2006 IEEE Congress on Evolutionary Computation
Sheraton Vancouver Wall Centre Hotel, Vancouver, BC, Canada
July 16-21, 2006

The Lévy Particle Swarm

Toby J. Richer, *Member, IEEE,* and T. M. Blackwell, *Member, IEEE*

***Abstract*— Many foragers and wandering animals have been shown to follow a Lévy distribution of steps and it is conjectured here that this distribution may be useful for optimization algorithms too. This paper investigates the effectiveness of replacing the particle dynamics within Particle Swarm Optimization (PSO) by random sampling from a Lévy distribution. The resulting Lévy PSO is demonstrated to perform as well, or better, than a standard PSO or equivalent Gaussian models over a range of benchmark problems.**

I. Introduction

Bumble bees, wandering albatrosses, wild and fenced deer and other foraging animals are known to follow a random walk (or flight) in their search for food. A strategy of random foraging in circumstances where there is no a priori information on the location of food sources might indeed be expected to be optimal, but it is surprising that many biological searchers actually follow a Lévy, or scale-free distribution of steps [1][2][3][4][5][6]. Lévy walks or flights are distinguished from Brownian walks by their absence of a characteristic scale: the second moment of the Lévy distribution is infinite as a consequence of the power law distribution of flight lengths in the tail of this distribution. This means that very large step lengths are possible.

Some insight on the origins of this power law behavior has been provided in a mathematical model of animal searching patterns [7]. In this analysis, Benichy and co-workers, incorporate two distinct phases of behavior, a phase of quick motion from one location to another, followed by a phase of slow, random "diffusion". Such behavior, typically observed in a dog searching for a buried bone in the garden, could usefully be employed in human search problems too, for example in the search for avalanche victims or people lost at sea.

Other suggestions on the optimality of Lévy rather than Gaussian or Poisson distributions of walks and flights [6] has included observation that the probability of returning to a previously visited sight is less, and the number of new sights visited is greater, for Lévy walkers than it is for Brownian walkers. In fact Lévy walkers diffuse so rapidly that there is a greatly reduced competition for food around their place of origin.

The absence of prior information in global optimization is somewhat analogous to the problem faced by animal foragers. The design of evolutionary search algorithms also involves a balance between exploration and exploitation, and it might be wondered if an insight from foragers might be useful in this domain. In fact, Lee and Yao [8] have shown the favorability of Lévy probability distributions over Gaussian in the optimization of multi-modal functions by evolutionary programming (EP). The reason for this, as suggested by the authors, is that Lévy distributions generate offspring further away from its parent than the commonly employed Gaussian mutation.

Particle Swarm Optimization (PSO) is another evolutionary optimization technique. In PSO, particles (which represent problem solutions) fly over a problem landscape. The particles share information about promising solutions with other particles in the swarm; each particle is attracted to the best location that it has found, and also the best location found by other particles in its topological neighborhood. Particle velocity is constricted so that convergence is possible [9] and the resulting algorithm is versatile and efficient at real space optimization. Analysis of particle flights around the overall (global) best solution resembles a Gaussian distribution, but with a higher proportion of outliers [10][11]. Replacement of particle movement by a Gaussian probability distribution has a deleterious effect on the algorithm, a situation that can be somewhat improved, but in an arbitrary way, with the inclusion of "bursts" of outliers [11].

This paper considers the possibility of replacing, within PSO, particle motion with a Lévy flight. The purpose of this research is to incorporate the Lévy distribution into the algorithm, and determine if Lévy-based PSO can perform optimization better, or more simply, than Gaussian based or standard PSO (PSO using a uniform probability distribution). The motivation is that the power law behavior of the Lévy distribution at large step length ("fat tails") will induce exploration at any stage of the convergence, enabling escape from local minima. The Lévy PSO should reproduce the "'Gaussian with Bursts"' idea, but within a simpler and more intuitive scheme.

The next section introduces the Lévy distribution, and describes an algorithm to generate random numbers within this distribution. Section 3 describes the set of PSO variants that serve as the basis for the new Lévy-based PSO variants. Section 4 describes the method used to apply the Lévy distribution to Gaussian-based PSO, outlines the resulting Lévy-based PSO variants, and describes the experiments used to determine the best form of Lévy distribution for use in the PSO. Sections 5 and 6 describe the comparison of Lévy-based PSO with Gaussian-based and standard PSO and illustrate convergence properties and Section 7 describes the general conclusions of this work.

Toby J. Richer is with the Department of Computing, Goldsmiths College, New Cross, London, SE16 6NW, UK (email: t.richer@gold.ac.uk)

T. M. Blackwell is with the Department of Computing, Goldsmiths College, New Cross, London, SE16 6NW, UK (email: t.blackwell@gold.ac.uk)

0-7803-9487-9/06/$20.00/©2006 IEEE

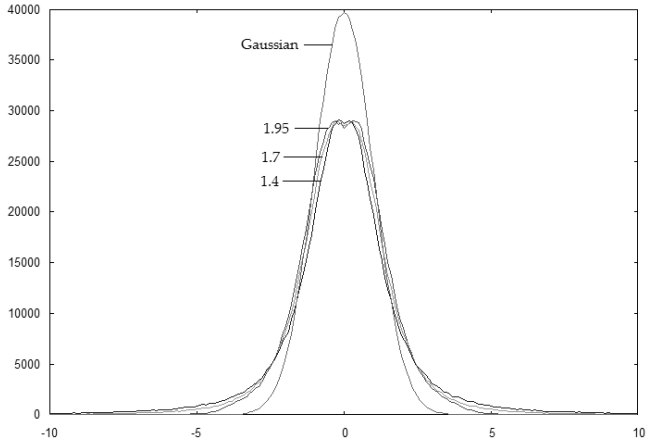

Fig. 1. Gaussian and Lévy distributions

II. The Lévy Distribution

The central limit theorem states that the sum of a large number of samples of any probability distribution will converge to the normal distribution, provided the distribution has finite moments [12]. Lévy found that probability distributions with power-law behavior at high values and thus infinite moments do not obey the central limit theorem. The sum of samples from these distributions converge to the Lévy distribution [13].

The Lévy distribution has the probability density:

$$L_{\alpha,\gamma}(z) = \frac{1}{\pi}\int_0^\infty \exp(-\gamma q^\alpha)\cos(qz)dq \tag{1}$$

The parameters characterizing the distribution are α, controlling the sharpness of the graph, and γ, controlling the scale unit of the distribution. For $\alpha = 2$, the distribution is equivalent to the Gaussian distribution. For $\alpha = 1$, the distribution is equivalent to the Cauchy distribution. Graphs of distribution for varying values of α are shown in Figure 1. As shown, the Lévy distribution for $\alpha < 2$ is similar to the Gaussian distribution, but with fatter tails. The scale parameter γ can be set to one without loss of generality [8]. Henceforth, $L_{\alpha,1}$ will be denoted L_α. The power law behavior of the Lévy distribution, crucial for the appearance of fatter tails, is apparent at large z, where Equation 1 has the limiting form

$$L_\alpha \approx \frac{1}{y^{(\alpha+1)}}, |y| >> 1 \tag{2}$$

The limiting form of the distribution demonstrates that the second moment is infinite for $0 < \alpha < 2$, and the scaling law $L_\alpha(by) = bL_\alpha(y)$, for $b \in \Re$.

Since the analytic form of Equation 1 is not known for general α, a generator of Lévy random numbers is commonly used [13]. In the first stage of this algorithm, two normal stochastic variables x and y with standard deviations $\sigma_x$and σ_y are generated. $\sigma_x$and σ_y are dependent on α and each other, so σ_y is set to 1 and σ_x is set to a value dependent on α. The variable v is generated according to

$$v = \frac{x}{|y|^{1/\alpha}}$$

A variable in the Lévy distribution, w, can then be generated using the non-linear transform

$$w = \{[K(\alpha)-1]\exp(-v/C(\alpha))+1\}v$$

To obtain Lévy distributions with a scale factor γ other than 1, the linear transformation

$$z = \gamma^{1/\alpha}w \tag{3}$$

is applied. The values of σ_x,$K(\alpha)$ and $C(\alpha)$ for specified α are given in [13]. The distribution created with this transform converges rapidly and accurately to the Lévy distribution.

III. Algorithms for Test

The purpose of this paper is to introduce the Lévy PSO and to compare its effectiveness to other PSOs. There are a range of PSO algorithms which replace the particle dynamics of standard PSO by sampling from a Gaussian distributions. Each one of these Gaussian PSOs can become a Lévy-based PSO by replacing the Gaussian by a Lévy distribution, and this is the basis of our approach. The standard and Gaussian PSOs are described in this section.

A. Standard PSO

For the purposes of this paper, the Clerc-Kennedy PSO [9] will be taken as the standard form. This version of PSO has the advantage of a stability proof and supersedes earlier PSOs which had variable inertial weight. The general form of the algorithm is given in Algorithm 1 and the particle update rules are given in Equations 4 and 5.

Algorithm 1 General process for Particle Swarm Optimization

```
FOR each time-step
  FOR each particle
    update particle position x_t
    calculate particle fitness f(x)
    update p_t, g_t
  END FOR
END FOR
```

$$\begin{aligned} v_{t+1} &= \chi(\omega v_t + c_1 U(0,\frac{\psi}{2})(p_t - x_t) \\ &\quad + c_2 U(0,\frac{\psi}{2})(g_t - x_t)) \end{aligned} \tag{4}$$

$$x_{t+1} = x_t + v_{t+1} \tag{5}$$

Here, p_t refers to the individual memory carried by the particle (the best position it has attained), and g_t is the best of these memories amongst the topological neighborhood of the particle, which we will take to be the *gbest* network. In *gbest*, there is global information sharing so that any

particle is informed by all the others. The inertial weight, w, is fixed at 1.0 in standard PSO, as are the spring constants c_1,c_2. The stability proof of Clerc and Kennedy is satisfied by $\psi = 4.1$, and this commonly used value is taken in this paper. This value of ψ in turn sets the constriction factor χ at 0.729843788 [9]. U is a random number drawn from the interval $\left[0, \frac{\phi}{2}\right]$, and Equations 4 and 5 are applied to each component of the particle's position and velocity.

B. Gaussian Bare Bones PSO

As described in Section 1, the distribution of particles in a PSO around a fixed local and global best resemble a Gaussian distribution, but with a greater number of particles at the outliers. Due to this similarity, the Bare Bones PSO was proposed [10][11] to test if a Gaussian distribution alone could substitute for individual particle velocity and momentum.

The algorithm for the Bare Bones PSO is identical to Algorithm 1 with no velocity update. At each step, the new position of each particle in each dimension is randomly selected from a Gaussian distribution with a mean at the midpoint of the particle and neighborhood best positions, and a standard deviation equal to the distance between the particle and neighborhood best positions. The position is updated according to the equation:

$$x_{t+1} = N\left(\frac{p_t + g_t}{2}, |p_t - g_t|\right)$$

C. Gaussian PSO

Gaussian PSO [14][15] is identical to standard PSO, except that the uniform distributions used to determine the attraction to the local and global bests are replaced by two Gaussian distributions. The mean value of the local and global velocities are equal to the distance between the particle and the local and global bests, respectively. The standard deviation of the local and global velocities is equal to half the distance between the particle and the local and global bests, respectively. The new velocity update step is:

$$\begin{aligned} v_{t+1} \quad = \quad & \chi(v_t + N\left(p_t - x_t, \frac{|p_t - x_t|}{2}\right)(p_t - x_t) \\ & + N\left(g_t - x_t, \frac{|g_t - x_t|}{2}\right)(g_t - x_t)) \end{aligned}$$

D. Gaussian Burst

Gaussian with bursts is an extension of Gaussian Bare Bones PSO designed to account for the greater number of the outliers and their tendency to appear in bursts [11]. This algorithm operates in 2 modes; for each step of the algorithm, the algorithm has a fixed probability of switching between these modes. In normal mode, the algorithm behaves identically to the Gaussian Bare Bones algorithm. In burst mode, the distribution is altered by taking the relative position of each point to the mean of a randomly determined power. This produces small bursts of points that are far away from the current center of the search space. The algorithm produces similar results to the standard PSO.

E. Gaussian Pivot

Gaussian pivot is a spherically symmetric adaptation of the Bare Bones Gaussian. The new position of each particle is determined by constructing two hyperspheres around the best local point and best neighborhood point. Each hypersphere has a radius equal to the distance between the local and neighborhood best point. A random point is generated (with uniform probability) on the surface of each hypersphere. Each point is weighted according to the fitness of the center point of that sphere, higher fitness giving a greater weight. The weighted average of the two points is the next location of the particle.

IV. Algorithms using the Lévy Distribution

The Bare Bones Gaussian, Gaussian PSO, and Gaussian Pivot were converted to Lévy-based algorithms by substituting the Lévy distribution for each Gaussian distribution. These alterations result in algorithms that are denoted, in turn, Lévy Bare Bones, Lévy PSO and Lévy with Pivot.

A. Applying Lévy distribution to Gaussian Algorithms

The PSO algorithms based on the Gaussian distribution control the search by altering the mean and standard deviation of the appropriate Gaussian distribution. The effective search area is linked to the standard deviation: particles which are close to p_t or g_t will produce a small standard deviation so that regions that are close to these points will be explored, and convergence is possible. But the Lévy distribution has an infinite standard deviation and this mechanism cannot be employed. Instead, another property of the distribution must be manipulated if convergence is to occur.

Two parameters govern the Lévy distribution, α and γ. Although α could, in principle, be adjusted during a run, σ_x, $K(\alpha)$ and $C(\alpha)$ would have to be recalculated and this would slow the search significantly. The scale factor γ, however, can be used to alter the Lévy distribution as described in Equation 3, and so control the search.

A simple methodology was proposed that would imitate the good convergence properties of the Gaussian approach, but would retain the fat tail property of the Lévy distribution. A Gaussian Optimizer sets the standard deviation of the distribution to a value calculated from the current state of the search (for example, the distance between p_t and g_t). A Lévy Optimizer could set the scale value to a value calculated from the same attributes of the search, but transformed in such a manner so as to produce similar behavior to the Gaussian Optimizer.

To do this, we define an "'effective standard deviation"', or σ_{eff}. Given a particular Lévy optimization algorithm, σ_{eff} is the value that the standard deviation would be set to in the equivalent Gaussian optimization algorithm. For example, $\sigma_{eff} = |p_t - g_t|$ for a Gaussian PSO. As the particles converge, $p_t \rightarrow g_t$ and $\sigma_{eff} \rightarrow 0$; the algorithm explores a smaller area of the search space in more detail. To enable the Lévy algorithm based on this Gaussian algorithm to converge in the same manner, the Lévy scale γ is set such

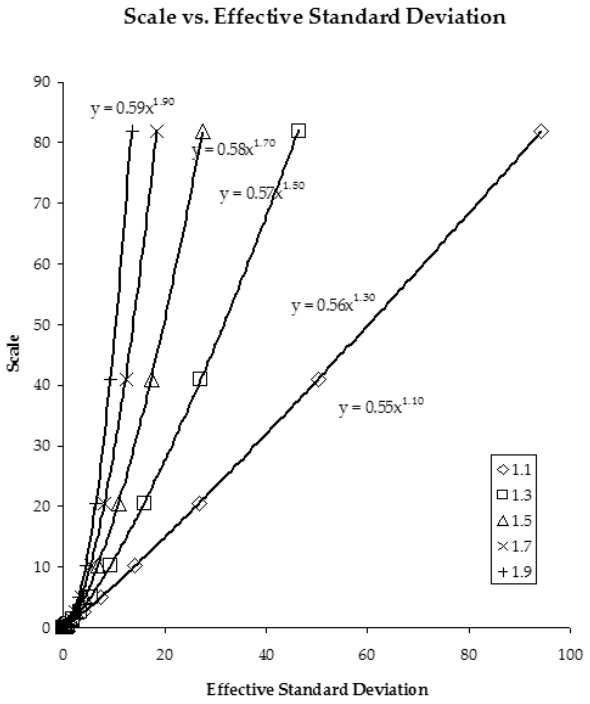

Fig. 2. Relationship between Effective Standard Deviation and Scale (Lines show Best-Fit Power Series)

TABLE I
CONSTANTS IN RELATIONSHIP BETWEEN GAUSSIAN STANDARD DEVIATION AND LÉVY SCALE

α	1.2	1.3	1.4	1.5	1.6
$K(\alpha)$	0.557	0.562	0.565	0.568	0.572
α	1.7	1.8	1.9	1.95	1.99
$K(\alpha)$	0.576	0.580	0.585	0.590	0.633

that a similar proportion of the particles is close to the mean. This can be done by solving

$$\int_{-\sigma_{eff}}^{+\sigma_{eff}} L_{\alpha,\gamma}(z)dz = 0.68 \quad (6)$$

for γ, with a given α.

No analytic solution exists for Equation 6. However, empirical solutions of Equation 6 obey the power law

$$\gamma = K(\alpha)\sigma_{eff}^{\alpha} \quad (7)$$

well (see Figure 2). The values of $K(\alpha)$ for tested α are given in Table I. Adjusting γ according to Equation 7, with σ_{eff} set to $|p_t - g_t|$ (as for the Gaussian PSO), results in a Lévy optimization algorithm that converges in the same manner as a Gaussian optimization algorithm, but with velocities or positions in the Lévy distribution.

B. Parameter Selection

Although it is proposed to adjust the scale factor according to Equation 7 in order to control convergence, the value of α is still undetermined. α determines the probability of obtaining Lévy random numbers in the tail (Equation 2) and might have a significant effect on search. The best value of a was selected using tests on three benchmark problems; Rosenbrock(f_3), Rastrigin(f_5) and Griewank(f_7). Rastrigin and Griewank are two difficult multimodal problems and Rosenbrock is a unimodal problem, but its shape makes finding the minima difficult for many algorithms. For each of these functions, N = 30; The bounds of these functions are shown in Table VI.

TABLE II
RESULTS FOR LÉVY BARE BONES SWARM

α	Griewank - Mean	Rastrigin - Mean	Rosenbrock - Mean
1.99	-0.795	141	24.5
1.95	-0.902	138	30.5
1.9	-0.843	135	27.9
1.8	-0.808	127	24.2
1.7	-0.814	129	26.0
1.6	-0.915	127	28.9
1.5	-0.764	131	29.6
1.4	-0.842	128	21.3
1.3	-0.867	136	28.2
1.2	-0.988	137	27.5

$$f_3(x) = \sum_{i=1}^{N-1} \{100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2\}$$

$$f_5(x) = \sum_{i=1}^{N} \{x_i^2 - 10\cos(2\pi x_i) + 10\}$$

$$f_7(x) = \frac{1}{4000}\sum_{i=1}^{N} x_i^2 - \prod_{i=1}^{N} \cos(\frac{x_i}{\sqrt{i}}) + 1$$

Each value of alpha, for each algorithm, was tested 50 times on each benchmark. For each run, the initial positions of each particle were determined randomly. Each trial lasted 3000 iterations. To determine the effectiveness of each algorithm for each alpha value and each landscape, the best fitness value found after 3000 iterations was averaged over the set of tests for that alpha value and benchmark. These were compared by alpha value, across the three benchmarks. The best α value was selected by ranking the fitnesses for each landscape, summing the ranks, and taking the value that had the lowest summed rank, provided that value of α performed acceptably (in the top half of the rankings) in all tests.

1) Test Results: The results for the Bare Bones Lévy Swarm are given in Table II. The Lévy Bare Bones Swarm produced very similar fitnesses for a range of values of alpha. The value of alpha that produced the best results in this case was 1.4.

The results for the Lévy PSO are given in Table III. Unlike the Lévy Bare Bones Swarm, the average fitness for the Lévy Particle Swarm was highly sensitive to the choice of alpha. Alpha values less than 1.4 produced poor results on all landscapes, and alpha values greater than 1.6 produced poor results on the Rastrigin landscape. The only alpha values that performed well on all landscapes were 1.5 and 1.6. The best value of alpha was 1.5, as it produced a significantly better

TABLE III
RESULTS FOR LÉVY PSO

α	Griewank - Mean	Rastrigin - Mean	Rosenbrock - Mean
1.99	-0.499	123	46.3
1.95	-0.546	112	42.8
1.9	-0.548	118	34.3
1.8	-0.559	104	34.5
1.7	-0.916	82	33.7
1.6	-0.977	44	37.1
1.5	-0.978	27	47.6
1.4	1.93	123	12735
1.3	117	594	1118466
1.2	141	724	1394551

TABLE IV
RESULTS FOR LÉVY PIVOT SWARM

α	Griewank - Mean	Rastrigin - Mean	Rosenbrock - Mean
1.99	-0.0869	220	26.9
1.95	-0.0869	212	26.2
1.9	-0.0844	213	24.3
1.8	-0.0713	225	29.5
1.7	-0.0576	229	28.1
1.6	-0.0369	224	24.6
1.5	-0.0175	222	25.8
1.4	-0.0108	221	27.2
1.3	0.0109	227	28.2
1.2	0.0249	220	30.5

result than all other values for the Rastrigin function while producing good values for the other landscapes.

The results for the Lévy Pivot Swarm are given in Table IV. The value of alpha that produces the best results is 1.9.

For each of the algorithms, a different value of α produces the best performance. As each algorithm covers the search space in a different manner, it is not surprising that different values of α suit each algorithm. In principle, each algorithm might have an optimal alpha for each test function. Although each Lévy swarm could be tuned to give optimal performance on each benchmark, this paper seeks to establish the overall feasibility of the Lévy approach. Hence an *effective* alpha for each algorithm, namely the alpha obtained from the trial experiments on three benchmarks, is used in the subsequent experiments.

C. Analysis of Distributions

The Lévy distribution, as shown in Figure 1, is 'sharper' than the Gaussian distribution. Compared to a Gaussian distribution, it will produce a greater number of points at very small distances and very large distances to the mean, with less points in between these two extremes. When considered in terms of the two main aspects of the optimization task, exploration and exploitation, the Lévy distribution can be said to specialize more. It produces more points at large distances from the mean, and thus dedicated to exploring the search space, and more points at very small distances to the mean, thus dedicated to exploiting the area around the current mean.

In a previous application of the Lévy algorithm to genetic algorithms, Lee and Yao [8] have analyzed the possible results of applying the Lévy distribution to Evolutionary Programming. This showed that the mean-square displacement of mutations is theoretically infinite for all generations. Through a series of simulations, they compared Lévy mutations and Gaussian mutations. The Lévy mutations covered a much greater range of values in the search space than the Gaussian mutations. They propose that this helps the EP algorithm escape from local optima. The movements of particles in PSO are more complicated than the mutation operations of EP, and cannot be analyzed as extensively.

As a guide to the effect that switching from a Lévy distribution to a Gaussian distribution could have on PSO, each of the PSO algorithms were ran for 1000 iterations with fixed local and global bests. The Lévy algorithm used the best α value determined above. For each particle, the local best was held to (-10,-10) and the global best was held to (10,10). To reduce the amount of data required, one point was plotted for each algorithm, for each iteration. These points are shown in Figure 3. The Lévy PSO produces a greater number of outliers than either the Gaussian and Canonical PSOs, and some of these outliers are very high distances from the core of the search - in this case, some are well outside of the bounds of the problem. Canonical PSO has less outliers than Lévy PSO, but more than Gaussian PSO. These outlier points could allow Lévy to escape from local optima more often than Gaussian or Canonical PSO.

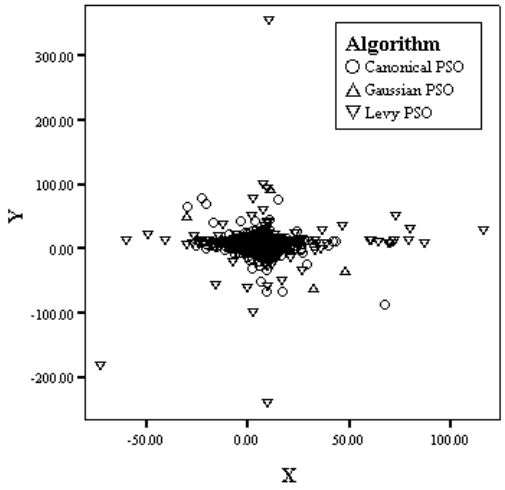

Fig. 3. Points Generated by PSO Algorithms with Fixed Local and Global Best Values.

V. ALGORITHM COMPARISON

To determine whether PSO algorithms based on the Lévy distribution can be as effective as other forms of PSO, Gaussian-based PSO algorithms, Lévy-based PSO algorithms and the standard PSO were all compared by running a series of experiments on a problem set. The problem set used to compare is a set of 9 functions previously used as a benchmark in [8]. These functions are given in Table V. The sub-functions y_i and $u(x)$ are defined as follows:

TABLE V
BENCHMARK FUNCTIONS FOR ALGORITHM COMPARISONS

$f_1(x) =$	$\sum_{i=1}^{N} x_i^2$
$f_2(x) =$	$\sum_{i=1}^{N} (\sum_{j=1}^{i} x_j)^2$
$f_3(x) =$	$\sum_{i=1}^{N-1} \{100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2\}$
$f_4(x) =$	$\sum_{i=1}^{N} x_i \sin(\sqrt{\lvert x_i \rvert})$
$f_5(x) =$	$\sum_{i=1}^{N} \{x_i^2 - 10\cos(2\pi x_i) + 10\}$
$f_6(x) =$	$-20exp\{-0.2\sqrt{\frac{1}{N}\sum_{i=1}^{N} x_i^2}\}$
	$-exp\{\frac{1}{N}\sum_{i=1}^{N} \cos(2\pi x_i)\} + 20 + e$
$f_7(x) =$	$\frac{1}{4000}\sum_{i=1}^{N} x_i^2 - \prod_{i=1}^{N} \cos(\frac{x_i}{\sqrt{i}}) + 1$
$f_8(x) =$	$\frac{\pi}{N}\{10\sin^2(\pi y_i) + \sum_{i=1}^{N-1} (y_i - 1)^2$
	$[1 + 10\sin^2(\pi y_{i+1})] + (y_n - 1)^2\}$
	$+ \sum_{i=1}^{n} u(x_i, 10, 100, 4)$
$f_9(x) =$	$0.1\{\sin^2(3\pi x_1) + (x_N - 1)^2$
	$\{1 + \sin^2(2\pi x_N)\} + \sum_{i=1}^{N} u(x_i, 5, 100, 4)$

TABLE VI
MAGNITUDE OF UPPER AND LOWER BOUNDS OF BENCHMARK FUNCTIONS

f_1, f_2	f_3	f_4	f_5	f_6	f_7	f_8, f_9
± 100	± 30	± 500	± 5.12	± 32	± 600	± 50

$$y_i = 1 + \frac{1}{4}(x_i + 1)$$

$$u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m, & x_i > a, \\ 0, & -a \leq x_i \leq a, \\ k(-x_i - a)^m, & x_i < -a. \end{cases}$$

Functions f_1, f_2 and f_3 are unimodal; functions f_4 to f_9 are multi-modal. Each algorithm is run 100 times on each problem. Each algorithm uses 20 particles, with a mesh topology (each particle is connected to every other) to determine the global best value. In each run, the particles used in the algorithm start in new, randomly-generated positions. To minimize the effect of any bias toward the origin, the initial positions of all particles were constrained to one half of the total search space in each dimension. Each run lasts 3000 iterations; the best fitness for each run is recorded. The average fitness for each algorithm on each problem can then be compared to determine the effect of the Lévy distribution on the problem-solving capability of the algorithm.

A. Results

The mean best fitness by algorithm and problem, with standard error, are shown in Table VII.

To determine if the differences in mean between algorithms are significant, the mean values for each problem were analyzed using an ANOVA (ANalysis Of VAriance) with 0.05 as the level of significance. If ANOVA showed statistically significant differences in the mean best fitness of each algorithm, the results of each algorithm were compared. This was a pairwise comparison of the means for each algorithm, using Tukey's honestly significance difference test with 0.05 as the level of significance. To determine which algorithms could be reliably said to be the most effective for each problem, the algorithm were ranked. Algorithms that were not statistically different to each other were given the same rank; algorithms that were not statistically different to more than one other group of algorithms were ranked with the best-performing of these groups. The resulting rankings are shown in Table VIII.

For one of the nine problem functions, f_1, there was no statistically significant difference between the performance of the algorithms. For five more of the nine problem functions, there was no statistically significant difference between the majority of the algorithms, and the algorithms that were significantly different performed less effectively than this group. The majority of the algorithms performed similarly well for a majority of the problem functions.

The best-performing algorithm over the problem set is the Lévy-based Particle Swarm Optimization algorithm. For each of the problem functions, it performs as well as any other algorithm or performs significantly better than other algorithms. The next best-performing algorithm was the bare bones algorithm based on the Lévy distribution. The other Particle Swarm Optimization algorithms generally performed less well.

As stated in Section 2, the Lévy distribution automatically produces outliers, and this might be beneficial for functions such as f_5 which have a large number of local optima. As expected, Lévy PSO outperformed Gaussian PSO, which has thin tails. Surprisingly, Lévy PSO also outperforms Canonical PSO which is known to also produce outliers. Lévy Bare Bones outperformed other Bare Bones algorithms. This would support the proposition that algorithms based on the Lévy distribution perform well on problems such as f_5 because its particles move to a greater number of outlying positions, and the algorithm is therefore better able to escape local minima.

VI. CONVERGENCE PROPERTIES OF LÉVY AND GAUSSIAN ALGORITHMS

While these results show that Lévy PSO is the most effective of these algorithms across the testbed, it provides little information as to why this should be so. To gather more information about Lévy PSO's relative strengths, each algorithm was run another 10 times on problem f_5, and the fitness measured at every 100 iterations. The average fitness at each iteration, for each algorithm, is shown in Figure 4. Problem f_5 was chosen as the results previously obtained for this problem showed more statistical differences between algorithm results than any other problem. In the early stages of the search, all of the algorithms apart from the pivot algorithms improve at roughly the same rate. All of these algorithms are able to find good values quickly. The Bare Bones algorithms improve most quickly during this phase. In later stages of the search, the rate of each algorithm's improvement decreases, until they stop finding better solutions. All of the algorithms, bar the pivot algorithms and the Lévy PSO, stop finding better solutions before 1000 iterations. The Lévy algorithm is able to continue to improving for much longer than the other algorithms, and thus ends up

TABLE VII
MEAN AND STANDARD ERROR BY ALGORITHM AND PROBLEM (BEST RESULTS IN BOLD)

Algorithm	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9
PSO	2.683E-11	17.463	49.065	-8514.355	92.551	2.372	.087	.229	.307
(Std. Err.)	(1.597E-10)	(25.991)	(37.100)	(447.179)	(21.626)	(1.918)	(.236)	(.500)	(.881)
Gaussian PSO	2.50E-4	91.986	36.352	-7478.383	125.335	11.400	.579	.661	1.005
	(.002)	(341.975)	(35.486)	(566.152)	(28.576)	(3.192)	(1.094)	(.859)	(1.536)
Gaussian Bare Bones	2.436E-41	10.554	33.034	-8503.117	79.258	1.276	.019	.208	.134
	(1.204E-40)	(15.569)	(30.145)	(495.830)	(19.357)	(1.215)	(.028)	(.363)	(.536)
Bare Bones with Burst	6.830E-42	8.296	37.843	-8589.113	82.900	1.460	.013	.231	.083
	(3.945E-41)	(6.995)	(33.908)	(471.355)	(16.876)	(2.216)	(.017)	(.374)	(.285)
Gaussian Pivot	**9.743E-53**	110.943	**31.700**	-5800.832	170.184	.559	**.009**	.084	**.002**
	(4.368E-52)	(137.795)	(44.095)	(1328.624)	(68.489)	(.738)	(.011)	(.349)	(.003)
Lévy PSO	1.806E-11	22.161	43.626	**-9478.158**	**47.560**	.398	.029	**.049**	.018
	(9.931E-11)	(20.143)	(37.900)	(421.141)	(16.984)	(2.200)	(.040)	(.165)	(.098)
Lévy Bare Bones	8.273E-44	**4.240**	42.056	-9233.437	62.613	**.012**	.013	.139	.007
	(3.059E-43)	(3.743)	(38.303)	(440.618)	(16.861)	(.116)	(.014)	(.277)	(.019)
Lévy Pivot	7.666E-26	2279.634	37.964	-5346.042	204.425	.496	.011	.319	.008
	(5.917E-25)	(1356.519)	(58.287)	(1474.744)	(50.731)	(.749)	(.026)	(.810)	(.004)

TABLE VIII
RANKING BY ALGORITHM AND PROBLEM

Algorithm	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9
PSO	=1	=1	8	=3	=3	7	=1	=1	7
Gaussian PSO	=1	=1	=1	6	6	8	8	8	8
Gaussian Bare Bones	=1	=1	=1	=3	=3	5	=1	=1	=1
Bare Bones with Burst	=1	=1	=1	=3	=3	6	=1	=1	=1
Gaussian Pivot	=1	=1	=1	7	7	=1	=1	=1	=1
Lévy PSO	=1	=1	=1	=1	1	=1	=1	=1	=1
Lévy Bare Bones	=1	=1	=1	=1	2	=1	=1	=1	=1
Lévy Pivot	=1	8	=1	8	8	=1	=1	7	=1

with the best solution. It is hypothesized that the fat tails of the distribution mitigate against premature convergence.

The f_1 problem has a single minima, and is a therefore a test of the ability of these algorithms to converge once an optimum has been found. In the tests of f_1 the algorithms were statistically indistinguishable, suggesting that all PSO variants are equally capable hill climbers. It is more likely that the advantage of Lévy PSO at f_5 is because of an ability to better escape local minima, rather than a greater ability to exploit a single minima.

VII. CONCLUSION

Evidence that many foraging animals use power-law governed search strategies is accumulating. Such strategies typically include a long step (fast phase) followed by a phase of diffusion (short steps, slow phase), and are well modeled by the Lévy distribution. These strategies might be expected to be useful in environments where no prior knowledge is available, the "targets" are difficult to detect, and the distribution of targets is sparse. Foragers such as birds, lizards, planktivorous fish and even pet dogs might spend some time carefully examining a local region, and then quickly relocating to a previously unscanned region [7]. These tactics have been shown to be superior to random search based on a Brownian walk.

The extension of these findings into global optimization techniques is the observation that the fat tails of the Lévy distribution will generate candidate solutions that are far from

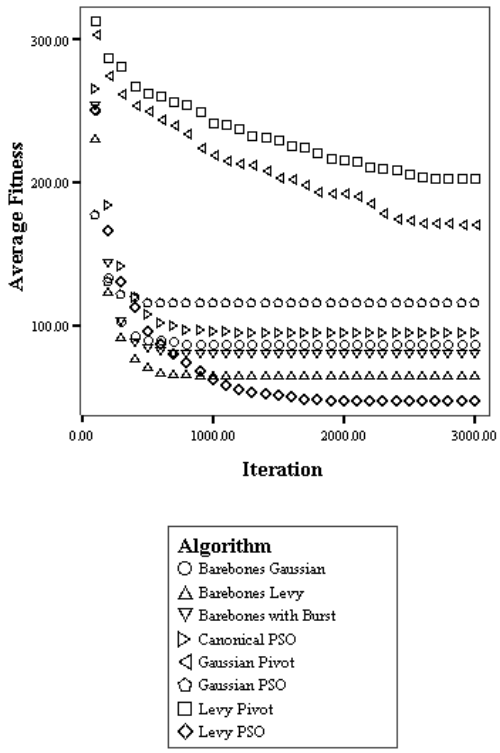

Fig. 4. Average Fitness by Iteration.

a current optimum, and hence promote continued exploration at any stage of the search. This might improve algorithm performance by allowing escape from a local optimum, and the adoption of Lévy mutation in evolutionary programming has been shown to be advantageous [8]. This paper examines the use of Lévy flights in Particle Swarm Optimization, a technique that was inspired by the collective behavior of animal groups. The question is to what extent the search patterns of solitary animals can be integrated with the collective intelligence of the animal group.

In standard PSO, particles are dynamical entities, possessing position and velocity. Particles are updated deterministically in a discrete kinematics by calculating an acceleration, adding this to the velocity, and adding the updated velocity to the position. (Certain parameters governing the particle accelerations are random, but the underlying scheme is deterministic.) Attempts have been made to replace the particle update rule of standard PSO by eliminating velocity and simply placing particles at positions taken from a Gaussian distribution. This scheme has the disadvantage that positions far from an optimum are unlikely to be selected, and the swarm can become stuck in a local optimum. The situation is remedied somewhat by including particle bursts, although the solution is somewhat arbitrary.

This paper has examined the extension of these ideas to the Lévy distribution, a distribution with fatter tails than the Gaussian. This paper has demonstrated that, over a range of standard test functions, a simple replacement of the particle update rule with a Lévy distribution (the so-called Lévy Bare Bones PSO) performs consistently well, equaling or bettering standard PSO. It also performs better or as well as the Gaussian PSOs.

Another Lévy variant, the Lévy PSO, modifies standard PSO by replacing the uniform distributions used to determine the attraction of each particle to the local and global bests with a Lévy distribution. This is the most consistent optimizer of all: it is the best algorithm of all tested in this study on the unimodal functions and performs well on the multimodal functions. In fact it is never bettered by standard PSO or any of the Gaussian PSOs.

These results strengthen a general finding that the deterministic particle update rule of PSO is not necessary for successful optimization. The replacement of the deterministic, but chaotic, dynamics with a probabilistic sampling should promote theoretical understanding of PSO since individual particle motion becomes unimportant.

It can be conjectured that the essence of PSO lies in the communication of information throughout the swarm, and not in the details of the particle motion, a conjecture that would appear to be supported by the findings of this paper.

ACKNOWLEDGMENT

This research was supported by EPSRC grant GR/T11272/01. The authors would like to thank the members of the eXtended Particle Swarms (XPS) project and the anonymous reviewers for their comments on this paper.

REFERENCES

[1] G. Viswanathan, S. Buldyrev, S. Havlin, M. da Luz, E. Raposo, and H. Stanley, "Optimizing the success of random searches," *Nature*, vol. 401, pp. 911–914, 1999.

[2] G. Viswanathan, S. Buldyrev, E. Murphy, P. Prince, and H. Stanley, "Lévy flight search patterns of wandering albatrosses," *Nature*, vol. 381, pp. 413–415, 1996.

[3] M. Levandowsky, J. Klafter, and B. White, "Swimming behavior and chemsensory responses in protozoan microzooplankton as a function of hydrodynamic regime," *Bulletin of Marine Science*, vol. 43, pp. 758–763, 1988.

[4] F. Schuster and M. Levandowsky, "Chemosensory responses of acanthamoeba castellanii: Visual analysis of random movement and responses to chemical signals." *Journal of Eukaryotic Microbiology*, vol. 43, pp. 150–158, 1996.

[5] B. Cole, "Fractal time in animal behaviour: the movement activity of drosophilia." *Animal Behaviour*, vol. 50, pp. 1317–1324, 1995.

[6] G. M. Viswanathan, V. Afanasyev, S. V. Buldyrev, S. Havlin, M. G. E. da Luz, E. P. Raposo, and H. E. Stanley, "Statistical physics of random searches," *Brazilian Journal of Physics*, vol. 31, no. 1, pp. 102–107, 2001.

[7] O. Benichou, M. Coppey, M. Moreau, P. Suet, and R. Voituriez, "Optimal search strategies for hidden targets," April 2005. [Online]. Available: http://arxiv.org/abs/cond-mat/0504107

[8] C.-Y. Lee and X. Yao, "Evolutionary progrmaming using mutations based on the lévy probability distribution," *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 1, 2004.

[9] M. Clerc and J. Kennedy, "The particle swarm: Explosion, stability, and convergence in a multi-dimensional complex space." *IEEE Transactions on Evolutionary Computation*, vol. 6, 2002.

[10] J. Kennedy, "Bare bones particle swarms," in *the 2003 IEEE Swarm Intelligence Symposium (SIS'03)*, 2003, pp. 80–87.

[11] ——, "Probability and dynamics in the particle swarm," in *the 2004 Congress on Evolutionary Computation (CEC2004)*, vol. 1, 2004, pp. 340–347.

[12] O. Kallenberg, *Foundations of Modern Probability*, ser. Springer Series in Statistics, Probability and Its Applications. Springer, 1997.

[13] R. N. Mantegna, "Fast, accurate algorithm for numerical simulation of lévy stable stochastic processes," *Physical Review E*, vol. 49, no. 5, pp. 4677–4683, 1994.

[14] M. Clerc, "The swarm and the queen: towards a deterministic and adaptive particle swarm optimization," in *1999 ICEC*, Washington DC, 1999, pp. 1951–1957.

[15] B. Secrest and G. Lamont, "Visualizing particle swarm optimization - gaussian particle swarm optimization," in *the 2003 IEEE Swarm Intelligence Symposium (SIS'03)*, 2003, pp. 198–204.