

How Perturbation Strength Shapes the Global Structure of TSP Fitness Landscapes

Paul McMenemy^{0000-0002-5280-425X}, Nadarajen Veerapen⁰⁰⁰⁰⁻⁰⁰⁰³⁻³⁶⁹⁹⁻¹⁰⁸⁰, and
Gabriela Ochoa⁰⁰⁰⁰⁻⁰⁰⁰¹⁻⁷⁶⁴⁹⁻⁵⁶⁶⁹

Computing Science and Mathematics, University of Stirling, Stirling, United Kingdom
{paul.mcmenemy,nadarajen.veerapen,gabriela.ochoa}@stir.ac.uk

Abstract. Local optima networks are a valuable tool used to analyse and visualise the global structure of combinatorial search spaces; in particular, the existence and distribution of multiple funnels in the landscape. We extract and analyse the networks induced by Chained-LK, a powerful iterated local search for the TSP, on a large set of randomly generated (Uniform and Clustered) instances. Results indicate that increasing the perturbation strength employed by Chained-LK modifies the landscape’s global structure, with the effect being markedly different for the two classes of instances. Our quantitative analysis shows that several funnel metrics have stronger correlations with Chained-LK success rate than the number of local optima, indicating that global structure clearly impacts search performance.

Keywords: Local Optima Network, Travelling Salesman Problem, Chained-LK, Perturbation Strength, Combinatorial Fitness Landscape

1 Introduction

Characterising the global structure of combinatorial search spaces remains a challenge, partly due to the lack of tools to study their complexity. Local optima networks (LONs) [1, 2] help to fill this gap by providing a modelling tool that compresses the fitness landscape into a network, where nodes are local optima and edges are possible search transitions between the local optima. Thus, LONs model the distribution and connectivity pattern of local optima. They help to characterise the underlying global landscape structure with a new set of metrics and visualisation tools [3].

Recent results using LONs challenge the existence of a ‘big-valley’ global structure on travelling salesman landscapes induced by iterated local search (ILS). Compelling evidence suggests instead that the big-valley decomposes into multiple valleys (or funnels) [4, 5]. This multi-funnel structure helps to explain why ILS can quickly find high-quality solutions, but fails to consistently reach the global optimum.

The notion of a *funnel* was introduced within the protein folding community to describe “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” [6]. It has been suggested that the energy landscape of proteins is characterised by a single deep funnel, a feature that underpins their ability to fold to their native state. In contrast, some shorter polymer chains (e.g., polypeptides) that misfold are expected to have other funnels that can act as traps. Similarly, recent studies on TSP [4, 5] show that landscapes with more than one funnel, where the global optimum is located in a deep, narrow funnel, are significantly harder to solve for ILS.

Iterated local search is a simple yet powerful search strategy. It works by alternating an intensification stage (local search) with a diversification stage (perturbation). *Chained Lin-Kernighan* (Chained-LK) [7] is an effective ILS for TSP, combining the powerful Lin-Kernighan local search [8] with a type of 4-exchange perturbation (*double-bridge*, depicted in Fig. 1b). A key factor in any ILS implementation is the strength of the perturbation, which is related to the number of solution components that are modified simultaneously [9]. A recent study based on local optima networks from NK fitness landscapes shows that a properly selected perturbation strength can help overcome the effect of ILS becoming trapped in clusters of local optima (which are related to funnel structures) [10]. This has implications for the design of effective ILS approaches, where normally only small perturbations or complete restarts are applied, with the middle ground of intermediate perturbation strengths largely unexplored.

The main goal of this study is to model the LONs induced by Chained-LK on two classes of randomly generated TSP instances (Uniform and Clustered cities) for increasing perturbation strengths. More specifically, the contributions are:

1. A rigorous, empirical characterisation and comparison of the global structure of TSP instances with increasing perturbation strength.
2. Identification of the most effective perturbation strength for different TSP instance classes and sizes.
3. A correlation study identifying connections between empirical search performance and the global structure of TSP instances with increasing perturbation strength.

2 Definitions and Algorithms

The search space for a TSP instance with m cities is the set of permutations of legitimate Hamiltonian tours of these m cities; the number of tours is factorial in m . The fitness function f is given by the length of the tour, which is to be minimised for optimality.

The Lin-Kernighan (LK) heuristic is a powerful local search algorithm, one which is based on the idea of k -exchange moves: take the current tour and remove k different links; these are then reconnected in a new way to achieve a legal tour. A tour is considered to be ‘ k -opt’ if no k -exchange exists which decreases its fitness value. LK applies 2, 3 and higher-order k -exchanges: Fig. 1a illustrates a 2-exchange move. The order of an exchange is not predetermined, rather k is

increased until a stopping criterion is met. Thus many kinds of k -exchanges and all 3-exchanges are included.

Chained Lin-Kernighan (Chained-LK) is an iterated local search where the current solution obtained by LK is perturbed (or kicked) and LK is reapplied. If the new local optimum solution is an improvement, the old solution is discarded and the new one retained; otherwise, the search continues with the old tour and ‘kicks’ it again. The kick or escape operator in Chained-LK is a type of 4-exchange (depicted in Fig. 1b), named *double-bridge* by Martin et al. [11].

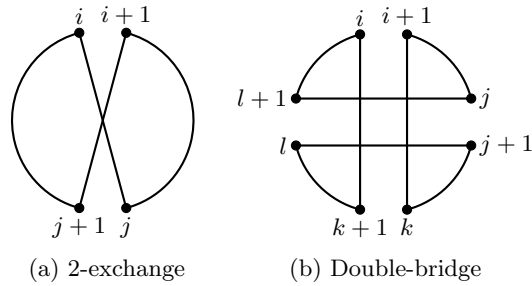


Fig. 1: Illustration of tours obtained after 2-exchange and double-bridge moves.

Fitness Landscape. A landscape [12] is a triplet (S, N, f) where S is a set of potential solutions, i.e., a search space; N is a neighbourhood structure, $N : S \rightarrow 2^S$, a function that assigns to every $s \in S$ a set of neighbours $N(s)$; $f : S \rightarrow \mathbb{R}$ is a fitness function.

Local Optima. A local optimum is a solution s^* such that $\forall s \in N(s^*), f(s^*) \leq f(s)$. The set of local optima, which corresponds to the set of nodes in the network model, is denoted by L . Since the whole set of local optima cannot be determined in realistic search spaces, such as those considered here, a process of sampling is required to estimate L .

Escape Edges. These are directed edges, where there exists an *escape* edge from local optimum x to local optimum y , only if y can be obtained after a double-bridge move applied to x and is then followed by LK. The type of edge is dependent on the algorithm used to sample the landscape. While escape edges are natural edges for ILS algorithms, other edge types may be used such as crossover and mutation edges in the case of an evolutionary algorithm [13]. The set of escape edges obtained is denoted by E . Note that during the sampling process only edges that correspond to monotonically improving transitions are stored (see Algorithm 1).

Local Optima Network (LON). This is the graph $LON = (L, E)$ where all nodes are the local optima L , and edges E are the directed escape edges.

In some cases, neutrality may be observed at the LON level, i.e., there exists connected nodes that share the same fitness. This leads to an even coarser model of the landscape [14], where these LON plateaus are compressed into single nodes.

The set of LON plateaus, $CLp = \{clp_1, clp_2, \dots, clp_n\}$ corresponds to the node set of a compressed local optima network. A node without neighbours that share the same fitness is also considered a LON plateau.

Compressed Local Optima Network (CLON). The weighted, oriented local optima network $CLON = (CLp, E)$ is the graph where the nodes $clp_i \in CLp$ are the LON plateaus. Weighted edges correspond to the aggregation of the multiple edges from nodes in a plateau to single edges in the compressed network.

Sink. A sink is a $CLON$ node without outgoing edges.

Funnel. The funnel of some sink is empirically defined as the set of nodes for which there exists a path of monotonically decreasing fitness between the nodes and the sink.

```

Data:  $I$ , TSP instance;  $z$ , the kick strength
Result:  $L$ , set of local optima;  $E$ , set of escape edges
 $L \leftarrow \{\}$ ;  $E \leftarrow \{\}$ 
for  $i \leftarrow 1$  to 1000 do                                /* loop across 1000 runs */
     $s_{start} \leftarrow \text{LK}(\text{initialFeasibleSolution}(I))$ 
     $L \leftarrow L \cup \{s_{start}\}$ ;  $j \leftarrow 0$ 
    while  $j < 10000$  do                                    /* termination criterion of run */
         $s_{end} \leftarrow \text{applyKick}(s_{start}, z)$           /* apply kick of strength  $z$  */
         $s_{end} \leftarrow \text{LK}(s_{end})$ 
         $j \leftarrow j + 1$ 
        if  $\text{Objective}(s_{end}) \leq \text{Objective}(s_{start})$  then
             $L \leftarrow L \cup \{s_{end}\}$                     /* aggregate the nodes */
             $E \leftarrow E \cup \{(s_{start}, s_{end})\}$       /* aggregate the edges */
             $s_{start} \leftarrow s_{end}$ 
             $j \leftarrow 0$ 
        end
    end
end

```

Algorithm 1: LON sampling aggregating 1 000 runs of Chained-LK.

3 Empirical Methodology

We empirically examine the global structure of two classes of synthetic instances by considering a sample of the search space. Clearly, exhaustive enumeration of

the search space is not possible when instances of non-trivial sizes are considered. We therefore choose to sample high quality local optima obtained. Local optima networks were constructed using the Chained Lin-Kernighan heuristic as described above, and implemented with the Concorde TSP solver [15, 16].

3.1 Instances

The instances considered here are generated using the DIMACS TSP instance generator¹. Two classes of synthetic instances were generated: ones where the cities are uniformly distributed (prefixed by ‘E’) and ones where the cities are clustered (prefixed by ‘C’). These two classes are further subdivided into three non-trivial instance sizes: 506, 755 and 1010 cities [4]. For each class-size combination, thirty different instances are generated; therefore a total of 180 instances is considered. A subset of these instances, and their associated networks, was created in the context of our previous work [4].

3.2 Sampling Method

The algorithm used to sample the search space is an instrumented version of the Chained-LK implementation found in Concorde [15, 16]. Concorde is currently the state-of-the-art in exact TSP solvers and uses the Chained-LK heuristic to generate a good upper bound for its branch-and-bound process. In addition, we used Concorde to compute a global optimum (and its associated fitness value) for each instance.

The Chained-LK implementation is modified [4] to record all the local optima found during a run (set L), as well as the order in which the local optima were generated (set E), i.e., which local optimum was obtained after some other local optimum was perturbed and then improved through LK. The termination criterion of the algorithm is 10 000 consecutive non-improving moves. The information for a total of 1 000 runs is subsequently aggregated in order to generate the corresponding LON. The pseudocode is given in Algorithm 1.

For each instance, we separately sample the landscapes for ten perturbation strengths ($z \in \{1, 2, \dots, 10\}$). The value of z is the number of consecutive double-bridge kicks that are applied in order to perturb a local optimum. The perturbations are therefore fairly disruptive, even at low strength. The total sampling process was relatively computationally expensive, requiring approximately 50 000 hours of computation time on Intel Xeon X5650 2.66 GHz processors.

3.3 Performance and Network Metrics

There are multiple performance and network metrics that can be computed and used to understand search difficulty and landscape structure; however, not all of them are relevant. We explored a number of metrics and selected a subset that allows us to meaningfully describe the relationship between search difficulty and

¹ dimacs.rutgers.edu/Challenges/TSP/download.html

landscape structure. Table 1 summarises each of these metrics, and which are described in some additional detail below.

Table 1: Definitions of performance and network metrics

performance metrics	<i>success</i>	Success rate of finding the <i>a priori</i> global optimum
	<i>iters</i>	Mean no. of Chained-LK iterations to find global optimum
network metrics	<i>n</i>	Number of local optima within the LON
	<i>n_{sinks}</i>	Number of sinks within the LON
	<i>d</i>	Relative in-strength of the globally optimal sinks
	<i>fun</i>	Proportion of nodes comprising the globally optimal funnels
	<i>u_{fit}</i>	Proportion of local optima with unique fitness values

Algorithmic performance is measured through its *success* rate; i.e., the proportion of runs that reached a global optimum, and through the mean number of iterations (*iters*) required to reach a global optimum.

The structure of a LON is assessed using five characteristics. The number of local optima (*n*), or nodes, within the LON describes its size. A sink describes a solution in which the search becomes trapped at the end of a funnel; the corresponding metric we record is *n_{sinks}*, the number of sinks.

In addition, the weighted incoming degree, or in-strength, of each sink is computed and the relative in-strength of the globally optimal sinks (*d*) is reported. This corresponds to the ratio of the sum of the in-strengths of the globally optimal sinks to the sum of the in-strengths for all sinks.

The nodes in a funnel, in the uncompressed LON, are identified by selecting one of the nodes that constitutes a sink and performing a breadth-first search. We thus define *fun* as the proportion of nodes of the LON within funnels that lead to a global optimum.

Finally, *u_{fit}* is the proportion of local optima with unique fitness values, i.e., the number of unique fitness values divided by the total number of unique local optima.

4 Results and Analysis

In this section, we examine the structure of the sampled LONs and the interplay with search difficulty. This is initially achieved through visualisation, to intuitively reveal structural differences; and then through a more traditional analysis of the performance and network metrics.

4.1 Visualisation

One advantage of modelling fitness landscape as LONs is the possibility of network visualisation, a valuable first approach in exploring and understanding their structure. Figure 2 illustrates LONs for two representative instances with 755 cites. Specifically, we considered one Uniform (E755.81) and one Clustered (C755.81) instance, and visualise the networks for three different perturbation strengths, $z \in \{1, 5, 10\}$. Each node corresponds to an LK optimum, and edges represent search transitions with the corresponding perturbation strength. Note that other instances were examined, and produced similar results.

In order to produce images of manageable size, each plot represents a sub-network of the sampled network. Nodes were kept whose fitness was within 5% of the evaluation of the global optimum. This visualises the connectivity pattern of solutions nearby the global optimum, which is arguably the most interesting part of the search space, and any competent heuristic method should attain this portion of the landscape.

Network plots were produced using the R statistical language together with the **igraph** package [17]. Graph layouts employ *force-directed* methods, and resultant visualisations are decorated to reflect features relevant to search dynamic. It is important to remember that a LON can be seen as a representation of the stochastic process of a search algorithm for a particular problem instance. In Fig. 2, red nodes correspond to local optima belonging to the funnel containing the global optimum, whereas blue nodes indicate optima belonging to suboptimal funnels. The bottom of the funnels (sinks) are highlighted with a black outline and a node size proportional to their incoming strength (weighted degree). Nodes which are not sinks are visualised with a grey outline and fixed size.

A visual inspection of the LONs in Fig. 2 reveals clear structural differences between the global shape of the studied landscapes, as well as the perturbation strengths. For the Uniform instances (left plots), the global structure and overall success rate of Chained-LK seems very robust to increasing perturbation strength. Interestingly, the success rate achieved with the standard single double-bridge kick ($z = 1$) is significantly lower than that achieved when $z = 5$. This can be explained by the existence of a suboptimal funnel seen in blue (Fig. 2(a)) which is disjoint from the global sink in red, and whose sink in blue has incoming strength of similar magnitude (visualised as the size of the node). Therefore this suboptimal funnel acts as a trap for the search process. Once the algorithm reaches the blue sink, it cannot escape using a single kick and thus fails to locate the global optimum. The intermediate kick strength ($z = 5$) shown in Fig. 2(c) provides the best success rate for this instance, supported by the single funnel observed. Thus the $z = 5$ perturbation seems to fuse the previous two funnels into one. However, too strong a perturbation deteriorates performance (Fig. 2(f)), which can be explained by the appearance of several suboptimal sinks (blue nodes), albeit each of small incoming strength.

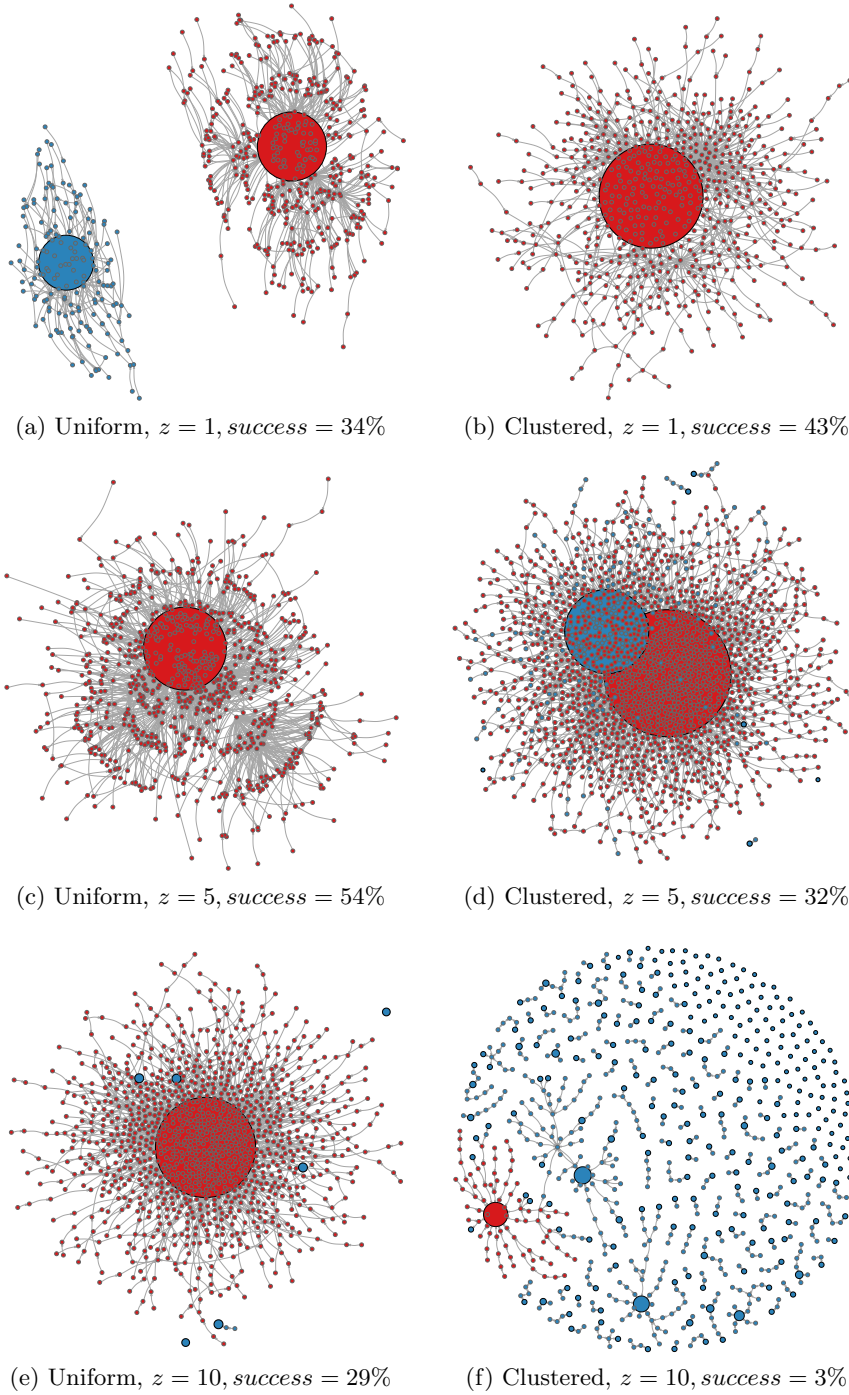


Fig. 2: LONs for two representative TSP instances, E755.81 (Uniform, left) and C755.81 (Clustered, right), and three perturbation strengths $z \in \{1, 5, 10\}$. $success$ percentages of Chained-LK are indicated. Nodes are LK local optima and edges are transitions using the respective perturbation strength. Red nodes belong to the global optimal funnel, while blue nodes to suboptimal funnels. The size of sink nodes is proportional to their incoming strength.

4.2 Performance & Network Metrics Results

The metrics previously defined in Table 1 are now examined in detail. Figure 3 provides boxplots of the metric values for the six TSP class-size instances. The metrics are shown grouped by class and are vertically descending by size (506, 755, 1010). Results for each metric presented in Fig. 3 are discussed hereafter:

success A significant difference is observed between the distributions of *success* for each class-size combination. The *success* rates of Uniform instances for each of the three sizes can be approximately characterised as concave, that is they follow an increasing then decreasing behaviour as z increases. However, the Clustered results all decay from maximum values of *success* when z is low, diminishing towards zero as $z \rightarrow 10$. The variability of *success* results also narrow as instance size increases for both instance classes.

iters For each class-size combination, an approximately monotonic increase in the iterations of Chained-LK is observed as the strength of z increases. However, the Clustered 1010 results peak when $z = 7$; the *iters* begin to decrease when $z \geq 8$. This is a consequence of the very low success rate for these instances, which is close to zero for these values of z . When Chained-LK does obtain the global optimum for these instances, it does so quickly due to the small size of the globally optimal funnel (c.f. Fig. 2(f)). The Uniform classes all exhibit lower *iters* values almost without exception when compared with the Clustered classes; only when z is low is there any overlap in the Uniform and Clustered interquartile boxes between comparable results.

d The relative in-strengths of the funnels leading to the global optima exhibit similar behaviours to that of the corresponding success rates for both Uniform and Clustered classes. This is particularly apparent when comparing the boxplots of *success* and d for instance size 1010. Note that the magnitude of d decreases as instance size increases.

n All plots in Fig. 3 exhibit values of local optima that decrease as kick strength increases. As detailed in the sampling method (Section 3.2), a perturbation away from the current local optimum employs z double-bridge moves; as z increases, this process becomes increasingly destructive of the TSP tour relevant to the current local optimum. This results in the perturbation stage relocating the search to a location on the fitness landscape that is increasingly distant from the current solution. This results in Chained-LK becoming less likely to discover a non-deteriorating solution after each iteration. Thus, the number of new local optima generated would reduce as z increased. Note that the Clustered instances again return greater numbers of local optima when compared to the Uniform results.

fun As with d , the proportional size of a funnel containing the global optimum follows a similar decay pattern to *success* as z increases.

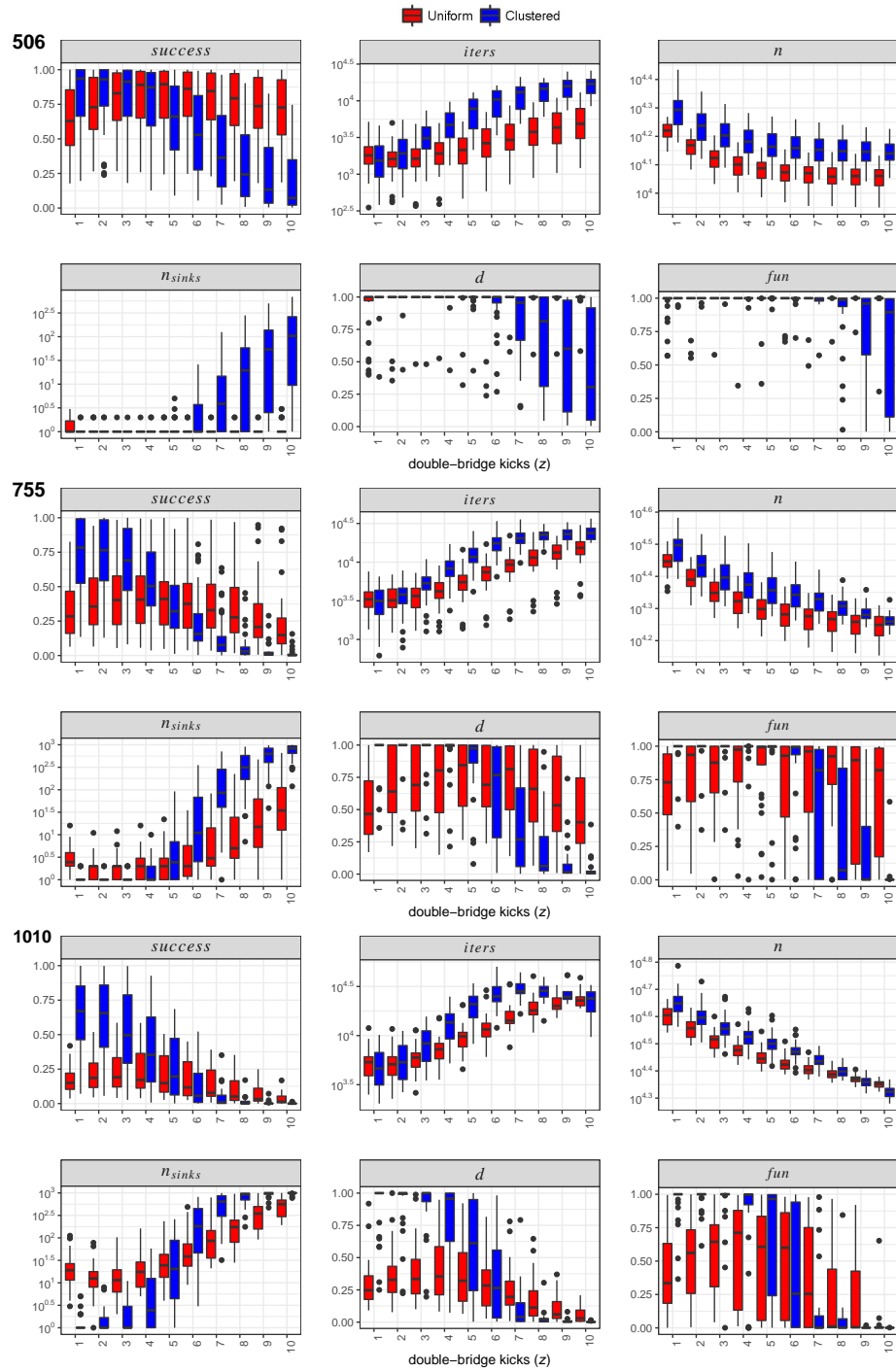


Fig. 3: Metrics of TSP Uniform and Clustered instances, of size 506, 755 and 1010, with increasing double-bridge kicks (z). Note the \log_{10} vertical scale on some of the plots.

n_{sinks} We first consider the number of sinks for the Uniform class as z increases. When instance size is 506, only a small number of sinks are observed; however, as instance size increases to 755 and 1010, the value of n_{sinks} also increases. In actuality, the number of sinks decreases from $z = 1$ to $z = 3$, and then begins to increase as $z \rightarrow 10$. These results are in accordance with the visualisations in Fig. 2, which also demonstrate that small perturbation strengths could meld together large but few sinks. The number of sinks within the Clustered instances all follow a logistic growth pattern as perturbation strength z increases. Both instance classes approximate reciprocal n_{sinks} behaviours to that of the performance metric *success*; therefore it is expected that there would exist a strong, negative correlation between these metrics.

4.3 Impact of Perturbation Strength on Success Rate

Considered together, the three upper left plots for each size shown in Fig. 3 provide indicators as to how the *success* rates alter as TSP instance size increases. The median values from the distributions of *success* are reproduced in Fig. 4; graphed by type (Uniform and Clustered), and then as connected lines by instance size (506, 755, 1010).

It is apparent from Fig. 4 that, as instance size increases, the rate of *success* decreases for all class types and all double-bridge kick strengths. The E506 (Uniform) median *success* rates all exhibit relatively large values, ranging from a high of 89.5% for kick strength $z = 5$ to a lowest rate of 62.9% when $z = 1$. This could be a consequence of using too low a perturbation z , one that is not strong enough to kick sufficiently ‘far’ away from the current optimum and so failing to escape to a new, non-deteriorating local optimum, even if the current optimum is in a relatively shallow funnel. Thus, for low values of z , the perturbation stage of Chained-LK is not destructive enough of the current solution to escape shallow traps, and continue on to find the global optimum.

A marked decrease in the median *success* values are observed for Uniform TSP instance sizes increasing from 755 to 1010. This increase is primarily a consequence of the increased magnitude of the solution space. The maximum median *success* for each of E506, E755, E1010 occur when double-bridge kick strengths are at $z = 5$ (median = 0.895), $z = 5$ (median = 0.410) and $z = 3$ (median = 0.190) respectively.

Similar analysis of the Clustered TSP instances shows notably different behaviour in the values of median *success* values for $z \in \{1, 2, \dots, 10\}$. Where the Uniform TSP instances’ median *success* values exhibit an approximately concave shape, the Clustered median *success* values conform to a complemented sigmoid pattern. This locates each Clustered instance’s maximum, median value when $z = 1$; the median *success* values then decay towards zero as $z \rightarrow 10$.

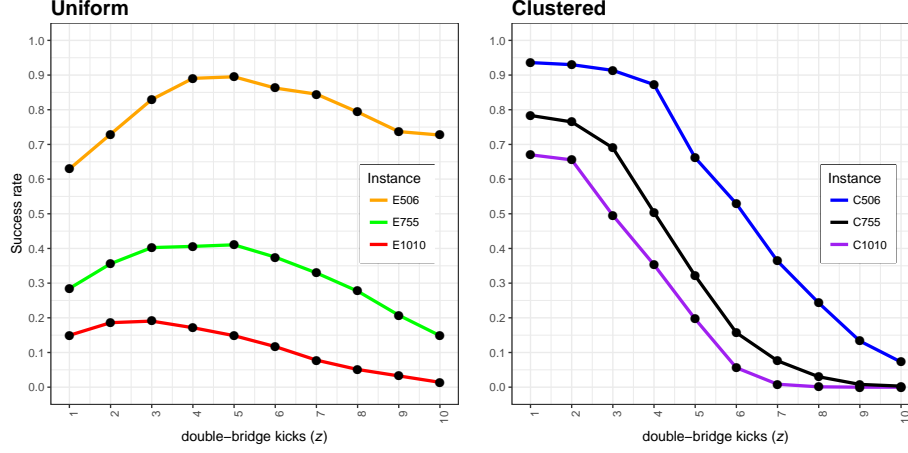


Fig. 4: Success rates by TSP instance class for increasing instance sizes. Points shown are medians of *success* for each class-kick combination.

As TSP instance size increases, both Uniform and Clustered classes experience a significant reduction in the median *success* rates, as well as a phase shift of their patterns towards the vertical axis. Extrapolating from these behaviours we would reasonably expect that, as TSP instance size increases beyond 1010, *success* rates would continue to degrade; maximum, median *success* would thus occur for lower values of z . Conversely, for smaller TSP instance sizes, *success* rates would logically approach 100%, with close to 100% *success* rates expected for reasonable values of z as TSP size decreased.

4.4 Correlation Analysis

Analysis of correlations between the performance metric *success* and the other metrics *iters*, n , n_{sinks} , d , *fun* and u_{fit} was undertaken. Figure 5 shows the correlation matrix for these metrics (for C755, or Clustered TSP instances of size 755, only), with the upper triangle of the matrix providing Spearman correlation coefficients for each metric pair combination. Asterisks next to each correlation coefficient denote the p-value significance, where *** indicates a p-value < 0.001 , ** a p-value < 0.01 , and a single * denotes p-value < 0.05 . The lower triangle of Fig. 5 shows pairwise scatter plots of each metric-metric combination. Pairwise scatter points shown are aggregated together from each instance and kick. Correlations matrices for all 6 type-size combinations were produced but not shown; however, the correlations for C755 are indicative of all instances, and so is the only matrix shown here for brevity.

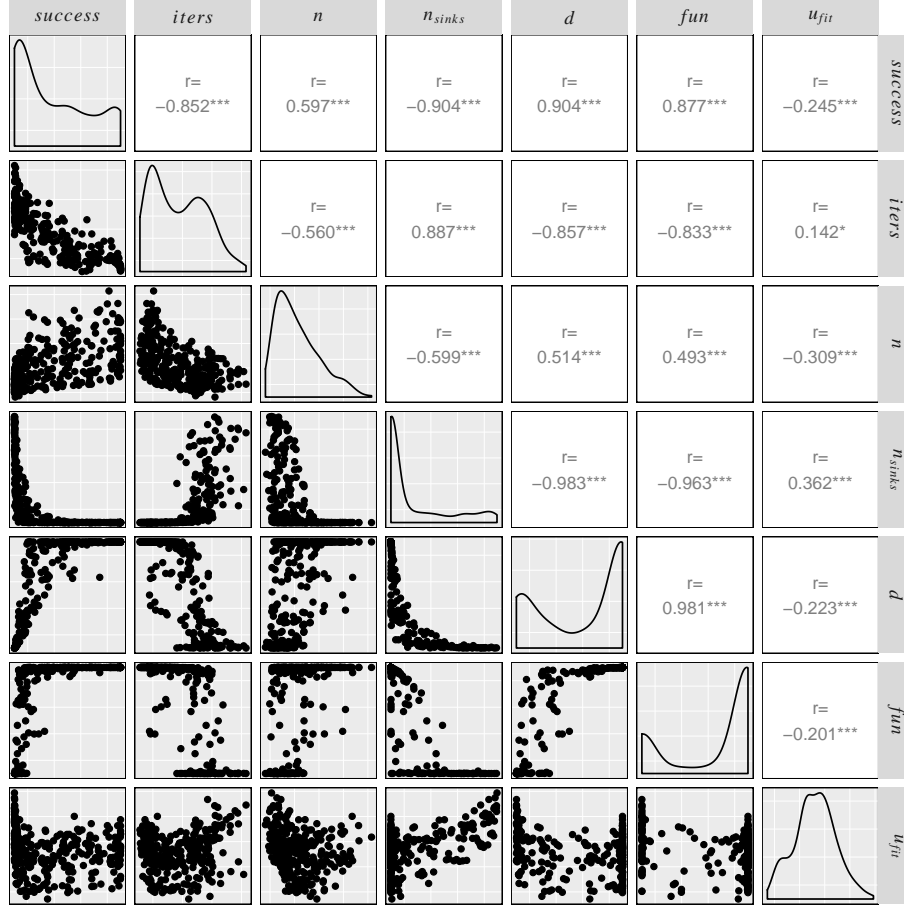


Fig. 5: Correlation matrix for combined Clustered instances C755.

We are primarily interested in correlations between *success* and the other metrics, and so need only consider coefficients shown in the first row of Fig. 5 and scatter plots within its first column. In the first row, we observe high positive correlations between *success* and the network metrics *d* and *fun*: the relative in-strength and funnel sizes of the global optima respectively. Strong negative correlations with *success* are seen with *iters* (iterations to global optimum) and *n_{sinks}* (number of sinks). Only *success* correlations with number of local optima (*n*) and the proportion of nodes with unique fitness (*u_{fit}*) exhibit coefficient values that could be described as being of moderate significance.

Note that all correlation coefficients provided here were calculated using Spearman's approach rather than Pearson's; thus coefficients reflect correlations based on monotonically increasing or decreasing relationships rather than linear interactions only. However, all correlations were recalculated using Pearson's

methodology, and no significant differences can be reported except for an across-the-board small reduction in the magnitudes of the coefficients.

4.5 Correlation Variance Between Instance Classes

Correlations of success rates in finding a global optimum (*success*) with the other metrics are grouped by class (Uniform, Clustered) and by TSP size and are shown as parallel plots in Fig. 6. This allows easy comparison of *success* correlates between the TSP classes. For both Uniform and Clustered classes, similar behaviours transpire for each of the metrics correlated to *success*. Increasing from 506 to 755 then to 1010 TSP sizes, we observe positive, strengthening *success* correlations with d , fun ; strengthening, negative correlations are observed with $iters$ and n_{sinks} . Both the number of local optima (n) and neutrality at the local optima level (u_{fit}) undergo changes in the directions of their correlation as TSP size increases, both transiting coefficient values of zero and thus emphasising their weak correlation with *success* rate.

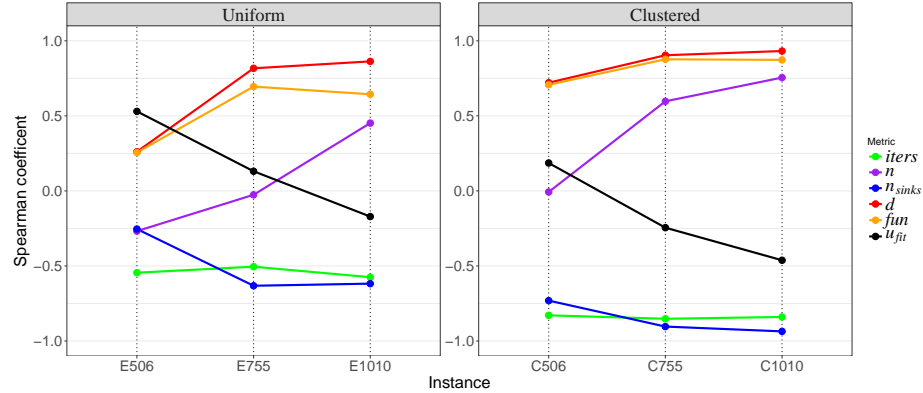


Fig. 6: Spearman correlation coefficient values of *success* versus other metrics by class.

Extrapolating these correlation behaviours beyond the TSP sizes we have analysed here would be inappropriate. We observed an unexpected decrease in the number of iterations for the C1010 instance (c.f. Fig. 3) as $z \rightarrow 10$, something that could not have been readily anticipated from the C506 and C755 results. Therefore, making assumptions about trends from these results as TSP size increases would not be advisable; studies of larger TSP sizes would be required to further understand the relationships between algorithm performance and LON structure.

5 Conclusions

We have extracted and analysed the local optima networks induced by Chained-LK with increasing perturbation strengths on two classes of randomly generated TSP instances, with Uniform and Clustered cities. Particularly, we characterised, visualised and measured the multi-funnel global structure of the underlying fitness landscapes. We have also measured the performance of Chained-LK on the studied instances with increasing perturbation strengths. Our results reveal that the landscape global structure (i.e., the funnel structure) changes with increasing perturbation strength, and these changes are markedly different in the two instance classes. For the Clustered instances, a smoother landscape (a single or a low number of funnels) is achieved with a minimal perturbation, whereas a strong perturbation is ineffective and breaks the landscape into a multitude of suboptimal funnels. In contrast, for the Uniform instances, the lowest perturbation induces a multi-funnel landscape, while a moderate perturbation can smooth the landscape and improve performance. This has implications for algorithm design: the amount of perturbation for an effective search is related to the instance class. However, we found that the standard perturbation strength of Chained-LK (a single double-bridge move) is robust across the instance classes and sizes studied.

Provided the most effective perturbation strength is selected for each instance class, the Clustered instances are generally easier to solve (have a higher success rate) across all instances sizes. However, the Clustered instances reveal a larger number of local optima according to our sampling, as compared to the Uniform instances across all sizes. If the number of local optima was an accurate predictor of search difficulty, we would expect that the Clustered instances would be harder to solve than the Uniform instances. The opposite is the case, which confirms that the distribution of optima is more important than their number for explaining search difficulty. Indeed, we found that the funnel metrics (the number of funnels, the size and strength of the global optima funnel) have a stronger correlation with search difficulty than the number of local optima. Global structure definitely impacts search performance.

Future work will consider other classes of TSP instances including real-world datasets. Additional problem domains will also be considered, as well as the impact on global structure of alternative perturbation and recombination operators.

Acknowledgements. This work was supported by the Leverhulme Trust [award number RPG-2015-395] and by the UK’s Engineering and Physical Sciences Research Council [grant number EP/J017515/1]. Results were obtained using the EPSRC-funded ARCHIE-WeSt High Performance Computer (www.archie-west.ac.uk, EPSRC grant EP/K000586/1).

Data Access. All data generated for this research are openly available from the Stirling Online Repository for Research Data (<http://hdl.handle.net/11667/104>).

References

1. Ochoa, G., Tomassini, M., Verel, S., Darabos, C.: A Study of NK Landscapes' Basins and Local Optima Networks. In: Proceedings of the Genetic and Evolutionary Computation Conference (GECCO). pp. 555–562. ACM (2008)
2. Verel, S., Ochoa, G., Tomassini, M.: Local Optima Networks of NK Landscapes With Neutrality. *IEEE Transactions on Evolutionary Computation* 15(6), 783–797 (2011)
3. Newman, M.E.J.: *Networks: An Introduction*. Oxford University Press, Oxford, UK (2010)
4. Ochoa, G., Veerapen, N.: Mapping the global structure of TSP fitness landscapes. *Journal of Heuristics* pp. 1–30 (2017)
5. Ochoa, G., Veerapen, N.: Deconstructing the Big Valley Search Space Hypothesis. In: *Evolutionary Computation in Combinatorial Optimization, EvoCOP 2016*. Lecture Notes in Computer Science, vol. 9595, pp. 58–73. Springer (2016)
6. Doye, J.P.K., Miller, M.A., Wales, D.J.: The double-funnel energy landscape of the 38-atom Lennard-Jones cluster. *Journal of Chemical Physics* 110(14), 6896–6906 (1999)
7. Applegate, D., Cook, W., Rohe, A.: Chained Lin-Kernighan for Large Traveling Salesman Problems. *INFORMS Journal on Computing* 15, 82–92 (2003)
8. Lin, S., Kernighan, B.W.: An Effective Heuristic Algorithm for the Traveling-Salesman Problem. *Operations Research* 21, 498–516 (1973)
9. Lourenço, H.R., Martin, O.C., Stützle, T.: Iterated Local Search. In: *Handbook of Metaheuristics*, pp. 320–353. Kluwer Academic Publishers, Boston (2003)
10. Herrmann, S., Herrmann, M., Ochoa, G., Rothlauf, F.: Shaping communities of local optima by perturbation strength. In: *Genetic and Evolutionary Computation Conference, GECCO*. pp. 266–273 (2017)
11. Martin, O., Otto, S.W., Felten, E.W.: Large-step markov chains for the TSP incorporating local search heuristics. *Operations Research Letters* 11, 219–224 (1992)
12. Stadler, P.F.: Fitness landscapes. *Appl. Math. and Comput* 117, 187–207 (2002)
13. Veerapen, N., Ochoa, G., Tinós, R., Whitley, D.: Tunnelling Crossover Networks for the Asymmetric TSP. In: Handl, J., Hart, E., Lewis, P.R., López-Ibáñez, M., Ochoa, G., Paechter, B. (eds.) *Parallel Problem Solving from Nature — PPSN XIV*, Lecture Notes in Computer Science, vol. 9921, pp. 994–1003. Springer, Cham (2016)
14. Ochoa, G., Veerapen, N., Daolio, F., Tomassini, M.: Understanding Phase Transitions with Local Optima Networks: Number Partitioning as a Case Study. In: Hu, B., Lopez-Ibez, M. (eds.) *Evolutionary Computation in Combinatorial Optimization*. Lecture Notes in Computer Science, vol. 10197, pp. 233–248. Springer, Cham (2017)
15. Applegate, D.L., Bixby, R.E., Chvátal, V., Cook, W.J.: *The Traveling Salesman Problem: A Computational Study*. Princeton University Press (2007)
16. Applegate, D., Bixby, R., Chvátal, V., Cook, W.: Concorde TSP solver (2003), <http://www.math.uwaterloo.ca/tsp/concorde.html>
17. Csardi, G., Nepusz, T.: The igraph software package for complex network research. *InterJournal Complex Systems*, 1695 (2006), <http://igraph.org>