Theoretical and Empirical Study of Particle Swarms with Additive Stochasticity and Different Recombination Operators

Jorge Peña
Institut de mathématiques appliquées
Université de Lausanne, Switzerland
jorge.pena@unil.ch

ABSTRACT

Standard particle swarms exhibit both multiplicative and additive stochasticity in their update equations. Recently, a simpler particle swarm with just additive stochasticity has been proposed and studied using a new theoretical approach [14]. In this paper we extend the main results of that study to a large number of existing particle swarm optimisers by defining a general update rule from which actual algorithms can be instantiated via the choice of specific recombination operators. In particular, we derive the stability conditions and the dynamic equations for the first two moments of the sampling distribution during stagnation, and show how they depend on the used recombination operator. Finally, the optimisation efficiency of several particle swarms with additive stochasticity is compared in a suite of 16 benchmark functions.

Categories and Subject Descriptors

I.2.8 [**Artificial Intelligence**]: Problem Solving, Control Methods, and Search

General Terms

Algorithms

Keywords

Swarm Intelligence, Optimization, Recombination Operators, Theory, Empirical Study

1. INTRODUCTION

In the standard formulation of the Particle Swarm Optimiser (PSO), the movement of the particles is ruled by a set of second order difference equations with both multiplicative and additive stochasticity [2]. In [14], Poli et al. proposed SPSO, a simpler PSO involving only additive stochasticity, and studied it using a moment analysis technique firstly introduced in [15].

GECCO'08, July 12–16, 2008, Atlanta, Georgia, USA.

Poli et al.'s SPSO is not the only particle swarm with additive stochasticity existing in the literature. In an effort to obtain simpler algorithms, several researchers have proposed PSOs that, intentionally or not, also remove multiplicative randomness from their update rules. Examples of these algorithms are Kennedy's Bare-bones PSO [7], GDPS [8] and Essential FIPS [9], as well as Peña et al.'s PSO-DR [12] (further studied by Bratton and Blackwell [3, 4]).

In this paper, a general formulation for particle swarms with additive stochasticity that encompasses all of these algorithms is proposed (Section 2). Single instances that correspond to actual particle swarms can be derived by defining appropriate *recombination operators* (Section 3). In Section 4 this family of algorithms is formally studied, extending the main results presented in [14] for SPSO. In particular, stability conditions and the dynamic equations describing the first two moments of the sampling distribution during stagnation are derived. Then, the optimisation efficiency of several members of this family of methods is compared in Section 5. Finally, conclusions are drawn (Section 6).

2. PARTICLE SWARMS WITH ADDITIVE STOCHASTICITY

In Standard PSO, position updates for each dimension are given by

$$x_{t+1} = x_t + w\,(x_t - x_{t-1}) + \sum_{k=1}^{K} \frac{\phi}{K} u_k\,(p_k - x_t)\,, \quad (1)$$

where p_k is the personal best of the k-th *informer*, K is the number of informers, $u_k \sim U\,[0,1]$ is a random variable uniformly distributed in the interval $[0,1]$, w is the inertia weight, and ϕ is the acceleration coefficient. In the best-of-neighbourhood (BN) model of influence, the set of informers is constituted by the particle itself and the neighbourhood best. In the fully informed (FI) model [10], all neighbours are considered as informers[1]. The specification of a population topology, of a model of influence and of the values for the parameters w and ϕ defines an instance of Standard PSO.

Notice that Eq. 1 can be rewritten as:

[1]We will also use the acronyms BNPS (for BN particle swarms) and FIPS (for FI particle swarms). Standard BNPS (resp. Standard FIPS) refers to Standard PSO with a BN (resp. FI) model of influence.

$$x_{t+1} + a_t x_t + b x_{t-1} = c_t, \tag{2}$$

with

$$\begin{cases} a_t &= \sum_k \frac{\phi}{K} u_k - w - 1 \\ b &= w \\ c_t &= \sum_k \frac{\phi}{K} u_k p_k \end{cases}. \tag{3}$$

Since a_t and c_t are random variables, Standard PSO is ruled by second order difference equations with both additive and multiplicative stochasticity [2].

Consider now the update rule for Standard BNPS

$$x_{t+1} = x_t + w\left(x_t - x_{t-1}\right) + \sum_{k=1}^{2} \frac{\phi}{2} u_k \left(p_k - x_t\right), \tag{4}$$

and rewrite it as:

$$x_{t+1} = x_t + w\left(x_t - x_{t-1}\right) + \frac{\phi}{2}\left(u_1 + u_2\right)\left(q - x_t\right), \tag{5}$$

with

$$q = \frac{u_1 p_1 + u_2 p_2}{u_1 + u_2}. \tag{6}$$

Looking for a PSO easier to analyse while still retaining Standard PSO's key properties, the authors of [14] argued for the replacement of the random variable $u_1 + u_2$ by its mean value $E\left[u_1 + u_2\right] = 1$ and of Eq. 6 by

$$q = u p_1 + \left(1 - u\right) p_2, \tag{7}$$

with $u \sim U\left[0, 1\right]$.

These changes led to SPSO's update rule:

$$x_{t+1} = x_t + w\left(x_t - x_{t-1}\right) + \frac{\phi}{2}\left(u p_1 + \left(1 - u\right) p_2 - x_t\right). \tag{8}$$

In this paper, we study the more general model:

$$x_{t+1} = x_t + w\left(x_t - x_{t-1}\right) + \alpha\left(q - x_t\right), \tag{9}$$

where α needs not to be related to the original parameter ϕ, and q can be derived from any imaginable *recombination operator* (see Section 3). Furthermore, informers can be those specified by either a BN or a FI model of influence with $K \geq 2$.

Remark that Eq. 9 can still be rewritten as Eq. 2, but the coefficients now take the values

$$\begin{cases} a &= \alpha - w - 1 \\ b &= w \\ c_t &= \alpha q \end{cases}. \tag{10}$$

Since c_t is the only coefficient involving a random variable, only additive stochasticity is present in Eq. 9. Hereafter we refer to a PSO with such an update rule as a Particle Swarm with Additive Stochasticity (PSAS).

3. RECOMBINATION OPERATORS

Recombination is a variation operator usually used in Evolution Strategies (ES) and other evolutionary computation techniques in order to combine already tested good solutions [1]. In the context of particle swarms, we define a recombination operator as a mapping between the set of informers' personal bests and the stochastic variable q (see Eq. 9). Below we define 5 types of recombination operators for particle swarms: *standard*, *rectangular*, *discrete-1*, *discrete-2* and *gaussian*. For the sake of clarity, we limit ourselves to the case $K = 2$, giving $k = 1$ to the best of the two informers.

The *standard* (S) and *rectangular* (R) recombination operators have been already presented in the last section and are respectively given by Eq. 6 and Eq. 7. The *discrete-1* (D1) recombination operator[2] is given by

$$q = \eta_d p_1 + \left(1 - \eta_d\right) p_2, \tag{11}$$

and the *discrete-2* (D2) recombination operator by

$$q = \frac{\left(1 + \eta_d\right) p_1 + \left(1 - \eta_d\right) p_2}{2}, \tag{12}$$

where $\eta_d \sim U\left\{0, 1\right\}$ is a discrete random number drawn from a Bernoulli distribution with success probability 1/2.

Finally, we define the *gaussian* (G) recombination operator as

$$q = N\left(\frac{p_1 + p_2}{2}, \gamma\left|p_1 - p_2\right|\right), \tag{13}$$

where $\gamma > 0$ is a real constant, and $N\left(\mu, \sigma\right)$ denotes a random variable from a normal distribution with mean μ and standard deviation σ.

Actual PSASs are recovered by coupling the general formula of Eq. 9 with the definition of q via a recombination operator. Hereafter, we refer to a PSAS with a recombination operator X as PSAS-X. Table 1 show the original formulations of several existing PSOs and their equivalent formulations as recombinant PSASs. Little algebra is required to pass from one formulation to the other.

Two comments should be made here about the *original formulations* presented in Table 1. Firstly, even though some of the listed methods were originally designed with a particular model of influence in mind (e.g. FI in Essential FIPS, BN in SPSO), all of them have been generalised so that p_1 and p_2 are the personal bests of any set of two informers. Secondly, the formulations have been slightly modified so that only the personal bests of these two informers are taken into account. The introduced modifications are most obvious in the cases of GDPS and Essential FIPS, for which the $frange$ and $rangep$ variables do not involve differences with respect to the self's p vector (as proposed in the original papers), but only the difference between the personal bests of the two informers. Notice also that Bare-bones, GDPS and Essential FIPS are essentially the same algorithm (PSAS-G) with different values for w, α and γ. In the case of the different PSO-DR formulations, PSO-DR and PSO-DR M2 correspond to a velocity-based and a velocity-free PSAS-D2, whereas PSO-DR M3 is a velocity-free PSAS-D1.

The expected value and the variance of q ($E\left[q\right]$ and $Var\left[q\right]$), as well as the expected value of q^2 ($E\left[q^2\right]$) will be found in the equations presented in the next section. $E\left[q\right] = \frac{p_1 + p_2}{2}$ for S, R, D1 and G, and $E\left[q\right] = \frac{3p_1 + p_2}{4}$ for D2. $Var\left[q\right]$

[2]Compare to the *dominant* or *discrete* recombination used in ES [1].

Table 1: Existing PSOs and their equivalent formulations as PSASs with recombination. Parameter values are those suggested by the cited original papers.

Algorithm	Original formulation	Equivalent formulation
Bare-bones [7]	$x_{t+1} = N\left(\frac{p_1+p_2}{2}, \lvert p_1 - p_2\rvert\right)$	PSAS-G $w = 0,\ \alpha = 1,\ \gamma = 1$
GDPS [8]	$frange = \lvert p_1 - p_2\rvert,$ $x_{t+1} = x_t + w_1\left(x_t - x_{t-1}\right) + w_2\left(\frac{p_1+p_2}{2} - x_t\right)$ $+N(0,1)\frac{frange}{2},$ $w_1 = 0.7298,\ w_2 = 2.187$	PSAS-G $w = w_1 = 0.7298,\ \alpha = w_2 = 2.187,$ $\gamma = \frac{1}{2w_2} = 0.2286$
Essential FIPS [9]	$rangep = \lvert p_1 - p_2\rvert$ $x_{t+1} = x_t + \chi\left(x_t - x_{t-1}\right) + \frac{\phi}{2}\frac{(p_1-x_t)+(p_2-x_t)}{2}$ $+\frac{1}{2}N(0,1)\,rangep$ $\chi = 0.7298,\ \phi = 4.1$	PSAS-G $w = \chi = 0.7298,\ \alpha = \frac{\phi}{2} = 2.05,$ $\gamma = \frac{1}{\phi} = 0.2439$
PSO-DR [12, 3, 4]	$r = \eta_d p_1 + (1-\eta_d)\,p_2$ $x_{t+1} = x_t + w\left(x_t - x_{t-1}\right) + \frac{\phi_1}{2}(r - x_t) + \frac{\phi_2}{2}(p_1 - x_t)$ $w = 0.5,\ \phi = 2.0,\ \eta_d \sim U\{0,1\}$	PSAS-D2 $w = 0.5,\ \alpha = 2.0$
PSO-DR M2 [4]	$r = \eta_d p_1 + (1-\eta_d)\,p_2$ $x_{t+1} = x_t + \frac{\phi}{2}(r - x_t) + \frac{\phi}{2}(p_1 - x_t)$ $\phi = 1.6,\ \eta_d \sim U\{0,1\}$	PSAS-D2 $w = 0.0,\ \alpha = \phi = 1.6$
PSO-DR M3 [4]	$r = \eta_d p_1 + (1-\eta_d)\,p_2$ $x_{t+1} = x_t + \phi\,(r - x_t)$ $\phi = 1.2,\ \eta_d \sim U\{0,1\}$	PSAS-D1 $w = 0.0,\ \alpha = \phi = 1.2$
SPSO [14]	$x_{t+1} = x_t + \chi\left(x_t - x_{t-1} + \frac{\phi}{2}\left((p_1 - p_2)\,u + p_2 - x_t\right)\right)$ $\chi = 0.7298,\ \phi = 4.1,\ u \sim U[0,1]$	PSAS-R $w = \chi = 0.7298,\ \alpha = \frac{\chi\phi}{2} = 1.4961$

Table 2: Statistical characterisation of different recombination operators

	$E\left[q^2\right]$	$\frac{Var[q]}{(p_1-p_2)^2}$
S	$(\log 2 - 1)\left(p_1^2 + p_2^2\right) + (2\log 2 - 1)\,p_1 p_2$	$\frac{3}{4} - \log 2$
R	$\frac{1}{3}\left(p_1^2 + p_1 p_2 + p_2^2\right)$	$\frac{1}{12}$
D1	$\frac{1}{2}p_1^2 + \frac{1}{2}p_2^2$	$\frac{1}{4}$
D2	$\frac{5}{8}p_1^2 + \frac{1}{4}p_1 p_2 + \frac{1}{8}p_2^2$	$\frac{1}{16}$
G	$\left(\gamma^2 + \frac{1}{4}\right)\left(p_1^2 + p_2^2\right) + \left(\frac{1}{2} - 2\gamma^2\right)p_1 p_2$	γ^2

is proportional to $(p_1 - p_2)^2$ for all the recombination operators. The values of $E\left[q^2\right]$ and $\frac{Var[q]}{(p_1-p_2)^2}$ are reported in Table 2 for each of the introduced recombination operators.

4. ANALYSIS

In order to analyse the search behaviour of PSASs, we will make use of a technique originally introduced by Poli and Broomhead in [15] to study the sampling distribution of a PSO during stagnation. This technique has already been used by the authors of [14] to analyse SPSO. Instead of defining actual PSASs (by specifying a recombination operator beforehand) and then deriving the results for each particular case, we will give all the results in terms of the stochastic variable q. Finally, although the analysis can be extended to higher moments [13, 14], we limit ourselves to the mean and the variance of the sampling distribution.

Let us start by collecting terms and rewrite Eq. 9 as

$$x_{t+1} = \theta x_t - w x_{t-1} + \alpha q, \tag{14}$$

where $\theta = 1 + w - \alpha$. Applying expectations to both sides of Eq. 14, we obtain

$$E\left[x_{t+1}\right] = \theta E\left[x_t\right] - w E\left[x_{t-1}\right] + \alpha E\left[q\right]. \tag{15}$$

This recursion is the same as the one derived for Standard PSO with the variable substitutions $E\left[q\right] = \frac{p_1+p_2}{2}$ and $\alpha = \frac{\phi}{2}$ (see [15]). Thus, PSASs with $E\left[q\right] = \frac{p_1+p_2}{2}$ and $\alpha = \frac{\phi}{2}$ preserve the behaviour of the mean of the sampling distribution of Standard PSO. Eq. 15 can be easily solved analytically, giving

$$\begin{aligned} E\left[x_t\right] \;=\; & c_1\left(\theta - \sqrt{\theta^2 - 4w}\right)^t 2^{-t} + \\ & c_2\left(\theta + \sqrt{\theta^2 - 4w}\right)^t 2^{-t} + E\left[q\right], \end{aligned} \tag{16}$$

where c_1 and c_2 depend on initial conditions.

Now, multiplying both sides of Eq. 14 by x_t and taking their expected values:

$$E\left[x_{t+1}x_t\right] = \theta E\left[x_t^2\right] - w E\left[x_{t-1}x_t\right] + \alpha E\left[q\right] E\left[x_t\right], \tag{17}$$

where we used $E\left[qx_t\right] = E\left[q\right] E\left[x_t\right]$ because of the statistical independence of q and x_t during stagnation. Taking the square of Eq. 14, expanding the resulting expression and applying the expectation operator to both sides:

$$\begin{aligned} E\left[x_{t+1}^2\right] \;=\; & \theta^2 E\left[x_t^2\right] + 2w\theta E\left[x_{t-1}x_t\right] + \\ & w^2 E\left[x_{t-1}^2\right] + 2\alpha\theta E\left[q\right] E\left[x_t\right] - \\ & 2\alpha w E\left[q\right] E\left[x_{t-1}\right] + \alpha^2 E\left[q^2\right]. \end{aligned} \tag{18}$$

Solving analytically the recursions for $E\left[x_t x_{t-1}\right]$ and $E\left[x_t^2\right]$ is less straightforward than for $E\left[x_t\right]$. They can however be integrated numerically after specifying initial conditions. Finally, a recursion for the variance of x_t can be constructed from Eq. 18 and 15 using $Var\left[x_t\right] = E\left[x_t^2\right] - \left(E\left[x_t\right]\right)^2$.

4.1 Stability analysis

In order to analyse the stability of the system, we again follow Poli and Broomhead [15] and write Eq. 15, 17 and 18 as the following system of first order equations

$$\mathbf{z}_{t+1} = M\mathbf{z}_t + \mathbf{b}, \tag{19}$$

where

$$\mathbf{z}_t = \left(E\left[x_t\right], E\left[x_{t-1}\right], E\left[x_t^2\right], E\left[x_{t-1}^2\right], E\left[x_t x_{t-1}\right]\right)^T. \tag{20}$$

The state update matrix M is given by

$$M = \begin{pmatrix} \theta & -w & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 2\alpha\theta E[q] & -2\alpha w E[q] & \theta^2 & w^2 & -2w\theta \\ 0 & 0 & 1 & 0 & 0 \\ \alpha E[q] & 0 & \theta & 0 & -w \end{pmatrix}, \quad (21)$$

and the forcing vector by

$$\mathbf{b} = \left(\alpha E[q], 0, \alpha^2 E\left[q^2\right], 0, 0\right)^T. \quad (22)$$

In order for $E[x_t]$, $E\left[x_t^2\right]$ and $E[x_t x_{t-1}]$ to converge to stable fixed points, the magnitudes of all the eigenvalues of M must be less than one. In this case, the first two moments of the sampling distribution converge and the system is said to be *order-2 stable* [13]. If only the first moment converges, the system is called *order-1 stable* [13].

The eigenvalues of M can be shown to be given by

$$\begin{cases} \lambda_{1,2} = \frac{\theta \pm \sqrt{(w-\alpha)^2 - 2\alpha - 2w + 1}}{2} \\ \lambda_3 = w \\ \lambda_{4,5} = \frac{1 - 2\alpha + (\phi - w)^2 \pm \theta\sqrt{1 - 2(\alpha + w) + (\alpha - w)^2}}{2} \end{cases}. \quad (23)$$

Since none of the eigenvalues depend on q, *stability is independent of the choice of the recombination operator*, as long as $E[q]$ and $E\left[q^2\right]$ take finite values. An empirical analysis of these eigenvalues shows that the region of order-2 stability is given by

$$\begin{cases} |w| < 1 \\ 0 < \alpha < 2(1 + w) \end{cases}. \quad (24)$$

It is noteworthy that the region of order-2 stability for PSASs coincides with their region of order-1 stability, which is not the case for Standard PSO [15]. With the appropriate variable substitutions, the region of convergence given by Eq. 24 can be shown to be equivalent to the ones derived by Poli et al. for SPSO [14] and by Blackwell and Bratton for PSO-DR [3]. Fig. 1 compares the regions of order-2-stability for Standard PSO and for PSASs. Notice that, for the mean preserving case $\alpha = \frac{\phi}{2}$, the region of convergence of PSASs goes beyond the corresponding to Standard PSO.

4.2 Fixed points of $E[x_t]$ and $Var[x_t]$

Following the algebraic procedure used in [15], it can be shown that the fixed point $E[x]^*$ for $E[x_t]$ is given by

$$E[x]^* = E[q], \quad (25)$$

whereas the fixed points $E\left[x^2\right]^*$ and $E[xx]^*$ for $E\left[x_t^2\right]$ and $E[x_t x_{t-1}]$ are respectively given by

$$E\left[x^2\right]^* = \frac{2\left(1 - \alpha - w^2\right)E[q]^2 + (1 + w)\alpha E\left[q^2\right]}{(w-1)[\alpha - 2(1 + w)]} \quad (26)$$

and

$$E[xx]^* = \frac{\left(2 - 2\alpha - \alpha^2 - w^2\right)E[q]^2 + \alpha\theta E\left[q^2\right]}{(w-1)[\alpha - 2(1 + w)]}. \quad (27)$$

The fixed point for the variance of the sampling distribution can be calculated using $Var[x]^* = E\left[x^2\right]^* - (E[x]^*)^2$. After little algebra:

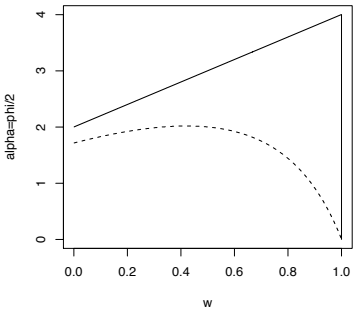

Figure 1: Curves limiting the regions of order-2-stability for Standard PSO (dotted line) and PSASs (solid line).

$$Var[x]^* = \frac{\alpha(1 + w)}{(w-1)[\alpha - 2(1 + w)]} Var[q]. \quad (28)$$

As said before, order-2 stability does not depend on the specific recombination operator used. Nevertheless, for a stable system, *the fixed points of the first two moments of the sampling distribution are determined by the choice of the recombination operator*. Notice also that $Var[q]$ is proportional to $(p_1 - p_2)^2$ for all of the presented recombination operators (see the last column of Table 2), which allows us to rewrite Eq. 28 as

$$Var[x]^* = \frac{\alpha(1 + w)}{(w-1)[\alpha - 2(1 + w)]} \delta_q (p_1 - p_2)^2 \quad (29)$$

where $\delta_q = \frac{Var[q]}{(p_1 - p_2)^2}$ is a constant depending on the recombination operator. Thus, the fixed point of the variance for PSASs is proportional to $(p_1 - p_2)^2$, as it was already the case for Standard PSO (see [15]).

Figure 2 shows a comparison of the multiplicative factors accompanying $(p_1 - p_2)^2$ in the expression for $Var[x]^*$ for Standard PSO and PSAS-S ($\delta_q \approx 0.0569$), PSAS-R ($\delta_q \approx 0.0833$) and PSAS-D1 ($\delta_q = 0.25$). Observe that, when fixing w, larger values of ϕ or α lead to larger variances. Notice also how larger values of δ_q move the lines of constant $Var[x]^*$ to the bottom, meaning that PSASs with larger δ_q will have larger variances for any given w and α.

4.3 Transient analysis of $E[x_t]$

The transient behaviour of the first moment of the sampling distribution has been suggested to be important for the performance of PSOs, since oscillations of the mean allow an amount of extrapolation that could be beneficial in some functions [14]. We studied empirically the transient behaviour of the mean of the sampling distribution generated by Standard PSO (resp. PSASs) by measuring the percentage overshoot and settling time of their step responses for different values of w and ϕ (resp. α). We define the step input as a sudden change in any of the informers' personal bests at $t = 0$ after an infinitely long period of stagnation. Without loss of generality, we assumed that the system had reached its fixed points corresponding to $p_1 = p_2 = 0$ (thus, $E[x_0] = 0$) and integrated numerically the equations for the

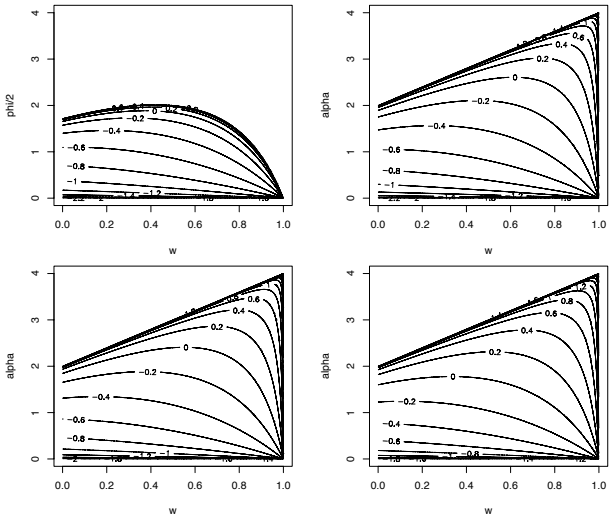

Figure 2: Contour plots of $\log_{10} \frac{Var[x]^*}{(p_1-p_2)^2}$ for Standard PSO (*top left*) and PSAS-S (*top right*), PSAS-R (*bottom left*) and PSAS-D1 (*bottom right*). Eq. 26 of [14] was used to generate the contour plots for Standard PSO.

new pair of values $p_2 = 0$ and $p_1 = 1$. This represents a situation in which the best informer changed its p from 0 to 1, but the worst informer kept its old value.

PO, the percentage overshoot of $E[x_t]$, can be calculated with

$$PO := \frac{\max\left(E[x_t] - E[x]^*\right)}{E[x]^*} \times 100\%. \tag{30}$$

and the settling time of $E[x_t]$ with

$$ST := \min\left\{t_s : |E[x_t] - E[x]^*| < \epsilon |E[x]^* \mid \forall t > t_s\right\}, \tag{31}$$

where $0 < \epsilon < 1$ gives a band of error.

Notice that both PO and ST depend on $E[x_t] - E[x]^*$. Subtracting Eq. 25 from Eq. 16 gives

$$\begin{aligned} E[x_t] - E[x]^* &= c_1 \left(\theta - \sqrt{\theta^2 - 4w}\right)^t 2^{-t} + \\ &\quad c_2 \left(\theta + \sqrt{\theta^2 - 4w}\right)^t 2^{-t} \end{aligned}, \tag{32}$$

which is independent of q. Thus, *both PO and ST are independent of the choice of the recombination operator.*

Figure 3 shows contour plots of PO and $\log_{10}(ST)$ for both Standard PSO and PSASs. Observe that the larger region of order-2-stability of PSAS allows stable systems with larger overshoots of the mean. Remark how, in general, larger values of ϕ or α correspond to larger values of PO and larger values of w correspond to larger values of ST.

4.4 Sampling behaviour of particular PSASs

In the light of what has been presented in this section, neither order-2 stability nor the transient behaviour of $E[x_t]$ depend on q. The use of a particular recombination operator, however, determines the fixed points of the first two moments of the sampling distribution. It is then possible

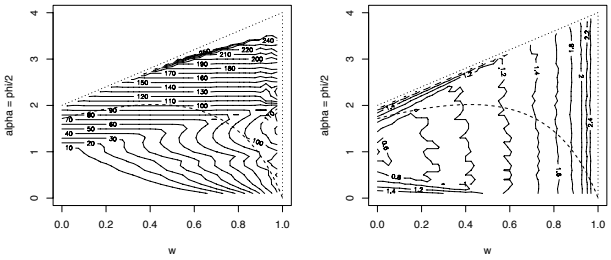

Figure 3: Contour plots of PO (*left*) and $\log_{10}(ST)$ (*right*). For ST, a band error $\epsilon = 0.02$ was used. Limits of the region of order-2 stability for Standard PSO (*dashed line*) and PSASs (*dotted line*) are also shown.

Table 3: Characteristics of the sampling distribution for different PSOs

Algorithm	$\frac{Var[x]^*}{(p_1-p_2)^2}$	PO	ST
Standard PSO [6, 5]	1.0868	84.57	26
PSAS-S	0.2773	84.57	26
SPSO [14]	0.4064	84.57	26
PSAS-D1-1	0.3750	50.00	10
PSO-DR [12, 3, 4]	0.3750	100.00	11
PSO-DR M2 [4]	0.2500	60.00	7
PSO-DR M3 [4]	0.3750	20.00	2
Bare-bones [7]	1.0000	0.00	0
GDPS [8]	0.5749	118.70	25
Essential FIPS [9]	0.5538	105.00	23

to design the transient behaviour of the mean by choosing appropriate values for w and ϕ and then design the fixed points of the mean and the variance via a given recombination operator.

Table 3 characterises the search properties of different PSOs in terms of $Var[x]^*$, PO and ST. For the already introduced PSASs, parameters are those listed in Table 1. Standard PSO uses $w = 0.7289$ and $\phi = 2.9924$ (equivalent to the standard constricted PSO with $\chi = 0.7298$ and $\varphi = 4.1$ [6, 5]). PSAS-D1-1 is a PSAS-D1, as PSO-DR M3, but is not velocity-free, having $w = 0.5$ and $\alpha = 1.0$. PSAS-S has the same w and α as SPSO. Since $\alpha = \frac{\phi}{2}$, the behaviour of the mean of the sampling distribution is the same for Standard PSO, PSAS-S and SPSO. The fixed point of the variance for PSAS-S is however smaller than that for SPSO because of its smaller δ_q (see Fig. 2).

As it can be seen from Table 3, search strategies of the listed PSOs are very diverse. Standard PSO is characterised by large amounts of exploration (large $Var[x]^*$) and extrapolation (large PO and ST). PSAS-S and SPSO maintain the same amounts of extrapolation as Standard PSO but are more exploitation-biased. Bare-bones has good exploration but no extrapolation at all, whereas the other two algorithms with gaussian recombination have important amounts of exploration and the largest amounts of extrapolation in terms of PO. Finally, notice that the velocity-free algorithms (PSO-DR M3 and Bare-bones) have the smallest PO.

5. EXPERIMENTAL RESULTS

In the previous sections we have analysed the most salient properties of the sampling distributions of PSASs with different recombination operators and have compared them with those of Standard PSO. In this section, we study empirically the optimisation efficiency of the PSOs listed in Table 3.

All of the PSAS listed in Table 3 have already been the subject of empirical studies [7, 8, 9, 12, 14, 3, 4], except for PSAS-S and PSAS-D1-1 which have been introduced in this paper. The subject of these investigations has been mainly the comparison of these methods to Standard PSO. Experiments, however, have been performed under very variable conditions. For instance, the author of [8] used one population topology for GDPS (a randomly generated network called "gr.2ed2") and a different one for Standard PSO ("Square"). In [12, 3, 4] local PSO-DR algorithms with ring topologies (which are basically biased FIPS) were compared to Standard BNPS, but not to Standard FIPS. These setups hinder conclusions made about the efficiency of the proposed update rules, since differences in the results may be due to the different topologies or models of influence. Additionally, a number of other parameters such as the number of particles, the termination conditions, and the used benchmark functions vary across different papers, making the comparison of one PSAS to another rather difficult.

We have tried to avoid these shortcomings by running all the studied algorithms under the same conditions. Our experimental setup follows closely the one proposed by Bratton and Kennedy in [5] as a standard for comparing different PSOs. All tested particle swarms used populations of 50 particles connected according to a ring topology. BN and FI versions were coded for each algorithm. In both cases neighbourhoods are open (i.e. the self is not considered to calculate the neighbourhood best in BNPS or as an informer in FIPS).

Algorithms were run on the suite of benchmark functions shown in Table 4. Function f_7 was optimised both in 10 and in 30 dimensions; we denote these instances as $f_7^{(10)}$ and $f_7^{(30)}$. Functions belong to three distinguishable groups: $f_1 - f_3$ are unimodal, $f_4 - f_9$ are complex high-dimensional with many local minima, and $f_{10} - f_{15}$ are low-dimensional ($f_{10} - f_{14}$ have few local minima and f_{15} has many local minima).

In order to remove any centrist bias, both the *region scaling* and the *centre offset* techniques [11] were used for all functions, except for f_4 where only region scaling was applied. The centre offset technique was implemented by shifting the function by a vector of uniform random values in $U[-0.25l, 0.25l]$ for each run, being l the size of the search space in each dimension. Particles flying out of the feasible bounds were not evaluated. Table 5 shows the dimensionality, feasible bounds, location of the optimum and initialization ranges for each function. Finally, the error $|f(x) - f(x^*)|$ found after 300000 function evaluations was used as the measure for algorithm performance, where $f(x^*)$ is the value of the objective function at the global minimum. Values less than 10^{-8} were rounded to 0.0. For the statistical analyses we used t-tests corrected with a modified Bonferroni procedure (see [5]).

With few exceptions, FIPSs performed better than BNPSs in the multimodal functions, while BNPSs did better in the unimodal functions. Given that one of the main reasons for using stochastic population-based optimisation techniques such as PSO is to cope with multimodal problems, we decided to focus our analysis on the FI variants of the mentioned methods. Only the results of the BN variant of Standard PSO are reported for comparison reasons. Differences in results for GDPS and Essential FIPS were statistically insignificant in all functions, except for f_2 and $f_7^{(10)}$ where Essential FIPS was better. As both these algorithms are basically the same PSO (a velocity-based PSAS-G) we only report results of Essential FIPS.

Table 6 displays the 95% confidence intervals of the mean performance over 30 runs for all functions. Given the large number of function evaluations, convergence plots for some of the benchmark functions are also shown in Fig. 4. All methods were able to effectively minimise the easy f_1 and f_{10} functions, and almost all of them were able to achieve mean objective function values below 10^{-8} for $f_7^{(30)}$, f_8, f_9 and f_{11}. Exceptions were PSAS-S for f_8, PSO-DR M2 for f_8 and f_9, and both PSO-DR M3 and Bare-bones for $f_7^{(30)}$, f_9 and f_{11}. These methods seem to be hindered by their low exploration

Table 4: Benchmark Functions

$f_1 = \sum_{i=1}^{D} x_i^2$

$f_2 = \sum_{i=1}^{D} \left(\sum_{j=1}^{i} x_j\right)^2$

$f_3 = \sum_{i=1}^{D-1} \left\{100\left(x_{i+1} - x_i^2\right)^2 + (x_i - 1)^2\right\}$

$f_4 = -\sum_{i=1}^{D} \sin\left(\sqrt{x_i}\right)$

$f_5 = \sum_{i=1}^{D} \left\{x_i^2 - 10\cos(2\pi x_i) + 10\right\}$

$f_6 = -20\exp\left\{-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D} x_i^2}\right\} - \exp\left\{\frac{1}{D}\sum_{i=1}^{D}\cos(2\pi x_i)\right\} + 20 + e$

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

$f_8 = \frac{\pi}{D}\left\{10\sin^2(\pi y_i) + \sum_{i=1}^{D-1}(y_i - 1)^2\left\{1 + 10\sin^2\left(\pi y_{i+1}\right)\right\} + (y_D - 1)^2\right\} + \sum_{i=1}^{D}\mu(x_i, 10, 100, 4)$

$f_9 = 0.1\{\sin^2(3\pi x_i) + \sum_{i=1}^{D-1}(x_i - 1)^2\left\{1 + \sin^2\left(3\pi x_{i+1}\right)\right\} + (x_D - 1)^2 \times \left\{1 + \sin^2(2\pi x_D)\right\}\} + \sum_{i=1}^{D}\mu(x_i, 5, 100, 4)$

$f_{10} = 4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1x_2 - 4x_2^2 + 4x_2^4$

$f_{11} = \left\{1 + (x_1x_2 + 1)^2\left(19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2\right)\right\} \times \left\{30 + (2x_1 - 3x_2)^2\left(18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2\right)\right\}$

$f_{12} = -\sum_{i=1}^{5}\left\{\sum_{j=1}^{4}\left(x_j - a_{ij}\right)^2 + c_i\right\}^{-1}$

$f_{13} = -\sum_{i=1}^{7}\left\{\sum_{j=1}^{4}\left(x_j - a_{ij}\right)^2 + c_i\right\}^{-1}$

$f_{14} = -\sum_{i=1}^{1} 0\left\{\sum_{j=1}^{4}\left(x_j - a_{ij}\right)^2 + c_i\right\}^{-1}$

$f_{15} = 0.5 + \frac{\sin^2\sqrt{x_1^2 + x_2^2} - 0.5}{\left(1 + 0.001\left(x_1^2 + x_2^2\right)\right)^2}$

Table 5: Dimensionality, Feasible Bounds, Optima and Initialization Ranges for the Benchmark Functions

	D	Feasible Bounds	Optimum	Initialization
f_1	30	$(-100, 100)^D$	0.0^D	$(50, 100)^D$
f_2	30	$(-100, 100)^D$	0.0^D	$(50, 100)^D$
f_3	30	$(-30, 30)^D$	1.0^D	$(15, 30)^D$
f_4	30	$(-500, 500)^D$	420.9687^D	$(-500, -250)^D$
f_5	30	$(-5.12, 5.12)^D$	0.0^D	$(2.56, 5.12)^D$
f_6	30	$(-32, 32)^D$	0.0^D	$(16, 32)^D$
f_7	10, 30	$(-600, 600)^D$	0.0^D	$(300, 600)^D$
f_8	30	$(-50, 50)^D$	-1.0^D	$(25, 50)^D$
f_9	30	$(-50, 50)^D$	1.0^D	$(25, 50)^D$
f_{10}	2	$(-5, 5)^D$ $(0.0898, -0.7126)$	$(-0.0898, 0.7126)$,	$(2.5, 5)^D$
f_{11}	2	$(-2, 2)^D$	$(0, -1)$	$(1, 2)^D$
f_{12}	4	$(0, 10)^D$	4.0^D	$(7.5, 10)^D$
f_{13}	4	$(0, 10)^D$	4.0^D	$(7.5, 10)^D$
f_{14}	4	$(0, 10)^D$	4.0^D	$(7.5, 10)^D$
f_{15}	2	$(-100, 100)^D$	0.0^D	$(50, 100)^D$

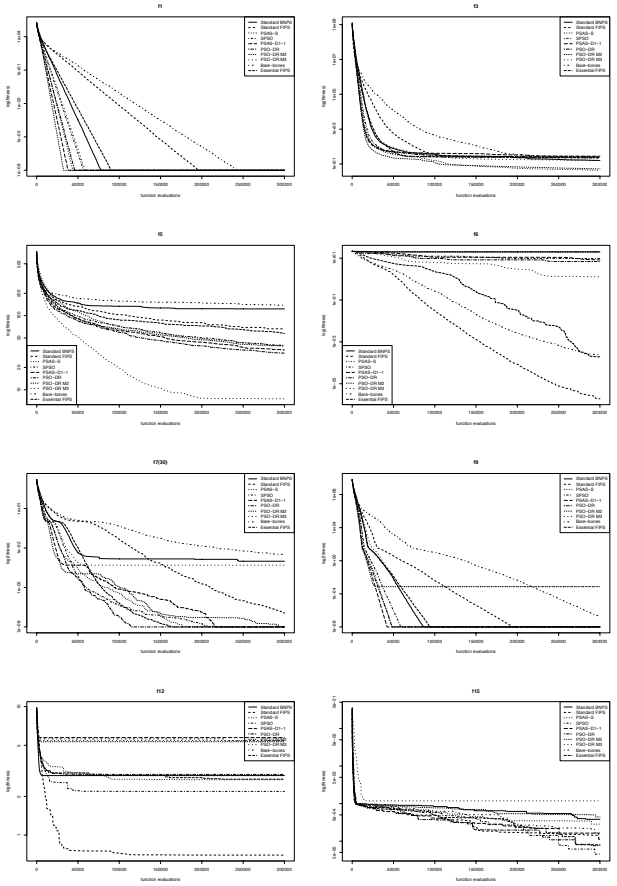

Figure 4: Convergence plots for selected functions

and/or extrapolation (see Table 3). In particular, $f_7^{(30)}$ and f_{11} seem to be better optimised by methods exhibiting important amounts of extrapolation, whereas methods with very low variances perform the worst in f_8. This said, notice that the best methods for the subset of unimodal functions were precisely those with the lowest variances (PSAS-S and PSO-DR M2). f_6 seemed to favour methods with larger variances irrespective of their amounts of extrapolation (Standard FIPS, Bare-bones and Essential FIPS being the best), while f_4 is better minimised by methods with both large exploration and extrapolation. In f_5 and $f_7^{(10)}$, PSOs with moderate extrapolation and exploration got the best scores. As a whole, low-dimensional multimodal functions with few minima favoured large variances and percentage overshoots of the mean.

In general, at least one of the PSASs achieved equal or better performance than Standard FIPS for all the unimodal functions, all of the high-dimensional multimodal functions except for f_4 and f_6, and all of the low-dimensional functions except for f_{12} and f_{13}. Standard FIPS, however, was the most consistent algorithm across different functions, achieving the best results for 9 of the 13 multimodal functions.

6. CONCLUSIONS

In this paper we have proposed a general formulation for particle swarms with additive stochasticity from which actual PSOs can be derived by specifying particular recombination operators. It was shown how several PSOs previously introduced in the literature (such as Bare-bones, GDPS, Essential FIPS, PSO-DR and SPSO) can be thought as particular instances of this general class of algorithms. Following the moment analysis technique by Poli and Broomhead [15], we studied the stability and behaviour of the first two moments of the sampling distribution of these PSOs during stagnation and compared them to those of Standard PSO. By treating algorithms as particular cases of the same model, we were able to extend to all PSASs some of the results derived for SPSO in [14] and to study the effect of different recombination operators. It was found that the choice of a particular recombination operator does not affect the conditions for order-2-stability nor the transient behaviour of the mean of the sampling distribution during stagnation, but that it defines the fixed points of both the mean and the variance. We tested several instances of PSASs and compared them to Standard PSO on a suite of 16 benchmark problems, showing how the sampling characteristics of the algorithms matched the specificities of different functions.

We focused on five types of recombination operators (standard, rectangular, discrete-1, discrete-2 and gaussian), and gave definitions of them for $K = 2$. However, most of the presented results are independent of the particularities of the recombination operator being used. Thus, results should still be valid for generalisations of the presented operators to an arbitrary number of informers, and also to other recombination operators.

From the point of view of the performance, and in the light of the presented empirical results no recombination operator seems to be superior to another. Recombination operators interact with parameters w and ϕ for determining a PSO with specific sampling distribution characteristics that could be beneficial for some functions, but detrimental for others. Having the possibility of specifying different recombination operators, however, helps the design of these algorithms by introducing an additional parameter that allows for fixing independently the required amounts of extrapolation and exploration.

7. ACKNOWLEDGMENT

This research was supported by the Future and Emerging Technologies programme IST-STREP of the European Community, under grant IST-034632 (PERPLEXUS). The author would like to thank Isis Fuchs, Andrés Pérez-Uribe, Henri Volken, the members of IMA-UNIL and two anonymous reviewers for their comments on this paper.

8. REFERENCES

[1] H.-G. Beyer and H.-P. Schwefel. Evolution strategies: A comprehensive introduction. *Natural Computing*, 1(1):3–52, Mar. 2002.

[2] T. Blackwell and D. Bratton. Origin of bursts. In *GECCO '07: Proceedings of the 2007 GECCO Conference Companion on Genetic and Evolutionary Computation*, pages 2613–2620, New York, NY, USA, 2007. ACM.

[3] D. Bratton and T. Blackwell. Understanding particle swarms through simplification: a study of recombinant pso. In *GECCO '07: Proceedings of the 2007 GECCO Conference Companion on Genetic and Evolutionary*

Table 6: 95% confidence intervals of the mean performance after 300000 function evaluations over 30 runs. The symbol §(resp. ‡) means that the algorithm performed significantly better (resp. worse) than Standard FIPS for the same function.

Algorithm	f_1	f_2	f_3	f_4	f_5	f_6	$f_7^{(10)}$	$f_7^{(30)}$
Standard BNPS	**0** (±0)	0.3479§ (±0.06333)	15.96 (±8.894)	3.272e+03‡ (±54.48)	1.232e+02‡ (±8.943)	19.46‡ (±0.2917)	0.0213‡ (±5.38e-03)	1e-03 (±9.5e-04)
Standard FIPS	**0** (±0)	1.153e+02 (±14.41)	24.71 (±0.1125)	**2.095e+03** (±52.37)	66.13 (±3.873)	**0** (±0)	7.2e-03 (±3.44e-03)	**0** (±0)
PSAS-S	**0** (±0)	2e-04§ (±5e-05)	**4.342**§ (±2.757)	3.332e+03‡ (±46.07)	38.86§ (±4.665)	20.41‡ (±0.05166)	1e-03§ (±9.5e-04)	**0** (±0)
SPSO	**0** (±0)	0.051§ (±0.01082)	21.99 (±6.109)	2.742e+03‡ (±40.18)	39.64§ (±3.237)	9.616‡ (±3.904)	2.3e-03 (±1.82e-03)	**0** (±0)
PSAS-D1-1	**0** (±0)	0.0471§ (±8.98e-03)	27.92 (±10.29)	2.843e+03‡ (±88.45)	34.33§ (±3.482)	8.248‡ (±3.807)	2.3e-03 (±1.51e-03)	**0** (±0)
PSO-DR	**0** (±0)	0.0142§ (±4.27e-03)	25.36 (±9.054)	2.7e+03‡ (±90.26)	31.13§ (±3.784)	6.864‡ (±3.687)	**5e-04**§ (±7e-04)	**0** (±0)
PSO-DR M2	**0** (±0)	**0**§ (±0)	5.147§ (±6.21)	3.435e+03‡ (±28.45)	38.95§ (±5.781)	20.27‡ (±0.04715)	4.1e-03 (±1.7e-03)	**0** (±0)
PSO-DR M3	**0** (±0)	5.418e+02‡ (±1.793e+02)	16.16 (±6.954)	2.985e+03‡ (±79.11)	**7.562**§ (±1.248)	1.282 (±1.819)	5.6e-03 (±2.91e-03)	5e-04 (±7e-04)
Bare-bones	**0** (±0)	3.391e+03‡ (±3.994e+02)	26.69‡ (±0.5263)	3.171e+03‡ (±91.41)	1.365e+02‡ (±4.185)	2e-04 (±2.5e-04)	0.0255‡ (±0.01162)	3.1e-03‡ (±1.93e-03)
Essential FIPS	**0** (±0)	18.77§ (±6.134)	22.88§ (±0.4379)	2.332e+03‡ (±66.29)	57.38 (±4.713)	2e-04 (±2.6e-04)	2.4e-03 (±1.49e-03)	**0** (±0)

Algorithm	f_8	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}	f_{15}
Standard BNPS	**0** (±0)	**0** (±0)	**0** (±0)	**0** (±0)	2.905‡ (±1.046)	1.403‡ (±0.8836)	0.3575 (±0.5078)	3.818e-04 (±1.69e-04)
Standard FIPS	**0** (±0)	**0** (±0)	**0** (±0)	**0** (±0)	**0.699** (±0.6773)	**1.602e-04** (±0)	1.263e-04 (±0)	1.435e-04 (±1.262e-04)
PSAS-S	0.02764 (±0.03928)	**0** (±0)	**0** (±0)	**0** (±0)	2.695‡ (±0.9569)	3.623‡ (±0.98)	0.7149 (±0.6921)	2.848e-04 (±1.621e-04)
SPSO	**0** (±0)	**0** (±0)	**0** (±0)	**0** (±0)	2.187 (±1.027)	1.052 (±0.7993)	0.8533 (±0.7892)	**4.26e-05** (±3.01e-05)
PSAS-D1-1	**0** (±0)	**0** (±0)	**0** (±0)	**0** (±0)	5.735‡ (±0.3739)	2.112‡ (±1.062)	2.241‡ (±1.05)	1.121e-04 (±9.57e-05)
PSO-DR	**0** (±0)	**0** (±0)	**0** (±0)	**0** (±0)	2.966‡ (±1.078)	0.7016 (±0.6792)	1.085 (±0.8239)	7.59e-05 (±9.34e-05)
PSO-DR M2	0.08314 (±0.08556)	7.325e-04 (±1.041e-03)	**0** (±0)	**0** (±0)	5.326‡ (±0.2373)	3.613‡ (±0.9815)	4.172‡ (±0.9748)	4.417e-04‡ (±1.674e-04)
PSO-DR M3	**0** (±0)	7.325e-04 (±1.041e-03)	**0** (±0)	10.8 (±10.46)	5.448‡ (±0.3289)	5.75‡ (±0.3372)	5.978‡ (±0.3127)	1.181e-03‡ (±2.784e-04)
Bare-bones	0 (±0)	2e-07‡ (±1e-07)	**0** (±0)	5.4 (±7.674)	5.425‡ (±0.3126)	3.861‡ (±0.9751)	4.6187‡ (±0.8975)	1.662e-04 (±1.167e-04)
Essential FIPS	**0** (±0)	**0** (±0)	**0** (±0)	**0** (±0)	2.724‡ (±1.128)	0.3659 (±0.5198)	**1.262e-04** (±1e-07)	8.03e-05 (±5.42e-05)

Computation, pages 2621–2628, New York, NY, USA, 2007. ACM.

[4] D. Bratton and T. Blackwell. A Simplified Recombinant PSO. *Journal of Artificial Evolution and Applications*, 2008.

[5] D. Bratton and J. Kennedy. Defining a standard for particle swarm optimization. In *Proceedings of the IEEE Swarm Intelligence Symposium*, pages 120–127, 2007.

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

[7] J. Kennedy. Bare bones particle swarms. In *Proceedings of the IEEE Swarm Intelligence Symposium*, pages 80–87, 2003.

[8] J. Kennedy. Dynamic-probabilistic particle swarms. In *GECCO '05: Proceedings of the 2005 Conference on Genetic and Evolutionary Computation*, pages 201–207, New York, NY, USA, 2005. ACM.

[9] J. Kennedy. In search of the essential particle swarm. In *Proceedings of the IEEE Congress on Evolutionary Computation*, pages 1694–1701, 2006.

[10] R. Mendes, J. Kennedy, and J. Neves. The fully informed particle swarm: Simpler, maybe better. *IEEE Transactions on Evolutionary Computation*, 8(3):204–210, 2004.

[11] C. K. Monson and K. D. Seppi. Exposing origin-seeking bias in PSO. In *GECCO '05: Proceedings of the 2005 Conference on Genetic and Evolutionary Computation*, pages 241–248, New York, NY, USA, 2005. ACM.

[12] J. Peña, A. Upegui, and E. Sanchez. Particle swarm optimization with discrete recombination: An online optimizer for evolvable hardware. In *Proceedings of the First NASA/ESA Conference on Adaptive Hardware and Systems, AHS*, pages 163–170, Los Alamitos, CA, USA, 2006. IEEE Computer Society.

[13] R. Poli. On the moments of the sampling distribution of particle swarm optimisers. In *GECCO '07: Proceedings of the 2007 GECCO Conference Companion on Genetic and Evolutionary Computation*, pages 2907–2914, New York, NY, USA, 2007. ACM.

[14] R. Poli, D. Bratton, T. Blackwell, and J. Kennedy. Theoretical derivation, analysis and empirical evaluation of a simpler particle swarm optimiser. In *Proceedings of the IEEE Congress on Evolutionary Computation*, pages 1955 – 1962, 2007.

[15] R. Poli and D. Broomhead. Exact analysis of the sampling distribution for the canonical particle swarm optimiser and its convergence during stagnation. In *GECCO '07: Proceedings of the 9th Annual Conference on Genetic and Evolutionary Computation*, pages 134–141, New York, NY, USA, 2007. ACM Press.