VSP International
Science Publishers
P.O. Box 346, 3700 AH Zeist
The Netherlands

Lecture Series on Computer and Computational Sciences
Volume 1, 2004, pp. 868-873

UPSO: A Unified Particle Swarm Optimization Scheme

K.E. Parsopoulos[1] and M.N. Vrahatis[2]

Department of Mathematics, University of Patras Artificial Intelligence Research Center (UPAIRC), University of Patras, GR–26110 Patras, Greece

Received 31 July, 2004; accepted in revised form 5 September, 2004

Abstract: We introduce Unified Particle Swarm Optimization, a new scheme that harnesses the local and global variants of the standard Particle Swarm Optimization algorithm, combining their exploration and exploitation abilities. Convergence in probability can be proved for the new approach in unimodal cases and preliminary results justify its superiority against the standard Particle Swarm Optimization.

Keywords: Optimization, Particle Swarm Optimization, Stochastic Algorithms, Swarm Intelligence

Mathematics Subject Classification: 90C26, 90C30, 90C59

1 Introduction

Particle Swarm Optimization (PSO) is a stochastic, population–based optimization method. Up–to–date it has been applied successfully on a plethora of test problems in diverse scientific fields [1, 5, 6, 7]. Its efficiency can be attributed to the information exchange among the search points that constitute the population. There are two main variants of PSO with respect to the information exchange scheme that is used, each with different exploration and exploitation characteristics. Practitioners usually select the most proper variant based on their experience as well as on the special characteristics of the problem at hand. Unified Particle Swarm Optimization is a new scheme that harnesses the two variants of PSO in a unified scheme that combines their exploration and exploitation capabilities. Under assumptions, convergence in probability can be proved for the new approach. Preliminary results on a widely used set of benchmark problems are indicative of the new scheme's efficiency.

2 Unified Particle Swarm Optimization

Emergent behavior in socially organized colonies constituted a great source of inspiration for computer scientists. Ant colonies, bird flocks and fish schools that could tackle efficiently combinatorial and numerical optimization problems were modeled and applied successfully on numerous benchmark and real–life problems, giving rise to the class of *swarm intelligence* algorithms [3].

PSO is a swarm intelligence optimization algorithm developed by Eberhart and Kennedy [3]. It employs a population called a *swarm*, $\mathbb{S} = \{x_1, \ldots, x_N\}$, of search points called *particles*, $x_i = (x_{i1}, x_{i2}, \ldots, x_{in})^\top$, $i = 1, \ldots, N$, which probe the search space, $S \subset \mathbb{R}^n$, simultaneously. The

[1]Corresponding author. E-mail: kostasp@math.upatras.gr
[2]E-mail: vrahatis@math.upatras.gr

algorithm works iteratively. Each particle is initialized to a random position in the search space. Then, at each iteration, each particle moves with an adaptable *velocity*, $v_i = (v_{i1}, v_{i2}, \ldots, v_{in})^\top$, while retaining in a memory the best position, $p_i = (p_{i1}, p_{i2}, \ldots, p_{in})^\top \in S$, it has ever visited in the search space. In minimization problems, best positions have lower function values. The particle's movement is also influenced by the experience of the rest particles, i.e., by their best positions. This is performed through the concept of neighborhood. More specifically, each particle is assigned a neighborhood which consists of some prespecified particles. Then, the particles that comprise the neighborhood share their experience by exchanging information. There are two main variants of PSO with respect to the number of particles that comprise the neighborhoods. In the *global* variant, the whole swarm is considered as the neighborhood of each particle, while, in the *local* variant, smaller neighborhoods are used. Neighboring particles are determined based rather on their indices than their actual distance in the search space.

Let g_i be the index of the best particle in the neighborhood of x_i, i.e., the index of the particle that attained the best position among all the particles of the neighborhood. The particles are considered in a ring topology. Thus, their indices are considered in a cyclic order, i.e., 1 is the index that follows after N. At each iteration, the swarm is updated according to the equations [1, 7],

$$v_i^{(k+1)} = \chi \left[v_i^{(k)} + c_1 r_1 \left(p_i^{(k)} - x_i^{(k)} \right) + c_2 r_2 \left(p_{g_i}^{(k)} - x_i^{(k)} \right) \right], \tag{1}$$

$$x_i^{(k+1)} = x_i^{(k)} + v_i^{(k+1)}, \tag{2}$$

where $i = 1, \ldots, N$; k is the iterations' counter; χ is a parameter called *constriction factor* that controls the velocity's magnitude; c_1 and c_2 are positive acceleration parameters, called *cognitive* and *social* parameter, respectively; and r_1, r_2 are random vectors that consist of random values uniformly distributed in $[0, 1]$. All vector operations in Eqs. (1) and (2) are performed componentwise. A stability analysis of PSO, as well as recommendations regarding the selection of its parameters are provided in [1, 7].

The performance of a population–based algorithm depends on its ability to perform global search of the search space (exploration) as well as more refined local search (exploitation). Proper balance between these two characteristics result in enhanced performance. In the global variant of PSO, all particles are attracted by the same overall best position, converging faster toward specific points. Thus, it has better exploitation abilities. On the other hand, in the local variant, the information of the best position of each neighborhood is communicated slowly to the other particles of the swarm through their neighbors. Therefore, the attraction to specific best positions is weaker, hindering the swarm from getting trapped in locally optimal solutions. Thus, the local variant of PSO has better exploration ability. Proper selection of the neighborhood's size affects the trade–off between exploration and exploitation. The selection of the most proper neighborhood size is an open problem. In practice, it is up to the practitioner and it is based solely on his experience.

Unified Particle Swarm Optimization (UPSO) is a new scheme that harnesses the global and the local variant of PSO, thereby combining their exploration and exploitation capabilities. Let $\mathcal{G}_i^{(k+1)}$ denote the velocity update of the ith particle, x_i, in the global PSO variant, while $\mathcal{L}_i^{(k+1)}$ denotes the corresponding velocity update for the local variant. Then, according to Eq. (1),

$$\mathcal{G}_i^{(k+1)} = \chi \left[v_i^{(k)} + c_1 r_1 \left(p_i^{(k)} - x_i^{(k)} \right) + c_2 r_2 \left(p_g^{(k)} - x_i^{(k)} \right) \right], \tag{3}$$

$$\mathcal{L}_i^{(k+1)} = \chi \left[v_i^{(k)} + c_1 r_1' \left(p_i^{(k)} - x_i^{(k)} \right) + c_2 r_2' \left(p_{g_i}^{(k)} - x_i^{(k)} \right) \right], \tag{4}$$

where k denotes the iteration number; g is the index of the best particle of the whole swarm (global variant); and g_i is the index of the best particle in the neighborhood of x_i (local variant). These

two search directions can be combined in a single equation, resulting in the main UPSO scheme,

$$U_i^{(k+1)} = u\,\mathcal{G}_i^{(k+1)} + (1-u)\,\mathcal{L}_i^{(k+1)}, \tag{5}$$

$$x_i^{(k+1)} = x_i^{(k)} + U_i^{(k+1)}, \tag{6}$$

where $u \in [0,1]$ is a parameter called the *unification factor*, which determines the influence of the global and local components in Eq. (5). For $u = 1$, Eq. (5) is equivalent to the global PSO variant, while for $u = 0$ it is equivalent to the local PSO variant. For all intermediate values, $u \in (0,1)$, we obtain composite variants of PSO that combine the exploration and exploitation characteristics of its global and local variant.

UPSO can be further enhanced by incorporating a stochastic parameter in Eq. (5) that imitates the mutation of evolutionary algorithms, however, it is directed toward a direction which is consistent with the PSO dynamics. Thus, Eq. (5) can be written as,

$$U_i^{(k+1)} = r_3\,u\,\mathcal{G}_i^{(k+1)} + (1-u)\,\mathcal{L}_i^{(k+1)}, \tag{7}$$

which is mostly based on the local variant or, alternatively,

$$U_i^{(k+1)} = u\,\mathcal{G}_i^{(k+1)} + r_3\,(1-u)\,\mathcal{L}_i^{(k+1)}, \tag{8}$$

which is mostly based on the global variant, where $r_3 \sim \mathcal{N}(\mu, \sigma^2 I)$ is a normally distributed parameter, and I is the identity matrix.

A proof of convergence in probability can be given for the schemes of Eqs. (7) and (8). The proof follows the analysis of Matyas [4] for stochastic optimization algorithms. Assume that $F : S \to \mathbb{R}$ is a unimodal objective function, x_{opt} is its unique minimizer in S, and $F_{\text{opt}} = F(x_{\text{opt}})$. Also, let $x_i^{(k)}$ be the ith particle of the swarm and $p_i^{(k)}$ be its best position in the kth iteration. The proof does not take into consideration the index i, therefore we will refer to them as $x^{(k)}$ and $p^{(k)}$, respectively. The level set of F at a constant value, K, is defined as $G[K] = \{x : F(x) < K\}$. We assume that $G[K] \neq \emptyset$, for all $K > F_{\text{opt}}$. Let $A^{(k+1)} = (1-u)\,\mathcal{L}^{(k+1)}$, $B^{(k+1)} = u\,\mathcal{G}^{(k+1)}$, and $f(z)$ be the probability distribution of $r_3\,B^{(k)}$, with $r_3 \sim \mathcal{N}(\mu, \sigma^2 I)$. The choice of the Normal distribution as the probability distribution of r_3, guarantees that $f(z) \neq 0$, for all z, although the proof holds for any choice of probability distribution for r_3, as long as this relation holds. We define as a *successful step* of UPSO at iteration k, the fact that $F\left(x^{(k+1)}\right) < F\left(p^{(k)}\right) - \varepsilon$, for a prescribed $\varepsilon > 0$. The probability of a successful step from $x^{(k)}$ is given by

$$P_F(x) = \int_{G[F(p)-\varepsilon]} f(z-x)dz.$$

Then, based on the analysis of Matyas [4], the following theorem is straightforwardly proved:

Theorem 1 *Let $F(x)$ have a unique minimum in S, $G[K] \neq \emptyset$, for all $K > F_{\text{opt}}$, and $f(z) \neq 0$ for all z. Then, at least one sub–sequence of best positions, $\{p^{(k)}\}$, of any particle, x, of the swarm in UPSO tends in probability to x_{opt}.*

Proof. Let $\delta(x) = \{z : \varrho(z,x) < \delta\}$, $\delta > 0$, be the δ–neighborhood of a point x. We will prove that for any $\delta > 0$ it holds that,

$$\lim_{k\to\infty} P\left\{\varrho\left(p^{(k)}, x_{\text{opt}}\right) > \delta\right\} = \lim_{k\to\infty} P\left\{p^{(k)} \notin \delta(x_{\text{opt}})\right\} = 0,$$

i.e., the probability that the distance $\varrho\left(p^{(k)}, x_{\text{opt}}\right) > \delta$, or equivalently that $p^{(k)} \notin \delta(x_{\text{opt}})$, tends to zero. If we denote by F_δ the minimum value of F on the boundary of $\delta(x_{\text{opt}})$, we shall have

$F_\delta > F_{\text{opt}}$. We can now define $\varepsilon = \varepsilon(\delta)$ such that $0 < \varepsilon(\delta) < F_\delta - F_{\text{opt}}$. For all previous best positions, $p \notin \delta(x_{\text{opt}})$, of the particle under consideration, the inequality $F(p) - \varepsilon > F_{\text{opt}}$, is valid. Furthermore, from the assumptions of the theorem, $G[F(p) - \varepsilon]$ is a non–empty region. Since $f(z) > 0$ for all z, there will exist an $\alpha > 0$, such that $P_F(x) \geqslant \alpha$, i.e., the probability of a successful step from x is positive (although in some cases it may become very small).

Let $F\left(x^{(1)}\right) = F\left(p^{(1)}\right)$ be the initial function value of x and p, respectively (recall that the initial position and the initial best position of a particle coincide). We denote $\tau = \left(F\left(p^{(1)}\right) - F_\delta\right)/\varepsilon$, and $m = \lfloor \tau \rfloor$, i.e., m is the largest integer less than τ. From the design of the PSO and UPSO algorithm, if even $m+1$ steps turn out to be successful, then all the subsequent points of the sequence $\left\{p^{(k)}\right\}$ lie in $\delta(x_{\text{opt}})$. Consequently, the probability $P\left\{p^{(k)} \notin \delta(x_{\text{opt}})\right\}$ is less than or equal to the probability that the number of successful steps does not exceed m, i.e.,

$$P\left\{p^{(k)} \notin \delta(x_{\text{opt}})\right\} \leqslant P\left\{\sum_{i=1}^{k} y^{(i)} \leqslant m\right\},$$

where, $y^{(i)} = 1$, if there was a successful step in iteration i, and $y^{(i)} = 0$, otherwise. The latter probability increases with a decrease in the probability of successful steps, and since $P_F(x) \geqslant \alpha$, it obeys the well–known Newton's theorem (on the binomial probability distribution),

$$P\left\{\sum_{i=1}^{k} y^{(i)} \leqslant m\right\} \leqslant \sum_{i=0}^{m} \binom{k}{i} \alpha^i (1-\alpha)^{k-i},$$

where k is the number of steps (iterations) taken. Further, when $k > 2m$ and $\alpha < 0.5$,

$$\sum_{i=0}^{m} \binom{k}{i} \alpha^i (1-\alpha)^{k-i} < (m+1)\binom{k}{m}(1-\alpha)^k = \frac{m+1}{m!} k(k-1)(k-2)\cdots(k-m+1)(1-\alpha)^k$$

$$< \frac{m+1}{m!} k^m (1-\alpha)^k.$$

Consequently, $P\left\{\varrho(p^{(k)}, x_{\text{opt}}) > \delta\right\} < \frac{m+1}{m!} k^m (1-\alpha)^k$. Thus, for $\alpha > 0$, it is clear that,

$$\lim_{k \to \infty} k^m (1-\alpha)^k = 0,$$

and the theorem is proved. ■

We must note that the best position p of the particle x may be replaced by the overall best position of the whole swarm, with minor modifications in the proof.

3 Experimental Results and Discussion

The performance of UPSO was investigated on the test set used by Trelea in [7], and it consists of the Sphere, Rosenbrock, Rastrigin, and Griewank function in 30 dimensions, as well as the Schaffer's function in 2 dimensions. The test functions are denoted as F_1, F_2, F_3, F_4, and F_5, respectively. For comparison purposes, the PSO configuration reported in [7] was also adopted here. Specifically, two sets of parameters were used, denoted as Set 1 and Set 2, respectively. Set 1 consists of $\chi = 0.6$ and $c_1 = c_2 = 2.833$, which is the equivalent of the set $a = 0.6$ and $b = 1.7$ in [7], and Set 2 consists of $\chi = 0.729$, $c_1 = c_2 = 2.05$, which is the equivalent of the set $a = 0.729$ and $b = 1.494$ in [7]. The maximum number of iterations was 10000. The swarm was initialized in the range $[-100, 100]^{30}$ for the Sphere function, $[-30, 30]^{30}$ for the Rosenbrock function, $[-5.12, 5.12]^{30}$ for the Rastrigin function, $[-600, 600]^{30}$ for the Griewank function, and $[-100, 100]^2$ for Schaffer's

Table 1: Results for UPSO. Success rates (in parenthesis) and expected number of function evaluations are reported.

Success Rates & Expected Number of Function Evaluations								
	$u = 0.9$				$u = 0.1$			
	$\mu = 0$		$\mu = 1$		$\mu = 0$		$\mu = 1$	
N	Set 1	Set 2	Set 1	Set 2	Set 1	Set 2	Set 1	Set 2
F_1 (Sphere). Best value reported in [2, 7] is (1.00)10320								
15	(1.00)**3078**	(1.00)3754	(1.00)**3981**	(1.00)5268	(1.00)**2892**	(1.00)3658	(1.00)**3876**	(1.00)5401
30	(1.00)6588	(1.00)6924	(1.00)6852	(1.00)8807	(1.00)6278	(1.00)7011	(1.00)6579	(1.00)8427
60	(1.00)12174	(1.00)14679	(1.00)11370	(1.00)14238	(1.00)12771	(1.00)14640	(1.00)11433	(1.00)14202
F_2 (Rosenbrock). Best value reported in [2, 7] is (0.50)15930								
15	(1.00)**2138**	(1.00)2413	(1.00)**4950**	(0.95)10451	(1.00)**2193**	(1.00)2570	(1.00)**3656**	(1.00)10333
30	(1.00)4385	(1.00)4835	(1.00)6765	(1.00)17714	(1.00)4223	(1.00)5330	(1.00)8601	(1.00)11466
60	(1.00)8505	(1.00)9936	(1.00)12186	(1.00)16626	(1.00)8040	(1.00)9765	(1.00)15450	(1.00)14688
F_3 (Rastrigin). Best value reported in [2, 7] is (0.90)4667								
15	(0.90)16541	(1.00)8412	(1.00)**4373**	(1.00)11315	(1.00)3820	(1.00)**2889**	(1.00)**6846**	(1.00)7506
30	(1.00)8766	(1.00)**7233**	(1.00)8744	(1.00)11673	(1.00)5805	(1.00)9294	(1.00)8196	(1.00)10256
60	(1.00)11475	(1.00)11640	(1.00)15882	(1.00)19437	(1.00)8433	(1.00)10293	(1.00)14013	(1.00)17934
F_4 (Griewank). Best value reported in [2, 7] is (1.00)9390								
15	(1.00)**3161**	(1.00)3275	(1.00)**4157**	(1.00)5473	(1.00)**3312**	(1.00)3620	(1.00)**4343**	(1.00)5383
30	(1.00)6507	(1.00)7229	(1.00)6914	(1.00)8114	(1.00)6009	(1.00)6870	(1.00)6516	(1.00)8300
60	(1.00)11844	(1.00)14019	(1.00)10725	(1.00)13230	(1.00)12150	(1.00)14640	(1.00)10839	(1.00)13458
F_5 (Schaffer). Best value reported in [2, 7] is (0.75)6440								
15	(0.90)24153	(0.85)20927	(0.95)21922	(0.95)37092	(1.00)37790	(1.00)**18591**	(1.00)36664	(0.95)**19234**
30	(1.00)**16716**	(1.00)19437	(1.00)22254	(1.00)36020	(1.00)25433	(1.00)26211	(1.00)29874	(1.00)22680
60	(1.00)25425	(1.00)24072	(1.00)21531	(1.00)**19203**	(1.00)33411	(1.00)20115	(1.00)21456	(1.00)22764

function, while the corresponding error goals were 0.01, 100, 100, 0.1, and 10^{-5} [7]. For each function, 20 experiments were performed, using three different swarm sizes, 15, 30, and 60. The particles were allowed to move anywhere in the search space without constraints on their velocity. Regarding the configuration of UPSO, two cases were investigated, namely the case of Eq. (8) with unification factor $u = 0.9$, which is closer to the global PSO variant used in [7], and Eq. (7) with $u = 0.1$, which is closer to the local PSO variant. For both cases, two different configurations of the distribution of r_3 were considered, namely, one with mean value $\mu = 0$, and one with mean value $\mu = 1$. The standard deviation of r_3 was $\sigma = 0.01$ in all cases, to avoid wide deterioration of the PSO dynamics. The neighborhood radius for the determination of the local PSO search direction, $\mathcal{L}_i$, in Eqs. (7) and (8), was 1, i.e., the neighbors of the ith particle, x_i, were the particles x_{i-1} and x_{i+1}. This selection was made in order to take full advantage of the properties of the local version, since the larger the neighborhoods the closer is the local variant to the global one.

For each parameter configuration and test function, the success rate of UPSO, namely the fraction of the experiments in which the error goal was achieved, as well as the corresponding expected number of function evaluations, defined as (Number of Particles)×(Average Number of Iterations)/(Success Rate) [7], were recorded and they are reported in Table 1. The results were very promising. The success rate never fell under 0.90 (i.e., 90%), while (in most cases) the expected number of function evaluations was smaller than the values reported in [7] and [2]. UPSO achieved success rates of 100% even in cases where the plain PSO had very low success rates. For example, in the cases of F_4 and F_5 and a swarm of 15 particles, the plain PSO with the parameters of Set 1 had a success rate of just 0.35 and 0.45, respectively, as reported in [7], while UPSO's success rate was higher than 0.90 for any set of parameters. Furthermore, the success rate and the lowest expected number of function evaluations for the test functions F_1–F_5, as reported in [7], was (1.0)10320, (0.50)15930, (0.90)4667, (1.00)9390, (0.75)6440, respectively. In UPSO, the

corresponding numbers were (1.00)2892, (1.00)2138, (1.00)2889, (1.00)3161, and (1.00)16716 (for each case the lowest expected number of function evaluations is bold faced in the table). We must notice that in the case of F_5, the number 6440 reported in [7] corresponds to a success rate equal to 0.75, while the number 16716 of UPSO corresponds to a success rate equal to 1.00.

Regarding the different configurations of UPSO, the version of Eq. (7) with $u = 0.1$ and $\mu = 0$ had the better overall performance, probably due to the better exploration ability of the local PSO variant, which is favored in this scheme. Moreover, the Set 1 proposed by Trelea in [7] outperformed the (most popular) Set 2. Also, UPSO with $\mu = 0$ outperformed that with $\mu = 1$, in most problems.

Further experiments were performed using the plain UPSO scheme of Eq. (5), both in static and dynamic optimization problems, revealing that the values $u = 0.5$ and $u = 0.2$ result in enhanced performance of the algorithm. The mathematical properties behind this effect are still under investigation, along with possible correlations between the UPSO scheme and PSO variants with adaptive neighborhood size.

4 Conclusions

A Unified Particle Swarm Optimization (UPSO) that aggregates the local and the global variant of PSO in a unified scheme has been introduced. The proposed approach seems to exploit the good properties of both variants and preliminary experiments on the test set used in [7] justify its efficiency. Further investigation is required to analyze the dynamics of UPSO. A self–adaptive scheme that will exploit knowledge of the characteristics of the objective function, as well as the performance of the algorithm, to control the unification factor is currently under development, along with an analysis of its convergence rates.

Acknowledgment

The authors wish to thank the anonymous referees, as well as Dr. M. Clerc and Dr. I.C. Trelea for their careful reading of the manuscript and their fruitful comments and suggestions.

References

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

[2] Eberhart, R.C. and Shi, Y., Comparing inertia weights and constriction factors in particle swarm optimization, In *Proc. 2000 IEEE CEC*, pp. 84–88, Piscataway, NJ, 2000. IEEE Service Center.

[3] Kennedy, J. and Eberhart, R.C., *Swarm Intelligence*, Morgan Kaufmann Publishers, 2001.

[4] Matyas, J., Random optimization, *Automatization and Remote Control*, **26**, pp. 244–251, 1965.

[5] Parsopoulos, K.E. and Vrahatis, M.N., Recent approaches to global optimization problems through particle swarm optimization, *Natural Computing*, **1**(2–3), pp. 235–306, 2002.

[6] Parsopoulos, K.E. and Vrahatis, M.N., On the computation of all global minimizers through particle swarm optimization, *IEEE Transactions on Evolutionary Computation*, **8**(3), pp. 211–224, 2004.

[7] Trelea, I.C., The particle swarm optimization algorithm: Convergence analysis and parameter selection, *Information Processing Letters*, **85**, pp. 317–325, 2003.