Shaping Communities of Local Optima by Perturbation Strength

Sebastian Herrmann
University of Mainz
Department of Information Systems and Business Administration
Jakob-Welder-Weg 9, 55128 Mainz
s.herrmann@uni-mainz.de

Matthias Herrmann
University of Kaiserslautern
Microelectronic Systems Design Research Group
Erwin-Schrödinger-Straße, 67663 Kaiserslautern
herrmann@eit.uni-kl.de

Gabriela Ochoa
University of Stirling
Department of Computing Science and Mathematics
Stirling, FK9 4LA, Scotland
goc@cs.stir.ac.uk

Franz Rothlauf
University of Mainz
Department of Information Systems and Business Administration
Jakob-Welder-Weg 9, 55128 Mainz
rothlauf@uni-mainz.de

ABSTRACT

Recent work discovered that fitness landscapes induced by Iterated Local Search (ILS) may consist of multiple clusters, denoted as funnels or communities of local optima. Such studies exist only for perturbation operators (kicks) with low strength. We examine how different strengths of the ILS perturbation operator affect the number and size of clusters. We present an empirical study based on local optima networks from NK fitness landscapes. Our results show that a properly selected perturbation strength can help overcome the effect of ILS getting trapped in clusters of local optima. This has implications for designing effective ILS approaches in practice, where traditionally only small perturbations or complete restarts are applied, with the middle ground of intermediate perturbation strengths largely unexplored.

CCS CONCEPTS

•**General and reference** → **General conference proceedings;**

KEYWORDS

Fitness Landscapes, Local Optima Networks, Iterated Local Search, Perturbation Strength, Community Detection

ACM Reference format:
Sebastian Herrmann, Matthias Herrmann, Gabriela Ochoa, and Franz Rothlauf. 2017. Shaping Communities of Local Optima by Perturbation Strength. In *Proceedings of GECCO '17, Berlin, Germany, July 15-19, 2017,* 8 pages.
DOI: http://dx.doi.org/10.1145/3071178.3071243

1 INTRODUCTION

The concept of fitness landscapes is frequently used to examine the structure of problems and the behavior of metaheuristics [25, 33, 36]. This is especially relevant for understanding and predicting the performance of metaheuristic search on different problems. Ochoa et al. [27] introduced local optima networks (LONs), which allow us to examine the behavior of metaheuristics by network analysis. A local optima network is a representation of the stochastic process of a search algorithm in a particular problem instance. LONs have been used frequently to predict the performance of different metaheuristics [7, 9, 13, 16]. A number of studies indicate that local optima in fitness landscapes are distributed among multiple clusters. Such structures have been observed in the field of continuous optimization [20, 22, 24], denoted as "funnels". Earlier [5, 12], as well as recent work with local optima networks [28, 29], show that this phenomenon occurs in combinatorial optimization too.

A study by Herrmann et al. [14] shows that there is a strong relationship between the occurrence of a multi-cluster structure and the performance of iterated local search (ILS). The fraction of independent runs in which ILS finds the[1] global optimum (hit rate) correlates with the size of the cluster of the global optimum (peak cluster), and also with the total number of clusters. ILS is a search strategy which combines local search with a perturbation operator to overcome the situation where the algorithm gets stuck in a local optimum. A key factor of using ILS is the strength of the perturbation, which is the number of solution components that are modified at once [23]. Herrmann et al. used a fixed, low perturbation strength for their study. Thus, it is unclear how the strength of the perturbation operator influences the occurrence of a cluster structure. Here, we intend to overcome this limitation and present results from a follow-up experiment based on local optima networks. For our experiment, we study a large number of LONs with escape edges [40]. Escape edges reflect the perturbations of ILS. As fitness landscapes, we used instances of the Kauffman NK model [17, 19], which is a binary combinatorial optimization problem.

Another limitation in the literature is that we can only explain the hit rate of metaheuristics by network features of LONs (e.g. by the centrality of the global optimum [7, 9, 13, 16], or the number of

GECCO '17, Berlin, Germany

978-1-4503-4920-8/17/07…$15.00
DOI: http://dx.doi.org/10.1145/3071178.3071243

[1] Many optimization problems have multiple global optima. We assume that there is a single global optimum. The Kauffman NK model used in [14] and in our study has "a single sequence which is the only and global optimum" [19, p.218].

clusters [15]). Yet, no network feature is known which sufficiently explains another important measure of performance: the search costs necessary to find the global optimum. Our study also provides some indication that it is possible to predict this performance measure with LONs.

Our paper is structured as follows: Sections 2–3 describe the fundamentals[2]. In Section 2, we explain the concept of fitness landscapes and local optima networks with escape edges. Section 3 summarizes the principle of iterated local search. Section 4 describes our experimental setup, i.e. the Kauffman NK model, the methodology for analyzing local optima networks, and our approach to determine the performance of ILS empirically. Our results are presented and discussed in Section 5. A brief summary and our conclusions are in Section 6.

2 FITNESS LANDSCAPES & LOCAL OPTIMA NETWORKS

The notion of *fitness landscapes* was introduced to describe the dynamics of adaptation in nature [42]. Fitness landscapes are also well suited to describe the structure of problems and the dynamics of heuristic search in combinatorial optimization. Formally, a fitness landscape is a triplet $\{S, N, f\}$, where S is the search space, f is a fitness function, and $N(S)$ a neighborhood structure. The search space S contains all valid candidate solutions. The fitness function $f : S \rightarrow \mathbb{R}_{\geq 0}$ assigns a fitness value[3] to each $s \in S$. The neighborhood function $N : S \rightarrow \mathcal{P}(S)$ assigns a set of neighbors $N(s)$ to every $s \in S$. The neighborhood is usually defined as the set of solutions that are reachable by single applications of a given move operator.

Local search algorithms iteratively try to improve a solution by applying small changes. A simple implementation of local search is the best improvement hill climber as outlined by Algorithm 1.

The algorithm usually starts with a random solution. It scans the neighborhood of the current solution and selects the best neighbor with a superior fitness as the next solution. This procedure is repeated until no better neighbor is found. Then, the algorithm has reached a local optimum and terminates.

A *local optimum* is a solution with higher fitness than all its neighbors. A greedy search method moving from a solution to a neighboring solution and accepting only improvements, cannot overcome local optima [11]. A higher number of local optima leads to a landscape that is more "rugged", which often indicates a higher search difficulty for local search [41].

A local optimum is surrounded by a *basin of attraction*, i.e., the set of solutions that will converge to the given optimum after running hill climbing starting from them. The basin around a local optimum lo is defined as a function

$$B : lo \rightarrow \mathcal{P}(S \backslash LO) \tag{1}$$

which assigns an element from the set of all subsets (power set $\mathcal{P}$) over the solutions in the search space to each local optimum $lo \in LO$ (the set of all local optima).

[2]For the reader's convenience, we wanted this paper to be self-contained and as comprehensible as possible. In the introductory sections, we included descriptions and formal definitions following the explanations in [15].

[3]We assume that the fitness function returns non-negative values.

Algorithm 1 Best Improvement Hill Climbing (hillClimb)

Require: Solution space S,
Fitness function $f(S)$,
Neighborhood function $N(S)$,
Initial solution s_0

```
i ← 0
repeat
    choose x ∈ N(s_i) s.t. f(x) = max_{x∈N(s_i)}(f(x))
    if f(x) > f(s_i) then
        s_{i+1} ← x
    else
        s_{i+1} ← s_i
    end if
    i ← i + 1
until s_i is a local optimum: {x ∈ N(s_i) : f(x) > f(s_i)} = ∅
return s_i
```

A *local optima network* (LON; [27]) is a compressed representation of a fitness landscape that can be studied using the complex-network analysis framework. Complex networks have been used to study the structure and dynamics of systems arising in almost every aspect of our life, made of numerous interacting components [1, 4]. Studies on the dynamics in networks include the influence of nodes (centrality) as well as information flow and diffusion [6, 38]. LONs are a novel way to examine the trajectory of algorithms in fitness landscapes.

Mathematically, a network is just a graph $G = (V, E)$ with the set of vertices V and the set of edges E. In LONs, the vertex set V represents all local optima of the fitness landscape. An edge exists between two nodes (local optima), if there is a potential transition between them with a given search operator. The edges are directed and weighted. The edge weights $w_{x,y}$ represent the probability that a search algorithm moves from local optimum lo_x to a solution in the basin around lo_y, assuming that the current state is lo_x. Verel et al. [40] introduced the concept of escape edges, which are defined according to the distance function of the fitness landscape d (minimal number of moves between two solutions). An escape edge is defined as follows: there exists a directed edge e_{xy} (escape edge) from local optimum lo_x to lo_y if there is a solution s with

$$d(s, lo_x) = D \tag{2}$$

and

$$s \in B(lo_y). \tag{3}$$

The weight w_{xy} of edge e_{xy} is the probability that a search algorithm can escape from the local optimum lo_x into the basin around lo_y. The constant $D > 0$ determines the distance that is used for the attempt to escape from a local optimum basin. A LON with escape edges is a model to describe the stochastic process of iterated local search in a fitness landscape [13].

3 ITERATED LOCAL SEARCH

Iterated local search is a simple but successful search strategy. It operates by iteratively alternating between applying a move operator to the incumbent solution and restarting local search from the perturbed solution. The local search stage focuses on promising areas of the search space, whereas the perturbation stage diversifies and explores new areas [34]. This search principle has been rediscovered multiple times, within different research communities and with various names, such as the Chained Lin-Kernighan heuristic for the traveling salesman problem [2]. The term iterated local search (ILS) was proposed in [23]. ILS has been used in several local optima network studies, as the escape-edges closely resemble the dynamics of this metaheuristic [9, 13, 29, 40].

Algorithm 2 outlines the functionality of ILS. The process starts with a randomly selected solution $s_0 \in S$, and follows with a hill climbing procedure with best improvement as selection rule (algorithm 1): from the neighborhood $N(s)$, the solution with highest fitness is selected. The neighborhood $N(s)$ is the set of solutions that can be reached by performing an incremental change to s. The effort of this step depends on the size of the neighborhood $|N(s)|$, as it requires a scan of the whole neighborhood of s. This local search step (hill climbing) is repeated until it reaches a local optimum s^*, i.e no further improvement is possible. Then, ILS performs a diversification step by applying a perturbation with strength γ to the local optimum, resulting in s'. The strength is the number of solution components that are changed during this step, e.g. the number of bits that are randomly flipped. As a next step, hill climbing is started again from s', until the next local optimum $s^{*'}$ is reached. If the new local optimum $s^{*'}$ is different from the previous s^* and has higher fitness, the algorithm has "escaped" to a new local optimum, and the change is accepted. Otherwise, another perturbation is applied to s^*. This procedure is repeated until a termination condition is met, e.g. a fixed number of escapes without any further improvement.

Algorithm 2 Iterated Local Search (ILS)

```
Require: Solution space S,
    Fitness function f(S),
    Neighborhood function N(S),
    Perturbation strength γ,
    Stopping threshold t
Choose initial random solution s0 ∈ S
s* ← hillClimb(s0)
i ← 0
repeat
    s' ← perturbation(s*, γ)
    s*' ← hillClimb(s')
    if f(s*') > f(s*) then
        s* ← s*'
        i ← 0
    end if
    i ← i + 1
until i ≥ t
return s*
```

4 EXPERIMENTAL SETUP

4.1 Kauffman NK Model

For our experiments, we calculated the local optima networks for 1,200 instances of the Kauffman NK model [18]. The NK model is a combinatorial optimization problem from the class of pseudo-Boolean functions. An instance is defined by two parameters: N is the number of binary optimization variables, and K is the number of co-variables per variable. The size of the search space S is $|S| = 2^N$. The fitness function

$$f_{NK} : \{0,1\}^N \rightarrow [0.0, 1.0] \tag{4}$$

assigns a value to each combination of bits. The fitness $f_{NK}(s)$ of a solution s is the average of the values of N sub-functions (one for each bit). Each sub-function f_i assigns a fitness contribution for each bit i, depending on the value of bit i and K other bits (co-variables) that were randomly selected before instantiation:

$$f_i : \{0,1\}^{K+1} \rightarrow [0.0, 1.0]. \tag{5}$$

The parameter K determines the number of co-variables per optimization variable (epistasis). All values of the fitness function f_{NK} are normalized to values between 0.0 and 1.0, with $f_{NK}(s_{opt}) = 1$ as the fitness of the global optimum s_{opt}. In general, a higher value of K leads to a landscape that has more local optima and is more rugged [41]. Landscapes with higher values of K are more difficult for evolutionary algorithms [26]. The distance between two solutions $x, y \in S$ is calculated by the Hamming distance $d(x,y) = \sum_{i=0}^{n} |x_i - y_i|$, i.e., the number of bits that are set to different values when comparing two solutions.

We randomly generated $3x400$ NK fitness landscapes with $N = 15$ optimization variables and different values of $K \in \{3, 6, 9\}$. Thus, we have 400 problems instances each for the three levels of epistasis K. The size N of our problem instances is relatively low, since the computational effort for the experiments grows factorially by the problem size N (especially calculating the local optima networks is time-consuming).

4.2 Analysis of Local Optima Networks

For each problem instance, we extracted 15 local optima networks with escape edges, one for each value of the distance parameter $D = [1, 2, ..15]$. In order to study the influence of the distance D, we analyzed and compared the structural properties of the different LONs. We calculated the following measures:

- *#cl*: the number of clusters detected by a community detection algorithm (MCL, see below),
- cl_{peak}: the size of the global optimum's cluster (i.e., the number of nodes) over the total number of local optima,
- *rwdist*: the average random walk distance in the LON (considering edge weights) until the global optimum is found (the distance is the number of nodes traversed),
- *dens*: the graph density of the LON, i.e. the number of actual edges (arcs) by the number of potential edges.

To detect the clusters in the LONs, we used the **M**arkov **cl**uster algorithm (MCL [39]) for community detection, which has already been tested in earlier studies on LONs [8, 15]. *Community detection* is a relaxed variant of graph partitioning [37] and a rather exploratory method of network analysis. A community detection algorithm is free in determining the number of communities or the number of nodes per community. A very general definition of a community is a group of nodes that have more links among each other than to nodes in other communities. However, the definition of a community depends on the discipline applied and there exists a variety of algorithms that have been validated for different purposes [10, 31]. MCL is an appropriate method since it is based on the idea of stochastic flows in a network. This matches recent findings [13] that a LON represents the stochastic process of an algorithm in a particular fitness landscape.

Algorithm 3 Markov Cluster Algorithm (MCL)

Require: Graph G,
Power Parameter p,
Inflation Parameter r
$E = AdjacencyMatrix(G)$
repeat
  {Expand Matrix}
  $E = E^p$
  for all $A_i \in E$ **do**
    {Inflate Columns}
    $E_i = A_i^r$
    $E_i = normalizeVector(E_i)$
  end for
until E has converged
return E

A description of MCL is given in Algorithm 3. Let G be the graph of a LON. E is the adjacency matrix containing the weights of the directed edges in G. Since the edge weights are the non-negative probabilities to move from a given local optimum into the basin around another local optimum, the probabilities of all columns in E sum up to 1. Thus, E is a stochastic matrix, i.e., a transition matrix in a discrete-time Markov chain.

To identify communities in G, MCL applies two mechanisms: expansion and inflation. The expansion operator raises the adjacency matrix E to a non-negative power p. This ensures that different regions of the graph stay connected. The second mechanism is the inflation operator. Inflation raises each column E_i from the adjacency matrix E to a non-negative power r. This increases the weights of heavy-weighted edges, whereas the weights of low-weighted edges are reduced. Subsequently, a re-normalization ensures that each column again sums up to 1, which is a constraint for a stochastic matrix of a Markovian process. Both mechanisms are repeated until the algorithm converges, i.e., the transition matrix E reaches a steady state. For our NK landscapes with $N = 15$, this state is usually reached after $20 - 30$ iterations.

4.3 Determining Empirical Performance of ILS

We applied ILS to all our problem instances to study the behavior and performance by statistical analysis of a large sample. For each problem instance, we performed $15 \times 10,000$ independent runs. In each set of 10,000 runs, we used a fixed perturbation strength with $\gamma \in [1, 2, ..15]$. For the hill climbing steps procedure in ILS, we assumed that two solutions x, y are neighbors if their Hamming distance is equal to one ($d_{\max} = 1$). Thus, a local search step flips exactly one bit of the current solution.

In analogy to our network analysis, we determined the following empirical measures for each fitness landscape, and each value of γ:

- hit_{ils}: The fraction of independent runs which found the global optimum,
- $cost_{\text{ils}}$: The average number of fitness function evaluations used in those independent runs which found the global optimum[4],

We used a Java implementation to generate the NK landscapes, to extract the LONs, and to apply ILS. We ran the experiments on a cluster using 64 cores with 256 GB of RAM per node. The running time per fitness landscape was approx. 30 minutes in the case of low epistasis, and 1.5 hours in the case of high epistasis. We implemented the Markov cluster algorithm using *Numerical Python* [30]. Statistical analysis was done with *R* [32], visualizations of the networks in Figure 1 with *Gephi* [3].

5 RESULTS

5.1 Visual Inspection of the LONs and the Data

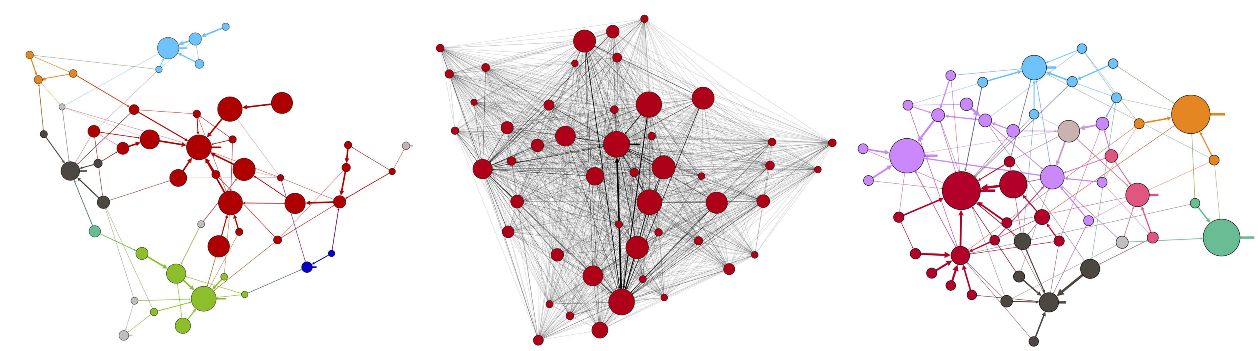
First, we take a look at our visualization of three LONs in Figure 1. All the three LONs have been extracted from the same NK fitness landscape, using different values for the escape distance D. The node color shows the clusters as obtained by community detection. Apparently, the number of clusters is high when D is low or high. An intermediate value of D leads to a LON in which all the local optima belong to the same, giant cluster (including the global optimum). Coincidentally, we observe a similar pattern for the number of edges: a LON extracted with low or high values of D has a sparse connection structure, whereas the network with $D \approx N/2$ exhibits a strongly interconnected structure with many edges. The lack of edges in the case where D is low or high offers a possible explanation for the occurrence of clusters: networks with few connections are more likely to decompose into several components. The data in Table 1 confirm our observations: for all values of K, the average network density $\langle dens \rangle$ first increases with D, reaching a maximum at $D = 8$ and then again decreases. The same holds for the average number of clusters $\langle \#cl \rangle$ and the average size of the global optimum's cluster $\langle cl_{\text{peak}} \rangle$.

A possible explanation for the curvilinear progression of the network density over the escape distance D is that the number of potentially possible bit flips follows a binomial distribution: $\binom{N}{D}$, with N as the number of binary optimization variables (here: $N = 15$), and D as the distance of the escape step used when computing the LON (here: $D \in [1, 2, ..15]$). This is an important finding, since the literature generally assumes that a stronger perturbation operator introduces more diversification [23], leading to a random search in the extreme cases. Our findings indicate that this does not necessarily apply to problems with a binary representation.

5.2 Hit Rate of ILS

Next, we study how the perturbation strength γ influences the probability to find the global optimum with ILS. We intend to explain the behavior of ILS by our results obtained from the analysis of the LONs. Figure 2 (top) shows the empirical average hit rate of ILS

[4]We did not consider the costs of the runs which did not find the global optimum. We used a large number of fitness function evaluations as a stopping criterion for ILS to simulate a theoretically unlimited search time available to the algorithm (10× the size of the search space). However, it is impossible to predict whether or not the algorithm will eventually terminate even after a large number of iterations. Thus, these runs had to be excluded from our cost statistics.



(a) LON with $D = 2$ (b) LON with $D = 8$ (c) LON with $D = 14$

Figure 1: Three LONs with escape edges, extracted from the same NK fitness landscape ($N = 15, K = 3$), but with different values for the escape distance D. Each node represents the basin around a local optimum. The nodes are colored by their community membership (the red clusters contain the global optimum). The node size represents the fitness of the local optimum. The networks have the highest graph density (number of edges by the number of potential edges) for $D \approx N/2$. The density decreases to both sides with a higher or lower value of D.

K	$\gamma = D$	$\langle hit_{\text{ils}} \rangle$	$\langle cost_{\text{ils}} \rangle$	$\langle \#cl \rangle$	$\langle cl_{\text{peak}} \rangle$	$\langle rwdist \rangle$	$\langle dens \rangle$
3	2	0.77	0.04	1.66	0.78	2.17	0.16
3	4	0.99	0.06	1.02	0.99	2.47	0.41
3	6	1.00	0.03	1.00	1.00	2.40	0.49
3	8	1.00	0.03	1.00	1.00	2.37	0.49
3	10	0.99	0.10	1.02	0.99	2.35	0.45
3	12	0.88	0.11	1.32	0.90	2.19	0.32
3	14	0.50	0.01	4.22	0.52	1.45	0.10
6	2	0.55	0.06	3.09	0.56	2.76	0.08
6	4	0.97	0.14	1.07	0.97	3.43	0.32
6	6	1.00	0.06	1.00	1.00	3.41	0.47
6	8	1.00	0.09	1.00	1.00	3.40	0.48
6	10	0.95	0.22	1.10	0.96	3.36	0.41
6	12	0.79	0.13	1.66	0.82	3.06	0.21
6	14	0.33	0.02	9.10	0.36	2.01	0.03
9	2	0.37	0.07	5.01	0.38	3.00	0.05
9	4	0.93	0.24	1.18	0.93	4.03	0.26
9	6	0.99	0.13	1.01	0.99	4.12	0.44
9	8	1.00	0.15	1.00	1.00	4.11	0.47
9	10	0.94	0.21	1.13	0.94	4.05	0.36
9	12	0.74	0.16	1.79	0.76	3.69	0.15
9	14	0.21	0.02	16.61	0.23	2.23	0.02

Table 1: For each combination of K and $\gamma = D$ (left columns), we show the data from the empirical performance of ILS (columns in the middle) and the data obtained from the analysis of the LONs (right columns). All values have been averaged across all landscapes, forming groups for all combinations of the parameter K and $\gamma = D$. To reduce the size of the table, we only show the data for even values of $\gamma = D$.

over the strength of the perturbation operator. We observe a distribution that exhibits a maximum for $\gamma \approx N/2$, and monotonously decreases towards high and low values of D.

For increasing K, the graph becomes more narrow, i.e. the average hit rate is constantly lower for any perturbation strength γ. In general, the hit rate is lower for higher values of K, which is as expected, since a higher K makes heuristic search more difficult (cf. Section 4.1). For intermediate values of γ, the average hit rates approach a plateau of 100% in all three cases. The curves in Figure 2 (bottom) show the average size of the global optimum's cluster and the number of clusters over the distance D in the LONs. We can observe a nearly identical shape of the curves. The multi-cluster structure disappears for intermediate values of D. Simultaneously, the average hit rate of ILS increases. Apparently, a lower number of clusters increases the chances to find the global optimum. On the other hand, a multi-cluster structure occurs when the perturbation operator is too weak (or too strong) to introduce enough diversification in order to scan the whole search space. The intermediately strong perturbation maximizes the diversification, connecting the landscape to a giant cluster and thus maximizing the chance to find the global optimum. The similarity between the global optimum's size and the hit rate is no surprise, as this close relationship has already been observed in [14].

Another observation is that the curves in Figure 2 are not perfectly symmetric, despite being an indirect consequence of the binomial distribution for the number of potential edges (arcs). For instance, we compare the global optimum's cluster size for $D = 1$ to its binomial counterpart[5], which is $D = N - 1 = 14$. We can see that all the values are higher for $D = 14$ (which holds for the clustering over γ, too). Even though we have not included further systematical statistical evidence, we can observe this effect for the other value pairs of D or γ that are binomial counterparts: a supposedly equally strong operator seems to achieve a slightly higher hit rate when it is chosen from the right side of the curve. As an explanation, we assume that a low perturbation strength kicks ILS into the same basin of attraction with a higher probability than a very strong operator. Thus, despite the nearly symmetric cluster structure, the hit rate is biased by properties of the landscape (e.g. locality).

[5] $\binom{N}{D} = \binom{N}{N-D}$

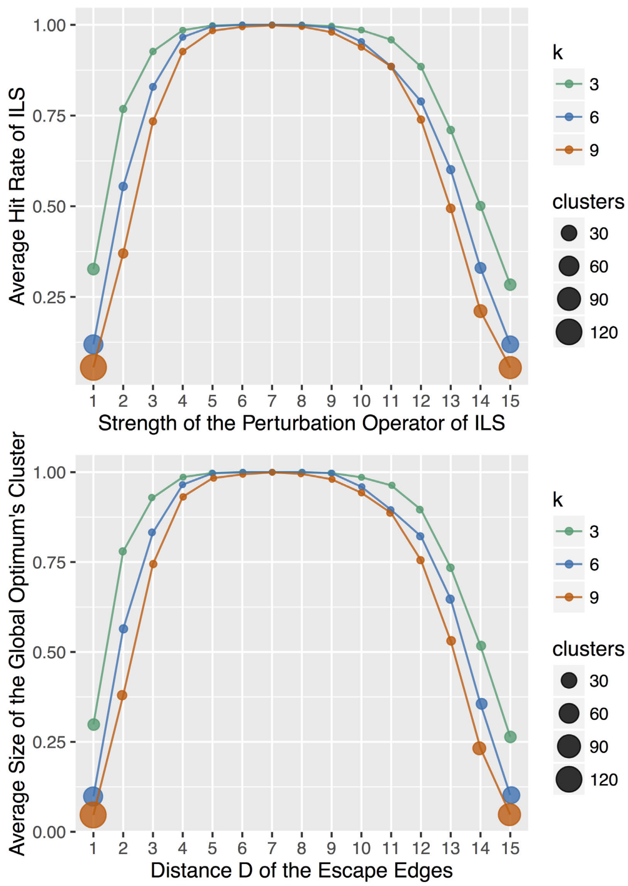


Figure 2: Top: The average hit rate ($hit_{\mathbf{ils}}$, fraction of runs which found the global optimum) over the strength of the perturbation operator γ of ILS. Bottom: the average size of the cluster containing the global optimum $cl_{\mathbf{peak}}$ over the escape distance D used to calculate the LONs with escape edges. The node size indicates the average number of clusters in the LONs ($\#cl$). Lines are for visual guidance.

We conclude that the best hit rate is achieved for $\gamma \approx N/2$ (ignoring search costs). Moreover, choosing a very high strength is superior to choosing a very low perturbation strength.

5.3 Search Costs of ILS

We will now have a look at the search costs of ILS (number of function evaluations), and again try to explain them by the results from our network analysis of the LONs. Figure 3 (top) shows the average search costs of the independent runs which found the global optimum. For low and high strengths of the search operator, the search costs are low. Moving towards the middle, the average costs increase at first, but then decrease again, leading to a minimum where $\gamma \approx N/2$. The higher the value of K, the higher are the average search costs. A possible explanation for this shape of the curves is that in the case of low and high γ, there is a sample bias, since we had to calculate the average costs from the runs which found the global optimum only. In an extreme case, the landscape is totally clustered, meaning that the global optimum will most likely be found by chance, rather than by heuristic search. In such cases, the costs will be very low. With increasing cluster size of the global optimum (or decreasing number of clusters), the chances to find the global optimum increases, but so do the search costs. In this region, there is a trade-off between the chance to find the global optimum and the necessary search costs. When the landscape merges into one giant cluster ($4 < D, \gamma < 10$), the perturbation operator with $\gamma \approx N/2$ seems to minimize the costs. We think that this is because in this case, the LON density is maximized, which probably reduces the average path lengths to the global optimum's cluster.

Before we have a deeper look into the network perspective, we can again observe a bias in the symmetry of the chart: for low strengths, the average costs to find the global optimum are lower than in the symmetric cases with high strengths. Like explained above, the strong perturbation operator probably kicks the search algorithm further away from the current local optimum than the corresponding symmetric weak perturbation operator. Because of the locality in the search space, it is more likely to be kicked into a basin different from the basin of the current local optimum. This means that more hill climbing needs to be performed to find the next local optimum than in a case where the perturbation operator kicks the algorithm somewhere in the middle of the basin around the current local optimum.

Last, we intend to find an explanation for the distribution of the search costs by a network metric. Towards this, we measured the average random walk distance to find the global optimum (Figure 3, bottom). Even though it takes some imagination to see a dent in the middle of the curves, we can see that the curves at the bottom have a quite similar shape to the curves at the top. A possible explanation why a minimum in the middle is not as pronounced as in the top figure could be because the average random walk distance is a very rough measure: it only considers the number of edges passed in the LON to reach the global optimum. The actual number of function evaluations necessary to perform hill climbing in ILS is not considered here. Instead, it is assumed that the costs are uniformly distributed among all potential kick operations. For a deeper analysis, we have calculated a linear regression between the logarithmized average search costs and our network measure, i.e. the average random walk distance. Figure 4 shows the result. We can see that there is a strong statistical relationship, since the average random walk distance explains nearly 68% of the variance of the average search costs. This is an interesting finding, since no network measure is known so far that predicts the expected search costs of metaheuristics with such high accuracy [9, 15].

6 CONCLUSION

We conducted an empirical study based on a large sample of instances of the Kauffman NK model. Using local optima networks and several measures from network analysis (e.g. community detection), we intended to gather a deeper understanding of the role of the perturbation strength of iterated local search. In particular, we compared statistical measures from our network analysis to the empirical performance of ILS (hit rate and search costs), depending on different levels of perturbation strength.

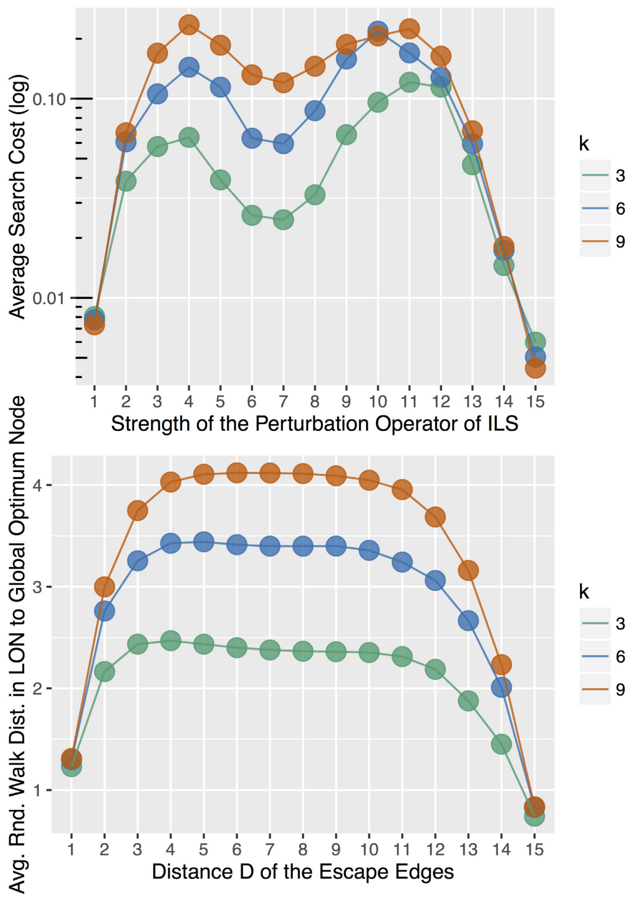


Figure 3: Top: The average search costs ($cost_{\mathbf{ils}}$, number of fitness function evaluations) over the strength of the perturbation operator γ of ILS. Bottom: the average random walk distance to the global optimum *rwdist* over the escape distance D used to calculate the LONs with escape edges. Lines are for visual guidance.

Our results suggest that the occurrence of clusters in fitness landscapes is a peculiarity of the perturbation strength used by ILS. Very low and very high strengths lead to a situation with many clusters, restricting the size of the global optimum's cluster, which is connected to a poor hit rate.

Our findings complement Lourenço et al. [23], stating that a stronger perturbation makes search completely random. In a binary search space, a very strong perturbation operator has (almost) the same effect as a weak perturbation operator. This is because the perturbation strengths that lead to a nearly equal number of potential perturbation moves (they are symmetric in the binomial distribution) cause an almost equally large second order neighborhood (the neighborhood of the perturbation operator). However, there is a bias because the strong operator has the potential to lead the search algorithm further away from the current local optimum. In the case of weak perturbation strengths, the local optima in the LON are connected to local optima that are close in the search space.

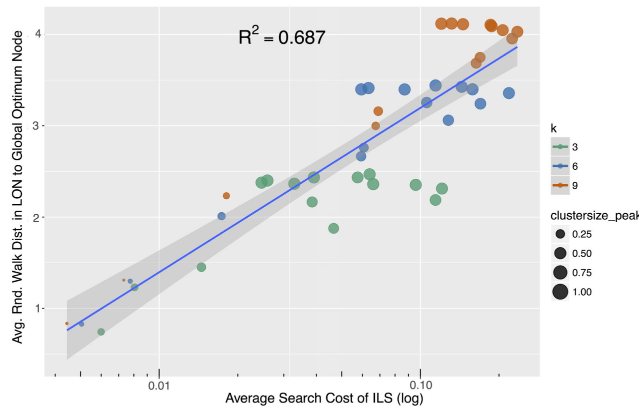


Figure 4: A linear regression model to predict the average search costs $cost_{\mathbf{ils}}$ by the average random walk distance to the global optimum *rwdist*. Each dot is a combination of K and D, cf. also Table 1.

In the case of strong perturbation, the local optima are connected to local optima that are further away. We assume that this effect might depend on the locality structure of the search space.

Considering search costs, we have shown that it is possible to predict these by statistical network measures, i.e. the random walk distance in a local optima network. When using a weak or very strong perturbation operator, there is a trade-off between the hit rate and the search costs. An intermediately strong perturbation operator seems to guarantee a high hit rate and minimizes the search costs in our experiments. However, we think that further analysis is necessary to understand this phenomenon. In summary, our results show that the role of perturbation strength is much more nuanced than previously thought.

7 ACKNOWLEDGEMENTS

G. Ochoa acknowledges support from the Leverhulme Trust [award number RPG-2015-395].

REFERENCES

[1] Réka Albert and Albert-László Barabási. 2002. Statistical mechanics of complex networks. *Reviews of Modern Physics* 74, 1 (jan 2002), 47–97. DOI: http://dx.doi.org/10.1103/RevModPhys.74.47

[2] David Applegate, William Cook, and Andre Rohe. 2003. Chained Lin-Kernighan for Large Traveling Salesman Problems. *INFORMS Journal on Computing* 15, 1 (2003), 82–92. DOI: http://dx.doi.org/10.1287/ijoc.15.1.82.15157

[3] Mathieu Bastian, Sebastien Heymann, and Mathieu Jacomy. 2009. Gephi: An Open Source Software for Exploring and Manipulating Networks. In *Third International AAAI Conference on Weblogs and Social Media*. 361–362. DOI: http://dx.doi.org/10.1136/qshc.2004.010033

[4] Stefano Boccaletti, Vito Latora, Yamir Moreno, Martín Chávez Hoffmeister, and Dong-Uk Hwang. 2006. Complex networks: Structure and dynamics. *Physics Reports* 424, 4-5 (feb 2006), 175–308. DOI: http://dx.doi.org/10.1016/j.physrep.2005.10.009

[5] Kenneth D. Boese, Andrew B. Kahng, and Sudhakar Muddu. 1994. A new adaptive multi-start technique for combinatorial global optimizations. *Operations Research Letters* 16, 2 (1994), 101–113. DOI: http://dx.doi.org/10.1016/0167-6377(94)90065-5

[6] Stephen P. Borgatti. 2005. Centrality and network flow. *Social Networks* 27, 1 (jan 2005), 55–71. DOI: http://dx.doi.org/10.1016/j.socnet.2004.11.008

[7] Francisco Chicano, Fabio Daolio, Gabriela Ochoa, Sébastien Verel, Marco Tomassini, and Enrique Alba. 2012. Local Optima Networks, Landscape Autocorrelation and Heuristic Search Performance. In *Parallel Problem Solving from*

Nature - PPSN XII: 12th International Conference, Carlos A. Coello Coello, Vincenzo Cutello, Kalyanmoy Deb, Stephanie Forrest, Giuseppe Nicosia, and Mario Pavone (Eds.), Vol. 7492 LNCS. Springer, Berlin and Heidelberg, 337–347. DOI: http://dx.doi.org/10.1007/978-3-642-32964-7_34 arXiv:arXiv:1210.4021v1
[8] Fabio Daolio, Marco Tomassini, Sébastien Verel, and Gabriela Ochoa. 2011. Communities of minima in local optima networks of combinatorial spaces. *Physica A: Statistical Mechanics and its Applications* 390, 9 (may 2011), 1684–1694. DOI: http://dx.doi.org/10.1016/j.physa.2011.01.005
[9] Fabio Daolio, Sébastien Verel, Gabriela Ochoa, and Marco Tomassini. 2012. Local optima networks and the performance of iterated local search. In *Proceedings of the fourteenth international conference on Genetic and evolutionary computation conference - GECCO '12*, Terence Soule (Ed.). ACM Press, Philadelphia, Pennsylvania, USA, 369. DOI: http://dx.doi.org/10.1145/2330163.2330217 arXiv:arXiv:1210.3946v1
[10] Santo Fortunato. 2010. Community detection in graphs. *Physics Reports* 486, 3-5 (feb 2010), 75–174. DOI: http://dx.doi.org/10.1016/j.physrep.2009.11.002 arXiv:0906.0612
[11] Fred Glover. 1986. Future paths for integer programming and links to artificial intelligence. *Computers & Operations Research* 13, 5 (jan 1986), 533–549. DOI: http://dx.doi.org/10.1016/0305-0548(86)90048-1
[12] Doug R. Hains, Darrel L. Whitley, and Adele E. Howe. 2011. Revisiting the big valley search space structure in the TSP. *Journal of the Operational Research Society* 62, 2 (feb 2011), 305–312. DOI: http://dx.doi.org/10.1057/jors.2010.116
[13] Sebastian Herrmann. 2016. Determining the Difficulty of Landscapes by PageRank Centrality in Local Optima Networks. In *European Conference on Evolutionary Computation in Combinatorial Optimization (EvoCOP 2016)*. Porto, 74–87. DOI: http://dx.doi.org/10.1007/978-3-319-30698-8_6
[14] Sebastian Herrmann, Gabriela Ochoa, and Franz Rothlauf. 2016. Communities of Local Optima As Funnels in Fitness Landscapes. In *Proceedings of the Genetic and Evolutionary Computation Conference 2016 (GECCO '16)*. ACM, New York, NY, USA, 325–331. DOI: http://dx.doi.org/10.1145/2908812.2908818
[15] Sebastian Herrmann, Gabriela Ochoa, and Franz Rothlauf. 2016. Communities of Local Optima as Funnels in Fitness Landscapes. In *Proceedings of the 2016 Genetic and Evolutionary Computation Conference - GECCO '16*. ACM Press. DOI: http://dx.doi.org/10.1145/2908812.2908818
[16] Sebastian Herrmann and Franz Rothlauf. 2015. Predicting Heuristic Search Performance with PageRank Centrality in Local Optima Networks. In *Proceedings of the 2015 Genetic and Evolutionary Computation Conference - GECCO '15*, Sara Silva (Ed.). ACM Press, Madrid, Spain, 401–408. DOI: http://dx.doi.org/10.1145/2739480.2754691
[17] Stuart A. Kauffman. 1993. *The origins of order*. Oxford University Press. DOI: http://dx.doi.org/10.1002/bies.950170412
[18] Stuart A. Kauffman and Simon Levin. 1987. Towards a General Theory of Adaptive Walks on Rugged Landscapes. *Journal of Theoretical Biology* 128, 1 (1987), 11–45. http://www.sciencedirect.com/science/article/pii/S0022519387800292
[19] Stuart A. Kauffman and Edward D. Weinberger. 1989. The NK model of rugged fitness landscapes and its application to maturation of the immune response. *Journal of Theoretical Biology* 141, 2 (1989), 211–245. DOI: http://dx.doi.org/10.1016/S0022-5193(89)80019-0
[20] Pascal Kerschke, Mike Preuss, Simon Wessing, and Heike Trautmann. 2015. Detecting Funnel Structures by Means of Exploratory Landscape Analysis. In *Proceedings of the 2015 Genetic and Evolutionary Computation Conference - GECCO '15*, Sara Silva (Ed.). ACM Press, Madrid, Spain, 265–272. DOI: http://dx.doi.org/10.1145/2739480.2754642
[21] S. Lin and B. W. Kernighan. 1973. An Effective Heuristic Algorithm for the Traveling-Salesman Problem. *Operations Research* 21, 2 (apr 1973), 498–516. DOI: http://dx.doi.org/10.1287/opre.21.2.498
[22] M. Locatelli. 2005. On the Multilevel Structure of Global optimization problems. *Computational Optimization and Applications* 30, 1 (2005), 5–22. DOI: http://dx.doi.org/10.1007/s10589-005-4561-y
[23] Helena R. Lourenço, Olivier C. Martin, and Thomas Stützle. 2003. Iterated Local Search. In *Handbook of Metaheuristics*. Kluwer Academic Publishers, Boston, 320–353. DOI: http://dx.doi.org/10.1007/0-306-48056-5_11
[24] Monte Lunacek and Darrell Whitley. 2006. The dispersion metric and the CMA evolution strategy. In *Proceedings of the 8th annual conference on Genetic and evolutionary computation - GECCO '06*. ACM Press, New York, New York, USA, 477. DOI: http://dx.doi.org/10.1145/1143997.1144085
[25] Katherine M. Malan and Andries P. Engelbrecht. 2014. Fitness Landscape Analysis for Metaheuristic Performance Prediction. In *Recent Advances in the Theory and Application of Fitness Landscapes*, Hendrik Richter and Andries Engelbrecht (Eds.). Springer, Berlin and Heidelberg, 103–129. DOI: http://dx.doi.org/10.1007/978-3-642-41888-4_9
[26] B. Naudts and L. Kallel. 2000. A comparison of predictive measures of problem difficulty in evolutionary algorithms. *IEEE Transactions on Evolutionary Computation* 4, 1 (apr 2000), 1–15. DOI: http://dx.doi.org/10.1109/4235.843491
[27] Gabriela Ochoa, Marco Tomassini, Sébastien Verel, and Christian Darabos. 2008. A study of NK landscapes' basins and local optima networks. In *Proceedings of the 10th annual conference on Genetic and evolutionary computation - GECCO '08*, Maarten Keijzer (Ed.). ACM Press, Atlanta, GA, USA, 555–562. DOI: http://dx.doi.org/10.1145/1389095.1389204
[28] Gabriela Ochoa and Nadarajen Veerapen. 2016. Deconstructing the Big Valley Search Space Hypothesis. In *Evolutionary Computation in Combinatorial Optimization: 16th European Conference, EvoCOP 2016, Porto, Portugal, March 30 – April 1, 2016, Proceedings*, Francisco Chicano, Bin Hu, and Pablo Garcia-Sanchez (Eds.). Springer, Porto, 58–73. DOI: http://dx.doi.org/10.1007/978-3-319-30698-8_5
[29] Gabriela Ochoa, Nadarajen Veerapen, Darrell Whitley, and Edmund K. Burke. 2016. The Multi-Funnel Structure of TSP Fitness Landscapes: A Visual Exploration. In *Artificial Evolution: 12th International Conference, Evolution Artificielle, EA 2015*. Springer International Publishing, Lyon, 1–13. DOI: http://dx.doi.org/10.1007/978-3-319-31471-6_1
[30] Travis E. Oliphant. 2007. Python for scientific computing. *Computing in Science and Engineering* 9, 3 (2007), 10–20. DOI: http://dx.doi.org/10.1109/MCSE.2007.58
[31] Mason a. Porter, Jukka-Pekka Onnela, and Peter J. Mucha. 2009. Communities in Networks. *Notices of the AMS* 486, 3-5 (2009), 1082–1097. DOI: http://dx.doi.org/10.1016/j.physrep.2009.11.002 arXiv:0902.3788
[32] R Development Core Team. 2009. *R: A Language and Environment for Statistical Computing*.
[33] Christian M. Reidys and Peter F. Stadler. 2002. Combinatorial Landscapes. *SIAM Rev.* 44, 1 (jan 2002), 3–54. DOI: http://dx.doi.org/10.1137/S0036144501395952
[34] Franz Rothlauf. 2011. *Design of modern heuristics: Principles and application*. Springer, Berlin and Heidelberg.
[35] Venu Satuluri, Srinivasan Parthasarathy, and Duygu Ucar. 2010. Markov clustering of protein interaction networks with improved balance and scalability. In *Proceedings of the First ACM International Conference on Bioinformatics and Computational Biology - BCB '10*. ACM Press, New York, USA, 247. DOI: http://dx.doi.org/10.1145/1854776.1854812
[36] Peter F. Stadler. 1996. Landscapes and their correlation functions. *Journal of Mathematical Chemistry* 20, 1 (1996), 1–45. DOI: http://dx.doi.org/10.1007/BF01165154
[37] E.G. Talbi and P. Bessière. 1991. A parallel genetic algorithm for the graph partitioning problem. In *Proceedings of the 5th international conference on Supercomputing - ICS '91*. ACM Press, New York, USA, 312–320. DOI: http://dx.doi.org/10.1145/109025.109102
[38] Thomas W. Valente. 1996. Network models of the diffusion of innovations. *Computational and Mathematical Organization Theory* 2, 2 (1996), 134. DOI: http://dx.doi.org/10.1007/BF00240425
[39] Stijn van Dongen. 2001. *Graph clustering by flow simulation*. Ph.D. Dissertation. Utrecht University.
[40] Sébastien Verel, Fabio Daolio, Gabriela Ochoa, and Marco Tomassini. 2012. Local optima networks with escape edges. In *Artificial Evolution*. Springer, Angers, France, 49–60. DOI: http://dx.doi.org/10.1007/978-3-642-35533-2_5
[41] Edward D. Weinberger. 1990. Correlated and uncorrelated fitness landscapes and how to tell the difference. *Biological cybernetics* 336 (1990), 325–336. http://link.springer.com/article/10.1007/BF00202749
[42] Sewall Wright. 1932. The roles of mutation, inbreeding, crossbreeding, and selection in evolution. In *Proceedings of the 6th International Congress of Genetics*, Donald F. Jones (Ed.). Morgan Kaufmann Publishers Inc., Ithaca, New York, 356–366. http://www.esp.org/books/6th-congress/facsimile/contents/6th-cong-p356-wright.pdf