# Markov Chain Models of Bare-Bones Particle Swarm Optimizers

Riccardo Poli
Department of Computer Science
University of Essex, UK
rpoli@essex.ac.uk

William B. Langdon
Department of Biological Sciences
University of Essex, UK
wlangdon@essex.ac.uk

## ABSTRACT

We apply a novel theoretical approach to better understand the behaviour of different types of bare-bones PSOs. It avoids many common but unrealistic assumptions often used in analyses of PSOs. Using finite element grid techniques, it builds a discrete Markov chain model of the BB-PSO which can approximate it on arbitrary continuous problems to any precision. Iterating the chain's transition matrix gives precise information about the behaviour of the BB-PSO at each generation, including the probability of it finding the global optimum or being deceived. The predictions of the model are remarkably accurate and explain the features of Cauchy, Gaussian and other sampling distributions.

## Categories and Subject Descriptors

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

## General Terms

Algorithms, Performance

## Keywords

Particle Swarm Optimisation, Bare-bones PSO, Theory

## 1. INTRODUCTION

PSO theorists have often been forced to make simplifying assumptions in order to obtain models that could be studied mathematically. These include: isolated single individuals, search stagnation (i.e., no improved solutions are found) and absence of randomness. So, the resulting mathematical models are approximate and need to be verified empirically.

The first attempt at providing a theoretical model of PSO was presented in [1]. The model was a simplified model of the type mentioned above, where the behaviour of one particle, in isolation, in one dimension, in the absence of stochasticity and during stagnation was considered. Also the personal best of a particle, $\vec{p}_i$, and and the swarm best, $\vec{p}_g$, were assumed to coincide, as is the case for the best particle in a neighbourhood. No inertia, velocity clamping nor constriction were used. The work was extended in [2] where multiple multi-dimensional particles were covered and $\vec{p}_i$ and $\vec{p}_g$ were not required to coincide.

Similar assumptions were used in [3]: one particle, one dimension, deterministic behaviour and stagnation. Under these conditions the swarm is a discrete-time linear dynamical system. The dynamics of the state of particle can be determined by finding the eigenvalues and eigenvectors of the state transition matrix. The model, therefore, predicts that the particle will converge to equilibrium if the magnitude of the eigenvalues is smaller than 1. Since the eigenvalues of the system are functions of the parameters of the PSO, the model can suggest which parameter settings would guarantee convergence.

A similar approach was used in [4], where, again, one particle, without randomness and during stagnation was modelled. As in previous work, an explicit solution for the trajectory of the particle was provided. [4] also suggested the possibility that particles may converge on a point that is neither the global optimum nor indeed a local optimum. This implies that a PSO is not guaranteed to be an optimiser.

A simplified model of particle was also studied in [5]. The assumptions were: one one-dimensional particle, stagnation and absence of stochasticity. Inertia was included in the model. Again an eigenvalue analysis of the resulting dynamical system was performed with the aim of determining for what parameter settings the systems is stable and what classes of behaviours are possible for a particle.

Blackwell [6] investigated how the spatial extent of a particle swarm varies over time. A simplified swarm model was adopted which is an extension of the one in [3] where more than one particle and more than one dimensions are allowed. This allowed particles to interact, in the sense that they could change their personal best. Constriction was included but not stochasticity.

Under the same assumptions as [3] and following a similar approach, [7] presented a lucid analysis of a 4-parameter family of particle models and identified regions in the parameter space where the model exhibits qualitatively different behaviours (e.g., stability, harmonic oscillations etc.).

The dynamical system approach proposed of [3] has recently been extended in [8, 9] where an extended PSO was studied. Under the assumption that no randomness is present, the resulting model is a discrete, linear and stationary dynamical system, for which [8, 9] formally expressed


*GECCO'07,* July 7–11, 2007, London, England, United Kingdom.

the free and forced responses. However, since the forced response depends inextricably on the specific details of the fitness function, only the free response was studied in detail.

To better understand the behaviour of the PSO during phases of stagnation, [10] analysed the distribution of velocities of one particle controlled by the standard PSO update rule with inertia and *stochastic forces*. In particular, a particle's new velocity was shown to be the sum of a forward force, a backward force and noise.

The stability of particles *in the presence of stochasticity* was studied in [11] by using Lyapunov stability analysis. The behaviour of a single particle – the swarm best – with inertia and during stagnation was considered. By representing the particle as a non-linear feedback system, a large body of knowledge from control theory could be applied. Sufficient conditions on the PSO parameters to guarantee convergence were derived, but, being Lyapunov theory very conservative, these conditions are very restrictive, effectively forcing the PSO to have little oscillatory behaviour.

In summary, despite valiant efforts, a comprehensive mathematical model of particle swarm optimisation has eluded researchers for almost a decade. By applying a simple idea borrowed from the field of solid modelling, in this paper we make a significant first step towards this goal for a special type of PSO: the bare-bones PSO [13]. In Sect. 2 we describe our finite element method Markov approach. Sect. 3 provides a generic model of bare-bones PSOs, which we apply in Sect. 4 to BB-PSOs with different sampling distributions. We compare the predictions of these models with actual runs in Sect. 6. We draw some conclusions in Sect. 7.

## 2. FINITE ELEMENT MODELS OF CONTINUOUS OPTIMISERS

In [12], we proposed a method to build discrete Markov chain models of continuous stochastic optimisers that, in principle, can approximate them on arbitrary continuous problems to any precision. The idea is to discretise the objective function using a finite element method grid which produces corresponding distinct states in the search algorithm. The discretisation makes it easy to compute the transition matrix for the system. Iterating the transition matrix gives precise information about the behaviour of the optimiser at each generation, including the probability of it finding the global optima, the expected run time, etc. In [12] we tested the approach on (1+1) evolutionary strategies, real-valued genetic algorithms and a bare bones PSO with Gaussian sampling distribution. In the case of the bare bones PSO, it was able to model the real system. I.e., a stochastic swarm with multiple interacting individuals on two simple fitness functions (a linear function and a deceptive function). Preliminary empirical results revealed that the predictions of the Markov chain are accurate.

The purpose of [12] was to introduce the idea and to test on a small but diverse group of optimisers. No in-depth analysis of the Markov chains nor of alternative parameter settings and sampling distributions was carried out. In this paper, instead, we concentrate on the bare-bones PSO, and study the resulting Markov chains in a variety of conditions. Our method works as follows.

We start by partitioning a continuous $N$-dimensional search space $\Omega$ into a finite number ($n$) of compact non-overlapping sub-domains $\Omega_i$. For simplicity, we take the fitness of each sub-domain $\Omega_i$ as the fitness at its centre. (I.e. $f_i = f(x_{c_i})$ where $x_{c_i}$ is the centroid of cell $i$) and we adopt the convention of ordering sub-domains by fitness so that $f_i \leq f_j$ for $i < j$. Again for simplicity, we consider $\Omega$ to be an $N$-dimensional cube, which we partition using a regular grid of hypercubic cells:

$$\Omega_i = [x_{c_{i_1}} - r, x_{c_{i_1}} + r] \times [x_{c_{i_2}} - r, x_{c_{i_2}} + r] \times \cdots \times [x_{c_{i_N}} - r, x_{c_{i_N}} + r] \quad (1)$$

where $r$ is the cell "radius" and $x_{c_{i_j}}$ is the $j$-th component of a lattice point $x_{c_i}$.

We then discretise the algorithm by only allowing it to be in discrete states, effectively disallowing all points in the search space except the centroids, $x_{c_i}$ of the domains.

## 3. MODELLING THE BARE-BONES PSO

We apply the technique in Sect. 2 to the bare-bones PSO [13]. This optimiser is inspired by the observation that, at least until a better location in the search space is sampled, the pseudo chaotic particle orbits in a PSO can be approximated by a fixed probability distribution centred on the point lying halfway between the particle best and the neighbourhood best. Its width is modulated by the distance between them. The exact nature of the distribution is not clear: it is bell shaped like a Gaussian distribution but the tails appear to be heavier, like a Cauchy distribution. So, [13] suggested to cut out the integration needed to find each particle's position, and instead draw it from a Gaussian distribution of mean $\frac{1}{2}(\vec{p}_i + \vec{p}_g)$ and standard deviation $|\vec{p}_i - \vec{p}_g|$. This means we no longer need to track each particle's position and velocity: we only need its personal best.

Let us consider a fully-connected bare bones PSO to start with. In this PSO the particles simply randomly sample the neighbourhood of their personal best and swarm best using a fixed probability density function. This continues until either their personal best or the swarm best is improved. When this happens, the parameters of the sampling distribution are recomputed and the sampling process is restarted.

In the unlikely event that more than one particle's personal-best fitness is the same as the best fitness seen so far by the whole swarm, we assume that swarm leadership is shared. That is, each particle chooses as its swarm best a random individual out of the set of swarm bests.

Naturally, at any given time the personal best for each particle and the swarm best will be located in some subdomain $\Omega_i$. In a discretised bare bones PSO both the particle best $x_p$ and swarm best $x_s$ can only take one of a discrete set of values, namely $x_p = x_{c_k}$ and $x_s = x_{c_j}$ for some $j$ and $k$ in $\{0, \cdots, n-1\}$. So, the discretised algorithm can only be in one of a finite set of states. However, we don't need to represent explicitly the swarm best, since the information is implicit in the fitness values $f_i$ of each centroid. So, if $P$ is the population size, there are $n^P$ such states – one for each particle's personal best – and we can represent states of the whole algorithm as $P$ dimensional vectors with integer elements, i.e. $s = (s_1, \cdots, s_P)$. Let us now focus on computing state transition probabilities.

Let $p(x|s_s, x_p)$ be the sampling probability density function when swarm best is $x_s$ and particle best is $x_p$. The standard sampling distribution used in the BB-PSO is a Gaussian distribution but our approach can also be applied to Cauchy or other distributions.

Normally in PSOs random numbers are chosen indepen-

dently for each dimension when computing velocity vectors. In a bare bones PSO we do the same thing. So each dimension of $p(x|x_s, x_p)$ can be calculated separately, and we can write the probability of a particle sampling $x$, given its personal best $x_s$ and the swarm best $x_p$, as $p(x|x_s, x_p) = \prod_{j=1}^{N} p(x_j|x_{s_j}, x_{p_j})$ where $x_j$, $x_{s_j}$, and $x_{p_j}$ are the $j$-th components of vectors $x$, $x_s$ and $x_p$, respectively.

The probability of sampling domain $\Omega_i$, $\Pr(\Omega_i)$, is given by the integral of $p(x|x_s, x_p)$ across $\Omega_i$. Since the subdomains $\Omega_i$ are arranged in a cubic lattice, cf. Equation 1, the probability of landing in cell $\Omega_i$ is given by the product of integrating along each dimension, i.e. we have

$$\Pr(\Omega_i|x_s, x_p) \quad = \quad \prod_j \int_{x_{c_{i_j}}-r}^{x_{c_{i_j}}+r} p(x_j|x_{s_j}, x_{p_j})\, dx_j. \quad (2)$$

Since particle personal bests can only change if there is a fitness improvement, only certain state transitions can occur. That is, a transition from state $s = (s_1, \cdots, s_P)$ to state $s' = (s'_1, \cdots, s'_P)$ is possible only if the fitness of at least one particle improves.

Typically in a fully-connected PSO there is only one particle with the best fitness value. However in a discretised PSO, it is not uncommon to have more than one. So, in general, we have a set $\mathcal{B}(s)$ of swarm bests: $\mathcal{B}(s) = \bigcup_{i:f(s_i)=f_m(s)}\{s_i\}$, where the union includes all members of the population with the maximum fitness $f_m(s) = \max_j f_{s_j}$. More generally, to allow other communication topologies, we need to allow each particle to have its own neighbourhood and so its own set of neighbourhood bests. We will denote these best sets as $\mathcal{B}(s, i)$, for $i = 1, \cdots, P$. Often the best particle in one neighbourhood will also be one of the best in an overlapping neighbourhood, so $\mathcal{B}(s, i)$ may overlap each other.

In a BB-PSO, at each iteration, the particles sample the search space independently. So, if the $i$-th particle's best is in domain $k$ (that is, $s_i = k$), then the probability of it changing to domain $l$ is given by:

$$\Pr(l|\mathcal{B}(s,i), k) = \quad (3)$$
$$\begin{cases} \frac{1}{|\mathcal{B}(s,i)|}\sum_{b\in\mathcal{B}(s,i)} \Pr(\Omega_l|x_{c_b}, x_{c_k}) & \text{if } k \neq l \text{ and } f_l > f_k, \\ 0 & \text{if } k \neq l \text{ and } f_l \leq f_k, \\ 1 - \sum_{l:f_l>f_k} \Pr(l|\mathcal{B}(s,i), k) & \text{if } l = k. \end{cases}$$

This ensures the particle remains in domain $k$ if any of the following conditions is met: 1) the new sample is in $\Omega_k$, 2) the new sample is in a lower fitness cell, 3) the sample is outside $\Omega$.

Because of the independence of the particles (over one time step), we can then write the state transition probability for the whole PSO as

$$m_{s,s'} \quad = \quad \prod_i \Pr(s'_i|\mathcal{B}(s,i), s_i) \quad (4)$$

where further decompositions can be obtained using Equations 3 and 2.

# 4. BB-PSO SAMPLING DISTRIBUTIONS

It is interesting to study how the behaviour and performance of the BB-PSO is affected by changes in its sampling distribution. In the following sub-sections we describe the distributions considered in this paper.

## 4.1 Uniform sampling distributions

Consider the simplest possible form of sampling distribution: a symmetric flat uniform distribution of width $(1 + 2\alpha)\Delta$, where $\Delta$ is the distance between the particle's best and its neighbourhood best and $\alpha \geq 0$ ensures we also sample outside the range between them. I.e. a distribution of the form:

$$p(x_j|x_{s_j}, x_{p_j}) = \begin{cases} 1/(1+2\alpha)\Delta & \text{if } x \in [x_{min} - \alpha\Delta, x_{max} + \alpha\Delta] \\ 0 & \text{otherwise,} \end{cases}$$

where $x_{min} = \min(x_{s_j}, x_{p_j})$, $x_{max} = \max(x_{s_j}, x_{p_j})$, $\Delta = x_{max} - x_{min}$ and $\alpha \geq 0$ is a constant. Note that this distribution becomes a Dirac delta function when the particle and neighbourhood best coincide, i.e. when $x_{s_j} = x_{p_j}$.

If the particle and neighbourhood best are different, i.e. $x_{s_j} \neq x_{p_j}$, we have that

$$\int_{x_{c_{i_j}}-r}^{x_{c_{i_j}}+r} p(x_j|x_{s_j}, x_{p_j})\, dx_j \quad = \quad \max\left(0, \frac{\beta_2 - \beta_1}{(1+2\alpha)\Delta}\right)$$

where $\beta_1 = \max(x_{c_{i_j}} - r, x_{min} - \alpha\Delta)$ and $\beta_2 = \min(x_{c_{i_j}} + r, x_{max} + \alpha\Delta)$. If the particle and neighbourhood best are the same, i.e. $x_{s_j} = x_{p_j}$, then the integral is 1 when the particle samples cell $i$ (i.e., $x_{p_j} = x_{c_{i_j}}$) and 0 otherwise.

In the rest of the paper we will limit our attention to the distributions obtained for $\alpha = 0$ and $\alpha = 0.5$, which we will term "uniform" and "extended uniform", respectively.

## 4.2 Gaussian sampling distributions

For a Gaussian sampling distribution the mean is half way between the swarm and particle bests, whilst we set the standard deviation equal to the distance between them, i.e.

$$p(x_j|x_{s_j}, x_{p_j}) = G\left(\frac{x_{s_j} + x_{p_j}}{2}, \left|x_{s_j} - x_{p_j}\right|\right).$$

Note that also this distribution becomes a Dirac delta function when $x_{s_j} = x_{p_j}$.

The integral of the Gaussian distribution is the erf function. So, if $x_{s_j} \neq x_{p_j}$, we have

$$\int_{x_{c_{i_j}}-r}^{x_{c_{i_j}}+r} p(x_j|x_{s_j}, x_{p_j})\, dx_j =$$
$$\frac{1}{2}\left(\mathrm{erf}\left(\frac{x_{c_{i_j}} + r - \frac{x_{s_j}+x_{p_j}}{2}}{|x_{s_j} - x_{p_j}|\sqrt{2}}\right) - \mathrm{erf}\left(\frac{x_{c_{i_j}} - r - \frac{x_{s_j}+x_{p_j}}{2}}{|x_{s_j} - x_{p_j}|\sqrt{2}}\right)\right)$$

while, again, if $x_{s_j} = x_{p_j}$ the integral is 1 when $x_{p_j} = x_{c_{i_j}}$ and 0 otherwise.

## 4.3 Cauchy sampling distribution

For a Cauchy sampling distribution we have

$$p(x_j|x_{s_j}, x_{p_j}) = \frac{\gamma}{\pi\left[\gamma^2 + (x-\mu)^2\right]}$$

where $\mu = \frac{x_{s_j}+x_{p_j}}{2}$ and $\gamma = |x_{s_j} - x_{p_j}|$. Note that also this distribution becomes a Dirac delta function when $x_{s_j} = x_{p_j}$.

So, for this distribution, when $x_{s_j} \neq x_{p_j}$, we have

$$\int_{x_{c_{ij}}-r}^{x_{c_{ij}}+r} p(x_j|x_{s_j},x_{p_j})\ dx_j =$$

$$\frac{1}{\pi}\left(\mathrm{tg}^{-1}\left(\frac{x_{c_{ij}}+r-\frac{x_{s_j}+x_{p_j}}{2}}{|x_{s_j}-x_{p_j}|}\right)-\mathrm{tg}^{-1}\left(\frac{x_{c_{ij}}-r-\frac{x_{s_j}+x_{p_j}}{2}}{|x_{s_j}-x_{p_j}|}\right)\right)$$

where $\mathrm{tg}^{-1}$ is the arc tangent function. If, instead, $x_{s_j} = x_{p_j}$, again the integral is 1 when $x_{p_j} = x_{c_{ij}}$ and 0 otherwise.

### 4.4 Non-collapsing Uniform, Gaussian and Cauchy distributions

We also consider BB-PSO sampling distributions which continue to explore even when the particle's best is also the best in the neighbourhood. This can easily be done by ensuring the width of the distribution never shrinks below $\nu$.

So, we propose a variation of the uniform sampling distribution where

$$p(x_j|x_{s_j},x_{p_j}) = \begin{cases} 1/\max(\alpha\Delta,2\nu) & \text{if } x \in [\bar{x}-\frac{1}{2}\max(\alpha\Delta,2\nu), \bar{x}-\\ & \frac{1}{2}\max(\alpha\Delta,2\nu)], \\ 0 & \text{otherwise,} \end{cases}$$

where $\bar{x} = (x_{min} + x_{max})/2$ and $\nu$ is a small positive constant. Also, we propose a variation of the Gaussian sampling distribution, where

$$p(x_j|x_{s_j},x_{p_j}) = G\left(\frac{x_{s_j}+x_{p_j}}{2}, \max(|x_{s_j}-x_{p_j}|,\nu)\right)$$

where $\nu$ is a small positive constant. Finally, we propose the following variant for the Cauchy distribution:

$$p(x_j|x_{s_j},x_{p_j}) = \frac{\tilde{\gamma}}{\pi\left[\tilde{\gamma}^2+(x-\mu)^2\right]}$$

where $\tilde{\gamma} = \max(\gamma,\nu)$. For all distributions we will concentrate on the case $\nu = 0.5$.

These changes prevent the sampling distributions from becoming a delta when $x_{s_j} = x_{p_j}$. This, as we will see, has important consequences.

## 5. SUCCESS PROBABILITY AND CONVERGENCE TIME

In a BB-PSO (and many other optimisers) each initial state is equally likely. That is, if $\pi_0$ represents the initial probability distribution over states, the components of $\pi_0$ are all $1/n^P$. Given $\pi_0$ and the transition matrix $M$, we can compute the probability distribution of a BB-PSO being in any particular state at generation $t$, from $\pi_t = M^t\pi_0$.

The success probability is obtained by summing the components of $\pi_t$ corresponding to states where at least one particle is in fittest domain. I.e., if $J$ is the set of such states, then the success probability is $\sum_{i\in J}\pi_{ti}$ at generation $t$. Naturally, $J$ could represent any other set of states of interest. E.g., if $J$ represents all the states where all particles are in the same domain (i.e., the PSO has converged), then $\sum_{i\in J}\pi_{ti}$ represents the probability of the BB-PSO being converged (whether on the global optimum or elsewhere) at generation $t$.

We can also estimate the expected time before the BB-PSO achieves a particular goal or settles in a particular state by computing the *expected waiting time* of the corresponding discrete Markov chain to visit a particular target state or set of states $J$. This is given by (see [14, pages 168–170])

$$EWT_J = \sum_{i\notin J}\pi_{0i}\eta_{i,J}$$

where $\pi_0$ is the distribution of initial states and $\eta_{i,J}$ is the *mean passage time* for going from state $i$ to the set of states $J$, given that it is currently outside the set, which is obtained by solving the following system of simultaneous equations:

$$\eta_{i,J} - \sum_{k\notin J} m_{i,k}\eta_{k,J} \quad = \quad 1 \qquad (5)$$

where $m_{i,j}$ are the elements of Markov matrix for the system.

If we represent Equations 5 in matrix notation as $\widetilde{M}_J\eta_J = \mathbf{1}$, it is clear that $\eta_J$ can only be computed if $\widetilde{M}_J$ is invertible for a particular set $J$. Note that this is the case only if the system can reach $J$ with non-zero probability from all initial conditions. Also, note that the matrix $\widetilde{M}_J$ may be invertible for some $J$ but not others (as we will see later).

Assuming invertibility, when the set of states $J$ represents all states where at least one particle is in the domain containing the global optimum, then $EWT_J$ effectively represents the expected time required for a BB-PSO to solve a problem (run-time). If instead $J$ represents all the states where all particles are in the same domain, then $EWT_J$ represents the expected time to convergence (somewhere, i.e., not necessarily on an optimum). If, finally, $J$ contains only the state where all particles are in the same domain *and* such a domain contains the global optimum, $EWT_J$ represents the expected time to convergence to the global optimum.

## 6. RESULTS

### 6.1 Test problems and set up

We used the domain $\Omega = [-5,5]$ and the following three 1–D maximisation problems to our models (see Figure 1): the linear function $f(x) = x$, the Rastrigin function $f(x) = 3 - 0.1x^2 + \cos(2\pi x)$, and the sphere function $f(x) = 3 - 0.1x^2$.

We tested our model of BB-PSO against real runs on these three problems and with eight distributions (see Sect. 4): uniform, Gaussian, Cauchy, extended-uniform, non-collapsing (NC) uniform, NC Gaussian, NC Cauchy and NC extended-uniform. We constructed models with resolutions, $n$, of 11, 21, 31, 41, 51, and 61 elements. To obtain accurate statistics, for each set up, we performed 5 000 independent runs of the BB-PSO. In all tests we focused on the case of a PSO with two particles.[1]

### 6.2 Model accuracy

Our Markov chain is an approximate model of a real, continuous BB-PSO, so we cannot expect it to *exactly* model its continuous counterpart for a large number of generations. We should expect, however, good accuracy for several generations. So, it is important to assess where the accuracy of

[1]We have demonstrated that it is possible to run models of larger swarms on an ordinary PC if the mesh is coarse [12]. Here we limit our attention to two particles, so that we can test finer resolutions.

the model becomes unacceptable. To do so we compared the model predictions vs. real runs at generations 10 and 50.[2]

Figure 2 shows a comparison between the probability of the whole population converging to the global optimum predicted by the chain vs. that obtained in real runs at generations 10. As one can see there is excellent agreement between model and real runs, with only a handful of very low resolution models ($n = 21$) diverging substantially from the main diagonal. A similar picture was obtained when we focused on the success probability (i.e., the probability that at least one member of the population visited the domain containing the global optimum), as shown in Figure 3.

Except for some off diagonal outliers, the predictions at generation 50 are still quite good, cf. Figure 4. Most of the poor predictions are BB-PSOs using Cauchy sampling. The Cauchy distribution is very peculiar. It has fat tails and an infinite variance. As a result, even when particles are very near to each other, there is a good chance of producing large displacements. So, in real runs nearly converged populations continue to progress slowly towards the global optimum. However this is something that the Markov chain cannot predict, since it violates the main assumption under which our finite elements model is built: that the elements are sufficiently small that we can describe the behaviour of particles within them with simple functions.

Overall, these results indicate that provided the resolution of the mesh is high-enough, our Markov chain model can predict the short term behaviour of a BB-PSO very accurately. The long term prediction accuracy can also be very good provided that the assumptions behind FEM are not violated.

## 6.3 Comparison of Distributions

Let us now consider our results by grouping them according to sampling distribution and problem type. Figure 5 shows how the probability of a member of the population sampling the domain containing the global optimum varies generation after generation. Similar results were obtained for the probability of converging to the global optimum. As one can see the predictions of the model are very accurate even over the 50 generations in many configurations. As indicated before exceptions are the Cauchy distributions, where after 10 or 15 generations the resolution of the model ($n = 41$ in the figure) shows its limits. Interestingly, on the Rastrigin function, the hardest to optimise, the model correctly predicts the dynamics of the system even with the Cauchy distribution. This is because the 10 local optima of the function can trap particles for longer. This means that the distance between them remains higher, for longer. This delays the violation of the FEM assumptions until later generations.

What can we learn from these plots? Firstly we can see that, on the two unimodal functions in our set, Sphere and Linear, the non-collapsing versions of the uniform, Gaussian and Cauchy sampling distributions invariably lead to sampling the domain of the global optimum. That is, for those functions the transition matrix of the chain is ergodic and so irrespective of where one starts from, the global optimum is guaranteed to be found.

For multimodal functions, such as Rastrigin, only the non-collapsing Gaussian and Cauchy guarantee convergence to

[2]Because the population is small and the test problems are 1–D, all PSOs exhibit rapid convergence.

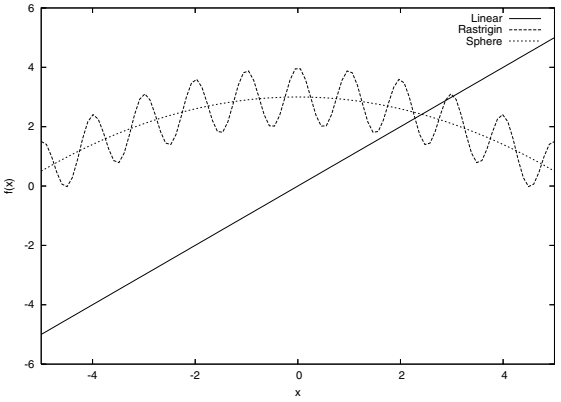



**Figure 1: Three fitness functions used in our tests.**

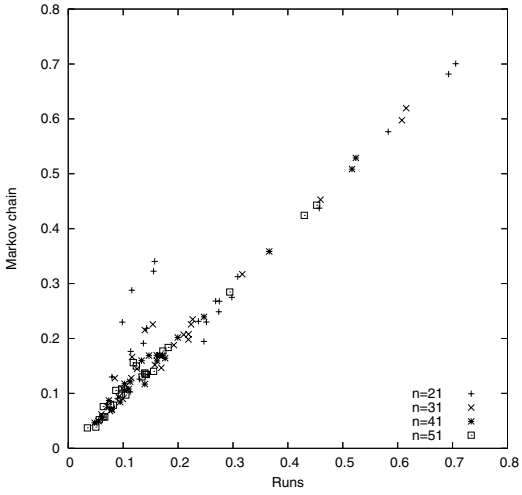



**Figure 2: Comparison between the probability of convergence to the global optimum predicted by the chain vs. that obtained in 5 000 independent runs of real BB-PSOs at generation 10. Data for all resolutions, fitness functions and distributions are plotted.**

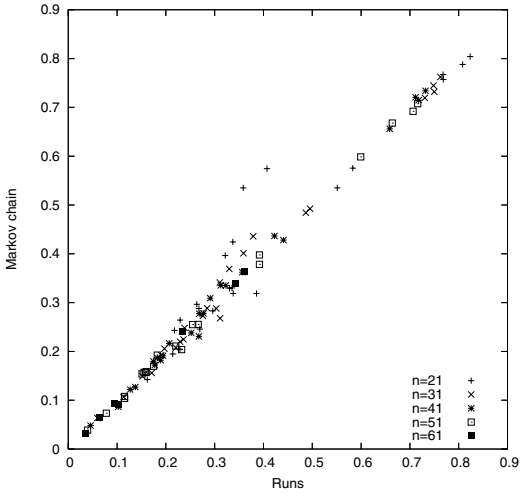



**Figure 3: Comparison between the success probability predicted by the chain vs. that obtained in 5 000 independent runs of real BB-PSOs at generation 10. Data for all resolutions, fitness functions and distributions are plotted.**

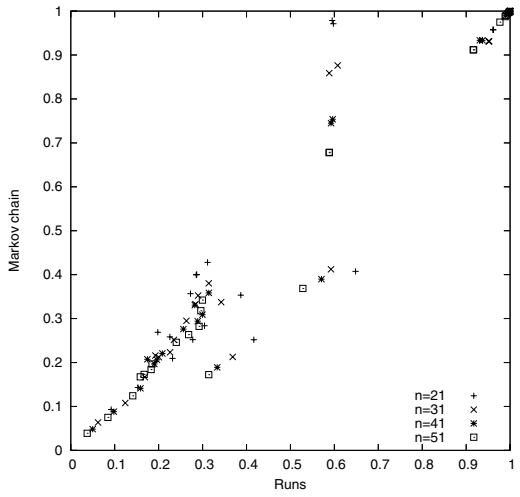



**Figure 4: Comparison between the probability of convergence to the global optimum predicted by the chain vs. that obtained in 5 000 independent runs of real BB-PSOs at generation 50. Data for all resolutions, fitness functions and distributions are plotted.**

the global optimum, at least with $\nu = 0.5$. With $\nu = 0.5$, if both particles reach the basin on attraction of a local optimum, they cannot escape it with the uniform and extended uniform sampling. With Gaussian and Cauchy, however, this is possible thanks to their infinite tails.

Naturally the "convergent" versions of the distributions can converge before the global optimum is sampled, leading to success probabilities $< 1$, for all fitness functions considered. It is surprising to see that even in the unimodal functions the population freezes in non-optimal states in a large proportion of runs, proving that also the BB-PSO is not guaranteed to be an optimiser as is the case for ordinary PSOs [4] (see Sect. 1). Such premature convergence is less frequent with the Cauchy and Gaussian distributions than with the uniform ones, and it is completely absent in the non-collapsing versions of Cauchy and Gaussian.

## 6.4 Solution time of BB-PSO

As explained in Sect. 5, the Markov chain model makes it possible to calculate the expected first hitting times for a variety of events. In particular, for the sampling distributions which are guaranteed to lead to sampling the global optimum, namely the non-collapsing Gaussian and Cauchy, one can compute the expected time to convergence to the global optimum, $EWT_J$, where $J$ is the set containing the state where all particles are in the domain of the global optimum.

We can also investigate $\eta_{i,J}$, the mean time for going from any state $i$ to the solution state $J$ (see Sect. 5). Large values of $\eta_{i,J}$ indicate initial conditions that are particularly hard for the optimiser. We show plots of $\eta_{i,J}$ for different problems and two BB-PSO sampling distributions in Figure 6.

Generally the non-collapsing Cauchy distribution is faster than the non-collapsing Gaussian. This is particularly true for the Rastrigin function. However, there are conditions where Gaussian beats Cauchy. Such as when at least one of the particles is near the global optimum (domain number 40). This suggests that starting the search with a non-

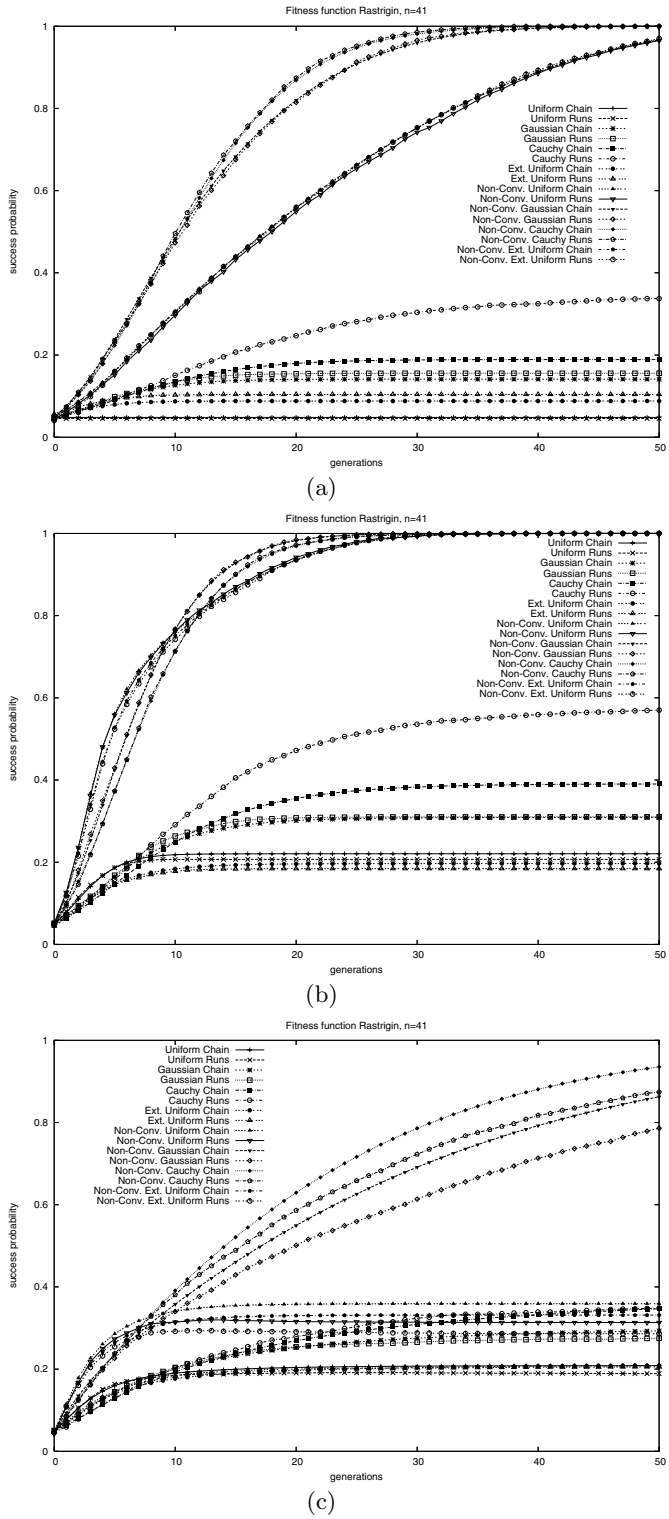



**Figure 5: Success probability for the Linear (a), Sphere (b) and Rastrigin (c) fitness functions for different sampling distributions. Grid resolution $n = 41$.**

collapsing Cauchy distribution and then progressively turning it into a non-collapsing Gaussian might be beneficial.

Note the raggedness of the plots for Sphere and Rastrigin. In the case of the Sphere there are regular ripples parallel to the main diagonal. These are because the function is symmetric and its optimum is in the middle of the search interval $[-5, 5]$. Therefore two particles with exactly the same fitness can either be at symmetric positions (say, $x$ and $-x$) or at the same position (say both $x$). Because in Figure 6 domains are ordered by fitness, these two distinct states are represented by neighbouring points. In the symmetric case the mode of the BB-PSO's sampling distribution is 0 while in the other it is $x$. Thus it is much easier for the BB-PSO to reach the origin and so solve the problem by starting from the $(x,-x)$ configuration than from $(x,x)$. Hence two neighbouring points in Figure 6 have very different heights. This is also true of the Rastrigin distribution. Here, however, we also have another important effect: particles can be temporarily trapped by local optima and this is easier the closer they are to a local optimum. This is the cause for the ridges parallel to the $x$ and $y$ axes in Figure 6 (bottom).

# 7. CONCLUSIONS

We have applied a novel theoretical tool to the analysis of different types of bare-bone particle swarm optimisers. This allows the creation of discrete Markov chain models which approximate the behaviour of a BB-PSO exploring a continuous space. Being Markov chains, the models allow one to compute everything that one needs to estimate about the distribution of states of a search algorithm for any fitness function. The models can be made arbitrarily accurate by increasing the resolutions of the discretisation mesh.

We have shown that even with the modest resolutions we adopted the models are very accurate over 10 or more generations (and, in fact, in many case for entire runs). The analysis of the models reveals important features of different sampling distributions. In particular, it reveals that a minor change to the sampling distribution — preventing its variance from ever becoming zero — transforms the Gaussian and Cauchy BB-PSO into global optimisers, at least for the three fitness functions we have studied.

The drawback of our models is their scalability. I.e., the size of the transition matrix grows quadratically with the number of domains in the discretisation grid and exponentially with the number of particles and the number of dimensions. So, we should not expect that this approach will solve all problems of theoretical interest. It is clear, however, that in principle the method overcomes many of the limitations presented by previous theoretical analyses: a) it allows the analysis of the dynamics of more than one particle, b) it can be applied to the case where fitness is present, c) it can be applied to any fitness function, d) it models stochasticity, e) it provides accurate predictions over multiple generations.

In future research we intend to apply this technique to study classical PSOs with velocity and inertia/constriction. Also, we want to adopt a sparse matrix representation to allow the study PSOs with larger populations and different communication topologies.

## Acknowledgements

The authors would like to acknowledge support by EPSRC XPS project (GR/T11234/01).

# 8. REFERENCES

[1] Ozcan, E. and Mohan, C. K. (1998). Analysis of a simple particle swarm optimization system. *Intelligent Engineering Systems Through Artificial Neural Networks* Vol. 8, pp. 253–258.

[2] Ozcan, E., and Mohan, C. (1999). Particle swarm optimization: surfing the waves. *Proc. 1999 Congress on Evolutionary Computation*, 1939–1944. IEEE.

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

[4] van den Bergh, F. (2002) *An Analysis of Particle Swarm Optimizers*. PhD thesis, Department of Computer Science, University of Pretoria, South Africa.

[5] K. Yasuda, A. Ide, and N. Iwasaki (2003) Adaptive particle swarm optimization. *Systems, Man and Cybernetics, 2003. IEEE International Conference on*, 2:1554–1559.

[6] T. M. Blackwell. Particle swarms and population diversity. *Soft Computing*, 9:793–802, 2005.

[7] Trelea, I. C. (2003) The particle swarm optimization algorithm: convergence analysis and parameter selection. *Information Processing Letters*, 85(6):317–325.

[8] Campana, E.F., Fasano, G., and Pinto, A. (2006) Dynamic system analysis and initial particles position in particle swarm optimization. In *IEEE Swarm Intelligence Symposium*, Indianapolis.

[9] Campana, E.F., Fasano, G., Peri, D., and Pinto, A. (2006) Particle swarm optimization: Efficient globally convergent modifications. In C. A. Mota Soares et al., editor, *Proceedings of the III European Conference on Conputational Mechanics, Solids, Structures and Coupled Problems in Engineering*, Lisbon, Portugal.

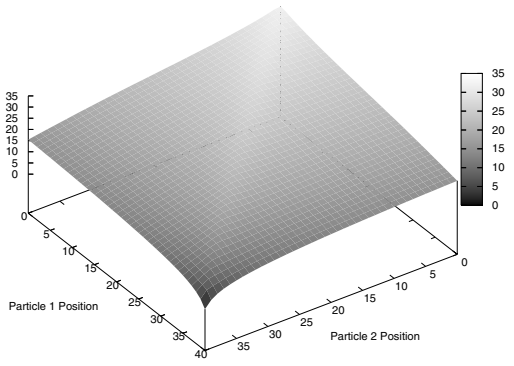
[10] Clerc, M. (2006) Stagnation analysis in particle swarm optimisation or what happens when nothing happens. Technical Report CSM-460, Department of Computer Science, University of Essex, August 2006.

[11] Kadirkamanathan, V., Selvarajah, K., and Fleming, P. J. (2006) Stability analysis of the particle dynamics in particle swarm optimizer. *IEEE Trans. Evolutionary Computation*, 10(3):245–255.

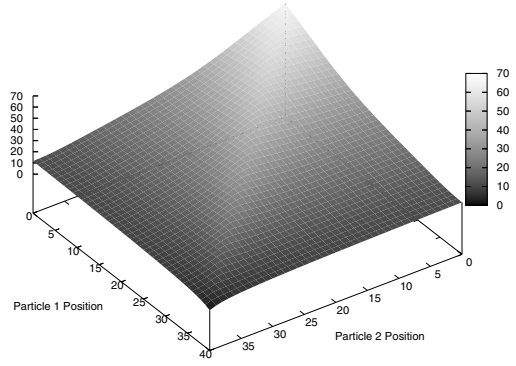
[12] Poli, R., Langdon, W. B., Clerc, M., and Stephens, C. R. (2007) Continuous optimisation theory made easy? Finite-element models of evolutionary strategies, genetic algorithms and particle swarm optimizers. Proceedings of the Foundations of Genetic Algorithms (FOGA) workshop. (Also available as Technical Report CSM-463, Department of Computer Science, University of Essex.)

[13] Kennedy, J. (2003) Bare bones particle swarms. *Proceedings of the IEEE Swarm Intelligence Symposium*, 80–87. Indianapolis.

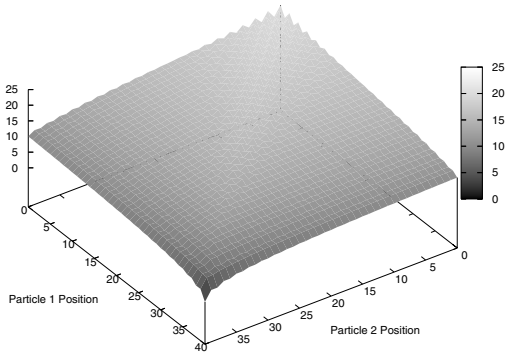
[14] W. M. Spears (1998) *The Role of Mutation and Recombination in Evolutionary Algorithms*. PhD thesis, George Mason University.



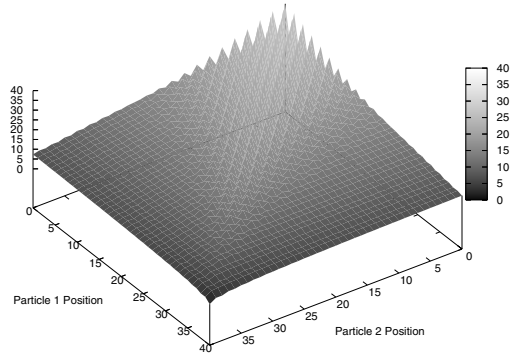

Linear Fitness – non-collapsing Cauchy Distribution



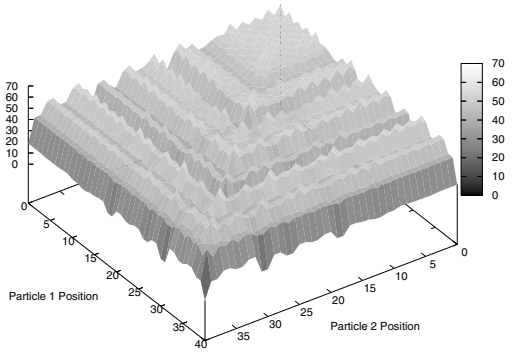

Linear Fitness – non-collapsing Gaussian Distribution



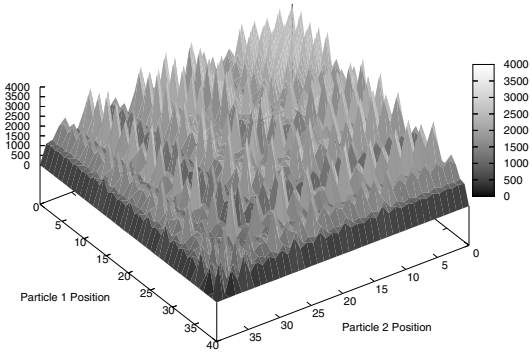

Sphere Fitness – non-collapsing Cauchy Distribution




Sphere Fitness – non-collapsing Gaussian Distribution




Rastrigin Fitness – non-collapsing Cauchy Distribution




Rastrigin Fitness – non-collapsing Gaussian Distribution

**Figure 6: Convergence times to the global optimum for a BB-PSO with a population of two particles estimated using a Markov chain model with $n = 41$ elements. Domains are numbered (0–40) in increasing fitness order.**