Sampling Local Optima Networks of Large Combinatorial Search Spaces: The QAP Case

Sébastien Verel[1](✉), Fabio Daolio[2], Gabriela Ochoa[2], and Marco Tomassini[3]

[1] EA 4491 - LISIC - Laboratoire d'Informatique Signal et Image de la Côte d'Opale, Université du Littoral Côte d'Opale, 62228 Calais, France
verel@univ-littoral.fr

[2] Computing Science and Mathematics, University of Stirling, Stirling, Scotland, UK

[3] Information Systems Department, Faculty of Business and Economics, University of Lausanne, Lausanne, Switzerland

Abstract. Local Optima Networks (LON) model combinatorial landscapes as graphs, where nodes are local optima and edges transitions among them according to given move operators. Modelling landscapes as networks brings a new rich set of metrics to characterize them. Most of the previous works on LONs fully enumerate the underlying landscapes to extract all local optima, which limits their use to small instances. This article proposes a sound sampling procedure to extract LONs of larger instances and estimate their metrics. The results obtained on two classes of Quadratic Assignment Problem (QAP) benchmark instances show that the method produces reliable results.

1 Introduction

Fitness landscapes are a commonly-used metaphor to describe heuristic search of a globally optimal, or at least of a satisfying solution, among the set of admissible solutions (see Richter and Engelbrecht [1] for a recent review of the state of the art in the field). The number and distribution of local optima in combinatorial fitness landscapes are known to have an impact on the performance of search heuristics. Local Optima Networks (LONs) have been recently proposed as a model of combinatorial landscapes that specifically captures these landscape features [2–4]. In this network model, the nodes are the local optima of the underlying optimisation problem and the edges account for transitions among them using a neighbourhood operator. Modelling combinatorial landscapes as networks brings a whole new set of metrics for capturing the topology and structure of combinatorial search spaces, and provides tools for estimating search difficulty. Most previous work with this model required the full enumeration of the search space in order to extract the nodes and edges of the local optima network, therefore it was applicable only to small problems. We present a sampling methodology for extracting local optima networks of large combinatorial problem instances, and estimating the relevant landscape network metrics

A. Auger et al. (Eds.): PPSN 2018, LNCS 11102, pp. 257–268, 2018.
https://doi.org/10.1007/978-3-319-99259-4_21

for benchmark instances of the Quadratic Assignement Problem (QAP). The fitness landscape of QAP have been studied several times (for example see the algebraic analysis of the autocorrelation function in Chicano *et al.* [5]). In this work, we propose to increase the number of relevant features to analyse large size fitness landscape that could be potentially be used for performance prediction of QAP algorithms.

The article is structured as follows. The next section briefly overviews previous work on local optima networks, describes the QAP, and the combinatorial landscapes considered. Section 3 describes local optima networks and the metrics employed as features. Section 4 outlines our approach for network sampling and Sect. 5 describes the empirical validation of the obtained estimates. Finally, Sect. 6 summarises our findings and suggest directions for future work.

2 Combinatorial Landscapes

Given a discrete optimization problem, a fitness landscape for its instances is defined as a finite set S of possible solutions, a neighbourhood $\mathcal{N}(s)$ given by the set of solutions that can be reached from any solution $s \in \mathcal{X}$ by applying a simple move operator, and a function $f : S \rightarrow \mathbb{R}$ that, given a solution, provides its objective value or fitness [6]. One can define a number of useful concepts such as global and local optima, and basins of attraction among others. It is also possible to define ways in which the search space can be traversed in a random or adaptive way in order to collect configuration space statistics or to improve the current solution.

Starting from the above notions, the *local optima network* (LON) model for combinatorial landscapes was first proposed in [2], with follow up work appearing in [3,7] using Kauffman's NK [8]. Subsequently, more complex and realistic search spaces were studied: the quadratic assignment problem [9], and the permutation flowshop problem [10] which are known to be NP-hard. In a LON, vertices correspond to solutions that are minima or maxima of the associated combinatorial problem, and edges correspond to weighted transitions among them. Initially, weighed edges represented an approximation to the probability of transition between the respective basins in a given direction [2,3,7]. This definition, although informative, produced densely connected networks and required exhaustive sampling of the basins of attraction. A second version, *escape edges* was proposed in [4], which does not require a full computation of the basins. Instead, these edges account for the chances of escaping a local optimum after a controlled mutation (*e.g.* 1 or 2 bit-flips in binary space) followed by hill-climbing. It is this later version that is used here. In order to demonstrate the methodologies proposed in this study, we consider the Quadratic Assignment Problem which is described below.

2.1 The Quadratic Assignment Problem

The Quadratic Assignment Problem [9] is a combinatorial problem in which a set of facilities with given inter-facilities flows has to be assigned to a set of locations with given inter-locations distances in such a way that the sum of the product of flows and distances is minimised. A solution to the QAP is generally written as a permutation π of the set $\{1, 2, ..., n\}$. The cost associated with a permutation π is given by:

$$C(\pi) = \sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij} b_{\pi_i \pi_j}$$

where n denotes the number of facilities/locations and $A = \{a_{ij}\}$ and $B = \{b_{ij}\}$ are referred to as the distance and flow matrices, respectively. The structure of these two matrices characterises the class of instances of the QAP problem.

The results presented in this article are based on two instance generators proposed in [11] which are in turn inspired by [12] included in the QAPLib. In [11] the generators were devised for the multi-objective QAP, but are adapted here for the single-objective QAP. In order to perform a statistical analysis of the extracted local optima networks, we consider 30 problem instances for each class and size combination.

- *Uniform generator*: produces uniformly random instances where all flows and distances are integers sampled from uniform distributions. The distances are random integer numbers between 0 and 99 (bounds included). The flow matrix is symmetric with random integer entries between 1 and 99. This leads to the kind of problems known in literature as *Tai*`nn`*a*, being `nn` the problem dimension [12].
- *Real-like generator*: makes instances where the distance and flow matrices have structured entries. To generate the symmetric distance matrix, N points (integer coordinates) are randomly distributed in a circle of radius 100, and the entries are given by the distances between these N points. The flow matrix is also symmetric with entries following the law $\lceil 10^r \rceil$ where r is a uniform random integer from $[L, U]$. This procedure generates non-uniformly random instances of type *Tai*`nn`*b* which have the so called "real-like" structure (see QAPLib) since they resemble the structure of QAP problems found in practical applications. The problem instances from the QAPLib are often unique example of a class of problems, so our study considers two parameterisations of real-like instances which allows a statically analysis: *rl1*: $L = -10$ and $U = 5$, *rl2*: $L = -2$ and $U = 4$.

3 Obtaining the Local Optima Networks

Our study considers the permutation representation for QAP solutions. In this case, the most basic neighbourhood structure in the search space is given by the pairwise exchange operation that exchanges any two positions in a permutation, thus transforming it into another permutation.

In what follows, we define how the LON graphs are obtained from the fitness landscapes corresponding to QAP instances.

Nodes: The nodes in the network are local optima (LO) in the search space. For a minimisation problem such as QAP, a solution $x \in \mathcal{X}$ is a local optimum iff $\forall x' \in \mathcal{N}(x), f(x) \leqslant f(x')$. Notice that in this work we do not target specifically neutral fitness landscape with large plateaus. However, this definition of local optima is still relevant for small amounts of neutrality. For fitness landscapes with high levels of neutrality, please refer to the definitions of previous work [13] where the nodes are local optima plateaus. LO are extracted using a best-improvement hill-climber (hc), as given in Algorithm 1. Thereby, when selecting the fittest neighbour (line 4), ties are broken at random.

Algorithm 1. Best-improvement hill-climbing (minimisation)

```
1: procedure HillClimbing
2:     x ← random initial solution
3:     while x ≠ Local Optimum do
4:         set x' ∈ N(x), s.t. f(x') = min_{y∈N(x)} f(y)
5:         if f(x') < f(x) then
6:             x ← x'
7:         end if
8:     end while
9: end procedure
```

Escape Edges: The edges in the network are defined according to a distance function $dist$ and a positive integer $D > 0$. The distance function represents the minimal number of moves between two solutions for a given search (mutation) operator. There is an edge e_{ij} between LO_i and LO_j if a solution x exists such that $dist(x, LO_i) \leqslant D$ and $hc(x) = LO_j$. In other words, if LO_j can be reached after mutating LO_i and running hill-climbing from the mutated solution. The weight $\tilde{w}_{ij}$ of this edge is $\tilde{w}_{ij} = \sharp\{x \in \mathcal{X} \mid dist(x, LO_i) \leqslant D \text{ and } hc(x) = LO_j\}$. That is, the number of LO_i mutations that reach LO_j after hill-climbing. This weight can be normalised by the total number of solutions, $\sharp\{x \in \mathcal{X} \mid dist(x, LO_i) \leqslant D\}$, within reach at distance D: $w_{ij} = \tilde{w}_{ij} / \sum_j \tilde{w}_{ij}$.

Local Optima Network: The weighted local optima network $G_w = (N, E)$ is the graph where the nodes $n_i \in N$ are the local optima, and there is an edge $e_{ij} \in E$, with weight w_{ij}, between two nodes n_i and n_j if $w_{ij} > 0$. According to the definition of weights, w_{ij} may be different than w_{ji}. Thus, two weights are needed in general, and we have a weighted, oriented transition graph.

3.1 Complex Network Metrics

The previous section described how to obtain the LONs. A number of models and statistical metrics have been proposed to study the structure and function of large networks [14]. The first section of Table 1 summarises the metrics for weighted networks considered in this study. First, we introduce some basic network notation before defining more advanced metrics. Let us denote a_{ij} an element of the graph's *adjacency matrix* A for a weighted oriented graph $G_w = (N, V)$, defined as $a_{ij} = 1$ if $w_{ij} > 0$, $a_{ij} = 0$ if $w_{ij} = 0$. Finally, $k_i = \sum_{j \neq i} a_{ij}$ is the degree of node i, whereas $s_i = \sum_{j \neq i} w_{ij}$ is a generalisation of a node's degree for weighted networks called the node's *strength*. From those basic definitions, we can define the average outdegree, $zout$, as the average of k_i for all nodes.

Disparity of a node n_i measures how heterogeneous are the contributions of the edges of node n_i to the total weight (strength):

$$Y_2(i) = \sum_{j \neq i} \left(\frac{w_{ij}}{s_i} \right)^2$$

Thus, the *average disparity* y_2 is defined as the average for all nodes of $Y_2(i)$.

The standard clustering coefficient [14] does not consider weighted edges. We thus use the *weighted clustering* $c^w(i)$ measure of a node n_i proposed in [15], which combines the topological information with the weight distribution of the network:

$$c^w(i) = \frac{1}{s_i(k_i - 1)} \sum_{j,h} \frac{w_{ij} + w_{ih}}{2} a_{ij} a_{jh} a_{hi}.$$

For each triple formed in the neighbourhood of the node n_i, $c^w(i)$ counts the weight of the two participating edges of the node n_i. The average weighted clustering coefficient wcc is defined as $wcc = 1/|N| \sum_{n_i \in N} c^w(i)$. The reader is referred to [15] for more details.

A network is said to show assortative mixing if the nodes in the network that have many connections tend to be connected to other nodes with many connections [16]. Assortativity can be measured using the Pearson correlation coefficient r of degree between pairs of linked nodes. Positive values of r indicate a correlation between nodes of similar degree, whereas negative values indicate relationships between nodes of different degree. We use here the *weighted assortativity*, denoted knn, which measures the nearest-neighbours degree correlation. This metric reflects the affinity to connect with high or low-degree neighbours.

The *fitness-fitness correlation* (fnn) measures the correlation between the fitness values of adjacent local optima. It is the Pearson correlation coefficient between the fitness value f_i of node n_i and the weighted-average of it nearest-neighbours fitness, defined as $f_{n,w}(i) = 1/s_i \sum_{j \neq i} w_{ij} f_j$.

4 Sampling Local Optima Networks

Most of the previous work has considered small search spaces (problem sizes up to 18 for binary spaces and up to 10 for permutation spaces), where it was possible to exhaustively enumerate and fully extract the local optima network models. For larger search spaces, i.e., those corresponding to realistic problem

Table 1. Set of features used for network characterisation and for sampling.

Network metrics	
fit	Average fitness of local optima in the network
wii	Average weight of self-loops
zout	Average outdegree, i.e., number of outgoing edges
y_2	Average disparity for outgoing edges
knn	Weighted assortativity
wcc	Weighted clustering coefficient. Measures "cliquishness" of a neighbourhood
fnn	Fitness-fitness correlation. Measures the correlation between the fitness values of adjacent local optima
snowball sampling metrics	
lhc	Average length of hill-climbing to local optima
mlhc	Maximum length of hill-climbing to local optima
nhc	Number of hill-climbing paths to local optima

sizes, a methodology for sampling the local optima networks is required. We propose here an original method for extracting a significant subset of the local optima and transition edges. Thereafter, the network metrics are estimated from the sampled network. The sampling follows a random walk over the local optima network coupled with a snowball process, also known as chain-referral [17]. Snowball sampling is a non-probabilistic technique used in sociology where existing subjects recruit future subjects from among their acquaintances. The sample population then grows like a rolling snowball, similarly to breadth-first search. In the computational implementation, the snowball procedure enlarges an original node sample by joining adjacent nodes. Two control parameters are required: the number of neighbours to consider m (how many acquaintances a recruit should name), and the depth of the sampling d (how many referral steps).

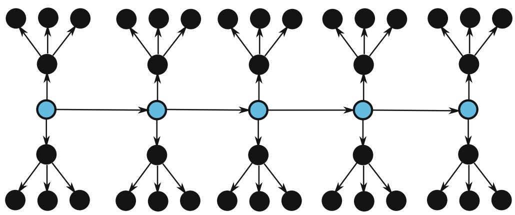

Fig. 1. Illustration of the sampling procedure, featuring a random walk of length $l = 5$, a number of sampled edges $m = 3$, and a sampling depth $d = 2$. The light circles in the center are solutions x_i on the random walk, while dark circles on the outside are solutions sampled during the snowball procedure.

Algorithm 2. Sampling methodology for local optima networks

```
procedure LONSampling(d, m, l)
    x_0 ← hc(x)                                  ▷ where x is randomly initialised
    N̂ ← {x_0}
    Ê ← ∅
    for t ← 0, ... l − 1 do
        Snowball(d, m, x_t)
        x_{t+1} ← RandomWalkStep(x_t)
    end for
end procedure

procedure Snowball(d, m, x)
    if d > 0 then
        for j ← 1, ... m do
            x' ← hc(op(x))
            N̂ ← N̂ ∪ {x'}                         ▷ Add node to the sample
            if (x, x') ∈ Ê then
                ŵ_{x,x'} ← ŵ_{x,x'} + 1
            else
                Ê ← Ê ∪ {(x, x')}                ▷ Add edge to the sample
                ŵ_{x,x'} ← 1
                Snowball(d − 1, m, x')
            end if
        end for
    end if
end procedure

procedure RandomWalkStep(x_t)
    neighbourSet ← {x : (x_t, x) ∈ Ê ∧ x ∉ {x_0, ..., x_t}}
    if neighbourSet ≠ ∅ then                     ▷ Randomly select a neighbour
        Select randomly x_{t+1} ∈ neighbourSet
    else                                         ▷ Restart from a random solution x
        x_{t+1} ← hc(x)
        N̂ ← N̂ ∪ {x_{t+1}}
    end if
    return x_{t+1}
end procedure
```

Figure 1 and Algorithm 2 illustrate the local optima network sampling procedure, which requires a hill-climbing algorithm (Algorithm 1), a mutation operator *op*, and a snowball sampling procedure. An initial local optimum is obtained using hill-climbing (*hc*) starting from a randomly generated solution. This initial local optimum is the starting point of the random walk, whose length is controlled by a parameter l indicating the number of steps. At each step of the walk, a snowball procedure is computed as follows: let x_t be the local optimum sampled at step t of the random walk. From x_t, a snowball sampling is performed, by applying m times the mutation operator *op* followed by hill-climbing to produce neighbouring local optima. Then, the edges and the corresponding weights from x_t are updated. Using recursion, the snowball procedure (with a decreasing depth) is invoked from each adjacent node. For the next step in the walk x_{t+1}, a neighbouring node of x_t that is not already in the walk is selected. If this is not possible (*i.e.* if all x_t adjacent nodes are already in the walk), then x_{t+1} is set as a local optimum obtained from a randomly generated solution, that is $x_{t+1} = hc(x)$ where x is a random solution. The second part of Table 1 summarises the main sampling metrics.

The random walk allows for the estimation of the network metrics that are based on the correlations between neighbouring nodes. The snowball procedure permits the estimation of metrics that require higher-oder interactions (that is, "neighbours of neighbours") such as the clustering coefficient. Moreover, along the sampling, a number of hill-climbing runs are performed, allowing us to opportunistically extract other metrics such as the average length of adaptive walks, the average maximum length of adaptive walks to reach each local optimum of the sampled set, and the corresponding average number of adaptive walks to each local optimum[1]. These sampled values have been used, together with network metrics, to predict the performance of metaheuristics on large problem instances (work to be presented elsewhere).

5 Empirical Validation

In order to validate the sampling methodology, we compared the estimated network metrics against their exact values obtained from previous work on small instances. Various sampling parameters and instance types were considered. We report here the QAP landscape experiments. Three instance types were tested, namely, uniform (*uni*) and two settings of real-like instances ($rl1$) and ($rl2$) as described in Sect. 2.1. For each type and size, 30 instances were generated. The depth of the snowball sampling procedure was set to $d = 2$. Table 2 summarizes the remaining parameters for both the small and larger instances.

Table 2. Parameters for the empirical validation of the sampling procedure. N: problem dimension. Sampling parameters, m: number of edges, l: size of the random walk.

	Small	Large
Problem size (N)	10	$30, 50, 70, 100, 150$
Sampling (m, l)	$(15, 50), (30, 50), (15, 100)$	$(30, 100), (60, 100), (30, 400)$
op strength (D)	2	4

[1] Here an adaptive walk means that from a given point the walk goes to a randomly chosen neighbor if the neighbor's fitness is better, otherwise it tries another random neighbor.

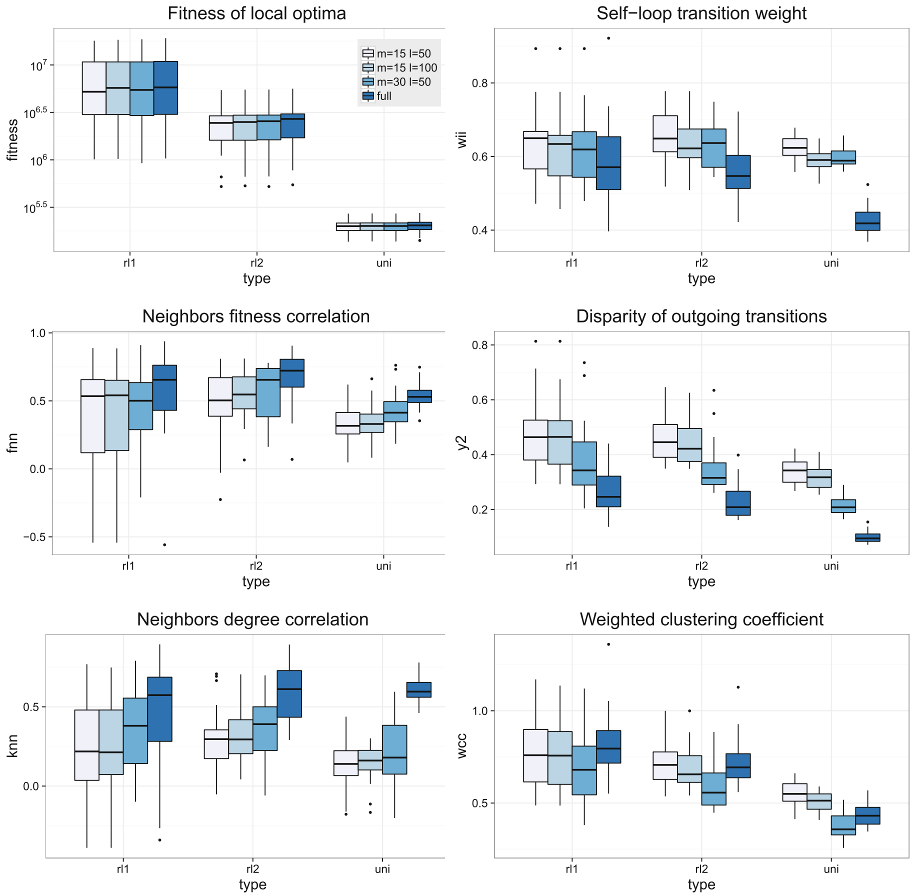

Fig. 2. Small QAP instances ($n = 10$). Estimated network metrics depending on instance type. Boxplots labelled 'full' correspond to the metrics calculated from the complete networks, whereas the remaining boxplots illustrate different sampling parameter pairs (m, l).

The plots in Fig. 2 show the network metrics: *fit, wii, fnn, y2, knn and wcc*, described in Table 1, for the small QAP instances. These metrics have been shown to be significantly related to search performance in previous work [18,19]. The colours in the plot refer to the different sampling parameter pairs (m, l), whereas the curves named 'full' correspond to the metrics calculated from fully enumerated networks, which is only possible for these small instances. The estimated metrics for the different instance classes follows the trend of the full metrics, and their values depend on the instance class. The sampling parameter m has a larger impact on the estimation with higher values producing better estimates. Some metrics are better estimated than others with fitness of local optima and weighted clustering coefficient (wcc) producing the best match.

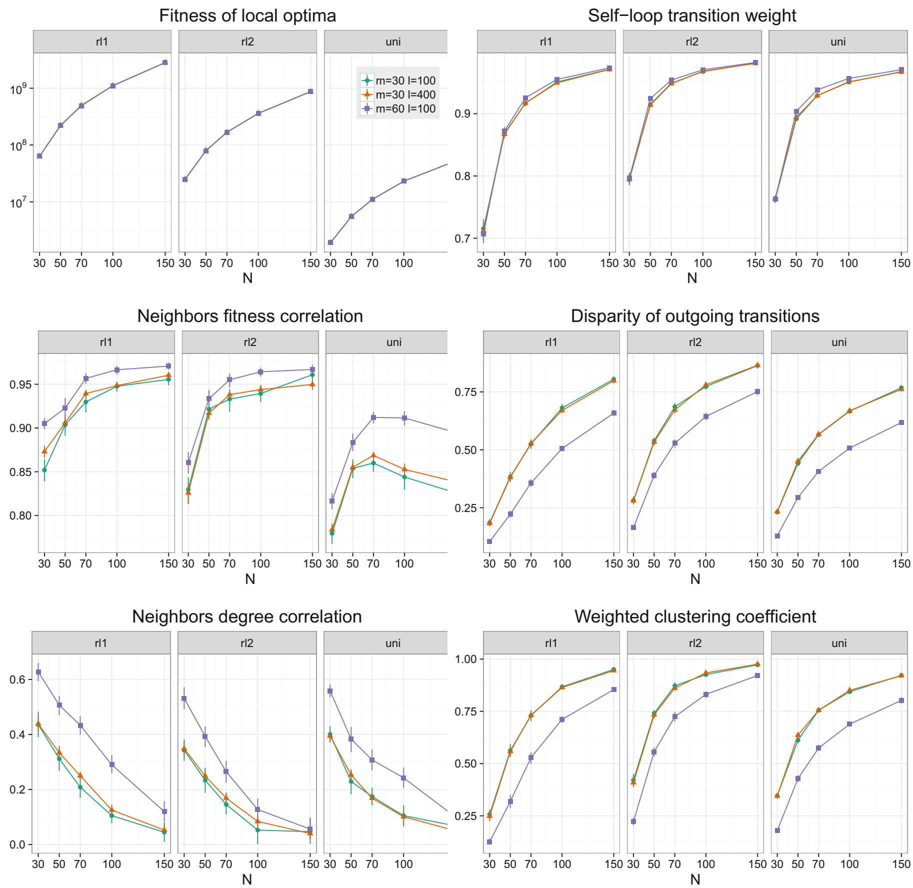

Fig. 3. Large QAP instances. Estimated network metrics as a function of the instance type (*uni, rl1, rl2*) and size ($n \in \{30, 50, 70, 100, 150\}$). The curves represent the sampling parameter pairs (m, l).

Figure 3 shows the estimated network metrics for the larger QAP instances. The curves represent the sampling parameter pairs (m, l), grouped from left to right according to the instance type: *uni, rl1, rl2*. The instance sizes explored, $n \in \{30, 50, 70, 100, 150\}$, are indicated in the X axis labels. The differences among the estimated metrics are small for the three sampling parameter pairs. As for the small instances, the m parameter has a larger impact on the estimation. The curves for $(m, l) \in \{(30, 100), (30, 400)\}$ are almost identical. Therefore, for a fixed computational cost it seems preferable to increase the sampling parameter m (number of edges) rather than the length of the random walk (parameter l). Beyond the quality of the estimation, the main metrics of LON are different according to the class of the QAP instances. For example, the weight clustering coefficient is lowest for uniform type of instance which corresponds to less dense network; or the self-loop transition weight is highest for the real-like instances of type 2. Without giving all details, the metrics seems to give useful information on the structure of the LON, and problem difficulty. The sampling methodology opens research directions on the performance prediction for large size instance.

6 Conclusions and Future Work

The fitness landscape metaphor has been often used in the context of metaheuristics to search for solutions to difficult problems. In the last few years, we have put forward a novel view of fitness landscapes based on a network called the Local Optima Network which captures fundamental features of the underlying fitness landscape, as well as information about the transitions among local optima basins. In previous work on relatively small instances of typical combinatorial problems we have been able to show that selected LON network statistics correlated with the problem instance difficulty and can be used to predict the performance of well-known metaheuristics on these search spaces. Although it is useful to show these capabilities in principle, the main limitation of the approach was that it required complete enumeration of all local optima in the search space, which of course can only be done for relatively small problem instances. In the present study we have shown that it is possible to sample larger search spaces without losing much in accuracy. This has been done by first comparing sampled and exhaustively enumerated spaces results for small instances, which give similar results, and then extending the procedure to larger sizes. The results obtained on uniformly random, as well as real-like QAP instances, are satisfactory and consistent.

In previous work with small instances, it was found that some network statistics were useful to predict performance [18,19]. As a follow-up work, a similar analysis could be conducted with larger sampled instances in order to find whether LON features could be more correlated with performance than basic fitness landscape features. This will allow to predict performance on larger instances using LON features. Finally, we note that the proposed methodology is not limited to sampling problem instance LON's. The same or a very similar approach could also be used to sample other features of a combinatorial search space. Many aspects remain to be studied and we intend to extend the methodology to other important combinatorial problems and their fitness landscapes. Future work is also planned to extend the sampling method to fitness landscapes that have a significant amount of neutrality.

References

1. Richter, H., Engelbrecht, A.E.: Recent Advances in the Theory and Application of Fitness Landscapes. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-642-41888-4
2. 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 2008, pp. 555–562. ACM (2008)

3. Tomassini, M., Verel, S., Ochoa, G.: Complex-network analysis of combinatorial spaces: the NK landscape case. Phys. Rev. E **78**(6), 066114 (2008)
4. Vérel, S., Daolio, F., Ochoa, G., Tomassini, M.: Local optima networks with escape edges. In: Hao, J.-K., Legrand, P., Collet, P., Monmarché, N., Lutton, E., Schoenauer, M. (eds.) EA 2011. LNCS, vol. 7401, pp. 49–60. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-35533-2_5
5. Chicano, F., Luque, G., Alba, E.: Autocorrelation measures for the quadratic assignment problem. Appl. Math. Lett. **25**(4), 698–705 (2012)
6. Reidys, C., Stadler, P.: Combinatorial landscapes. SIAM Rev. **44**(1), 3–54 (2002)
7. Ochoa, G., Verel, S., Tomassini, M.: First-improvement vs. best-improvement local optima networks of NK landscapes. In: Schaefer, R., Cotta, C., Kołodziej, J., Rudolph, G. (eds.) PPSN 2010. LNCS, vol. 6238, pp. 104–113. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-15844-5_11
8. Kauffman, S., Levin, S.: Towards a general theory of adaptive walks on rugged landscapes. J. Theor. Biol. **128**, 11–45 (1987)
9. Koopmans, T.C., Beckmann, M.: Assignment problems and the location of economic activities. Econometrica **25**(1), 53–76 (1957)
10. Daolio, F., Verel, S., Ochoa, G., Tomassini, M.: Local optima networks of the permutation flow-shop problem. In: Legrand, P., Corsini, M.-M., Hao, J.-K., Monmarché, N., Lutton, E., Schoenauer, M. (eds.) EA 2013. LNCS, vol. 8752, pp. 41–52. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-11683-9_4
11. Knowles, J., Corne, D.: Instance generators and test suites for the multiobjective quadratic assignment problem. In: Fonseca, C.M., Fleming, P.J., Zitzler, E., Thiele, L., Deb, K. (eds.) EMO 2003. LNCS, vol. 2632, pp. 295–310. Springer, Heidelberg (2003). https://doi.org/10.1007/3-540-36970-8_21
12. Taillard, E.D.: Comparison of iterative searches for the quadratic assignment problem. Locat. Sci. **3**(2), 87–105 (1995)
13. Verel, S., Ochoa, G., Tomassini, M.: Local optima networks of NK landscapes with neutrality. IEEE Trans. Evol. Comput. **15**(6), 783–797 (2011)
14. Newman, M.E.J.: Networks: An Introduction. Oxford University Press, Oxford (2010)
15. Barthélemy, M., Barrat, A., Pastor-Satorras, R., Vespignani, A.: Characterization and modeling of weighted networks. Phys. A **346**, 34–43 (2005)
16. Newman, M.E.: Assortative mixing in networks. Phys. Rev. Lett. **89**(20), 208701 (2002)
17. Goodman, L.: Snowball sampling. Ann. Math. Stat. **32**(1), 148–170 (1961)
18. Chicano, F., Daolio, F., Ochoa, G., Vérel, S., Tomassini, M., Alba, E.: Local optima networks, landscape autocorrelation and heuristic search performance. In: Coello, C.A.C., Cutello, V., Deb, K., Forrest, S., Nicosia, G., Pavone, M. (eds.) PPSN 2012. LNCS, vol. 7492, pp. 337–347. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-32964-7_34
19. Daolio, F., Verel, S., Ochoa, G., Tomassini, M.: Local optima networks and the performance of iterated local search. In: Genetic and Evolutionary Computation Conference, GECCO 2012, pp. 369–376. ACM Press (2012)