Particle Swarm Optimizer: The Impact of Unstable Particles on Performance

Christopher W Cleghorn
Department of Computer Science
University of Pretoria
Pretoria, South Africa
Email: ccleghorn@cs.up.ac.za

Andries Engelbrecht
Department of Computer Science
University of Pretoria
Pretoria, South Africa
Email: engel@cs.up.ac.za

***Abstract*—There exists a wealth of theoretical analysis on particle swarm optimization (PSO), specifically the conditions needed for stable particle behavior are well studied. This paper investigates the effect that the stability of the particle has on the PSO's actually ability to optimize. It is shown empirically that a majority of PSO parameters that are theoretically unstable perform worse than a trivial random search across 28 objective functions, and across various dimensionalities. It is also noted that there exists a number of parameter configurations just outside the stable-2 region which did not exhibit poor performance, implying that a minor violation of the conditions for order-2 stability is still acceptable in terms of overall performance of the PSO.**

I. Introduction

Particle swarm optimization (PSO) is a stochastic population-based search algorithm developed by Kennedy and Eberhart [1]. PSO has been effectively utilized to solve numerous real world optimization problems, a summary of which can be found in the work of Poli [2].

PSO has undergone a substantial amount of theoretical analysis, the majority of which focus on the conditions that are both necessary and sufficient for particle stability. Some of the more recent works are [3], [4], [5], [6], [7], [8]. Currently the conditions needed for order-1 and order-2 stability are well known. Order-1 and order-2 stability are respectively defined as convergence in expectation of the particle's position and convergence of the particle's variance.

This paper investigates the effect of selecting parameters that theoretically yield unstable particle behavior (non order-2 behavior). This paper considers a wide range of objective functions across various dimensionalities. Performance is also considered across differing iteration counts.

A description of PSO is given in section II. Section III contains an discussion about the theoretical PSO results directly relevant to this paper. The experimental setup is presented in section IV, followed by the experimental results and a discussion thereof in section V. Section VI presents a summary of the findings of this paper.

II. Particle Swarm Optimizer

Particle swarm optimization (PSO) was originally developed by Kennedy and Eberhart [1] to simulate the complex movement of birds in a flock. The standard variant of PSO this section focuses on includes the inertia coefficient proposed by Shi and Eberhart [9].

The PSO algorithm is defined as follows: Let $f : \mathbb{R}^d \to \mathbb{R}$ be the objective function that the PSO algorithm aims to find an optimum for, where d is the dimensionality of the objective function. For the sake of simplicity, a minimization problem is assumed from this point onwards. Specifically, an optimum $\boldsymbol{o} \in \mathbb{R}^d$ is defined such that, for all $\boldsymbol{x} \in \mathbb{R}^d$, $f(\boldsymbol{o}) \leq f(\boldsymbol{x})$. The analysis of this paper focuses on objective functions where the optima exist. Let $\Omega(t)$ be a set of N particles in $\mathbb{R}^d$ at a discrete time step t. Then $\Omega(t)$ is said to be the particle swarm at time t. The position $\boldsymbol{x}_i$ of particle i is updated using

$$\boldsymbol{x}_i(t+1) = \boldsymbol{x}_i(t) + \boldsymbol{v}_i(t+1), \tag{1}$$

where the velocity update, $\boldsymbol{v}_i(t+1)$, is defined as

$$\begin{aligned} \boldsymbol{v}_i(t+1) = w\boldsymbol{v}_i(t) + c_1\boldsymbol{r}_1(t) \otimes (\boldsymbol{y}_i(t) - \boldsymbol{x}_i(t)) \\ + c_2\boldsymbol{r}_2(t) \otimes (\hat{\boldsymbol{y}}_i(t) - \boldsymbol{x}_i(t)), \end{aligned} \tag{2}$$

where $r_{1,j}(t), r_{2,j}(t) \sim U(0,1)$ for all t and $1 \leq j \leq k$. The operator $\otimes$ is used to indicate component-wise multiplication of two vectors. The position $\boldsymbol{y}_i(t)$ represents the "best" position that particle i has visited, where "best" means the location where the particle had obtained the lowest objective function evaluation. The position $\hat{\boldsymbol{y}}_i(t)$ represents the "best" position that the particles in the neighborhood of the i-th particle have visited. The coefficients c_1, c_2, and w are the cognitive, social, and inertia weights, respectively.

A primary feature of the PSO algorithm is social interaction, specifically the way in which knowledge about the search space is shared amongst the particles in the swarm. In general, the social topology of a swarm can be viewed as a graph, where nodes represent particles, and the edges are the allowable direct communication routes. The social topology chosen has a direct impact on the behaviour of the swarm as a whole [10], [11], [12]. Some of the most frequently used are the star, ring, and Von Neumann topologies.

III. Theoretical Considerations

This section presents a brief description of the theoretical PSO results that are directly related to this paper are pre-

Algorithm 1 PSO algorithm

Create and initialize a swarm, $\Omega(0)$, of N particles uniformly within a predefined hypercube redof dimension k.
Let f be the objective function.
Let $\boldsymbol{y}_i$ represent the personal best position of particle i, initialized to $\boldsymbol{x}_i(0)$.
Let $\hat{\boldsymbol{y}}_i$ represent the neighborhood best position of particle i, initialized to $\boldsymbol{x}_i(0)$.
Initialize $\boldsymbol{v}_i(0)$ to $\mathbf{0}$.
repeat
  for all particles $i = 1, \cdots, N$ **do**
    if $f(\boldsymbol{x}_i) < f(\boldsymbol{y}_i)$ **then**
      $\boldsymbol{y}_i = \boldsymbol{x}_i$
    end if
    for all particles $\hat{i}$ with particle i in their nhb **do**
      if $f(\boldsymbol{y}_i) < f(\hat{\boldsymbol{y}}_{\hat{i}})$ **then**
        $\hat{\boldsymbol{y}}_{\hat{i}} = \boldsymbol{y}_i$
      end if
    end for
  end for
  for all particles $i = 1, \cdots, N$ **do**
    update the velocity of particle i using equation (2)
    update the position of particle i using equation (1)
  end for
until stopping condition is met

sented. This section also discusses what instability of a PSO particle implies.

There is a relatively large amount of theoretical work on PSO. Currently, the most general theoretical PSO model is currently from Bonyadi and Michalewicz [8], which is an extension of the work of Poli [4], relying on a recurrence relation of the form

$$\boldsymbol{z}_t = \boldsymbol{M}\boldsymbol{z}_{t-1} + \boldsymbol{b}, \tag{3}$$

where $\boldsymbol{z}_t, \boldsymbol{b} \in \mathbb{R}^q$ and $\boldsymbol{M}$ a matrix from $\mathbb{R}^q$ to $\mathbb{R}^q$.

The order-2 stability of a PSO particle relies upon the spectral radius of $\boldsymbol{M}$ being less than 1, where the spectral radius is defined as the maximum absolute value of the eigenvalues of the matrix $\boldsymbol{M}$, denoted $\rho(\boldsymbol{M})$. The spectral radius of M is less than 1 for PSO update equations (1) and (2) if the following criteria are satisfied (illustrated in figure 1):

$$c_1 + c_2 > 0, \qquad |w| < 1, \qquad c_1 + c_2 < \frac{24(1-w^2)}{7-5w} \tag{4}$$

The criteria for order-2 stability has also been empirically verified without the presence of simplifying assumptions by Cleghorn and Englebrecht [13].

An important aspect of an unstable particle is at what rate the positional variance would increase. For the sake of simplicity, consider M as symmetric, with $\rho(M) > 1$ (what follows still holds if M is not symmetric; however,

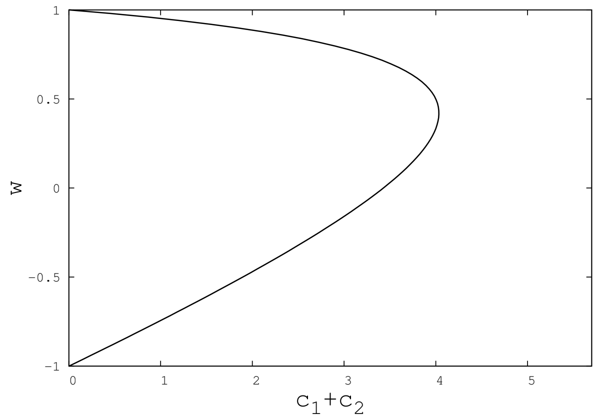

Fig. 1: Order-2 stable region

the technical detail would detract from the discussion). Now, unwinding of equation (3) leads to

$$\boldsymbol{z}_t = \boldsymbol{M}^{n-1}\boldsymbol{z}_1 + \sum_{j=0}^{n-2} \boldsymbol{M}^i \boldsymbol{b} \tag{5}$$

Since M is symmetric it is possible to represent any $\boldsymbol{z}_1$ as a weighted sum of $\boldsymbol{M}$'s eigenvectors, specifically $\boldsymbol{z}_1 = \sum_{i=0}^{q} \eta_i \boldsymbol{e}_i$, where $\boldsymbol{e}_i$ are the eigenvectors of M, which have the corresponding eigenvalues λ_i. Consider the first term of equation (5):

$$\boldsymbol{M}^{n-1}\boldsymbol{z}_1 = \boldsymbol{M}^{n-1}\sum_{j=0}^{q} \eta_j \boldsymbol{e}_j = \sum_{j=0}^{q} \eta_i \lambda_i^{n-1} \boldsymbol{e}_i \tag{6}$$

Since $\rho(\boldsymbol{M}) > 1$, at least one eigenvalue of $\boldsymbol{M}$ is greater than 1 so at least one term in the summation in equation (6) will diverge (assuming the corresponding $\eta_i \neq 0$). Not only will the term diverge, but it will do so exponentially.

The exponential divergence is very important to note as small increases in $\rho(\boldsymbol{M})$ could drastically increase the long term particle trajectory. The immediate question is how tolerant the PSO is to having control parameters selected that have slightly larger than 1 spectral radius. It would be ideal to derive the required conditions for different spectral radius's of $\boldsymbol{M}$. While the criteria of equation (4) appear relatively simple, they are not directly derivable from the condition $\rho(M) < 1$ even with assistance of a symbolic solver. Instead, necessary conditions where derived using a technique first used by Blackwell [5], and then empirically verified to hold for $\rho(M) < 1$ [8]. This approach was needed as the individual eigenvalues of M require over a 1000 characters to express. Unfortunately, this indicates that finding the explicit condition for $\rho(M) < s$, where s is any non-negative number is intractable.

IV. Empirical Setup

This section summarizes the experimental procedure used for this paper. The experiment aims to illustrate that most unstable parameter configurations actually result in such poor performance of PSO that a random search can outperform them.

The performance of the PSO was measured for each parameter configuration across the following region:

$$w \in [-1.1, 1.1] \text{ and } c_1 + c_2 \in (0, 5.5] \quad (7)$$

where step sizes of 0.1 were used for w and $c_1 + c_2$. This results in 1264 parameter configurations, of which 761 are unstable and 504 are stable according to equation (4). A fully connected star neighborhood topology was used. Velocities where initialized to $\mathbf{0}$. A population size of 20 was used. The results for each configuration were derived from 35 independent runs.

The performance of PSO is compared to that of a random search, with the premise that if a specific configuration of PSO does worse than a random search it is not effectively optimizing. The random search algorithm used, samples uniformly within a given objective function's defined domain. For the sake of comparison each iteration of the PSO algorithm is seen as comparable to 20 random samples of the search space, one for each of the particles in the PSO swarm.

The objective functions used in this paper used are presented in table I. Full definitions of the objective functions can be found in following works, [14], [15], [16]. Each objective function is tested in 5, 10, and 30 dimensions. The performance is measured at 500, 1000, 2000, and 5000 iterations.

TABLE I: Objective Functions

	Function name	Domain
F1	Absolute Value	$\boldsymbol{x} \in [-100, 100]^d$
F2	Ackley	$\boldsymbol{x} \in [-32, 32]^d$
F3	Alpine	$\boldsymbol{x} \in [-10, 10]^d$
F4	Bent Cigar	$\boldsymbol{x} \in [-100, 100]^d$
F5	Discus	$\boldsymbol{x} \in [-100, 100]^d$
F6	Egg Holder	$\boldsymbol{x} \in [-512, 512]^d$
F7	Elliptic	$\boldsymbol{x} \in [-100, 100]^d$
F8	Expanded F9+F19	$\boldsymbol{x} \in [-100, 100]^d$
F9	Griewank	$\boldsymbol{x} \in [-600, 600]^d$
F10	HappyCat	$\boldsymbol{x} \in [-2, 2]^d$
F11	HGBat	$\boldsymbol{x} \in [-2, 2]^d$
F12	Hyper Ellipsoid	$\boldsymbol{x} \in [-5.12, 5.12]^d$
F13	Katsuura	$\boldsymbol{x} \in [-5, 5]^d$
F14	Michalewicz	$\boldsymbol{x} \in [0, \pi]^d$
F15	Norwegian	$\boldsymbol{x} \in [-1.1, 1.1]^d$
F16	Quadric	$\boldsymbol{x} \in [-100, 100]^d$
F17	Quartic	$\boldsymbol{x} \in [-1.28, 1.28]^d$
F18	Rastrigin	$\boldsymbol{x} \in [-5.12, 5.12]^d$
F19	Rosenbrock	$\boldsymbol{x} \in [-2.048, 2.048]^d$
F20	Salomon	$\boldsymbol{x} \in [-100, 100]^d$
F21	Schaffer 6	$\boldsymbol{x} \in [-100, 100]^d$
F22	Schwefel 2.21	$\boldsymbol{x} \in [-100, 100]^d$
F23	Schwefel 2.22	$\boldsymbol{x} \in [-10, 10]^d$
F24	Shubert	$\boldsymbol{x} \in [-10, 10]^d$
F25	Spherical	$\boldsymbol{x} \in [-5.12, 5.12]^d$
F26	Step	$\boldsymbol{x} \in [-20, 20]^d$
F27	Vincent	$\boldsymbol{x} \in [0.25, 10]^d$
F28	Weierstrass	$\boldsymbol{x} \in [-0.5, 0.5]^d$

V. Empirical Results and Discussion

This section presents the results of the experiments described in section IV.

For each parameter configuration Mann-Whitney U tests, using a confidence level of 95%, were performed to determine, whether a given parameter configuration resulted in a PSO that was in fact better or worse than a random search, or if the PSO showed no statistical difference with the performance of the random search. This information is summarized into four categories for each test case:

- CP_BR: The percentage of parameters that are theoretically stable that resulted in the PSO performing better than random search.
- CP_NDR: The percentage of parameters that are theoretically stable that resulted in the PSO performing with no statistical difference to random search.
- DP_BR: The percentage of parameters that are theoretically unstable that resulted in the PSO performing better than random search.
- DP_NDR: The percentage of parameters that are theoretically unstable that resulted in the PSO performing with no statistical difference to random search.

For each dimensionality tested, a table summarizing the performance information is given. In which the performance results for 500, 1000, 2000, and 5000 iterations are reported.

The first thing to note is that, over all the results shown in tables II, III, and IV for 5, 10, and 30 dimensions respectively, the percentage of unstable parameter choices that were able to outperform random search were significantly low. The highest percentage of unstable parameter configurations able to outperform random search was 34.67% for Griewank in 5 dimensions, at 500 iterations. The performance on Griewank decreases with a increase in iteration count, down to 27.2% at 5000 iterations. The exact parameter configurations failing to beat random search can be seen clearly in figure 2. What is interesting is that all the unstable parameter configurations that did in fact outperform random search are in a region which appears to be a natural extension of the stability boundary of equation (4), specifically a region with a slightly larger spectral radius. Similar results can be seen for both Rosenbrock and Solomon in 5-dimensions at 5000 iterations as illustrated figures 4 and 3 respectively.

It might appear as if there is a fair degree of tolerance on the stability boundary of equation (4). However, a quick scan of tables II, III, and IV, shows numerous cases where the performance of unstable parameters is terrible, for example, only 6.18% of unstable parameters where able to outperform random search on Michalewicz in 30 dimensions at 1000 iterations, whereas 100% of stable parameter configurations outperformed random search. It can be seen in figure 5 that there is nearly a perfect relationship between convergent parameter configurations and the ability of the PSO to outperform random search when optimizing Michalewicz.

There are actually two objective functions that seem to illustrate how finely tuned PSO sometimes needs to be, namely Egg Holder and Elliptic. Over all tested dimensions even stable parameter configurations were more often than not outperformed by random search. This requirement of fine tuning can be seen in figure 6 for the Elliptic objective function in 5 dimensions at 5000. What is worth noting in figure 6 is that there are some unstable parameter configurations that outperform some stable parameters. However, all unstable parameter configurations that were successful are still near the apex of the stable region of equation (4). The performance of PSO was very poor for Egg Holder, with only less than 12% of parameter configurations (stable and unstable) outperforming random search in 5 dimensions at 500 iterations, and decreasing to less than 11%. Looking at figure 7 it very clear that only a very small number of parameter configurations are effective at optimizing the Egg Holder objective function. But, what is interesting is that once again the optimal parameter configurations appear to be clustered around the boundary of the stable region, however in the case of the Egg Holder objective function most are just slightly outside the stable region.

There is a very clear trend throughout all the data, namely that stable parameter configurations improve as the dimensionality of the problem increases. This behavior is observed on all tested objective functions except of the Happy Cat objective function. This is not surprising since a random search's performance is likely to degrade quickly as dimensionality increase. However, what is interesting is that the performance across nearly all the objective functions worsens for unstable parameter configurations. This implies that the higher the dimensionality, the more important selecting stable parameter configurations becomes.

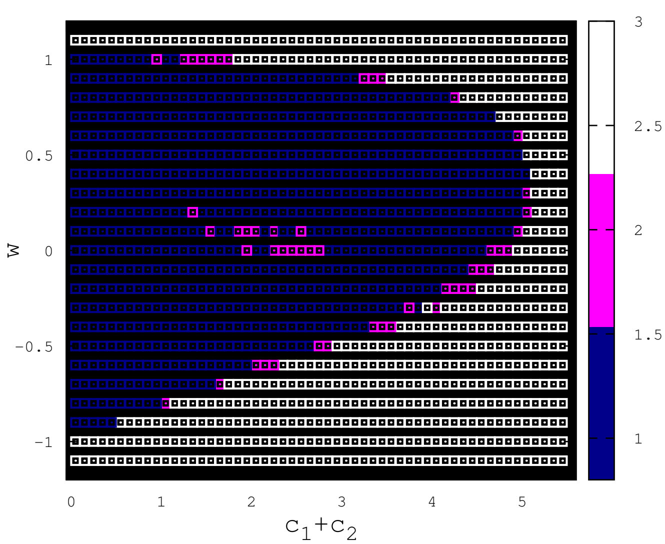

Fig. 2: Griewank, 5-dimensions, 5000 iterations
1 = performed better than random search, 2 = no statistical difference, 3 = random search performed better

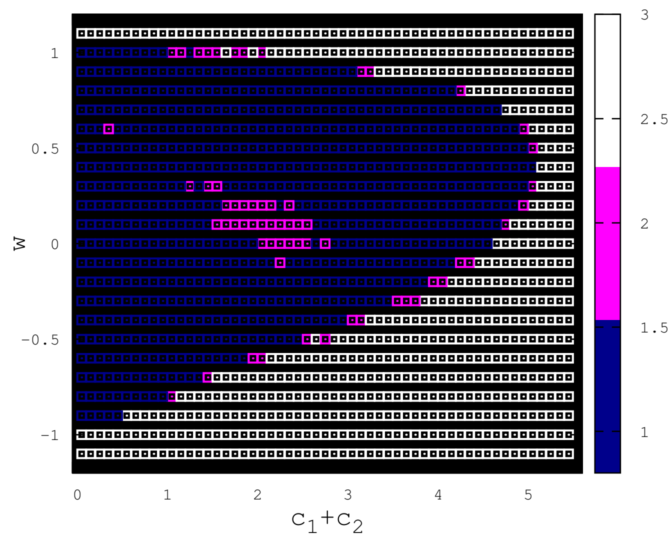

Fig. 3: Salomon, 5-dimensions, 5000 iterations
1 = performed better than random search, 2 = no statistical difference, 3 = random search performed better

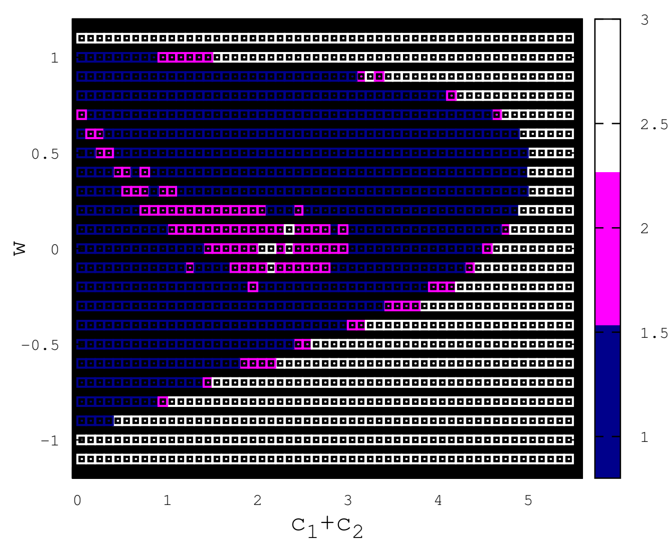

Fig. 4: Rosenbrock, 5-dimensions, 5000 iterations
1 = performed better than random search, 2 = no statistical difference, 3 = random search performed better

VI. Conclusion

It was shown that the majority of parameter configurations that are theoretically unstable perform worse than random search on all objective functions tested. It was also shown that there is a degree of tolerance from which parameters can be selected just outside of the convergent region without extreme performance degradation. However, the degree to which parameter values can be selected outside of the stable region is very problem dependent. For most tested

TABLE II: Performance of PSO versus Random Search: 5-Dimensions

	500 Iterations				1000 Iterations				2000 Iterations				5000 Iterations			
f	S_BR	S_NDR	US_BR	US_NDR	SP_BR	SP_NDR	US_BR	US_NDR	S_BR	S_NDR	US_BR	US_NDR	S_BR	S_NDR	US_BR	US_NDR
F1	100.00	0.00	32.72	9.46	100.00	0.00	30.62	8.67	100.00	0.00	29.70	4.99	99.21	0.79	27.07	4.86
F2	100.00	0.00	32.59	9.99	100.00	0.00	30.09	8.67	99.80	0.20	28.12	6.70	97.62	2.38	26.68	5.52
F3	98.21	1.79	21.81	4.73	86.31	13.49	20.24	3.42	79.17	20.24	19.97	2.89	72.42	24.40	19.97	2.50
F4	100.00	0.00	32.19	9.72	99.80	0.20	29.96	6.83	98.21	1.79	26.94	5.52	95.44	4.56	25.10	5.52
F5	41.87	50.20	20.50	4.34	35.32	31.15	20.24	2.89	30.75	15.28	19.58	2.23	28.97	9.52	19.84	1.58
F6	3.77	8.93	7.88	6.57	1.39	6.94	7.62	6.96	0.60	5.16	9.20	5.26	0.20	2.98	10.38	4.34
F7	30.75	5.95	17.87	2.76	30.56	5.36	18.40	1.45	29.37	6.15	18.27	1.18	28.97	4.56	18.40	0.92
F8	98.81	1.19	31.14	8.67	95.63	4.17	27.46	6.83	92.66	6.94	26.68	4.73	87.10	10.52	24.57	3.94
F9	100.00	0.00	34.69	9.72	100.00	0.00	30.62	6.96	99.40	0.60	29.30	5.78	97.22	2.78	27.20	4.86
F10	86.90	13.10	19.19	3.81	85.32	14.29	19.45	3.15	77.18	22.02	19.05	2.50	68.85	29.56	19.05	1.58
F11	14.48	41.87	13.67	6.57	9.52	20.24	14.19	4.20	8.53	9.52	14.32	3.29	6.94	5.95	14.98	2.50
F12	100.00	0.00	32.46	7.75	99.21	0.79	30.62	6.70	98.61	1.39	28.78	5.65	94.05	5.75	26.54	5.39
F13	100.00	0.00	28.78	6.70	100.00	0.00	26.02	5.78	99.40	0.60	24.84	4.99	92.06	7.94	24.05	3.29
F14	82.14	17.86	22.08	4.99	69.84	26.39	21.29	3.15	65.28	27.78	21.29	2.37	55.95	24.01	20.50	2.37
F15	29.56	29.96	15.24	5.39	29.96	27.58	16.82	2.89	27.58	24.60	17.74	1.31	25.60	21.63	17.87	1.58
F16	99.21	0.79	28.91	6.04	96.43	3.57	26.81	4.99	90.48	9.52	25.76	4.99	76.39	23.02	24.84	3.68
F17	98.41	1.59	29.57	8.54	95.63	3.97	27.99	5.12	94.64	4.76	27.07	3.81	88.49	9.52	24.97	3.81
F18	98.21	1.79	22.08	5.52	95.44	4.56	21.02	4.47	92.46	5.95	20.11	3.42	82.54	13.89	19.58	1.84
F19	99.40	0.60	28.25	8.94	97.42	2.58	26.81	5.65	94.44	5.56	25.49	4.73	84.92	14.09	23.39	3.94
F20	100.00	0.00	31.67	10.64	99.80	0.20	29.04	6.96	97.62	2.38	26.54	5.91	94.05	5.95	24.70	4.07
F21	100.00	0.00	19.32	3.42	99.80	0.20	18.92	2.50	97.02	2.98	18.92	1.97	85.91	14.09	18.79	1.84
F22	96.63	3.37	29.96	8.80	92.46	6.75	26.68	6.18	90.08	8.33	25.23	5.26	86.90	9.92	24.44	3.94
F23	100.00	0.00	31.80	8.67	100.00	0.00	29.30	6.96	99.21	0.79	27.60	4.60	96.03	3.97	25.49	4.20
F24	61.31	24.60	16.69	2.37	53.57	16.27	17.21	1.31	51.19	14.48	17.48	1.31	43.06	12.30	17.48	0.92
F25	100.00	0.00	32.72	9.59	99.21	0.79	30.35	7.49	97.62	2.38	28.91	5.91	95.24	4.37	26.94	4.60
F26	100.00	0.00	33.77	8.54	98.81	1.19	31.01	7.36	97.22	2.78	28.91	6.18	94.05	5.95	26.68	3.94
F27	68.65	16.07	19.45	2.89	66.87	11.71	19.19	2.50	65.48	9.13	19.19	1.84	63.29	5.75	18.66	1.58
F28	100.00	0.00	29.30	9.59	100.00	0.00	28.52	5.91	100.00	0.00	27.60	4.60	100.00	0.00	25.49	4.07

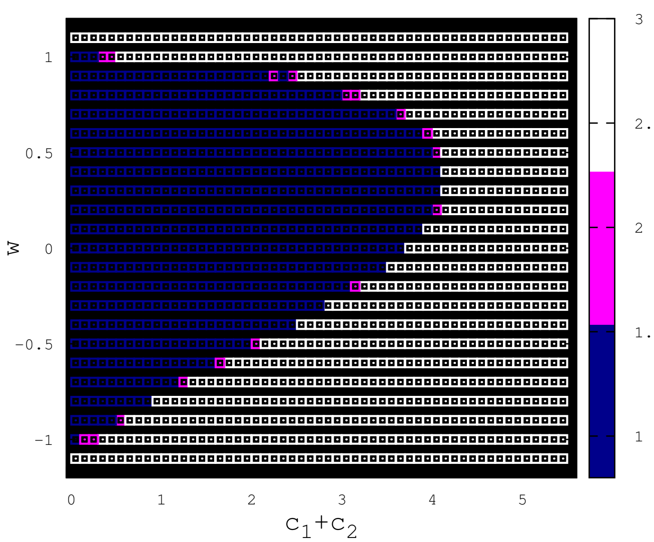

Fig. 5: Michalewicz, 30-dimensions, 1000 iterations
1 = performed better than random search, 2 = no statistical difference, 3 = random search performed better

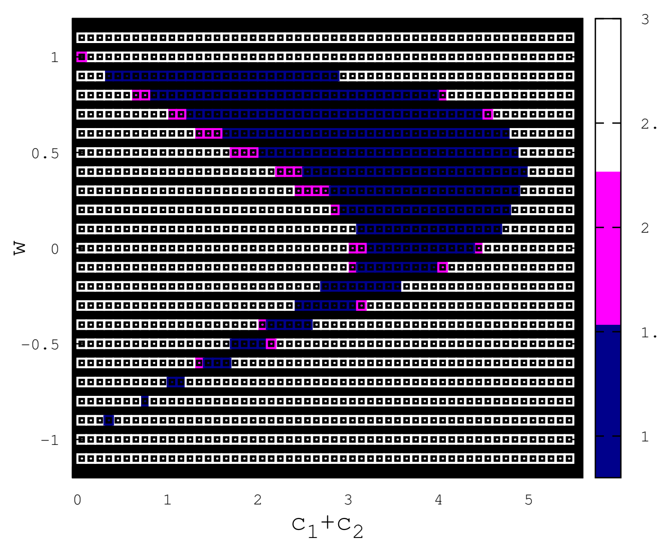

Fig. 6: Elliptic, 5-dimensions, 5000 iterations
1 = performed better than random search, 2 = no statistical difference, 3 = random search performed better

cases it was found that convergent parameter configurations drastically increased the chance of outperforming random search. However, selecting parameter values within the stable region near the apex was the best strategy to ensure that PSO was always superior to random search. It was also observed that the higher the dimensionality of the problem, the more important selecting stable parameter configurations became.

Future work could include an investigation into defining a region between two spectral radius values that encompasses the region around the apex where the PSO outperforms random search in all the tested objective function. While this region may be problem dependent to a degree, there appears

TABLE III: Performance of PSO versus Random Search: 10-Dimensions

	500 Iterations				1000 Iterations				2000 Iterations				5000 Iterations			
f	S_BR	S_NDR	US_BR	US_NDR	SP_BR	SP_NDR	US_BR	US_NDR	S_BR	S_NDR	US_BR	US_NDR	S_BR	S_NDR	US_BR	US_NDR
F1	100.00	0.00	33.25	9.33	100.00	0.00	30.22	8.28	100.00	0.00	27.33	6.04	100.00	0.00	25.10	4.86
F2	100.00	0.00	29.43	7.36	100.00	0.00	27.73	5.65	100.00	0.00	26.15	4.99	100.00	0.00	24.70	4.73
F3	100.00	0.00	19.45	6.70	100.00	0.00	18.40	4.47	100.00	0.00	17.35	3.94	100.00	0.00	16.95	2.50
F4	100.00	0.00	27.99	7.62	100.00	0.00	25.62	6.04	100.00	0.00	24.57	5.39	98.81	1.19	23.52	3.55
F5	85.91	14.09	16.95	4.20	59.52	37.70	16.03	2.76	44.05	34.13	15.90	1.97	36.11	20.04	15.77	1.71
F6	23.61	21.23	9.99	1.58	13.89	17.06	10.38	1.71	11.51	13.69	11.70	1.31	7.74	7.34	11.83	1.84
F7	29.96	8.13	12.88	1.71	28.37	6.75	13.53	1.05	27.18	5.75	13.80	1.31	25.20	6.15	14.72	0.66
F8	100.00	0.00	27.60	7.49	99.80	0.20	25.62	6.44	99.60	0.40	25.10	6.04	96.83	2.98	22.60	4.20
F9	100.00	0.00	32.19	8.02	100.00	0.00	29.43	7.49	100.00	0.00	28.91	5.26	100.00	0.00	26.54	5.39
F10	72.62	20.04	14.32	3.55	53.97	34.92	13.80	2.76	44.05	38.49	13.80	3.15	36.71	36.31	14.72	2.23
F11	100.00	0.00	25.89	8.94	99.60	0.40	20.89	7.62	98.61	1.39	18.92	6.04	48.21	51.59	15.77	3.81
F12	100.00	0.00	29.57	8.02	100.00	0.00	27.46	5.91	100.00	0.00	25.76	5.26	100.00	0.00	24.84	4.20
F13	100.00	0.00	22.08	5.91	100.00	0.00	20.76	5.12	100.00	0.00	19.19	4.34	100.00	0.00	19.58	3.29
F14	98.21	1.79	15.77	3.68	96.63	3.37	16.43	2.50	90.08	9.52	15.77	2.10	79.17	18.65	15.77	1.58
F15	68.06	9.52	14.98	3.42	67.66	5.75	14.98	2.76	67.06	3.57	15.37	1.84	66.87	2.98	15.77	1.18
F16	100.00	0.00	25.76	6.04	100.00	0.00	24.70	5.26	100.00	0.00	22.86	5.26	98.61	1.39	21.81	2.89
F17	100.00	0.00	25.89	6.96	99.80	0.20	24.31	5.65	98.81	1.19	23.13	4.73	94.44	5.16	21.55	2.76
F18	100.00	0.00	18.79	6.70	100.00	0.00	18.40	4.86	100.00	0.00	17.48	3.81	98.61	1.39	17.08	1.84
F19	100.00	0.00	26.81	7.62	100.00	0.00	23.78	6.31	100.00	0.00	23.26	5.78	99.21	0.79	22.34	3.42
F20	100.00	0.00	29.17	10.38	100.00	0.00	27.07	6.44	100.00	0.00	26.54	5.39	100.00	0.00	23.26	4.86
F21	100.00	0.00	13.27	3.02	100.00	0.00	13.80	1.97	100.00	0.00	14.19	1.71	100.00	0.00	13.80	2.76
F22	99.60	0.40	24.18	6.31	98.61	1.39	22.86	4.73	96.03	3.97	21.81	4.20	94.05	5.75	20.37	5.26
F23	100.00	0.00	24.84	6.83	100.00	0.00	22.86	5.78	100.00	0.00	22.21	4.20	100.00	0.00	21.81	3.42
F24	69.25	23.61	11.83	2.10	66.67	21.83	12.75	1.31	62.30	14.29	13.01	1.58	53.57	13.10	13.80	1.45
F25	100.00	0.00	30.75	8.94	100.00	0.00	29.04	6.96	100.00	0.00	26.15	5.91	100.00	0.00	24.57	5.39
F26	100.00	0.00	31.27	7.36	100.00	0.00	30.22	5.52	100.00	0.00	27.07	5.78	100.00	0.00	24.44	4.99
F27	70.24	11.31	14.06	2.76	68.65	11.11	14.59	2.63	66.47	10.12	15.51	1.45	63.89	7.34	15.51	1.18
F28	100.00	0.00	28.91	8.80	100.00	0.00	26.81	6.57	100.00	0.00	24.57	6.44	100.00	0.00	22.60	4.73

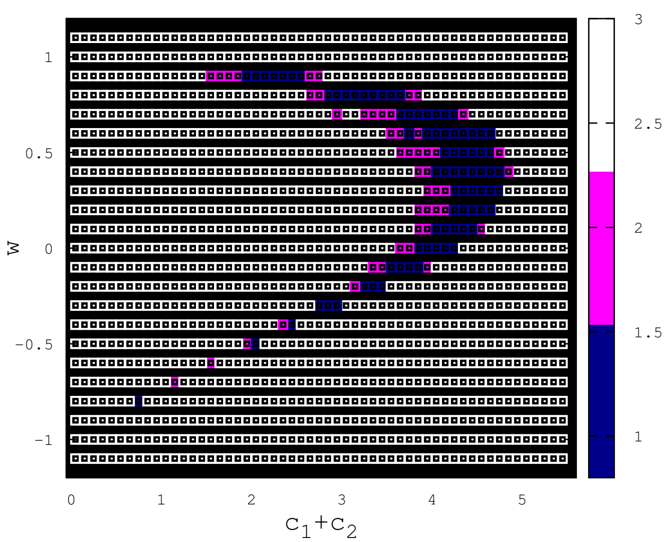

Fig. 7: Egg Holder, 5-dimensions, 5000 iterations
1 = performed better than random search, 2 = no statistical difference, 3 = random search performed better

to be enough of a pattern to warrant further investigation.

REFERENCES

[1] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Joint Conference on Neural Networks*. Piscataway, NJ: IEEE Press, 1995, pp. 1942–1948.

[2] R. Poli, "Analysis of the publications on the applications of particle swarm optimisation," *Journal of Artificial Evolution and Applications*, vol. 2008, pp. 1–10, 2008.

[3] R. Poli and D. Broomhead, "Exact analysis of the sampling distribution for the canonical particle swarm optimiser and its convergence during stagnation," in *Proceedings of the Genetic and Evolutionary Computation Conference*. New York, NY: ACM Press, 2007, pp. 134–141.

[4] R. Poli, "Mean and variance of the sampling distribution of particle swarm optimizers during stagnation," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 4, pp. 712–721, 2009.

[5] T. Blackwell, "A study of collapse in bare bones particle swarm optimzation," *IEEE Transactions on Evolutionary Computation*, vol. 16, no. 3, pp. 354–372, 2012.

[6] C. Cleghorn and A. Engelbrecht, "A generalized theoretical deterministic particle swarm model," *Swarm Intelligence*, vol. 8, no. 1, pp. 35–59, 2014.

[7] Q. Liu, "Order-2 stability analysis of particle swarm optimization," *Evolutionary Computation*, vol. doi: 10.1162, pp. 1–30, 2014.

[8] M. Bonyadi and Z. Michalewicz, "Stability analysis of the particle swarm optimization without stagnation assumption," *IEEE Transactions on Evolutionary Computation*, vol. PP, no. 1, pp. 1–7, 2015.

[9] Y. Shi and R. Eberhart, "A modified particle swarm optimizer," in *Proceedings of the IEEE Congress on Evolutionary Computation*. Piscataway, NJ: IEEE Press, 1998, pp. 69–73.

[10] J. Kennedy, "Small worlds and mega-minds: effects of neighborhood topology on particle swarm performance," in *Proceedings of the IEEE Congress on Evolutionary Computation*, vol. 3. Piscataway, NJ: IEEE Press, 1999, pp. 1931–1938.

[11] J. Kennedy and R. Mendes, "Population structure and particle performance," in *Proceedings of the IEEE Congress on Evolutionary Computation*. Piscataway, NJ: IEEE Press, 2002, pp. 1671–1676.

[12] A. Engelbrecht, "Particle swarm optimization: Global best or local best," in *Proceedings of the 1st BRICS Countries Congress on Computational Intelligence*. Piscataway, NJ: IEEE Press, 2013, pp. 124–135.

[13] C. Cleghorn and A. Engelbrecht, "Particle swarm convergence: An empirical investigation," in *Proceedings of the Congress on Evolutionary Computation*. Piscataway, NJ: IEEE Press, 2014, pp. 2524–2530.

TABLE IV: Performance of PSO versus Random Search: 30-Dimensions

	500 Iterations				1000 Iterations				2000 Iterations				5000 Iterations			
f	S_BR	S_NDR	US_BR	US_NDR	SP_BR	SP_NDR	US_BR	US_NDR	S_BR	S_NDR	US_BR	US_NDR	S_BR	S_NDR	US_BR	US_NDR
F1	100.00	0.00	30.75	4.73	100.00	0.00	28.38	6.44	100.00	0.00	26.54	7.49	100.00	0.00	24.31	6.31
F2	100.00	0.00	28.91	5.65	100.00	0.00	27.60	5.78	100.00	0.00	26.94	6.18	100.00	0.00	24.97	5.12
F3	100.00	0.00	15.64	6.18	100.00	0.00	14.98	4.73	100.00	0.00	14.45	3.68	100.00	0.00	13.80	3.81
F4	100.00	0.00	32.59	4.20	100.00	0.00	30.49	4.86	100.00	0.00	26.94	6.57	100.00	0.00	24.70	6.31
F5	100.00	0.00	7.10	3.68	99.01	0.99	7.49	3.02	94.44	5.56	7.75	2.76	80.95	18.85	8.41	2.37
F6	52.18	7.34	2.76	0.53	50.79	7.14	3.55	0.92	48.21	6.55	4.60	1.05	41.67	9.33	6.04	0.79
F7	32.54	6.55	3.42	0.92	30.36	6.75	3.81	1.58	29.56	5.16	4.73	1.18	26.79	4.76	5.78	0.79
F8	100.00	0.00	29.43	5.78	100.00	0.00	27.46	6.04	100.00	0.00	25.36	6.04	100.00	0.00	23.13	5.65
F9	100.00	0.00	30.88	5.91	100.00	0.00	31.01	4.86	100.00	0.00	30.22	5.26	100.00	0.00	27.86	5.91
F10	67.26	10.32	4.34	2.37	66.07	10.71	5.26	2.76	48.41	25.79	3.68	4.60	24.01	46.23	2.37	6.18
F11	100.00	0.00	32.33	4.34	100.00	0.00	30.09	5.26	100.00	0.00	28.12	6.31	100.00	0.00	26.81	6.04
F12	100.00	0.00	28.25	6.57	100.00	0.00	27.99	5.78	100.00	0.00	25.36	7.10	100.00	0.00	23.26	5.26
F13	100.00	0.00	16.16	6.96	100.00	0.00	15.90	5.39	100.00	0.00	14.32	4.73	100.00	0.00	13.67	3.81
F14	100.00	0.00	5.91	2.76	100.00	0.00	6.18	2.23	100.00	0.00	7.36	1.71	100.00	0.00	7.36	1.58
F15	99.80	0.20	9.20	3.55	100.00	0.00	9.46	3.02	100.00	0.00	8.80	3.55	99.80	0.20	9.33	2.50
F16	100.00	0.00	23.39	7.62	100.00	0.00	21.42	6.83	100.00	0.00	20.63	6.96	100.00	0.00	19.05	4.73
F17	100.00	0.00	22.86	6.70	100.00	0.00	21.68	5.91	100.00	0.00	20.63	5.52	100.00	0.00	19.71	4.47
F18	100.00	0.00	18.27	6.44	100.00	0.00	17.21	5.26	100.00	0.00	16.29	5.52	100.00	0.00	15.64	4.34
F19	100.00	0.00	27.86	5.78	100.00	0.00	25.49	6.70	100.00	0.00	24.05	6.96	100.00	0.00	22.73	6.44
F20	100.00	0.00	31.27	4.99	100.00	0.00	29.96	5.12	100.00	0.00	28.12	5.91	100.00	0.00	25.62	6.31
F21	99.01	0.79	4.20	2.50	99.80	0.20	4.73	2.50	100.00	0.00	5.12	2.63	100.00	0.00	6.18	1.58
F22	100.00	0.00	22.86	5.65	100.00	0.00	20.63	6.70	100.00	0.00	19.84	6.44	100.00	0.00	18.13	4.47
F23	100.00	0.00	17.61	6.44	100.00	0.00	16.82	5.39	100.00	0.00	16.69	4.07	100.00	0.00	15.77	2.89
F24	75.20	19.64	3.42	0.66	73.21	18.45	3.94	1.05	71.23	17.86	4.99	0.92	65.08	15.48	6.57	0.53
F25	100.00	0.00	30.75	6.31	100.00	0.00	30.22	5.78	100.00	0.00	27.46	5.65	100.00	0.00	27.33	5.78
F26	100.00	0.00	31.41	5.65	100.00	0.00	29.96	5.78	100.00	0.00	28.91	4.73	100.00	0.00	27.20	5.65
F27	78.97	17.26	5.65	1.97	75.99	16.07	6.31	1.97	73.41	14.48	6.96	1.45	68.65	11.11	7.88	1.31
F28	100.00	0.00	26.28	7.23	100.00	0.00	23.26	7.23	100.00	0.00	21.81	5.52	100.00	0.00	20.76	5.52

[14] A. Engelbrecht, “Particle swarm optimization: Global best or local best,” in *Proceedings of the 1st BRICS Countries Congress on Computational Intelligence*. Piscataway, NJ: IEEE Press, 2013, pp. 124–135.

[15] K. Malan, “Characterising continuous optimisation problems for particle swarm optimisation performance prediction,” *PhD thesis, Department of Computer Science, University of Pretoria, Pretoria, South Africa*, 2014.

[16] J. Liang, B. Qu, and P. Suganthan, “Problem definitions and evaluation criteria for the CEC 2014 special session and competition on single objective real-parameter numerical optimization,” Computational Intelligence Laboratory, Zhengzhou University and Nanyang Technological University, Tech. Rep. 201311, 2013.