Comparing Communities of Optima with Funnels in Combinatorial Fitness Landscapes

ECOM Track

ABSTRACT

The existence of sub-optimal *funnels* in combinatorial fitness landscapes has been linked to search difficulty. The exact nature of these structures - and how commonly they appear - is not yet fully understood. Improving our understanding of funnels could help with designing effective diversification mechanisms for a ‘smoothing’ effect, making optimisation easier. We model fitness landscapes as local optima networks. The relationship between *communities* of local optima found by network clustering algorithms and funnels is explored. Funnels are identified using the notion of *monotonic sequences* from the study of energy landscapes in theoretical chemistry. NK Landscapes and the Quadratic Assignment Problem are used as case studies. Our results show that communities are linked to funnels, but they are not the same structures. The analysis exhibits relationships between these landscape structures and the performance of trajectory-based metaheuristics such as Simulated Annealing (SA) and Iterated Local Search (ILS). In particular, ILS gets trapped in funnels, and modular communities of optima slow it down. The funnels contribute to lower success for SA. We show that increasing the strength of ILS perturbation helps with avoidance of the funnels and improves performance in multi-funnel landscapes.

KEYWORDS

Fitness Landscapes, NK Landscapes, Quadratic Assignment Problem, Local Optima Networks, Funnel Landscapes, Combinatorial Optimisation

ACM Reference format:
. 2017. Comparing Communities of Optima with Funnels in Combinatorial Fitness Landscapes. In *Proceedings of the Genetic and Evolutionary Computation Conference 2017, Berlin, Germany, July 15–19, 2017 (GECCO ’17),* 8 pages.
DOI: 10.475/123_4

1 INTRODUCTION

Until recently, the travelling salesman problem and other combinatorial optimisation problems were believed to have a globally convex ‘big valley’ fitness landscape structure [15], implying that local optima are close to the global optimum and clustered together. Under this view, many local optima may exist, but they are easy to escape and the coarse-grained gradient easily leads to the the global optimum. However, recent studies have revealed that some combinatorial landscapes decompose into more than one valley or *funnel* [8, 9, 15]. In the context of combinatorial optimisation, a rigorous definition of ‘funnels’ is still lacking. The intuition behind this notion is captured by Figure 1 where two funnels are depicted as two groups of local optima which are close in configuration space within a group, but well-separated between groups. In this study we consider the definition from the study of energy landscapes in theoretical chemistry [6], stating that a funnel is ‘a region of configuration space that can be described in terms of a set of downhill pathways that converge on a single low-energy structure or a set of closely-related low-energy structures’. When a landscape has at least one sub-optimal funnel, local search algorithms can be trapped there and never reach the global optimum. Indeed, recent research has conjectured that the existence of these structures in a landscape are the cause of sub-optimal performance from common metaheuristics [8, 9, 15]. The exact nature of multiple-funnel fitness landscapes, and how commonly they occur, is not yet fully understood.

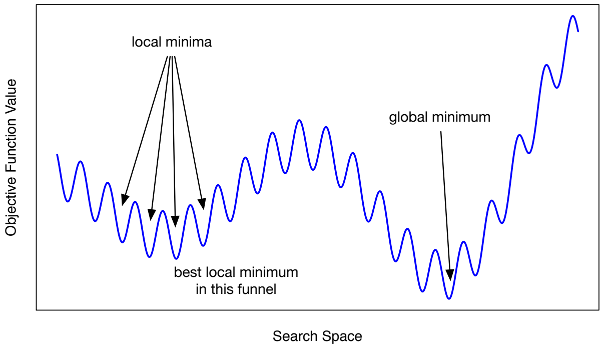


Figure 1: Schematic view of a double-funnel landscape, depicting a sub-optimal funnel (left) and the optimal funnel (right) in a minimisation problem.

We use the *Local Optima Network* (LON) model [18] to compress the connectivity of a full search space into a network, where nodes are the local optima, and edges represent possible transitions among basins with a given search operator. Techniques from the field of complex networks can be applied to LONs to analyse the landscape topology. We use two well-known families of combinatorial landscapes as test problems: NK Landscapes and the Quadratic Assignment Problem (QAP).

Applying two network community detection algorithms — the ‘Markov Clustering Algorithm’ and ‘Infomap’ — to the optima networks of these instances, we note that local optima are not always uniformly distributed: we can identify regions of densely connected sub-graphs, corroborating previous findings [5, 9]. We compare these communities with landscape ‘funnels’ exposed using the notion of monotonic sequences from the study of energy landscapes.

GECCO ’17, Berlin, Germany
© 2017 Copyright held by the owner/author(s). 123-4567-24-567/08/06…$15.00
DOI: 10.475/123_4

A monotonic sequence [3] is a sequence of local minima where the energy of minima is always decreasing. We use monotonic sequences with LONs, for the first time in these two test functions, to uncover their funnel structure. We found that funnels and communities of optima are related, but they do not exactly match. These two approaches expose landscape global features that can negatively impact optimisation. To gain empirical insight into this, two popular metaheuristics, Iterated Local Search (ILS) and Simulated Annealing (SA), are applied to the studied instances. Our analysis shows that both ILS and SA can get trapped in funnels. For ILS a slightly stronger perturbation helps to avoid the funnels for a significantly improved rate of success. Flow-based communities of optima do not noticeably affect success rate but if they are modular (strongly-linked within their own community) they do cause ILS to stagnate during search.

2 FITNESS LANDSCAPES AS NETWORKS

After some preliminary definitions, this section formalises the standard local optima network model. Thereafter, we describe our proposal for adapting monotonic sequences to the context of LONs and characterising funnel structures.

2.1 Preliminaries

A fitness landscape [17] is a triplet $\{S, V, f\}$ where S is the set of all possible solutions, $V : S \longrightarrow 2^{|S|}$, a neighbourhood structure, is a function that assigns to every $s \in S$ a set of neighbours $V(s)$, and f is a fitness (objective value) function such that f : S $\longrightarrow \mathbb{R}$, where the fitness value is a real number that can be viewed as the *height* of a given solution in the landscape.

Our study considers two combinatorial landscapes as discussed in Section 4. Therefore, two solution representations are used: binary strings (NK landscapes) and permutations (QAP). For each case, the most basic neighbourhood structure is considered: the single bit-flip operation that changes a single bit in a given binary string, and the pairwise exchange operation that exchanges any two positions in a permutation. A best-improvement local search (hill-climbing) algorithm is used to determine the local optima, with the neighbourhoods mentioned above for each of the studied representations.

2.2 Local Optima Networks

In order to describe the network model, we need to characterise the nodes and edges.

Nodes. A local optimum, is a solution lo such that $\forall s \in V(lo)$, $f(s) \leq f(lo)$ ($f(s) \geq f(lo)$, when minimising). Let us denote by $h(s)$ the stochastic operator that associates each solution s to its local optimum, i.e. the solution obtained after applying a best-improvement hill-climbing algorithm until convergence. The size of the landscape is finite, so we can denote the local optima by $lo_1, lo_2, \ldots, lo_p$. We denote the set of local optima by LO, which corresponds to the set of nodes in the network.

Escape edges. The escape edges are defined according to a distance function d (minimal number of moves between two solutions), and a positive integer $D > 0$. There is an edge e_{ij} between lo_i and lo_j if a solution s exists such that $d(s, lo_i) \leq D$ and $h(s) = lo_j$. The weight w_{ij} of this edge is $w_{ij} = |\{s \in S \mid d(s, lo_i) \leq D \text{ and } h(s) = lo_j\}|$. This weight can be normalised by the number of solutions, $|\{s \in S \mid d(s, lo_i) \leq D\}|$, within reach at distance D.

Local optima network (LON). The weighted local optima network $LON = (LO, E)$ is the graph where the nodes $lo_i \in LO$ are the local optima, and there is an edge $e_{ij} \in E$, with weight w_{ij}, between two nodes lo_i and lo_j if $w_{ij} > 0$. w_{ij} may be different than w_{ji}. Thus, two weights are needed in general, and we have an oriented transition graph.

2.3 Monotonic Sequences

Since we need to model monotonic sequences in order to characterise funnels, we consider a graph with a reduced number of edges. Given a LON, which is extracted after a full enumeration of the local optima and basins in the instances studied, we kept only the non-deteriorating escape edges. That is, those edges e_{ij} that arrive at a local optimum lo_j from a local optimum lo_i with worse (or equal) evaluation (i.e. $f(lo_j) \geq f(lo_i)$ when maximising or $f(lo_j) \leq f(lo_i)$ when minimising). The rest of the edges are removed from the network. We, therefore, define the following two related LON models:

Monotonic Local Optima Network (MLON). The directed (and weighted) graph $MLON = (LO, ME)$, where nodes $lo_i \in LO$ are the local optima, and $ME \subseteq E$ is the set of edges e_{ij} that depart from a node worse in evaluation than the arriving node.

Monotonic Local Optima Network pruned by weight (MLON).* Similar to the MLON model, with the difference that only the 50% 'heaviest-weighted' links between optima are retained. That is, there is an edge $e_{ij} \in ME$, with weight w_{ij}, between two optima lo_i and lo_j if w_{ij} is greater than the median link weight in the network. Recalling that the weight of a link is the probability of the escape move, we select this model to gain insight into the more 'likely' path a stochastic local search algorithm might follow.

2.4 Detecting Funnel Structures

To detect funnel structures we first identify the funnels' 'ends' or 'bottoms'. To do so, we take advantage of the Monotonic Local Optima Networks. MLONs are directed graphs without loops. In a directed graph, one can distinguish the outdegree (number of outgoing edges) from the indegree (number of incoming edges). In graph theory, a *source* node is a node with indegree zero, while a *sink* node is a node with outdegree zero. We consider the MLON sinks as the funnel bottoms.

Sinks. We thus define the *funnel sinks* as the MLON nodes that have at least one incoming edge but no outgoing edges. Therefore, a sink is a reachable node, but with no possibility of escaping from it to a better node. Notice that global optima are naturally sinks.

Once funnel sinks are detected, we can proceed to identify the funnel basins. This is done by finding all nodes in the MLON graph which are reachable from each funnel sink. Breadth-First-Search is used for this purpose. The set of unique nodes in the combined paths to a given funnel sink corresponds to the funnel basin. The cardinality of this set corresponds to the funnel size. Notice that the membership of a node to a funnel might be overlapping, that is, a node may belong to more than one funnel, in that there are paths from that node to more than one funnel sink.

3 COMMUNITY DETECTION ALGORITHMS

Synthetic randomly generated networks generally show little structure in the sense that there are few or no recognisable sub-networks. That is, if one looks at a picture of the network it appears to be rather homogeneous on a global scale. On the contrary, many real-world networks, especially those arising from social interactions, show the presence of clusters of nodes. These clusters are called *communities.* It is difficult, if not impossible, to give a precise and unique mathematical definition of a community. An intuitive definition is as follows: nodes belonging to a community are more strongly associated with each other than they are with the rest of the network. In other words, the intra-community connectivity is higher than the inter-community connectivity. This definition is somewhat circular but in the last few years a multitude of algorithms have been proposed for community detection [7]. The choice of algorithms to apply crucially depends on the objective of the study. We selected two algorithms based on network flow (Infomap and Markov clustering) as an alternative to more traditional algorithms based on maximising *modularity* (density of links). This is because the former focuses on how the topology of the network affects its dynamic behaviour [16], as opposed to the way the network was established. We conjecture that using algorithms based on flow might be more fruitful for this particular domain: heuristic search can be seen as a stochastic process flowing in local optima networks. Moreover, flow-based algorithms, specifically Markov clustering, have been previously used for finding communities in LONs [5, 9].

InfoMap. This algorithm looks for the partition of a graph from which one can most succinctly describe a path through it. First of all, every node in the network is assigned to its own cluster [16]. Then, nodes are iteratively moved (in a random order) into the neighbouring cluster which results in the largest decrease in theoretical network path description length. This continues until there exists no move that could decrease it further. Following this, the clusters from the previous step become *nodes* and the process repeats as before. This continues until the algorithm has converged and we have the partition of the graph from which we can infer a network path with the shortest description. Crucially, InfoMap supports weighted and oriented networks; clusters are defined as places where flow is likely to circulate for long periods.

Markov Clustering (MCL) Algorithm. Works by mimicking the flow between nodes in a network [19]. The aim is that flow should end up being thin between communities and heavy within them. This algorithm has two parameters: *inflation* and *expansion*. Expansion tries to replicate the dispersion of flow over the network, and inflation simulates the opposite: heavier-weighted links in the network get heavier, and lighter-weighted links get lighter, modelling *contraction*, i.e. flow circulating in densely-connected sub-graphs. Different inflation settings can result in different levels of coarseness in the clusterings. Consequently, there can be difficulty discerning which one is most indicative of the 'true' community structure [7]. In the present study, we use the default inflation value of 2, in order to avoid a too finely-grained partitioning of the graph, while still obtaining good *modularity* scores.

4 EMPIRICAL SETTING

4.1 Test Problems

Two well known combinatorial optimisation problems are considered as case studies. For both, we selected the largest possible problem sizes that allow exhaustive enumeration and thus extraction of the optima networks.

NK landscapes. In the NK landscape model [11] ruggedness can be adjusted by the parameter K, which induces the amount of epistasis in the binary encoded solution of length N. The number of local optima increases almost exponentially with increasing K, inducing a rugged and increasingly random landscape. We studied NK instances with $N = 18$ and $K \in \{2, 4\}$.

The Quadratic Assignment Problem (QAP). In the QAP a set of facilities with given flows have to be assigned to a set of locations with given distances in such a way that the sum of the product of flows and distances is minimised. A solution is encoded as a permutation of the set $\{1, 2, ..., n\}$. The structure of the flow and distance matrices characterises the class of instances. Our experiments are based on two instance generators proposed in [12], originally devised for the multi-objective QAP, but adapted for the single-objective QAP and used for LON analysis in [5]. The first generator produces uniformly random instances where all flows and distances are integers sampled from uniform distributions. The second generator produces flow entries that are non-uniform random values. In this case, clusters of points are placed in compact circular areas, and all of these clusters are enclosed in a large circle. These instances have the so-called "real-like" structure since they resemble the structure of QAP problems found in practical applications. We consider both uniform and real-like random instances of size 11.

4.2 Metaheuristics

In order to assess the relationship between heuristic search performance and LON and MLON metrics, two standard trajectory-based metaheuristics are considered. Namely, iterated local search and simulated annealing. Our ILS implementation [10] uses a best improvement local search with the most basic neighbourhood structure in the improvement stage, a single bit-flip (NK landscapes) or pairwise exchange (QAP). The perturbation stage uses 2 (regular) or 3 (increased strength) bit-flips for the NK landscapes (pairwise exchanges for the QAP). A deterministic acceptance condition accepting non-deteriorating solutions is used.

The simulated annealing implementation [10] uses the most basic neighbourhoods for each landscape and considers a standard exponential cooling scheme. The initial temperature is set according to the largest possible difference in fitness values. Table 1 summarises the SA parameters.

For both algorithms, search terminates at the global optimum, which is known a priori, or when reaching a pre-set limit of objective function evaluations FE_{max}. For NK landscapes this is 10% of the size of the search space, $FE_{max} = 0.1 \times 2^{20} = 26214$, while for QAP it is 2.5% of the search space size, $FE_{max} = 0.025 \times 11! \simeq 10^6$.

As performance measures, we consider the success rate, *SR*, measured as the proportion of runs (out of 1000) that attained the global optimum. To measure the speed of optimisation, we considered the time to success, *TS*, measured as the number of objective function evaluations to reach the global optimum for the successful runs.

Table 1: SA parameters. MaxT is the maximum #iterations at same temperature, between cooling steps.

Landscape	Start Temp	End Temp	alpha	MaxT
NK	1.4	0.0	0.8	262
QAP	14 426 950	0.0	0.8	10 000

5 RESULTS

5.1 Contrasting Funnels and Communities

Our study extracts the complete MLONs of the selected test problems. For each problem and class, 30 independent instances are generated and analysed. For each instance, we extract the funnel sinks, and compute the funnel memberships as discussed in Section 2.4. Similarly, we apply the two community detection algorithms to the original optima networks as discussed in Section 3. In order to identify the communities' most central nodes, we used the *PageRank* centrality metric. PageRank ranks a node in a network based on how many other nodes it has links to, and the quality of the vertices to which it links. We calculated scores for each node in the network, and cluster centres were identified as the nodes with the largest global PageRank value in their cluster [4].

Figure 2 shows the optima network of a selected NK Landscape instance, represented as three partitions found by MLON (funnel), Infomap and MCL (clusters), respectively. The different background colours — and also node colour — represent membership to a group. The cluster or funnel centres have a thickened border, while the global optimum is labelled *G*. Node size is proportional to the centrality of that optimum.

The groups of colour in Figure 2b and 2c suggest several distinct clusters of optima. Conversely, the presence of a single colour in Figure 2a indicates a single-funnel landscape. Therefore, this instance should largely have a 'big valley' topology. If this is true, the landscape would be easy to navigate through. Fittingly, ILS and SA have success rates of 0.996 and 0.963 on this instance, respectively. However, the 'big valley' might not be the complete picture. The separated clusters in 2b and 2c might imply that search could circulate within certain areas. Note also that there is some agreement between the two cluster algorithms. Homogeneity in separate algorithms' partitions can suggest the presence of 'true' community structure. Indeed, we provide evidence in Section 5.2 suggesting that modular clusters in our test landscapes slow down ILS. The instance in Figure 2 took more time to be solved than 75% of the other K = 2 instances, most of which have similar ruggedness. One possible explanation for this is that the search began in a sub-optimal cluster.

It appears there is a discrepancy between communities and funnels — both in their nature and their effect. Section 5.2 reports a statistical analysis of their empirical impact on metaheuristics. Table 2 contrasts funnels with clusters of optima. Metrics are reported (averaged over 30 instances) which show their structural differences. Namely, their respective quantities in the landscapes are reported. Recall that a sink is the end of a funnel; therefore the number of sinks is also the number of funnels. A comparison of the two types of central node was conducted. Sinks are the centres of funnels, and optima with the highest centrality in their cluster are the centres. If these are the same, funnels and optima clusters might be the same structures. The *#match* rows in Table 2 show how many of these were indeed the same node. Many of them weren't; therefore, the *distance* values show the mean distance from a cluster centre to the set of funnel sinks in the landscape. *Distance* here means binary bit-flips in the case of the NK Landscapes, and random swaps for the QAP, described in Section 2.1. These have been normalised by the maximum distance (18 for NK, 10 for the QAP).

Table 2 shows that there are more optima clusters than funnels, comparing *#clusters* with the *#sinks.* This holds across each landscape class. Therefore, the two structures seem inconsistent in terms of size.

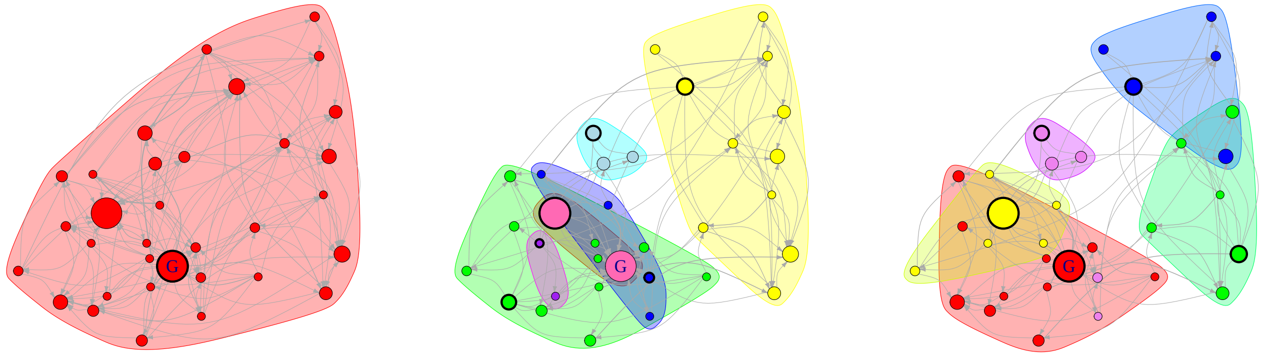
The majority of funnel sinks were also Markov Cluster centres (the relevant rows are in bold text). The likelihood of this arising by chance seems slim, in particular when considering the 'uniform random' QAP instances, which typically have over 1000 local optima. This suggests a connection between the centres of these two landscape structures. When observing the *distance* values in Table 2, note that for the K = 2 instances, the normalized mean distance between any two nodes in the optima networks is 0.42. Therefore, on average MCL non-sink centres seem much closer to funnel bottoms than to a given node in the network, being at a mean distance of 0.28 and 0.22 from the sinks for the two MLON models.

Overall, clusters of optima show signs of being related to landscape funnels, but they don't appear to be the same. We conjecture that network clustering algorithms used on optima networks might not be a reliable way to characterise funnels — and therefore local search success — in fitness landscapes. We next conduct a statistical study to gain insight on how landscape features relate to each other, and contribute to the behaviour of metaheuristics.

5.2 Impact on Search Performance

Figure 3 displays all pairwise associations between search performance metrics and the considered landscape features. Pairs are laid out in a matrix fashion, with scatter plots on the lower part, univariate density plots on the diagonal, and rank-correlation coefficients on the upper part. In particular, the first two rows and columns, 'SR_ils' and 'TS_ils', report on the success rate and the runtime of successful runs of iterated local search; the following two on the same metrics for simulated annealing.

In terms of success rate alone, NK landscapes with $K = 2$ are empirically easier to solve to optimality than NK landscapes with with $K = 4$ for both metaheuristics. On QAP instances though, SA is more successful with uniform-random instances than with real-like ones. The opposite is true for ILS. We conjecture that this different behaviour might be due to the fact that, in QAP random instances, fitness differences between neighbouring solutions are



(a) MLON (funnel) partition **(b) Infomap (community) partition** **(c) MCL (community) partition**

Figure 2: A representative LON for an NK landscape with N = 18, K = 2, visualised according to the three different partitions methods. Colours identify different clusters, and node size is proportional to PageRank centrality. The global optimum is labelled with 'G', and cluster centres show a thickened border. Half of the 'lightest' links are pruned off for visual clarity.

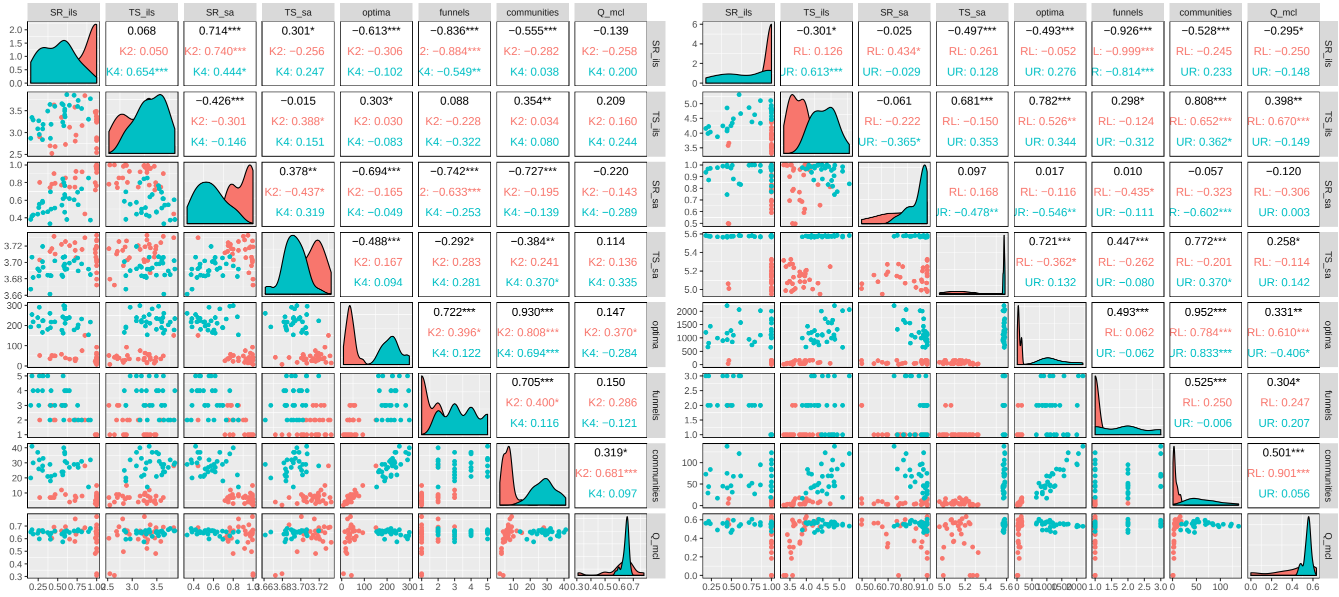
	NK				QAP			
	MLON		MLON*		MLON		MLON*	
	K = 2	K = 4	K = 2	K = 4	RL	UR	RL	UR
#optima	$42.97_{27.66}$	$220.63_{39.10}$	$42.97_{27.66}$	$220.63_{39.10}$	$74.83_{51.24}$	1241.17_{433}	$74.83_{51.24}$	1241.17_{433}
#sinks	$\mathbf{1.57}_{0.73}$	$\mathbf{3.43}_{1.07}$	$\mathbf{2.87}_{1.38}$	$\mathbf{8.00}_{1.76}$	$\mathbf{1.07}_{0.25}$	$\mathbf{1.87}_{0.78}$	$\mathbf{1.13}_{0.35}$	$\mathbf{2.53}_{1.17}$
MCL								
#clusters	$7.30_{4.84}$	$28.43_{6.64}$	$7.30_{4.84}$	$28.43_{6.64}$	$5.93_{4.85}$	$63.67_{31.87}$	$5.93_{4.85}$	$63.67_{31.87}$
#match	$\mathbf{1.47}_{0.68}$	$\mathbf{3.30}_{1.15}$	$\mathbf{2.60}_{1.28}$	$\mathbf{7.33}_{1.65}$	$\mathbf{1.00}_{0.37}$	$\mathbf{1.80}_{0.85}$	$\mathbf{1.07}_{0.50}$	$\mathbf{2.43}_{1.19}$
$\|\|distance\|\|$	$0.28_{0.09}$	$0.40_{0.03}$	$0.22_{0.07}$	$0.35_{0.04}$	$0.49_{0.23}$	$0.77_{0.02}$	$0.48_{0.22}$	$0.76_{0.31}$
Infomap								
#clusters	$5.47_{2.21}$	$19.37_{2.76}$	$5.47_{2.21}$	$19.37_{2.76}$	$2.83_{1.18}$	$54.33_{14.94}$	$2.83_{1.18}$	$54.33_{14.94}$
#match	$1.17_{0.59}$	$2.67_{1.21}$	$2.00_{1.08}$	$5.40_{1.16}$	$0.87_{0.43}$	$1.70_{0.65}$	$0.87_{0.48}$	$2.17_{1.12}$
$\|\|distance\|\|$	$0.29_{0.11}$	$0.41_{0.03}$	$0.23_{0.00}$	$0.35_{0.03}$	$0.25_{0.26}$	$0.76_{0.02}$	$0.25_{0.25}$	$0.76_{0.02}$

Table 2: Contrasting funnels with clusters of optima. Distances are normalised by the maximum possible distance between two solutions, 18 for NK and 10 for QAP. RL are the 'real-like' and UR are the 'uniform-random' QAP instances. Values are averages of 30 instances with standard deviations shown as subscripts.

comparatively lower than in real-like instances, which could make it easier for SA to traverse the landscape despite its higher ruggedness. In fact, the runtime distribution of SA runs on QAP instances is highly bimodal, with most of the successful runs terminating close to the FE_{max} budget in the case of random instances.

In most cases, there is a strong negative correlation between the number of funnels and the success rate. We can see this looking down from the 'funnels' columns, and checking against the success rate for ILS and SA. Overall, the association between the number of optima and success rate is lower than for funnel quantity. This suggests the ruggedness of a landscape cannot wholly predict its challenges. The 'Q_mcl' entries in the matrices denote the correlation between the *modularity* (strength) of community structure with performance. The crossover between these columns and the 'TS' runtime rows shows a weak positive association, in particular with ILS. Features inter-correlations are also displayed.

As we notice, the number of funnels and the number of communities are both associated with the number of local optima. The number of local optima, in turn, has an impact on performance and thus acts as a potential confounder. Therefore, if we want to assess the relationship between heuristic performance and the landscape global structure, we need to control for the number of local optima.



(a) NK landscapes

(b) QAP instances

Figure 3: correlation matrices of performance metrics and landscape features (see facet titles). Lower triangle: pairwise scatter plots. Diagonal: density plots. Upper triangle: pairwise Spearman's rank correlation, $^{***}p < 0.001$, $^{**}p < 0.01$, $^{*}p < 0.05$.

A classical way to achieve this is to include the confounder variable in a multiple regression analysis. However, if we want to draw generalisable inferences by grouping all the problem instances under study, the regression model needs to consider that our observations are not independent, but clustered according to the studied problem classes [1].

Therefore, we turn to hierarchical models to explain the observed performance variance by using the number of optima, the number of funnels, the number of communities, and their modularity scores as common predictors across all problem instances (*fixed effects*). Moreover, the models include *random effects* that are conditional to the problem type and class. More precisely, let us denote by y_{ijk} the performance (success rate or runtime) observed on instance i from problem j (NK or QAP) of class k ($K = 2$, $K = 4$, real-like or uniform-random). We can write the linear model:

$$y_{ijk} = \beta_0 + \sum_{c=1}^{p} \beta_c x_{cijk} + \alpha_j + \alpha_{jk} + \epsilon_{ijk}, \qquad \epsilon_{ijk} \sim \mathcal{N}(0, \sigma^2)$$

where x_{cijk} is the value of predictor c (e.g., number of local optima, number of funnels, etc.) of instance i from problem j of class k, β_c is its corresponding fixed effect on search performance, α_j and α_{jk} are the random effects conditional on problem j and problem class k, which represent random deviations from the common intercept β_0, and finally ϵ_{ijk} are the model residuals.

Table 3 reports the results of model fitting [2], namely coefficients estimation for the fixed effects and variance estimation of the random effects, for both algorithms and performance metrics. Note that, in order to better approach normality of residuals, the arcsin-squareroot transformation has been applied to success rate, whereas runtime has been logtransformed. In order to allow for a direct comparison of effect sizes, predictor variables have been rescaled to zero mean and unitary variance. That is, each estimated coefficient $\hat{\beta_c}$ can be interpreted as the effect a change in one standard deviation of the related predictor x_c would have on the transformed outcome y, all other predictors being held constant. The $\hat{\beta_0}$ coefficient represents the expected outcome when all predictors are at their mean value, conditionally on the random effects. The *conditional* R^2 gives the proportion of variance explained by the whole regression model. The *marginal* R^2 is the ratio of variance that is explained by the fixed effects [13].

Local optima networks with escape edges provide a view of the landscape that is consistent with iterated local search dynamics. Therefore, we expect LON features to explain the ILS performance variance better than that for SA. This shows in higher marginal R^2, in particular concerning success rate. Notably for ILS, it is only the number of funnels that has a significant impact on success across all considered problem instances, when controlling for other factors such as the number of local optima or their clustering into communities. That is, all things being equal, for ILS the number of funnels is all that matters for success; the higher their number, the lower the success rate. A similar inference can be made regarding the success rate of SA: the number of funnels is the predictor with the strongest conditional impact on SR. However, in this case, the part of variance explained by the fixed effects is comparatively much lower, and so are the effect sizes.

Regarding runtime, SA data fits very poorly the model and we can not draw any sound conclusion. Instead, it is interesting to notice that the more modular is the clustering of local optima, the longer it takes for ILS to find the global optimum. Furthermore, the significant negative association between ILS runtime and number

Table 3: Conditional impact of landscape features on the expected success rate and runtime of ILS and SA in a linear mixed-model with random effects for problem and problem class. Model coefficients estimates (and standard deviations) are reported.

	Iterated Local Search		Simulated Annealing	
	Success Rate	Time to Succeed	Success Rate	Time to Succeed
(Intercept $\hat{\beta}_0$)	**1.158** (0.063)***	**3.713** (0.373)***	**1.184** (0.088)***	**4.528** (0.826)***
#optima $\hat{\beta}_1$	−0.039 (0.085)	0.197 (0.127)	0.050 (0.079)	−0.025 (0.025)
#funnels $\hat{\beta}_2$	**−0.350** (0.029)***	**−0.138** (0.041)***	**−0.086** (0.025)***	0.001 (0.008)
#communities $\hat{\beta}_3$	0.078 (0.067)	0.034 (0.095)	−0.104 (0.059)	0.019 (0.018)
modularity $\hat{\beta}_4$	−0.034 (0.025)	**0.160** (0.035)***	**−0.059** (0.021)**	**−0.013** (0.006)*
R^2 marginal/conditional	0.672/0.751	0.168/0.813	0.278/0.624	0.000139/0.998
Var α_{jk}: class:problem (Intercept)	0.014	0.044	0.030	0.055
Var α_j: problem (Intercept)	0.000	0.254	0.000	1.338
Var ε_{ijk}: Residual	0.046	0.087	0.033	0.003

$^{***}p < 0.001$, $^{**}p < 0.01$, $^{*}p < 0.05$

of funnels can be explained by the fact that we are considering the runtime of successful runs only, that is, those that happen to start in the funnel containing the global optimum.

5.3 An Observational Study

Section 5.2 suggests that the strength of communities of optima can slow down ILS. Whether the *number* of clusters has an impact remains unaddressed. Consider the landscapes of a pair of instances which have comparable global topologies with the exception of community structure. Table 4 shows the features of pairs of such instances. There is a set of two for each of our four landscape classes (NK: $N = 18$ and $K \in \{2, 4\}$, QAP: 'real-like' and 'random-uniform'). Each set of two rows is a pair intended for comparison with one another. They were selected because they have similar numbers of local optima and the same number of funnels. Therefore, any difference in search performance could possibly be due to their different cluster structure.

Table 4: instances pairs with different community structure

class	#optima	#funnels	#mcl	#infomap	time
N18K2	18	1	1	2	661
N18K2	18	1	1	**6**	**1988**
N18K4	223	5	28	20	60 412
N18K4	254	5	**41**	22	**130 244**
QAPrl	162	1	3	9	6150
QAPrl	166	1	**8**	9	**15 535**
QAPu	993	2	33	**57**	**2 084 508**
QAPu	947	2	30	40	462 243

The *time* entries within the pairs in Table 4 — when viewed alongside the *#mcl* and *#infomap* entries — show that the instances with more optima clusters (the quantity shown in bold text) took longer to solve. This raises the question of whether community detection algorithms can uncover or estimate how many individual search space areas there are where stochastic flow could stagnate. Clearly, we cannot conclude anything statistically sound from these pairs, but they are interesting nonetheless.

5.4 Escaping from Funnels

Increasing the strength of ILS perturbation has been found to reduce the number of funnels and increase the success rate in a study of LONs for the travelling salesman problem [14]. We therefore, conducted experiments to assess whether searching on multi-funnel NK and QAP landscapes benefits from a stronger ILS perturbation.

Figure 4 shows ILS success rate with two perturbation strengths plotted against the number of local optima. Each dot represents an instance, both NK and QAP instances are grouped for a total of 120 dots in each plot. The perturbation strengths correspond to 2 and 3 moves, respectively (bit flips for NK, pairwise exchanges for QAP). The number of funnels in an instance is indicated by the dot colour and shape, as seen in the legend. On all single-funnel instances, the success rate is close to 100% (pale green diamonds), despite the variable number of local optima (ranging from 8 to 2003) in the single funnel instances. This suggests that the number of local optima is not a reliable predictor of search difficulty. Indeed, instances with a similar number of optima show a wide range of success rates. ILS can get trapped on instances with more than one funnel, with the lowest success rates observed for instances with 3 funnels or more. An increase in the perturbation strength from 2 to 3 basic moves (Figure 4 (right)), shows an increased success rate on the multi-funnel instances. This suggests that an increased or variable perturbation strength might be favourable when solving complex optimisation problems.

6 CONCLUSIONS

We have conducted an empirical study on two benchmark combinatorial problems to examine the similarity between clusters

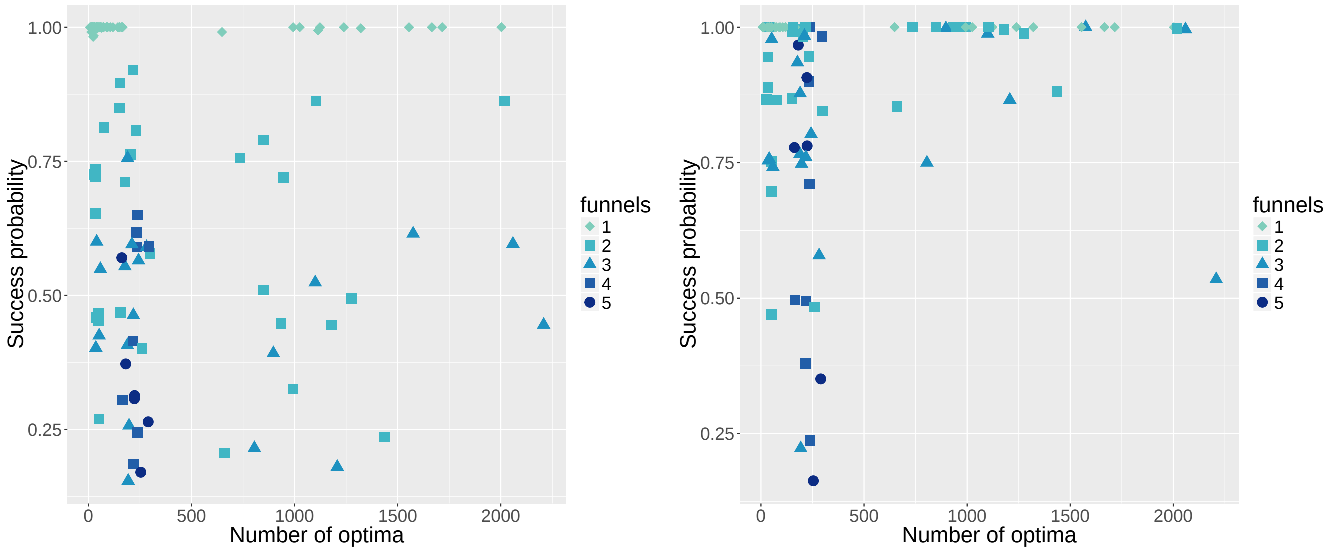


Figure 4: ILS success rates vs. number of local optima for all the studied instances and perturbation strengths: 2-moves (Left), and 3-moves (Right). The number of funnels in an instance is indicated by the dot colour and shape, as seen in the legend.

of optima and landscape funnels, and their respective impact on search. The results suggest that flow-based clusters of optima are not funnels and that the presence of multiple funnels in fitness landscapes contributes to lower success for common metaheuristics. Clusters of optima do not appear to impact success; however, if they are modular they seem to slow down optimisation for ILS. We argue that optima communities and funnels are related and hypothesize that there is scope for using monotonic sequences and cluster algorithms as landscape analysis tools for exposing search 'traps' in other types of landscapes. Clustering algorithms could possibly be used to estimate the runtime of a search — perhaps from quite a small sample of optima — because they identify areas where flow circulates. Our results imply that the number of local optima is not always a reliable indicator of search difficulty; their global distribution seems more important. Our next steps are to use this knowledge to construct a method for *predicting* funnel structure in combinatorial landscapes, such that a full enumeration of the search space is not required, as is the case with funnel detection using the local optima network model.

REFERENCES

[1] Thomas Bartz-Beielstein, Marco Chiarandini, Luís Paquete, and Mike Preuss. 2010. *Experimental methods for the analysis of optimization algorithms.* Springer.

[2] Douglas Bates, Martin Mächler, Ben Bolker, and Steve Walker. 2015. Fitting Linear Mixed-Effects Models Using lme4. *Journal of Statistical Software* 67, 1 (2015), 1–48. DOI: http://dx.doi.org/10.18637/jss.v067.i01

[3] R Stephen Berry and Ralph Breitengraser-Kunz. 1995. Topography and dynamics of multidimensional interatomic potential surfaces. *Physical review letters* 74, 20 (1995), 3951.

[4] Gabor Csardi and Tamas Nepusz. 2006. The igraph software package for complex network research. *InterJournal* Complex Systems (2006), 1695. http://igraph.org

[5] Fabio Daolio, Marco Tomassini, Sebastien Verel, and Gabriela Ochoa. 2011. Communities of minima in local optima networks of combinatorial spaces. *Physica A: Statistical Mechanics and its Applications* 390, 9 (2011), 1684–1694.

[6] J P K Doye, M A Miller, and D J Wales. 1999. The double-funnel energy landscape of the 38-atom Lennard-Jones cluster. *Journal of Chemical Physics* 110, 14 (1999), 6896–6906.

[7] Santo Fortunato. 2010. Community detection in graphs. *Physics reports* 486, 3 (2010), 75–174.

[8] DR Hains, L Darrell Whitley, and Adele E Howe. 2011. Revisiting the big valley search space structure in the TSP. *Journal of the Operational Research Society* 62, 2 (2011), 305–312.

[9] Sebastian Herrmann, Gabriela Ochoa, and Franz Rothlauf. 2016. Communities of Local Optima as Funnels in Fitness Landscapes. In *Proceedings of the 2016 on Genetic and Evolutionary Computation Conference.* 325–331.

[10] Jérémie Humeau, Arnaud Liefooghe, E-G Talbi, and Sébastien Verel. 2013. ParadisEO-MO: From fitness landscape analysis to efficient local search algorithms. *Journal of Heuristics* 19, 6 (2013), 881–915.

[11] Stuart Kauffman and Simon Levin. 1987. Towards a general theory of adaptive walks on rugged landscapes. *Journal of theoretical Biology* 128, 1 (1987), 11–45.

[12] J. Knowles and D. Corne. 2003. Instance Generators and Test Suites for the Multiobjective Quadratic Assignment Problem. In *Proceedings of the Evolutionary Multi-Criterion Optimization Conference (EMO 2003) (LNCS).* Springer, 295–310.

[13] Shinichi Nakagawa and Holger Schielzeth. 2013. A general and simple method for obtaining R2 from generalized linear mixed-effects models. *Methods in Ecology and Evolution* 4, 2 (2013), 133–142.

[14] Gabriela Ochoa and Nadarajen Veerapen. 2016. Additional Dimensions to the Study of Funnels in Combinatorial Landscapes. In *Proceedings of the 2016 on Genetic and Evolutionary Computation Conference (GECCO '16).* ACM, New York, NY, USA, 373–380. DOI: http://dx.doi.org/10.1145/2908812.2908820

[15] Gabriela Ochoa and Nadarajen Veerapen. 2016. Deconstructing the big valley search space hypothesis. In *European Conference on Evolutionary Computation in Combinatorial Optimization.* Springer, 58–73.

[16] Martin Rosvall, Daniel Axelsson, and Carl T Bergstrom. 2009. The map equation. *The European Physical Journal Special Topics* 178, 1 (2009), 13–23.

[17] Peter F Stadler. 2002. Fitness landscapes. In *Biological evolution and statistical physics.* Springer, 183–204.

[18] Marco Tomassini, Sebastien Verel, and Gabriela Ochoa. 2008. Complex-network analysis of combinatorial spaces: The NK landscape case. *Physical Review E* 78, 6 (2008), 066114.

[19] Stijn Van Dongen. 2000. A cluster algorithm for graphs. *Report-Information systems* 10 (2000), 1–40.