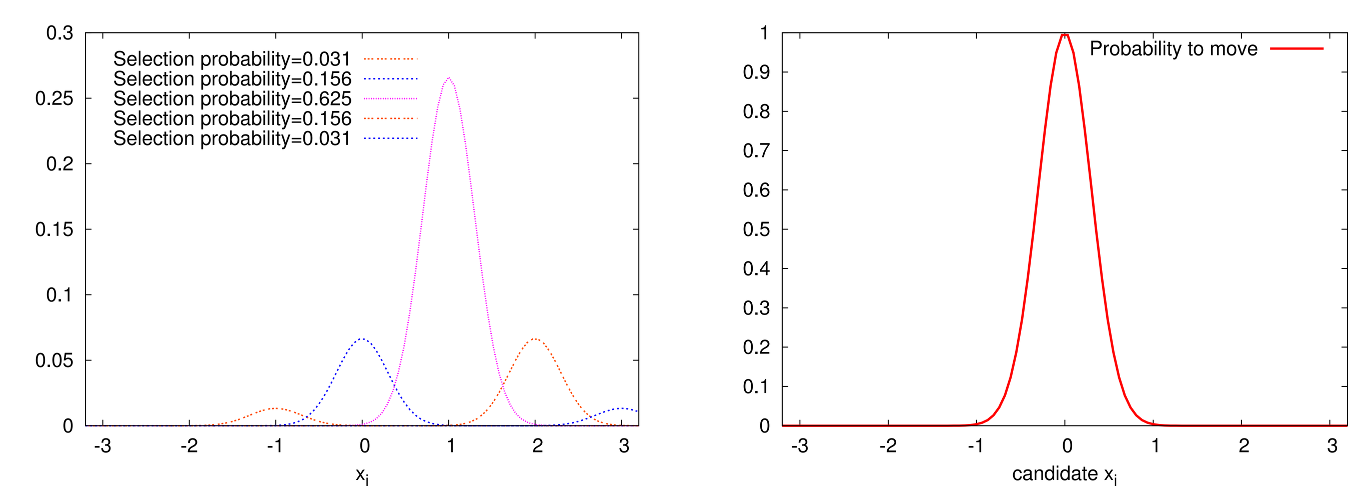
In particle swarm optimization algorithms, the swarm as a whole does not exploit its collective memory (composed by the set of vectors that represent the particles' previous best positions) to guide its search. This causes a re-exploration of what should already be known as bad regions of the search space, wasting costly function evaluations.



The estimation of distribution particle swarm optimizer (EDPSO)

EDPSO estimates the distribution of promising regions by using mixtures of weighted Gaussian functions. Weights represent the quality of different search regions. EDPSO works as a canonical particle swarm optimizer under the condition that particles do not try to move too far from good regions. In such a case, EDPSO substitutes the move with a sample from the probabilistic model.





$$w_l = \frac{1}{qk\sqrt{2\pi}} e^{-\frac{(l-1)^2}{2(qk)^2}}, \quad p_l = \frac{w_l}{\sum_{j=1}^{k} w_j}, \quad \sigma_l^i = \xi \sum_{j=1}^{k} \frac{|s_j^i - s_l^i|}{k-1}$$

Experiments and Results

Experimental Setup

We compared the canonical PSO and EDPSO on five well-known benchmark functions (Sphere, Rosenbrock, Rastrigin, Griewank, and Ackley).

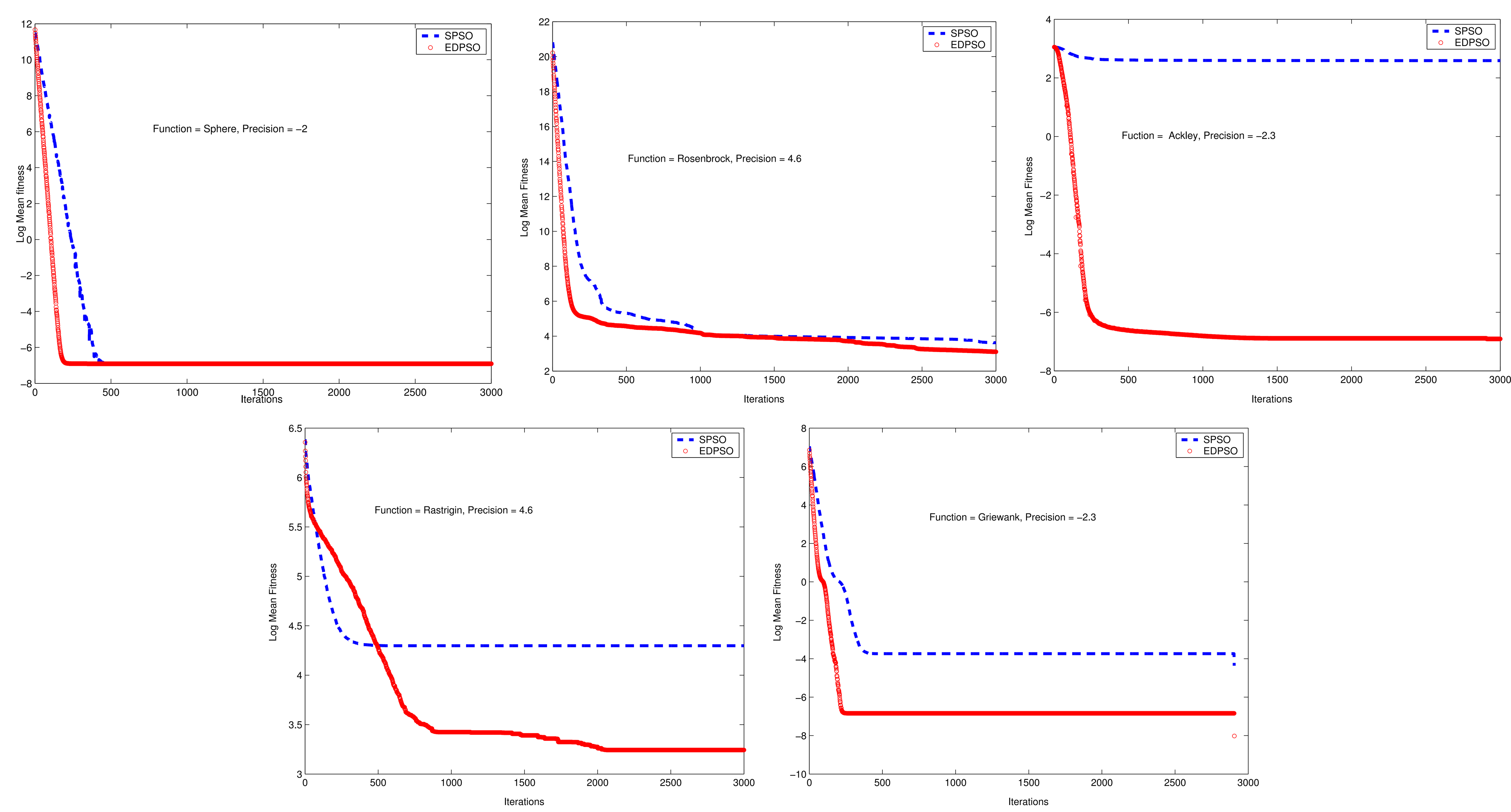
We ran 30 independent runs for each function in 30,40 and 50 dimensions for a maximum of 120 000, 160 000, and 200 000 function evaluations (i.e., 4000 iterations) respectively. The number of particles k was equal to 40.

The following table shows the algorithms' parameter settings used in our experiments.

Algorithm	Parameter	Value
Canonical PSO	φ_1	2.05
	φ_2	2.05
	χ	0.729
EDPSO	φ_1	2.05
	φ_2	2.05
	χ	0.729
	q	0.1
	ξ	0.85

The parameter $q > 0$ determines the degree of preferability of good solutions. The smaller q, the stronger the preference of the best solutions to guide the search.
The parameter $\xi > 0$ controls the wideness of the chosen Gaussian function. The smaller ξ, the narrower the range over which EDPSO searches.

Results



Conclusions

In EDPSO, the particle swarm's collective memory is used to contain the movement of the particles within the most promising regions of the search space. EDPSO operates as a canonical particle swarm optimizer whenever the estimation of distribution mechanism does not contain the movement of particles. When a particle tries to move too far from a good region, EDPSO substitutes the move with a sample from a probabilistic model. In many cases, EDPSO finds better solutions than the canonical particle swarm optimizer; however, speed is sometimes sacrificed.

Mudassar Iqbal acknowledges support from the Computing Laboratory, University of Kent and the EPSRC grant GR/T11265/01 (eXtended Particle Swarms). Marco A. Montes de Oca acknowledges support from the ANTS project, an *Action de Recherche Concertée* funded by the Scientific Research Directorate of the French Community of Belgium, and from Programme Alβan, the European Union Programme of High Level Scholarships for Latin America, through scholarship No. E05D054889MX.