Understanding Parameter Spaces using Local Optima Networks: A Case Study on Particle Swarm Optimization

Christopher W Cleghorn
School of Computer Science and Applied Mathematics
University of Witwatersrand
Johannesburg, South Africa
christopher.cleghorn@wits.ac.za

Gabriela Ochoa
Computing Science and Mathematics
University of Stirling
Stirling, Scotland, UK
gabriela.ochoa@cs.stir.ac.uk

ABSTRACT

A major challenge with utilizing a metaheuristic is finding optimal or near optimal parameters for a given problem instance. It is well known that the best performing control parameters are often problem dependent, with poorly chosen parameters even leading to algorithm failure. What is not obvious is how strongly the complexity of the parameter landscape itself is coupled with the underlying objective function the metaheuristic is attempting to solve. In this paper local optima networks (LONs) are utilized to visualize and analyze the parameter landscapes of particle swarm optimization (PSO) over an array of objective functions. It was found that the structure of the parameter landscape is affected by the underlying objective function, and in some cases by a considerable degree across multiple metrics. Furthermore, despite PSO's parameter landscape having a relatively simple macro structure, the LONs demonstrate that there is actually a considerable amount of complexity at a micro level; making parameter tuning harder for PSO than would have been initially thought. Apart from the PSO specific findings this paper also provides a formalism of parameter landscapes and demonstrates that LONs can be used as an effective tool in the analysis and visualization of parameter landscapes of metaheuristics.

CCS CONCEPTS

• **Computing methodologies** → **Artificial intelligence**; • **Theory of computation** → *Theory of randomized search heuristics.*

KEYWORDS

Local Optima Networks, Particle Swarm Optimization, Parameter Tuning, Fitness landscape analysis

ACM Reference Format:
Christopher W Cleghorn and Gabriela Ochoa. 2021. Understanding Parameter Spaces using Local Optima Networks: A Case Study on Particle Swarm Optimization. In *2021 Genetic and Evolutionary Computation Conference Companion (GECCO '21 Companion), July 10–14, 2021, Lille, France.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3449726.3463145

1 INTRODUCTION

A major challenge for a practitioner is the selection of effective control parameters for a metaheuristic. While there are often recommended control parameters for metaheuristics, the effectiveness on unseen problems is generally not that predictable, with the best performance of a metaheuristic requiring at least some degree of parameter turning. In an attempt to reduce the burden on the practitioner a number of approaches to parameter tuning have been proposed, a small subset of the fundamental ones are [1–3, 10, 12, 13]. While existing parameter tuning approaches have been effective, they tend to utilize a uniform approach to the parameter tuning, which ignores the underlying structure induced by the specific metaheuristic, leading to a potential loss in either optimality or efficiency. It is for this reason that gaining a better understanding of the structure of the space that parameter tuning operates on is needed.

In this paper we propose a formalism for parameter landscapes and demonstrate that local optima networks (LONs) can be effectively utilized to visualize and analyze parameter landscapes of metaheuristics, with the focus placed on the parameter landscapes of the particle swarm optimization (PSO) algorithm, specifically the inertia PSO [19].

We start with formally defining fitness landscapes and parameter landscapes in section 2. A brief description of PSO is given in section 3. A description of LONs and monotonic LONs is presented in section 4, as well as the alterations to the standard LON construction process used. The experimental setup is presented in section 7, followed by the experimental results, visualizations, and a discussion thereof in section 8. Section 9 presents a summary of the findings of this paper as well as potential areas of future work.

2 FITNESS AND PARAMETER LANDSCAPE DEFINITION

In order to better frame the focus of this paper it is worth formally defining what is meant by a parameter landscape, and how it differs, if only in form, from a traditional fitness landscape. To this end, a fitness landscape is first defined.

Definition 2.1. A **Fitness Landscape**, in a discrete search space context[1], is a triplet (S, V, f) where S is the set of elements in the search space, $V : S \rightarrow \mathcal{P}(S)$ maps each element, $s \in S$ to a set of neighbours $V(s) \subseteq S$, where $\mathcal{P}$ denotes the power set, and $f : S \rightarrow \mathbb{R}$ defines the fitness of the element $s \in S$.

A parameter landscape is inherently a type of fitness landscape where the fitness is the algorithm's performance. However, an explicit definition of a parameter landscape is worth creating as it allows for better disentanglement between the algorithmic performance and the fitness of a specific candidate solution on the

© ACM, 2021. This is the author's version of the work. It is posted here by permission of ACM for your personal use. Not for redistribution. The definitive version was published in GECCO '21, July 10–14, 2021, Lille, France 2021. ACM ISBN 978-1-4503-8350-9/21/07. https://doi.org/10.1145/3449639.3459288

[1]The definition can be extended to a continuous search space, but it is not needed for the purposes of the paper

underlying objective function. Specifically, the following definition is now introduced.

Definition 2.2. A **Parameter Landscape**, in a discrete search space context[1], is the tuple (P, f, D, A, C, V_c) where f is the underlying objective function being optimized over domain D by algorithm A, which is configured with control parameters, $c \in C$. The neighborhood function $V_c : C \rightarrow \mathcal{P}(C)$ maps each element, $c \in C$ to a set of neighbors $V_c(C) \subseteq C$. Lastly, P is a performance measure, $P_{f,A,D}(c)$, and is utilized to quantify how performant algorithm A was at optimizing objective function f on domain D using a configuration $c \in C$.

There are a number of reasonable ways to design the performance measure, P, particularly in the presence of a stochastic optimizer, as is typically the case with metaheuristics. For the purposes of this paper P is treated as a distribution, which is sampled for each repeated execution of algorithm A using a configuration $c \in C$. The performance of A would also clearly depend on the stopping condition used, adding further parameterization to P itself. However, for the purposes of this paper a fixed function evaluation budget as well as a fixed repeated execution count are used.

The LON variants described in section 4 are defined in terms of the standard fitness landscape triplet, it is therefore worth formalizing, for the sake of clarity, the linkage between the fitness landscape triplet and a parameter landscape from definition 2.2. The search space of a parameter landscape, (P, f, D, A, C, V_c) is the set C, the fitness value is the performance, $P_{f,A,D}(c)$ of configuration c of algorithm A over domain D on objective function f.

3 PARTICLE SWARM OPTIMIZATION

Particle swarm optimization (PSO) was originally inspired by the complex movement of birds in a flock. The variant of PSO this paper focuses on uses the inertia coefficient, as proposed by Shi and Eberhart [19], which is referred to as PSO from now on.

The PSO algorithm is defined as follows: let $f : \mathbb{R}^d \rightarrow \mathbb{R}$ 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 $\boldsymbol{x} \in \mathbb{R}^d$, $f(\boldsymbol{o}) \leq f(\boldsymbol{x})$. 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 $r_{1,k}(t), r_{2,k}(t) \sim U(0,1)$ for all t and $1 \leq k \leq d$. 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 full algorithm description as well as more details on possible neighborhood typologies choices can be found here [4].

4 LOCAL OPTIMA NETWORKS

Local optima networks were proposed by Ochoa *et al* [14], inspired by the work of Doye and Massen [9], as an effective tool for both the analysis and visualization of the global structure of a fitness landscape. A detailed overview of the utility of LONs on an array of combinatorial optimization problems can be found here [16]. In this paper the standard LON graph, the monotonic LON (MLON) graph [15], and the compressed monotonic LON (CMLON) [15] are used for analyzing parameter landscapes. In order to define both the LON and MLON graphs some intermediate definition are required, which are given in terms of the standard fitness landscape triplet (S, V, f). The following definitions contain minor alterations to the originals to allow for a more lenient treatment of continuous values given the floating point representational limits of the IEEE-754 standard used by modern CPUs. Specifically, an ϵ term is introduced, when ϵ is set to zero the original definitions are restored. Furthermore, minimization is assumed from this point onwards.

Definition 4.1. The floating point inequality, $x \leq_\epsilon y$, is true when $x < y$ or $|x - y| < \epsilon$, where $\epsilon \in \mathbb{R}^+$ is a sufficiently small value [2].

Definition 4.2. s^* is a *local optima*, in the context of minimization, if $\forall s \in V(s^*)$, $f(s^*) \leq_\epsilon f(s)$. The inequality is intentionally not strict, in order to cater for the neutral landscape case.

Definition 4.3. The *LON nodes* are the set, $L_{opt} = \{l_1, l_2 \cdots, l_n\}$, of all local optima under definition 4.2.

Definition 4.4. The *basin of attraction* of local optima, l_i, is defined as $b_i = \{s \in S \mid h(s) = l_i \text{ with probability } p_i(s) > 0\}$, where $h(s)$ represents the final state of a best-improvement hill-climber from the start position s. The basin's size is defined as $|b_i| = \sum_{s \in S} p_i(s)$.

Definition 4.5. The set of *LON edges*, E_{lon}, are all the weighted edges $w_{i,j}$ between local optima l_i and l_j where $w_{i,j}$ represents the probability of ending up in the basin of attraction b_j after perturbing l_i, by M random moves, and applying h. Specifically, $w_{i,j} = p(l_i \rightarrow b_j)$, where the probability of moving from any $s \in S$ to a basin b_j is calculated as $p(s \rightarrow b_j) = \sum_{s' \in b_j} p(s \rightarrow s') p_j(s')$ with $p(s \rightarrow s') = P(s' \in \{z \mid d(z,s) \leq M\})$, and d an appropriate distance metric.

Definition 4.6. A *LON* is the weighed graph $LON = (L_{opt}, E_{lon})$, with L_{opt} and E_{lon} defined in definitions 4.3 and 4.5 respectively.

Definition 4.7. An *MLON* is the weighed graph $MLON = (L_{opt}, E_{mlon})$, where $E_{mlon} = \{w_{i,j} \in E_{lon} \mid f(l_j) \leq_\epsilon f(i_i)\}$. Namely, only directed edges represent a non-deterioration in the fitness are included in the MLON graph.

Definition 4.8. A *CMLON* is the weighed graph $CMLON = (CL_{opt}, E_{cmlon})$, where the nodes $cl_i \in Cl_{opt}$ are the MLON plateaus. The MLON plateaus are derived from connected subgraph of *MLON* where each node has the equal fitness, and where equality is defined as $f(l_i) =_\epsilon f(l_j)$ when $|f(l_i) - f(l_j)| < \epsilon$. Weighted edges in the CMLON are aggregated for the edges of nodes in the MLON plateau.

[2]The value $\epsilon = 10^{-5}$ was used for this paper.

5 RELATED WORK

To the best of our knowledge, the properties of metaheuristic parameter landscapes have been investigated by only two previous articles. The first [18], investigates the effect on the landscape structure of varying individual numerical parameters while fixing the remaining parameters at optimized values. The authors consider different algorithm configuration scenarios, including three widely studied, NP-hard problems (SAT, MIP and TSP), 6 prominent algorithms for these and 5 well-known instance sets. Two basic landscape metrics were considered, the number of local minima, and the fitness distance correlation. They found that parameter landscapes on instance sets tend to be uni-modal and convex. However, for single instances, these responses are more rugged than their aggregate counterparts. This study only considers single-parameter slices, and indeed the authors indicate that their results do not preclude the possibility of complex parameter interactions that result in configuration landscapes with many local optima, leaving this to future work.

The second article to look at metaheuristic parameter landscapes [20], does consider parameter interactions and uses fitness distance correlation and sampled local optima networks to analyse and visualize the global structure of the landscapes. The study constructs the LON models from data extracted while running the ParamILS automatic algorithm configuration framework [11]. The metaheuristic used is a standard GA for solving five continuous benchmark problems. The search space consists of three parameters (crossover rate, mutation rate and population size) each covering a discretized range of possible values. The results indicate that the configuration spaces of interacting parameters of the same algorithm when solving different instances can vary widely. The landscapes show many local optima and, with the exception of the simplest Sphere benchmark function, the landscapes show multiple funnels. Funnels can be loosely defined as groups of local optima which are close in configuration space within a group, but well-separated between groups. They can make optimization harder.

The contribution of this paper is to continue with the use of LONs to analyze and visualize parameter landscapes where parameters interact. However, instead of sampling the space as it was done in [20], we conduct a full enumeration of two PSO parameters in a discretized range. PSO is more suitable as an optimizer in the continuous domain than a standard GA, and we covered a wider range of benchmark functions than those explored in [20].

6 DEFINING THE PARAMETER REGION UNDER CONSIDERATION

For the purposes of the paper the tuple $c = (c_1, c_2, w)$ represents a PSO configuration, where the configuration set, C, is derived from the following intervals:

$$w \in [-1.2, 1.2] \text{ and } c_1 + c_2 \in (0, 5] \tag{3}$$

where $c_1 = c_2$, with a sample point every 0.1 along w and $c_1 + c_2$. This results in a set of $|C| = 1250$ possible configurations. The region was selected to contain the parameter region that would result in order-2 stability of PSO particles, where order-2 stability is defined as convergence in expectation and variance [5]. It was shown by Cleghorn and Englebrecht [6] that the best performing configurations are near the boundary of the parameter region that would result in order-2 stability of PSO particles. Specifically, the following region is necessary and sufficient for order-2 stability of PSO particles:

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

with $c_1 = c_2$, as originally derived by Poli [17] and under minimal modeling assumption by Cleghorn and Engelbrecht [7]. A derivation without the restricted relationship between c_1 and c_2 was derived by Cleghorn and Stapelberg [8].

7 EXPERIMENTAL SETUP

A total of 26 well known benchmark functions are used in this paper, listed in table 1. For each objective function a parameter landscape is generated using PSO as A, and the objective function's domain as D. In order to obtain $P_{f,A,D}(c)$, for all $c \in C$, a PSO is run for 30 trials using an iteration based stopping condition of 2000, and a population size of 20, in 30 dimensions. In order to directly utilize the LON model, the expected value of the performance, $E[P_{f,A,D}(c)]$ is used (and estimated from the 30 trials) as the "fitness". The neighborhood function, V_c, allows step sizes of 0.1 provided the new configuration is still in C as defined in section 6.

Table 1: Objective functions

	Function name	Modality	Separable	Domain
F1	Ackley	Multi	No	$[32.768, 32.768]^n$
F2	Alpine	Multi	Yes	$[-10, 10]^n$
F3	Bent-Cigar	Uni	No	$[-100, 100]^n$
F4	Discus	Uni	No	$[-100, 100]^n$
F5	Egg_Holder	Multi	No	$[-512, 512]^n$
F6	Elliptic	Uni	Yes	$[-100, 100]^n$
F7	Expanded-G+R	Multi	No	$[-100, 100]^n$
F8	Griewank	Multi	No	$[-600, 600]^n$
F9	HappyCat	Multi	No	$[-2, 2]^n$
F10	HGBat	Multi	No	$[-2, 2]^n$
F11	Hyper-Ellipsoid	Uni	Yes	$[-5.12, 5.12]^n$
F12	Katsuura	Multi	Yes	$[-5, 5]^n$
F13	Michalewicz	Multi	Yes	$[0, \pi]^n$
F14	Norwegian	Multi	No	$[-1.1, 1.1]^n$
F15	Quadric	Uni	No	$[-100, 100]^n$
F16	Quartic	Uni	Yes	$[-1.28, 1.28]^n$
F17	Rastrigin	Multi	Yes	$[-5.12, 5.12]^n$
F18	Rosenbrock	Multi	No	$[-30, 30]^n$
F19	Salomon	Multi	No	$[-100, 100]^n$
F20	Schaffer-6	Multi	No	$[-100, 100]^n$
F21	Schwefel-2-21	Uni	Yes	$[-100, 100]^n$
F22	Schwefel-2-22	Uni	Yes	$[-10, 10]^n$
F23	Spherical	Uni	Yes	$[-5.12, 5.12]^n$
F24	Step	Multi	Yes	$[-100, 100]^n$
F25	Vincent	Multi	Yes	$[0.25, 10]^n$
F26	Weierstrass	Multi	Yes	$[-0.5, 0.5]^n$

For each objective function a LON, MLON, and a CMLON are constructed from the PSO's parameter landscape. From the constructed LONs the following metrics are derived:

- The number of local optima and edges within the LON, denoted as $\|L_{opt}\|$ and $\|E_{lon}\|$ respectively, where $\|\cdot\|$ represents the cardinality of the set.
- The number of self-loops within the LON, $\|Self(LON)\|$, where $Self(LON)$ is the set of self-loops in the LON.
- The number of globally optimal configurations within the LON, $\|G_\epsilon(L_{opt})\|$, where $G_\epsilon(LON)$ is the set of globally optimal configurations within an ϵ fitness threshold of the best configuration.
- The ratio of improving edges to all not self-loop edges, $I_{ratio} = \|I_\epsilon(LON)\|/(\|E_{lon}\| - \|Self(LON)\|)$, where $I_\epsilon(LON)$ is the set of edges that represent a transition improvement greater than an ϵ fitness threshold.
- The ratio of neutral edges to all not self-loop edges, $N_{ratio} = \|N_\epsilon(LON)\|/(\|E_{lon}\| - \|Self(LON)\|)$, where $N_\epsilon(LON)$ is the set of edges that represent a neutral transition improvement with an ϵ fitness threshold.
- The ratio of worsening edges to all not self-loop edges, $W_{ratio} = \|W_\epsilon(LON)\|/(\|E_{lon}\| - \|Self(LON)\|)$, where $W_\epsilon(LON)$ is the set of edges that represent a transition worsening greater with an ϵ fitness threshold.

From the constructed CMLONs the following metrics are derived:

- The number of local optima and edges within the CMLON, denoted as $\|CL_{opt}\|$ and $\|E_{cmlon}\|$ respectively.
- The number of sinks within the CMLON, denoted as $n_{sink} = \|S(CMLON)\|$, where $S(CMLON)$ is the set of nodes from $CMLON$ that have zero outgoing edges.
- The number of globally optimal sinks within the CMLON, denoted as $n_{globalsink} = \|G_\epsilon(S(CMLON))\|$, where $G_\epsilon(S(CMLON))$ is the set of globally optimal sinks within an ϵ fitness threshold of the best configuration.
- The funnel size of the global optimum sinks, denoted as $\|Funnel_\epsilon(CMLON)\|$, where $Funnel_\epsilon(CMLON)$ is the set of nodes that have a path in the CMLON to one or more of the globally optimum sinks [15]. $F_r = \|Funnel_\epsilon(CMLON)\|/\|CL_{opt}\|$ is the relative funnel size, which is reported to allows for a better comparison of the funnel size over objective functions.
- The aggregate incoming strength for optimal sinks (assuming ϵ equality) and the aggregate incoming strength of suboptimal sinks, which are denoted as $Str_{optimal}$ and $Str_{suboptimal}$ respectively.
- The compression percentage, $C_\epsilon(MLON)$, in terms of node count reduction, obtained when constructing the CMLON from the MLON under the floating point inequality from definition 4.1. The smaller the compression percentage is the less neutrality is present in the parameter landscape.
- The average fitness difference between all sub-optimal sinks and the global sink(s) as well as the corresponding standard deviation, which is denoted as $AveDiff_{Fitness}$.

8 EXPERIMENTAL RESULTS AND DISCUSSION

The recorded LON and CMLON metrics over the 26 objective functions under consideration are presented in tables 2 and 3 respectively. Additionally, for four interesting example objective functions, Ackley (F1), Egg-holder (F5), Hyper-Ellipsoid (F11), and Quartic (F16), a 3D visualization of the parameter surface, as well as 2D network visualization of the LON and CMLON models of the corresponding PSO parameter landscapes are presented in figures 2, 3, 4, and 5 respectively. The legend used for both LON and CMLON figures is show in figure 1.

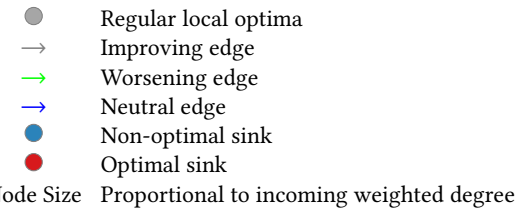
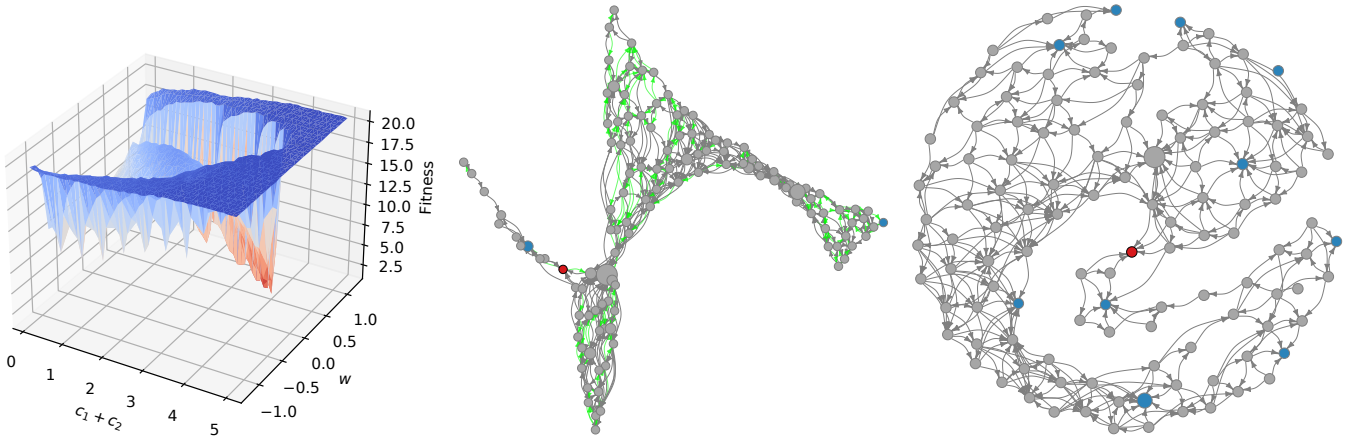


Figure 1: LON and CMLON network visualization legend.

The first notable finding, shown in table 2, is that the number of local optima is surprisingly high. The highest number of local optima present in the parameter landscape was found with the F9 objective function, with 187 local optima (14.96% of the search space). Even at the lower end of local optima count, where F7 and F26 tied, 109 local optima were still present (8.72% of the search space). This relatively high local optima count is already suggestive of a landscape that could easily trap a naive parameter tuning approach. Furthermore, the 6.21% range in number of local optima across the objective functions points to the fact that the underlying objective function is influencing the parameter landscape. The subsequent LON metrics will better serve at indicating how impactful these changes are on the fundamental properties of the parameter landscape.

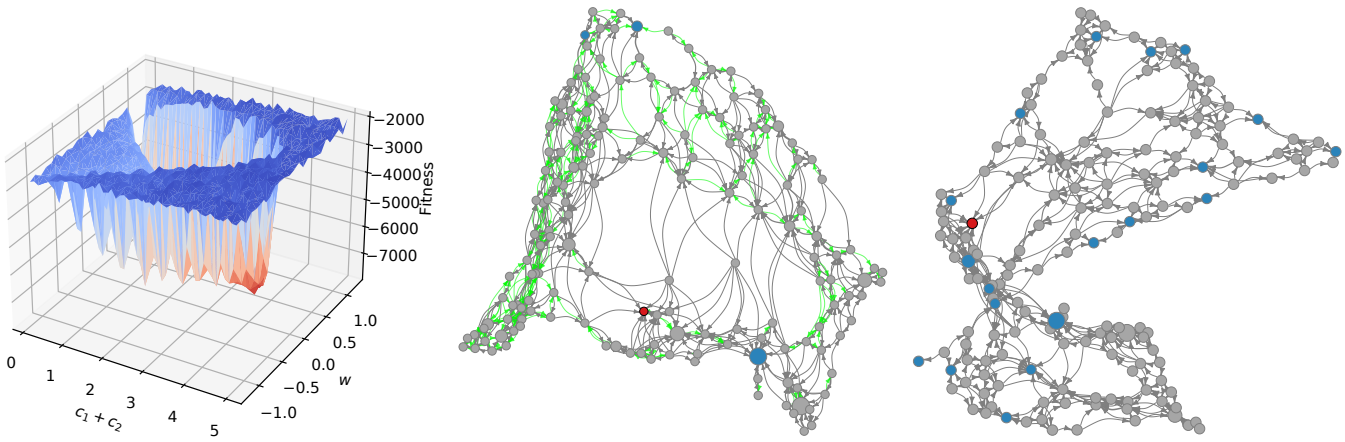
One noteworthy feature of the PSO parameter landscape, is the low level of neutrality. Specifically, the N_{ratio} metrics is 0 for 20 of the 26 functions in table 2. The low neutrality is also evident with how low the compression percentage, $C_\epsilon(MLON)$, is when constructing the CMLON from the MNLON models as seen in table 3.

It was also found that for the vast majority of objective functions there was only one globally optimal node, $\|G_\epsilon(LON)\|$, in the LON model (20 out of 26 objective function). However, there are some clear outliers with F16 having 42 globally optimal nodes within the LON model. What is interesting is that in the case of objective functions that resulted in multiple globally optimal nodes (F4, F11, F15, F16, F23, and F24), the corresponding CMLON model had only one globally optimal node. The presence of only one globally optimal node in the CMLON models indicate that while there may have been multiple globally optimal nodes in the LON they were in fact neutrally connected to each other (there existed at least one neutral edge connecting them), and not spread out over the search space. Of the objective functions that resulted in multiple globally optimal nodes in the LON model, all but one of the underlying continuous objective functions was unimodal, with the one multimodal



(a) 3D Plot of Ackley. (b) LON of Ackley. (c) CMLON of Ackley.

Figure 2: PSO parameter landscape plots for the Ackley objective function (F1) in 30-dimensions.



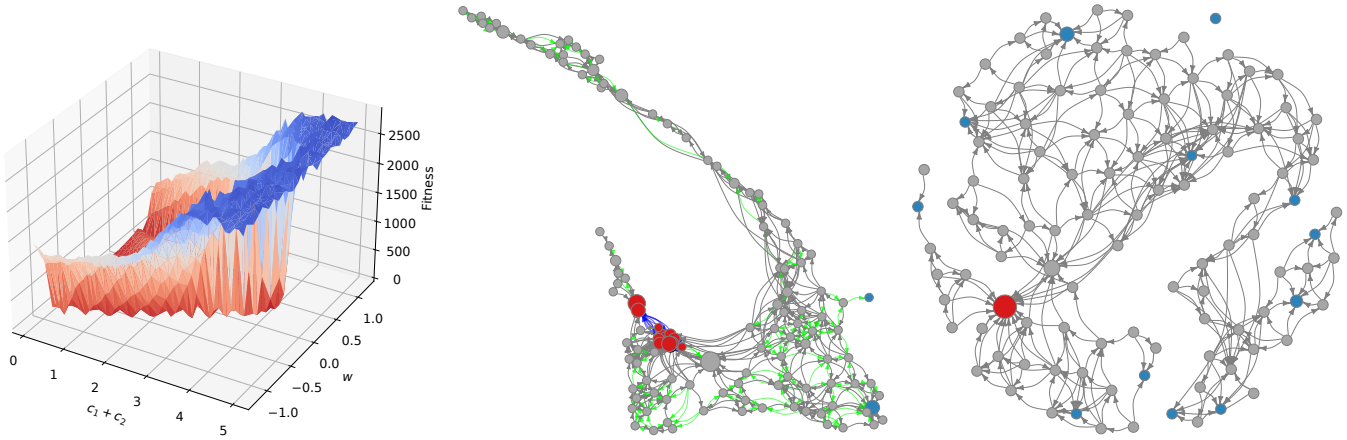
(a) 3D Plot of Egg-Holder. (b) LON of Egg-Holder. (c) CMLON of Egg-Holder.

Figure 3: PSO parameter landscape plots for the Egg-Holder objective function (F5) in 30-dimensions.

underlying continuous objective function (F24) only having two global optima. The finding that the PSO parameter landscape had only one global sink, as seen in table 3, held across all considered objective functions. This finding suggests that the single global sink structure is likely an interesting characteristic of the PSO itself that appears invariant to the underlying objective function.

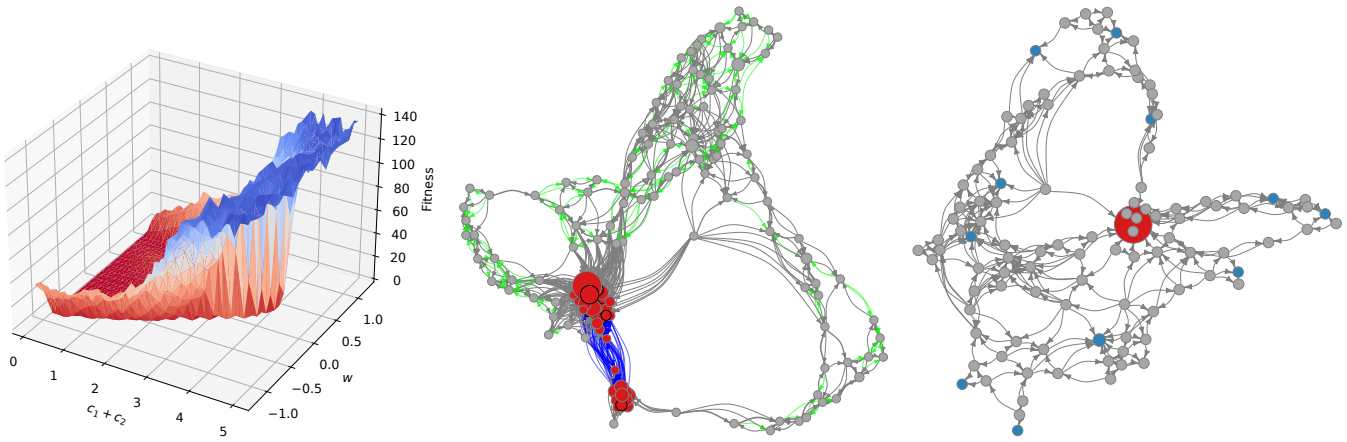
The CMLON metrics provide us with a good indication of how searchable the underlying parameter landscape is. Specifically, in table 3 there are numerous non-optimal sinks present for all objective functions, implying that there are a number of "traps" in the search space were even a non-naive tuner could easily get stuck. The number of non-optimal sinks is dependant on the underlying objective functions, with F1 and F16 having the least (11) and F20 and F25 having the most (23). The presence of non-optimal sinks is in principle only a problem, from a practical perspective, if there is a meaningfully degradation in performance between the non-optimal configurations and the globally optimal ones. However, upon inspection of the $AveDiff_{Fitness}$ it is clear that there is in fact a meaningful average degradation when selecting a parameter configuration that corresponds to a non-optimal sink as apposed to one from an optimal sink. Moreover, there is also a substantial amount of variance on the $AveDiff_{Fitness}$ metrics, indicating that there is a large range of differing performance levels that a tuner could get stuck at.

The next question worth asking is, for a given objective function, how likely are we to end up at the globally optimal sink(s). The F_r metric provides some illumination in this regard. Specifically, for 23 of the 26 objective functions the funnel size of the global sink contains over 30% of all the CMLON nodes, with F8 having the largest percentage, 52.13% of CMLON nodes within the global sink's funnel. This large funnel size is somewhat expected after



(a) 3D Plot of Hyper-Ellipsoid. (b) LON of Hyper-Ellipsoid. (c) CMLON of Hyper-Ellipsoid.

Figure 4: PSO parameter landscape plots for the Hyper-Ellipsoid objective function (F11) in 30-dimensions.



(a) 3D Plot of Quartic. (b) LON of Quartic. (c) CMLON of Quartic.

Figure 5: PSO parameter landscape plots for the Quartic objective function (F16) in 30-dimensions.

inspecting the 3D plots in figures 2a, 3a, 4a, and 5a. Specifically, in each case, there is a large sunken connected area, which corresponds, roughly, to the the order-2 stability criteria of equation (4). Unfortunately, the $Str_{optimal}$ metrics is rather low across all the objective functions considered, with an average value of only 0.0986. The low $Str_{optimal}$ metrics implies that despite the large funnel sizes the chances of moving towards the globally optimal sink is far lower than the chances of moving to a sub-optimal sink.

All these observations provide strong evidence that the parameter landscape of PSO is surprisingly intricate and multimodal in nature. This observation is in contrast to the findings of Pushak and Hoos [18], in which the parameter space of the considered metaheuristics were simple. There are two reasons for this discrepancy. The first is that in the work of Pushak and Hoos their analysis was done over a set of problem instances as opposed to focusing on problem specific parameter landscapes. Secondly, they did not cater for parameter interdependence if multiple were being tuned simultaneously (listed as future work). As such one of the big findings of this paper, is that the parameter space of metaheuristics, in the case of this paper PSO, can actually be a considerably challenging search space when considering multiple parameters and per-instance tuning. The per-instance tuning part of the findings, is rather practically pertinent, as most real world problems being solved are not sets of problems, but rather an important singular problem that needs solving.

Apart from the complexity of the found parameter landscapes, it is clear that while there are some constant trends across the PSO parameter landscapes, there is also a considerable amount of variability. For example, in the CMLON of the Egg-Holder function (F5), shown in figure 3c, there are many non-optimal sinks, with most having larger incoming strength than the globally optimal sink. Whereas in the case of the Quartic function (F16), shown in figure

Table 2: PSO parameter landscape LON metrics over 26 functions

	$\|\|L_{opt}\|\|$	$\|\|G_\epsilon(LON)\|\|$	$\|\|E_{lon}\|\|$	$\|\|Self(LON)\|\|$	I_{ratio}	N_{ratio}	W_{ratio}
F1	121	1	675	121	0.700361	0	0.299639
F2	120	1	605	120	0.711340	0	0.288660
F3	120	1	678	120	0.698925	0	0.301075
F4	172	3	931	170	0.653088	0.006570	0.340342
F5	162	1	864	162	0.665242	0	0.334758
F6	156	1	832	156	0.644970	0	0.355030
F7	109	1	554	109	0.674157	0	0.325843
F8	117	1	635	117	0.710425	0	0.289575
F9	187	1	1063	187	0.699772	0	0.300228
F10	152	1	894	152	0.683288	0	0.316712
F11	122	14	728	108	0.617742	0.137097	0.245161
F12	133	1	706	133	0.732984	0	0.267016
F13	150	1	844	150	0.707493	0	0.292507
F14	154	1	865	153	0.689607	0	0.310393
F15	114	2	595	112	0.641822	0.004141	0.354037
F16	149	42	1311	107	0.398671	0.478405	0.122924
F17	134	1	780	134	0.710526	0	0.289474
F18	126	1	735	126	0.706076	0	0.293924
F19	114	1	606	114	0.729675	0	0.270325
F20	177	1	995	177	0.711491	0	0.288509
F21	127	1	705	127	0.66609	0	0.333910
F22	137	1	716	137	0.677029	0	0.322971
F23	118	19	678	101	0.604853	0.159445	0.235702
F24	112	2	603	112	0.665988	0.004073	0.329939
F25	151	1	901	151	0.690667	0	0.309333
F26	109	1	544	109	0.708046	0	0.291954

5c, the globally optimal sink has a considerably larger incoming strength than any of the sub-optimal sinks as can be seen by the large red node. Furthermore, the Quartic function has a large number of sinks connected by neutral edges in the LON model, shown in figure 5b, which is not at all the case for the egg-Holder function (F5), shown in figure 3b. Most of the presented metrics provide evidence that the optimal tuning approach will likely depend not only the type of metaheuristic but also the underlying problem. This observation makes it clear that the problem of automatic algorithm configuration will need to consider both components to be highly effective.

9 CONCLUSION

In this paper a formalism for investigating parameter landscapes was provided and it was demonstrated that local optima networks can be effectively used for the visualization and analysis the parameter landscapes of PSO. The approach used is sufficiently general to be directly applied to other metaheuristics.

It was found that, despite earlier research in the realm of parameter landscape analysis suggesting otherwise, that parameter landscapes can be of significant complexity. Namely, objective function specific parameter landscapes can be highly multimodal with multiple sub-optimal sinks. Furthermore, the performance difference between sub-optimal sinks and the globally optimal sinks was found to be surprisingly large with a high degree of variance.

In terms of PSO specific findings that were stable across objective functions, it was observed, in the CMLON models, that all PSO parameter landscapes had only a single global sink with a relatively large funnel size. However, the global sink typically had low incoming strength compared to the suboptimal sinks. Generally it was found that the PSO parameter landscape was fundamentally influenced by the underlying objective function, and that each landscape was far from trivial in terms of searchablity.

REFERENCES

[1] B. Adenso-Diaz and M. Laguna. 2006. Fine-tuning of algorithms using fractional experimental design and local search. *Operations Research* 54, 1 (2006), 99–114.

[2] T. Bartz-Beielstein, C. Lasarczyk, and M. Preuss. 2005. Sequential parameter optimization. In *Proceedings of the IEEE Congress on Evolutionary Computation*. IEEE Press, Piscataway, NJ, 773–780.

[3] M. Birattari, Z. Yuan, P. Balaprakash, and T. Stützle. 2010. *Experimental Methods for the Analysis of Optimization Algorithms. In: Bartz-Beielstein, T. Chiarandini,M. Paquete ,L. Preuss, M. (eds.) Experimental Methods for the Analysis of Optimization Algorithms.* Springer, Heidelberg, Berlin. 311–336 pages.

[4] M. R. Bonyadi and Z. Michalewicz. 2016. Particle swarm optimization for single objective continuous space problems: a review. *Evolutionary Computation* 25, 1 (2016), 1–54. https://doi.org/10.1162/EVCO_r_00180

[5] C. W. Cleghorn. 2019. Particle Swarm Optimization: Understanding Order-2 Stability Guarantees. In *Proceedings of International Conference on the Applications of Evolutionary Computation*. Springer International Publishing, Switzerland, 535–549.

[6] C. W. Cleghorn and A. P. Engelbrecht. 2016. Particle Swarm Optimizer: The Impact of Unstable Particles on Performance. In *Proceedings of the IEEE Symposium Series on Swarm Intelligence*. IEEE Press, Piscataway, NJ, 1–7.

[7] C. W. Cleghorn and A. P. Engelbrecht. 2018. Particle swarm stability: a theoretical extension using the non-stagnate distribution assumption. *Swarm Intelligence* 12, 1 (2018), 1–22.

Table 3: PSO parameter landscape CMLON metrics over 26 functions

	$\|CL_{opt}\|$	$\|E_{cmlon}\|$	n_{sink}	$n_{globalsink}$	F_r	$Str_{suboptimal}$	$Str_{optimal}$	$C_\epsilon(MLON)$	$AveDiff_{Fitness}$	
F1	121	388	11	1	0.396694	0.906250	0.093750	0	12.1405	± 6.60391
F2	120	345	15	1	0.466667	0.903987	0.096013	0	52.7797	± 21.4647
F3	120	390	15	1	0.466667	0.890080	0.109920	0	3.52E+10	± 3.04E+10
F4	170	495	22	1	0.358824	0.946685	0.053315	1.16279	1.41E+07	± 8.72E+06
F5	162	467	20	1	0.308642	0.963568	0.036432	0	4412.26	± 8121.83
F6	156	436	18	1	0.448718	0.958206	0.041795	0	9.45E+07	± 2.67E+08
F7	109	300	17	1	0.449541	0.929120	0.070880	0	1.93E+16	± 1.50E+16
F8	117	368	11	1	0.521368	0.882391	0.117609	0	250.092	± 288.223
F9	187	613	19	1	0.368984	0.944856	0.055144	0	0.545946	± 0.815667
F10	152	507	20	1	0.309211	0.931560	0.068440	0	2.86165	± 8.19476
F11	109	340	13	1	0.467890	0.777390	0.222610	10.656	1096.01	± 1232.78
F12	133	420	19	1	0.406015	0.953891	0.046109	0	2.26E+11	± 1.61E+11
F13	150	491	13	1	0.513333	0.801275	0.198725	0	15.8415	± 24.0928
F14	154	491	19	1	0.441558	0.934627	0.065373	0	0.54155	± 0.85233
F15	113	310	13	1	0.522124	0.882598	0.117402	0.87719	8.15E+06	± 7.91E+06
F16	108	319	11	1	0.333333	0.876352	0.123648	27.5170	71.3308	± 58.1645
F17	134	459	15	1	0.462687	0.876853	0.123147	0	211.545	± 184.810
F18	126	430	16	1	0.166667	0.957257	0.042743	0	3368.25	± 3537.38
F19	114	359	17	1	0.394737	0.813019	0.186981	0	14.1320	± 12.1696
F20	177	582	23	1	0.197740	0.948212	0.051788	0	1.85260	± 10.4551
F21	127	385	14	1	0.425197	0.956649	0.043351	0	58.1227	± 31.8725
F22	137	392	19	1	0.423358	0.935140	0.064860	0	3.97E+12	± 4.10E+12
F23	102	304	12	1	0.392157	0.868619	0.131381	13.5593	113.835	± 82.3507
F24	111	321	14	1	0.468468	0.780858	0.219142	0.89286	954.105	± 1165.47
F25	151	518	23	1	0.158940	0.915089	0.084911	0	17.9657	± 32.4323
F26	109	308	13	1	0.449541	0.902103	0.097897	0	39.1675	± 13.9839

[8] C. W. Cleghorn and B. Stapelberg. 2020. Particle Swarm Optimization: Stability Analysis using N-Informers under Arbitrary Coefficient Distributions. *ArXiv* abs/2004.00476 (2020).

[9] J. P. Doye and C. P. Massen. 2005. Characterizing the network topology of the energy landscapes of atomic clusters. *The Journal of Chemical Physics* 122, 8 (2005), 1–13.

[10] F. Hutter, H. H. Hoos, and K. Leyton-Brown. 2011. Sequential model-based optimization for general algorithm configuration. *Learning and Intelligent Optimization, (LION). LNCS* 6683 (2011), 507–523.

[11] Frank Hutter, Holger H. Hoos, Kevin Leyton-Brown, and Thomas Stützle. 2009. ParamILS: An Automatic Algorithm Configuration Framework. *J. Artif. Int. Res.* 36, 1 (Sept. 2009), 267–306.

[12] F. Hutter, H. H. Hoos, and T. Stützle. 2007. Automatic algorithm configuration based on local search. In *Proceedings of the Twenty-Second Conference on Artifical Intelligence.* AAAI Press/MIT Press, Menlo Park, CA, 1152–1157.

[13] M. López-Ibánez, J. Dubois-Lacoste, L. Cáceres, and M. Birattari. 2016. The irace package: Iterated racing for automatic algorithm configuration. *Operations Research Perspectives* 3 (2016), 43–58.

[14] G. Ochoa, M. Tomassini, S. Verel, and C. Verel. 2008. A study of NK landscapes' basins and local optima networks. In *Proceedings of the Genetic and Evolutionary Computation Conference.* ACM Press, New York, NY, 555–562.

[15] G. Ochoa, N. Veerapen, F. Daolio, and M. Tomassini. 2017. Understanding Phase Transitions with Local Optima Networks: Number Partitioning as a Case Study. *Evolutionary Computation in Combinatorial Optimization, (EVOCOP). LNCS* 10197 (2017), 233–248.

[16] G. Ochoa, S. Verel, F. Daolio, and Tomassini M. 2014. *Local optima networks: A new model of combinatorial fitness landscapes. In: Richter, H., Engelbrecht, A. (eds.) Recent Advances in the Theory and Application of Fitness Landscapes, Emergence, Complexity and Computation.* Vol. 6. Springer, Heidelberg, Berlin. 233–262 pages.

[17] R. Poli and D. Broomhead. 2007. 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.* ACM Press, New York, NY, 134–141.

[18] Y. Pushak and H. H. Hoos. 2018. Algorithm Configuration Landscapes: More Benign Than Expected?. In *Parallel Problem Solving from Nature - PPSN XV (Lecture Notes in Computer Science, Vol. 11102).* Springer, 271–283. https://doi.org/10.1007/978-3-319-99259-4_22

[19] Y. Shi and R. C. Eberhart. 1998. A Modified Particle Swarm Optimizer. In *Proceedings of the IEEE Congress on Evolutionary Computation.* IEEE Press, Piscataway, NJ, 69–73.

[20] German Treimun-Costa, Elizabeth Montero, Gabriela Ochoa, and Nicolás Rojas-Morales. 2020. Modelling parameter configuration spaces with local optima networks. In *Proceedings of the 2020 Genetic and Evolutionary Computation Conference.* ACM, New York, NY, USA, 751–759. https://doi.org/10.1145/3377930.3390199