# Fractal Dimension and Perturbation Strength: A Local Optima Networks View

Sarah L. Thomson[1], Gabriela Ochoa[1], and Sébastien Verel[2]

[1]University of Stirling, United Kingdom
{s.l.thomson, gabriela.ochoa}@stir.ac.uk
[2]Université du Littoral Côte d'Opale (ULCO), France
verel@univ-littoral.fr

**Abstract.** We study the effect of varying perturbation strength on the fractal dimensions of Quadratic Assignment Problem (QAP) fitness landscapes induced by iterated local search (ILS). Fitness landscapes are represented as Local Optima Networks (LONs), which are graphs mapping algorithm search connectivity in a landscape. LONs are constructed for QAP instances and fractal dimension measurements taken from the networks. Thereafter, the interplay between perturbation strength, LON fractal dimension, and algorithm difficulty on the underlying combinatorial problems is analysed. The results show that higher-perturbation LONs also have higher fractal dimensions. ILS algorithm performance prediction using fractal dimension features may benefit more from LONs formed using a high perturbation strength; this model configuration enjoyed excellent performance. Around half of variance in Robust Taboo Search performance on the data-set used could be explained with the aid of fractal dimension features.

**Keywords:** Local Optima Network, Fractal Dimension, Quadratic Assignment Problem, QAP, Iterated Local Search, Perturbation Strength, Fitness Landscapes

## 1 Introduction

Many systems can be characterised by their *fractal* geometry. Fractals are patterns which contain parts resembling the whole [1]. This kind of geometry is non-Euclidean in nature and a non-integer dimension can be computed for a pattern — the *fractal* dimension. This is an index of spatial complexity and captures the relationship between the amount of detail and the scale of resolution the detail is measured with. Not all systems can be characterised by a single fractal dimension, however [2] and multiple fractal dimensions — a spectrum — can be obtained through *multifractal* analysis. If there is diversity within the spectrum, this is an indication that the pattern is multifractal; i.e., the spatial complexity may be heterogeneous in nature.

Local Optima Networks (LONs) [3] are a tool to study fitness landscapes. The nodes are local optima, and the edges are transitions between local optima

This is a post-peer-review, pre-copyedit version of a paper published in Rudolph, G., Kononova, A.V., Aguirre, H., Kerschke, P., Ochoa, G., Tušar, T. (eds) *Parallel Problem Solving from Nature – PPSN XVII. PPSN 2022*. Lecture Notes in Computer Science, vol 13398. Springer, Cham, pp. 532-574. The final authenticated version is available online at: https://doi.org/10.1007/978-3-031-14714-2_39

using a given search operator. Analysing the features of LONs can help explain algorithm search difficulty on the underlying optimisation problem. LONs have been subject to fractal analysis previously [4]; results have suggested that their fractal dimension, and extent of multifractality, may be linked to increased search difficulty.

The connection between perturbation strength and fractal dimension in LONs has not been studied before. We speculate that there may be some untapped knowledge concerning algorithm performance explanation in this area, and advance towards this aim in the present work.

The Quadratic Assignment Problem (QAP) — a benchmark combinatorial optimisation domain — is used for this study. We extract LONs with low and high perturbation strength, then compute fractal dimension features from them. Separately, two metaheuristics (iterated local search and robust taboo search) are executed on the QAP instances to collect algorithm performance information. The interplay between perturbation strength, fractal dimensions, and algorithm performance is then examined.

## 2 Methodology

### 2.1 Quadratic Assignment Problem

*Definition.* A solution to the QAP is generally written as a permutation $s$ of the set $\{1, 2, ..., n\}$, where $s_i$ gives the location of item $i$. Therefore, the search space is of size $n!$. The cost, or fitness function associated with a permutation $s$ is a quadratic function of the distances between the locations, and the flow between the facilities, $f(s) = \sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij} b_{s_i s_j}$, where $n$ denotes the number of facilities/locations and $A = \{a_{ij}\}$ and $B = \{b_{ij}\}$ are the distance and flow matrices, respectively.

*Instances.* We consider the instances from the QAPLIB[1] [5] with between 25 and 50 facilities; these are of moderate size, and yet are not always trivial to solve. Some of the instances in this group have not been solved to optimality; for those, we use their best-known fitness as the stand-in global optimum. In the rest of this paper, for simplicity we **refer to these as the global optimum**. According to [6,7], most QAPLIB instances can be classified into four types: uniform random distances and flows, random flows on grids, real-world problems, and random real-world like problems. All of these are present in the instance set used in this work.

### 2.2 Monotonic Local Optima Networks

**Monotonic LON.** Is the directed graph $MLON = (L, E)$, where nodes are the local optima $L$, and edges $E$ are the monotonic perturbation edges.

[1] `http://www.seas.upenn.edu/qaplib/`

**Local optima.** We assume a search space $S$ with a fitness function $f$ and a neighbourhood function $N$. A local optimum, which in the QAP is a minimum, is a solution $l$ such that $\forall s \in N(l)$, $f(l) \leq f(s)$. Notice that the inequality is not strict, in order to allow the treatment of neutrality (local optima of equal fitness), which we found to occur in some QAP instances. The set of local optima, which corresponds to the set of nodes in the network model, is denoted by $L$.

**Monotonic perturbation edges.** Edges are directed and based on the perturbation operator ($k$-exchange, $k > 2$). There is an edge from local optimum $l_1$ to local optimum $l_2$, if $l_2$ can be obtained after applying a random perturbation ($k$-exchange) to $l_1$ followed by local search, and $f(l_2) \leq f(l_1)$. These edges are called *monotonic* as they record only non-deteriorating transitions between local optima. Edges are weighted with estimated frequencies of transition. We estimated the edge weights in a sampling process. The weight is the number of times a transition between two local optima basins occurred with a given perturbation. The set of edges is denoted by $E$.

### 2.3 Multifractal Dimensions

A fractal dimension is the logarithmic ratio between amount of detail in a pattern, and the scale used to measure the detail: $\frac{ln(detail)}{ln(scale)}$. *Multifractal* analysis [2] can be used for systems where a single fractal dimension may not be sufficient to characterise the spatial complexity. With this approach, a spectrum of dimensions is instead produced. Study of the spectrum can provide information about the multifractality (i.e., the heterogeneity of fractal complexity), as well as dimensionality. We approach multifractal analysis using the *sandbox* algorithm [8] where several nodes are randomly selected to be sandbox 'centres'. Members of the sandboxes are computed as nodes which are $r$ edges apart from the centre $c$. After that the average sandbox size is calculated. The procedure is replicated for different values of $r$ which is the sandbox radius. To facilitate the production of a dimension spectrum the whole process is repeated for several arbitrary real-valued numbers which supply a parameter we call $q$.

The sandbox algorithm has been specialised and modified to suit LONs [4] and this is the process we use for our fractal analysis experiments. *Fitness* distance — as well as network edge distance — is considered. The comparison between two local optima fitness values is conducted through *logarithmic returns*: *fitness difference* $= |ln(f_1/f_2)|$ where $f_1$ and $f_2$ are the fitnesses of two local optima at the start and end of a LON edge. The resultant value can then be compared with a set fitness-distance maximum allowable threshold, $\epsilon$. Pseudo-code for the multifractal algorithm we use on the LONs is given in Algorithm 1. Sandbox centre selection is at Line 7 of the Algorithm. A node $n$ is included in the 'sandbox' of a central node $c$ (Line 15 of the pseudo-code) if *either* the LON edge distance $d(n, c) = 1$ *or* $d(n, c) = r - 1$ **and** the fitness-distance between the two local optima is less than a threshold: $|ln(\frac{f(n)}{f(c)})| < \epsilon$ (see Line 14).

**Algorithm 1** Multifractal Analysis of a LON

**Input:** $LON$, $q.values$, $radius.values$, $fitness.thresholds$, $number.centres$
**Output:** *mean sandbox size*

Initialisation:
$centre.nodes \leftarrow \emptyset, noncentre.nodes \leftarrow all.nodes$
$mean.sandbox.sizes \leftarrow \emptyset$
**for** $q$ in $q.values$ **do**
  **for** $r$ in $radius.values$ **do**
    **for** $\epsilon$ in $fitness.thresholds$ **do**
      $centre.nodes \leftarrow$ `RANDOM.SELECTION`($all.nodes$, $number.centres$)
      $sandbox.sizes \leftarrow \emptyset$
      **for** $c$ in $centre.nodes$ **do**
        $number.boxed \leftarrow 0$
        **for** $v$ in $all.nodes$ **do**
          $d \leftarrow$ `DISTANCE`$(c, v)$
          $j \leftarrow$ `DIFFERENCE`$(f(c), f(v))$
          **if** ( $d == 1$ ) **OR** ( $d == r$ - 1 **and** $j < \epsilon$ ) **then**:
            $number.boxed \leftarrow number.boxed + 1$
          **end if**
        **end for**
        $sandbox.sizes \leftarrow sandbox.sizes \cup \{[number.boxed]\}$
      **end for**
      $bs \leftarrow$ `MEAN`($sandbox.sizes$)
      $mean.sandbox.sizes[q][r][\epsilon] \leftarrow bs$
    **end for**
  **end for**
**end for**

At the end of each 'sandboxing' iteration conducted with particular values for the parameters $q$, $r$ and $\epsilon$, the associated fractal dimension is calculated:

$$fractal\ dimension = \frac{ln(detail^{q-1})}{(q-1) * ln(scale)} \tag{1}$$

where *detail* is the average sandbox size (as a proportion of the network size), $q$ is an arbitrary real-valued value, and *scale* is $\frac{r}{dm}$, with $r$ being the radius of the boxes and $dm$ the diameter of the network. The sandbox algorithm has a cubic time complexity and quadratic space complexity [9].

## 3 Experimental Setup

### 3.1 Iterated Local Search

We use Stützle's iterated local search (**ILS**) for both gathering performance data and as the foundation of LON construction [7]. The local search stage uses a first improvement hill-climbing variant with the pairwise (2-exchange) neighbourhood. This operator swaps any two positions in a permutation. The

perturbation operator exchanges $k$ randomly chosen items. We consider two perturbation strengths **for both constructing the LONs and computing the performance metrics**: $\frac{ND}{8}$ (we will henceforth refer to this as **low** perturbation) and $\frac{3ND}{4}$ (this will be referred to as **high** perturbation) with $ND$ being the problem dimension. These perturbation magnitudes were chosen because they have been studied previously for the QAP and ILS [10]; in that work, $\frac{ND}{8}$ is the lowest strength considered, while $\frac{3ND}{4}$ is the second-strongest (the strongest was a total restart, which we decided was too extreme for our purposes). Only local optima which have improved or equal fitness to the current are accepted. Worsening local optima are never accepted.

### 3.2 Robust Taboo Search

Robust Taboo Search (**ROTS**) [11] is a competitive heuristic for the QAP and is also executed on the instances in this study. ROTS is a best-improvement pairwise exchange local search with a variable-length taboo list tail. For each facility-location combination, the most recent point in the search when the facility was assigned to the location is retained. A potential move is deemed to be 'taboo' (not allowed) if both facilities involved have been assigned to the prospective locations within the last $y$ cycles. The value for $y$ is changed randomly, but is always from the range $[0.9n, 1.1ND]$, where $ND$ is the problem dimension.

**Algorithm Performance Metric.** We compute the *performance gap* to summarise ILS and ROTS performance on the instances. In the case of ILS, runs terminate when *either* the known best fitness is found or after 10,000 iterations with no improvement. For ROTS, runs complete when the best-known fitness is found or after 100,000 iterations. The performance gap is calculated over 100 runs for each, and is defined as the *mean* obtained fitness as a proportion of the best-known fitness.

### 3.3 LON Construction and Metrics

The LON models are constructed by aggregating the unique nodes and edges encountered during 100 independent ILS runs with the standard acceptance strategy (i.e. accepting improvements and equal solutions). Runs terminate after 10,000 non-improving iterations.

At this stage, **esc** instances are removed from the set: their local optima networks are uninteresting to study because there is a very high degree of LON neutrality. Removing these anomalies left us with the remaining moderate-size (between 25 and 50, inclusive) QAPLIB: 40 instances. There are two LONs per problem instance (for the two perturbation strengths), totalling 80 LONs.

For each LON, thousands of fractal dimensions are produced. The exact number depends on the diameter of the network: full parameter details are given in

the next Section. The measurements we compute from the set of fractal dimensions for a given LON are: the median fractal dimension (simply the median of all the dimensions calculated); the maximum fractal dimension (maximum of all dimensions); the dimension variance; the multifractality (measured by taking the absolute value of a fractal dimension at the end of the spectrum divided by the absolute value of a dimension at the beginning), and an excerpt dimension (randomly chosen from the spectrum).

We consider some other LON metrics too: the flow towards global optima (computed as the incoming network edge strength to global optima in the LON); the number of local optima (simply the number of nodes in the LON); and the number of global optima (number of LON nodes with the best-known fitness).

### 3.4 Multifractal analysis

We implement the multifractal analysis algorithm for LONs in C programming language and have made it publicly available for use; some of the code functionality was obtained from a monofractal analysis algorithm [12] available on Hernan A. Makse's webpage[2]. To generate *multifractal* spectra, a range of arbitrary real-valued numbers is needed. We set these as $q$ in the range $[3.00, 8.90]$ in step sizes of 0.1. The number of 'sandbox' centres in each iteration is set at 50 and the choice of these centres is randomised. A range of ten values is used for the local optima fitness-distance threshold: $\epsilon \in [0.01, 0.19]$ in step sizes of 0.02. The sizes of sandboxes are integers in the range $r \in \{2..diameter - 1\}$ where *diameter* is the LON diameter. Note that in the interest of reducing computation, we constrain the maximum considered box radius to eleven — that is, when the LON diameter exceeds twelve ($diameter - 1 > 11$), then the upper limit for $r$ is set to 11, to allow ten possible values $r \in \{2..11\}$.

### 3.5 Regression Models

Random Forest regression [13] is used. We separate LONs by the ILS perturbation strength which was applied during their construction; in this way, for modelling there are two distinct data-sets, each of them totalling 40 rows. Each observation is a set of LON features such as median fractal dimension (these are the independent variables) alongside performance metrics (the dependent variables). LONs formed using low perturbation are mapped to low-perturbation ILS performance runs, and high-perturbation LONs are mapped to high-perturbation ILS performance runs. The same Taboo Search performance metrics are used for both sets of LONs. The candidate independent variables are:

- Number of local optima
- Number of global optima
- Search flow towards global optima
- Median fractal dimension for the LON

[2] `https://hmakse.ccny.cuny.edu/`

- Variance of fractal dimension (proxy for *multifractality*)
- Maximum fractal dimension
- Variation in the multifractal spectrum (proxy for *multifractality*)

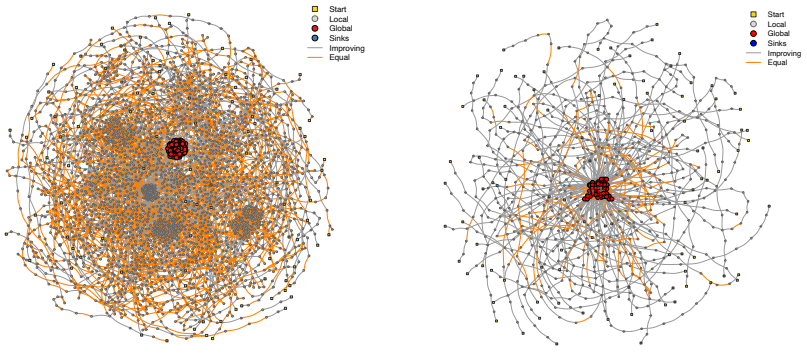
The manner of computing the metrics was described in Section 3.3. Iterated local search and Robust Taboo Search *performance gap* on the instances serve as response variables, making this a regression setting.

We aimed for models with as few independent variables as possible, owing to the limited number of eligible QAPLIB instances of moderate size. The *one-in-ten* rule [14] stipulates that roughly ten observations are required per independent variable. Our instance sets are each comprised of 40 instances — so we correspondingly set the maximum number of independents as four and conduct feature selection, as described now.

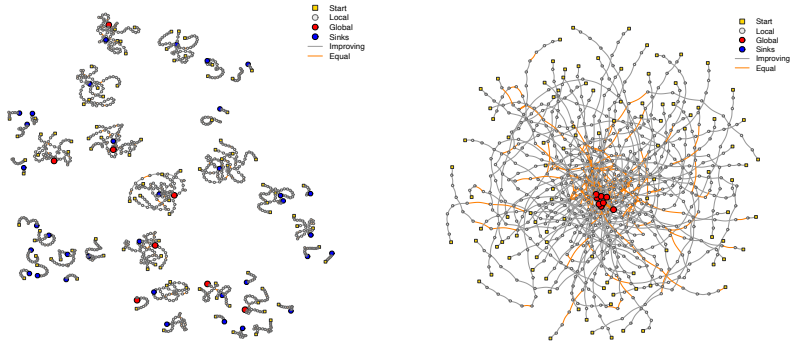
*Recursive Feature Elimination.* Backwards *recursive feature elimination* (RFE) was used to select model configurations with subsets of the predictors. We use Root Mean Squared Error (RMSE) as the quality metric for model comparisons. RMSE is the square-root of the MSE, which itself is the mean squared difference between the predicted values and true values. For the experiments, we configure RFE as follows. Random Forest is the modelling method. We consider feature subset sizes of one, two, three, and four from a set of eight candidates (listed earlier). The RFE cross-validation is set to 10-fold; model configurations are compared based on the mean RMSE over the 10 folds.

*Models using selected features.* After feature selection, Random Forest regression is conducted using the selected features only. There are several separate model configurations owing to the different ILS perturbations under scrutiny and the two optimisation performance algorithms. To attempt to mitigate the effect of the limited training set size — which is due to the available quantity of moderate-size QAPLIB instances — we *bootstrap* the selection of the training and validation sets. We consider an 80-20 split for training and validation with 1000 iterations. Quality metrics are computed on both the training set and also from the predictions made on the validation set. The first included measurement is the R-Squared (RSQ, computed as $1 - \frac{MSE}{variance(t)}$, where $t$ is the response variable). Also considered is the RMSE, as detailed already. The metrics are computed as the mean value over 1000 bootstrapping iterations, and their standard error is also included in the results. The standard error reported here is a measurement for how varied the means for RSQ and RMSE are across different random sub-samplings: it is the standard deviation of the means for these parameters.

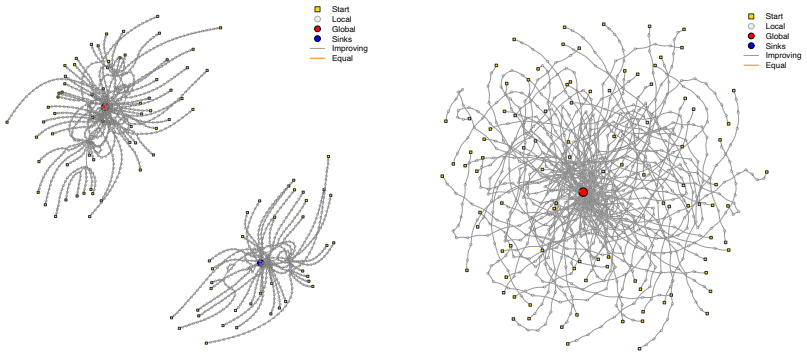
*Details.* For all feature selection and subsequent modelling, the default hyperparameters for Random Forest in R are used, namely: 500 trees; minimum size of terminal nodes set to five; a sample size set to the number of observations; re-sampling with replacement; features considered per split set to one-third of the number of features. Independent variables are standardised as follows:




(a) bur26a,$p$=low,$d$=(1.60, 2.10) (b) bur26a,$p$=high,$d$=(1.91, 5.63)




(c) nug25,$p$=low,$d$=(1.79, 2.50) (d) nug25,$p$=high,$d$=(2.16, 5.07)




(e) tai30b,$p$=low,$d$=(7.05, 21.91) (f) tai30b,$p$=high,$d$=(11.62, 167.32)

Fig. 1: Monotonic LONs for selected instances and the two perturbation strengths, $p = low$ (left) $p = high$ (right). The median and maximum fractal dimension are also indicated in the sub-captions as $d$ = (median, maximum)

$p = \frac{(p-E(p))}{sd(p)}$, with $p$ being the predictor in question, $E$ the expected value (mean), and $sd$ the standard deviation.

## 4 Experimental Analysis

### 4.1 Network Visualisation

Visualisation is a powerful tool to get insight into the structure of networks. Figure 1 illustrates MLONs for three representative QAP instances: a real-world instance **bur26**, a random flows on grids instance **nug25**, and a random real-world like instance **tai30b**. The networks in Figure 1 capture the whole set of sampled nodes and edges for each instance and perturbation strength. The two perturbation strengths, low and high, are shown. In the plots, each node is a local optimum and edges are perturbation transitions, either improving in fitness (visualised in grey) or leading to equal fitness nodes (visualised in orange). Node and edge decorations reflect features relevant to search. The edges colour reflect their transition type, to nodes with improving (grey) or equal fitness (orange). Global optima are highlighted in red. The start nodes (without incoming edges) are highlighted as yellow squares, while the sink nodes (without outgoing edges) are visualised in blue.

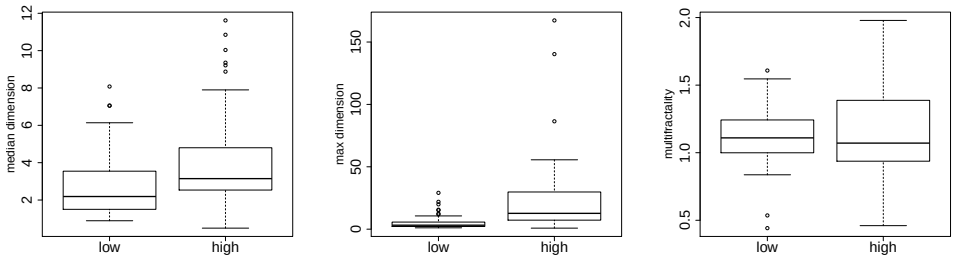
Figures 1a and 1b reflect the same problem instance (**bur26a**) but with LONs constructed using different perturbation strengths. Figure 1b has higher fractal dimensions and this is probably because of the lesser extent of neutrality at the local optima level (in the image, this can be seen through the amount of orange connections), as well as fewer connection patterns between local optima. There are also some long monotonic paths. All of these factors would result in higher fractal dimension, because they would lend to the fitness-distance and edge-distance boxing constraints in the multifractal analysis algorithm not being satisfied — and consequently, nodes remaining un-boxed, leading to a higher level of detail being computed and a higher fractal dimension (recall Section 2.3 for particulars on this process).

The LONs of instance **tai30b** (Figure 1e and 1f) have the highest fractal dimensions shown. This is probably because of the lack of LON neutrality (lack of orange edges in the plot), as well as long and separate monotonic pathways (notice the number of edge-steps forming some of the paths). Additionally, compared to the networks associated with the other two instances, the LON fitness ranges are quite large for here: the minimum LON fitness is around 73-74% of the maximum in the **tai30b** LONs, while for the other two instances shown in Figure 1a-1d, it is between approximately 92%-99%. This means that the fitness-distance boxing condition in the fractal algorithm will be satisfied much less for these LONs, resulting in higher fractal dimensions. This situation also implies that the monotonic pathways contain large fitness jumps.

### 4.2 Distributions

Figure 2 presents distributions for fractal dimension measurements, split by perturbation strength: *low* and *high*. In Figure 2a are median fractal dimensions for the LONs. Notice that the dimensions are noticeably higher — and more varied — in the high-perturbation group (on the right) when compared to the

low-perturbation group on the left. Next, in Figure 2b, are the *maximum* fractal dimensions for the LONs. The same trend is evident here; that is, high-perturbation LONs (on the right) have higher and more varied dimensions than the low-perturbation LON group.




(a) median fractal dimension (b) max fractal dimension (c) multifractality

Fig. 2: Distributions of fractal dimension measurements taken from local optima networks. Note the different scales on the $y$-axes.

Figure 2c shows the amount of multifractality (heterogeneity of fractal geometry), computed as the absolute value of a fractal dimension at the end of the spectrum divided by the absolute value of a dimension at the beginning. This time, in the plot, it cannot confidently be said that one group contains more multifractality than another; however, it seems clear that the high-perturbation group have more varied multifractality values.

### 4.3 Predictive Modelling

Table 1 presents the configuration and quality of regression models for algorithm performance prediction. Each column (columns two to four) is a model setting. The first two rows are configuration information: the ILS perturbation strength used to form the LONs whose features are used as predictors (*LON perturbation*) and the features selected for the model by recursive feature elimination. All the remaining rows convey data about the quality of the models. Provided are the RSQ, RMSE, and RMSE as a percentage of the range of the target variable. Each of these are reported for the training and validation data. In the Table, abbreviations are used for feature names. *Multifractality* is computed as the absolute value of a fractal dimension at the end of the spectrum divided by the absolute value of a dimension at the beginning; *FD* is short for fractal dimension; *flow GO* is the combined strength of LON edges incoming to global optima; and *var FD* is the variance of the fractal dimension.

Recall from Section 3.5 that the number of local optima and the number of global optima were candidate predictors. Notice from Table 1, row two, that

Table 1: Information about models with features selected by recursive feature elimination in a Random Forest setting.

| | Iterated Local Search | | Robust Taboo Search | |
|---|---|---|---|---|
| LON perturbation | low | high | low | high |
| selected features | [*multifractality*, *median FD* , *max FD*] | [*flow GO*, *multifractality*, *max FD*, *median FD*] | [*median FD*, *multifractality*, *flow GO*, *max FD*] | [*var FD*, *median FD*] |
| RSQ-*train* (SE) | 0.1663 (0.2650) | 0.7917 (0.2457) | **0.8610 (0.3081)** | 0.7523 (0.2790) |
| RMSE-*train* (SE) | 0.0001 (0.0001) | 0.0000 (0.0000) | 0.0005 (0.0042) | 0.0010 (0.0044) |
| RMSE%range-*train* | 0.1% (0.1%) | **0% (0%)** | 0.1% (0.7%) | 0.2% (0.8%) |
| RSQ-*validation* (SE) | 0.8661 (0.4630) | **0.9891 (0.4147)** | 0.5124 (1.1331) | 0.4973 (1.0439) |
| RMSE-*validation* (SE) | 0.0086 (0.0076) | 0.0007 (0.0021) | 0.0709 (0.0509) | 0.0720 (0.0482) |
| RMSE%range-*validation* | 7.9% (7.0%) | **0.3% (0.8%)** | 12.2% (8.8%) | 12.4% (8.3%) |

these are never selected from the pool. Instead, fractal dimension metrics and the incoming search flow to global optima (*flow GO*) are chosen by the RFE algorithm. We particularly note that *multifractality*, which captures how varied the fractal complexity in a LON is, appears in three of the four model setups. The median fractal dimension appears in all four, and maximum dimension in three. Differing quantities of predictors are selected. In two cases, there is the maximum allowable amount (recall Section 3.5) chosen from eight candidates: four. The remaining models, however, contain less selected features: two and three, respectively.

Bold text in the Table draw the eye to the best value within a row. RMSE values are not highlighted in this way because they do not have a common range (owing to different response variable distributions). Instead, the RSQ and RMSE as a percentage of the range are emphasised with emboldened text. Notice that the model built using features of high perturbation LONs and which is modelling ILS performance gap as a response seems to be the best of the four models; this can be seen by comparing the second model column with the other three. RMSE is very low on both training and validation data, suggesting that this is a good model. While the RSQ-*train* is lower than for the ROTS response using low-perturbation LONs modelling (in the next column along), the RSQ-*validation* is superior to that — and indeed, the others — by a large margin.

Using features of low-perturbation LONs to model ILS performance response results in a much weaker model (view this in the first model column). The RSQ for training data is poor — only approximately 0.17. Even though the RSQ for validation data is significantly higher (approximately 0.87), the low RSQ on training data suggests that it does not accurately capture the patterns. Comparing this model (low-perturbation LONs) with its neighbour in the Table (high-perturbation LONs), we observe that using a higher perturbation strength to construct LONs may result in fractal dimension metrics which are more useful in predicting ILS performance.

Focusing now on the two models which consider ROTS performance as response variables (model columns three and four), we can see that — on validation data — each of them explains around 50% of variance (RSQ-*validation* row). That being said, both RSQ means have very high standard errors (in brackets). This means that while the results hold true for this set of QAP instances, we would be cautious in extrapolating these specific results to other instance sets. A high standard error can occur with a limited sample size and with high diversity of training instances — both of which are present in our dataset. Nevertheless, the fact that some ROTS variance can be explained (at least for this specific dataset) is important because the LONs were not formed using a ROTS process; ILS was the foundation (Section 3.3).

The finding means that performance of a separate metaheuristic can be partially explained using ILS-built LON fractal dimension features, even when different perturbation strengths are used. Notice also that the low-perturbation model for ROTS is slightly better than the high-perturbation LON model. This might be because ROTS does not conduct dramatic perturbations on solutions. While RMSE is low on the training data (RMSE%range-*train*), it is much higher on validation data (although still not what might be considered 'high').

## 5 Conclusions

We have conducted a study of the relationship between Iterated Local Search (ILS) perturbation strength and fractal dimensions. The ILS perturbation strength is used when constructing Local Optima Networks (LONs), and fractal dimension can be computed from those LONs.

We found that higher-perturbation LONs also have higher fractal dimensions. Fractal dimension measurements drawn from LONs which were constructed using low and high perturbation strengths were related to algorithm performance on the underlying Quadratic Assignment Problems (QAPs). The results showed that ILS algorithm performance prediction using fractal dimension features may benefit more from LONs formed using a high perturbation strength; this model configuration enjoyed excellent performance. Around half of variance in Robust Taboo Search performance on the dataset used could be explained using predictors including fractal dimension features, and the model using the low-perturbation features was slightly stronger than the high-perturbation model.

The local optima networks are available online[3]; the fractal analysis algorithm for local optima networks is published here[4].

## References

1. Mandelbrot, B.B.: Possible refinement of the lognormal hypothesis concerning the distribution of energy dissipation in intermittent turbulence. In: Statistical models and turbulence, pp. 333–351. Springer (1972)

[3] `https://github.com/sarahlouisethomson/fractal-dimension-perturbation-strength`
[4] `https://github.com/sarahlouisethomson/compute-fractal-dimension-local-optima-networks`

2. Mandelbrot, B.B., Fisher, A.J., Calvet, L.E.: A multifractal model of asset returns. Cowles Foundation discussion paper (1997)
3. Ochoa, G., Tomassini, M., Vérel, S., Darabos, C.: A study of nk landscapes' basins and local optima networks. In: Proceedings of the 10th annual conference on Genetic and evolutionary computation. pp. 555–562. ACM (2008)
4. Thomson, S.L., Ochoa, G., Verel, S.: The fractal geometry of fitness landscapes at the local optima level. Natural Computing pp. 1–17 (2020)
5. Burkard, R.E., Karisch, S.E., Rendl, F.: QAPLIB – a quadratic assignment problem library. Journal of Global Optimization 10(4), 391–403 (1997)
6. Taillard, E.: Comparison of iterative searches for the quadratic assignment problem. Location Science 3(2), 87–105 (1995)
7. Stützle, T.: Iterated local search for the quadratic assignment problem. European Journal of Operational Research 174(3), 1519–1539 (2006)
8. Liu, J.L., Yu, Z.G., Anh, V.: Determination of multifractal dimensions of complex networks by means of the sandbox algorithm. Chaos: An Interdisciplinary Journal of Nonlinear Science 25(2), 023–103 (2015)
9. Ding, Y., Liu, J.L., Li, X., Tian, Y.C., Yu, Z.G.: Computationally efficient sandbox algorithm for multifractal analysis of large-scale complex networks with tens of millions of nodes. Physical Review E 103(4), 043303 (2021)
10. Ochoa, G., Herrmann, S.: Perturbation strength and the global structure of qap fitness landscapes. In: International Conference on Parallel Problem Solving from Nature. pp. 245–256. Springer (2018)
11. Taillard, É.: Robust taboo search for the quadratic assignment problem. Parallel computing 17(4-5), 443–455 (1991)
12. Song, C., Gallos, L.K., Havlin, S., Makse, H.A.: How to calculate the fractal dimension of a complex network: the box covering algorithm. Journal of Statistical Mechanics: Theory and Experiment 2007(03), P03006 (2007)
13. Breiman, L.: Random forests. Machine Learning 45(1), 5–32 (2001)
14. Harrell Jr, F.E., Lee, K.L., Califf, R.M., Pryor, D.B., Rosati, R.A.: Regression modelling strategies for improved prognostic prediction. Statistics in medicine 3(2), 143–152 (1984)