Search Difficulty of Two-Connected Ring-based Topological Network Designs
Beatrice Ombuki-Berman and Mario Ventresca

Abstract—Ring-based network design problems have many important applications, especially in the fields of logistics and telecommunications. This paper focuses on the recently proposed two-connected network with bounded rings problem. We investigate the search difficulty of both the Euclidean edge and unit edge length ring size bounds flavors of this problem by performing an information-theoretic fitness landscape analysis for several instances of the problem. Our results further confirm the hypothesis that the unit edge length version of the problem is indeed more difficult. The investigation also further reveals that smaller sized ring bounds lead to harder problems for Euclidean edge lengths. However, for unit edge lengths we did not establish such a relationship.

Index Terms—Landscape analysis, two-connected network, ring-based topology, search difficulty.

I. INTRODUCTION

The design of survivable cost-effective networks is a difficult task since the number of potential topologies for even small networks is extremely large [1]. In this paper, we study the Two-Connected Network with Bounded Rings (2CNBR) problem [2], [3], [4]. This problem is NP-hard [5], and abstracts many applications, especially in the fields of logistics and telecommunications.

The 2CNBR problem was first studied by Fortz et al. [2], [3] and involves designing a minimum cost network $T$ satisfying the conditions; (1) $T$ contains at least two node-disjoint paths between every pair of nodes, which is a connectivity constraint [6], and (2) each edge of $T$ belongs to at least one cycle whose length is bounded by a given constant $K$, which is a ring constraint [3]. Furthermore, two flavors of 2CNBR have been identified [2]. The first defines the ring constraint in terms of Euclidean edge lengths, and the second requires that each edge belongs to a cycle using at most $K$ edges.

Designing a survivable two-connected (so called low-connectivity) network at minimum-cost is one of the combinatorial optimization problems that has been widely studied [7], [8], [9], [10], and efficient methods for solving it are already available [11] (for a comprehensive survey of network design problems and their applications, see [12]). The extension of two-connected networks to include bounded rings was recently introduced by Fortz et al. [2], [13] who proposed adding ring constraints to the two-connected network such that the shortest cycle to which each edge belongs does not exceed a given maximum length $K$. This is a relevant extension since the ring constraints limit the region of influence of the traffic, which necessarily needs to be re-routed should there be a node or edge failure.

Furthermore, the emerging technology known as self-healing rings is applicable for re-routing, but only if the network satisfies the idea of bounded rings [3]. A self-healing ring is a cycle in the network equipped in such a way that any link failure is automatically detected and the traffic is re-routed using an alternative path in the cycle. When the self-healing ring technique is used, rings need to cover the network and their size must be limited, which is equivalent to setting a bound on the length of the shortest cycle including each edge, a property provided in the problem model studied in this paper.

Various researches have shown that incorporating domain knowledge into meta-heuristics search as evolutionary algorithms (EAs) can make them more effective. Thus, the more knowledge we have regarding a problem prior to a search algorithm design, the better we can exploit some of its characteristics to increase the efficiency of the algorithm [14]. For instance, in the development of an EA, knowledge about a given problem can be important in aiding the choice and design of crossover and mutation operators as well as the selection mechanism. Analyzing the search space may also help in predicting the expected performance. In this paper, we investigate the fitness landscape of the 2CNBR problem by utilizing information theoretic-based measures. Based on the results of these metrics we can provide insight into the search difficulty associated with the 2CNBR problem.

This paper is organized as follows. In Section II, we provide the formal definition of the 2CNBR problem. In Section III we discuss the fitness landscape analysis measures adopted in this work. We present the search operator used in Section IV and discuss our search space analysis for several instances of the 2CNBR problem in Section V. Section VI concludes the paper and outlines future works.

II. PROBLEM DESCRIPTION

In this section we provide a mathematical formulation of the 2CNBR problem based on that derived and used by Fortz et al. [2], [3] and adopted in [15].

Let $G = (V, E)$ be an undirected graph, where $V$ represents a set of vertices, and $E$ is the set of edges that represent possible pairs of vertices between which a direct link can be made. Each edge $e = (i, j) \in E$ (where $i$ and $j$ are any two vertices), has a non-negative cost $C_e = C_{ij}$, and a length $d_{ij}$. The constant $K$ defines the size of the shortest cycle (ring).
to which each edge belongs. Then, we can let the cost of a
network $T = (V, E_T)$ (where $E_T \subseteq E$ is a subset of possible
dges) be denoted by

$$c(E_T) = \sum_{e \in E_T} C_e. \quad (1)$$

Each subset $E_T$ is associated with an incidence vector $y = (x_e)_{e \in E_T} \in \{0, 1\}^{|E|}$ by setting $x_e = 1$ if $e \in E_T$, or $x_e = 0$ otherwise. On the other hand, each vector $y \in \{0, 1\}^{|E|}$ induces a subset $E_T = \{e \in E | x_e = 1\}$ of the edge set $E$. Then, for any subset of edges $E_T \subseteq E$

$$x(E_T) = \sum_{e \in E_T} x_e. \quad (2)$$

For each edge $e \in E$, we can define $\xi_e$, as the set of cycles in $G$ that include edge $e$ and whose length is less than or
equal to the constant $K$. That is, to differentiate which cycles are
used in imposing the ring constraint, a new variable is intro-
duced for each feasible network containing a given edge, for
all edges in the network. Hence, this represents a new
binary variable $Y_c^e$, where $c \in \xi_e$, such that

$$Y_c^e = \begin{cases} 1, & \text{if cycle } c \in E_T \text{ and covers edge } e \\ 0, & \text{otherwise.} \end{cases} \quad (3)$$

Then, given the graph $G = (V, E)$ and $V' \subseteq V$, the edge
set $\delta_G(V') = \{(i, j) \in E | i \in V', j \in V \setminus V'\}$ is called the
cut induced by $V'$. If we let $V - w = V \setminus \{w\}$ and $E - e =
E \setminus \{e\}$ be the subsets resulting from the removal one vertex
or one edge from the set of vertices or edges, respectively. Thus,
$G - w$ represents the graph $(V - w, E \setminus \delta(V))$, which is the
result of removing a vertex $w$ and its incident edges from $G$
[6].

Now, the 2CNBR problem can be mathematically formulated
as follows [3]:

$$\min \sum_{e \in E} C_e x_e \quad (4)$$

such that,

$$x(\delta(V')) \geq 2, \quad V' \subseteq V \quad (5)$$

$$x(\delta_G - w(V')) \geq 1, \quad w \in V, \quad V' \subseteq V \setminus \{w\} \quad (6)$$

$$\sum_{c \in \xi_e} Y_c^e \geq x_e, \quad e \in E \quad (7)$$

$$\sum_{c \in \xi_e, f \in \xi_e} Y_c^e \leq x_f, \quad e \in E, \quad f \in E \setminus \{e\} \quad (8)$$

$$x_e, Y_c^e \in \{0, 1\}, \quad e \in E, \quad c \in \xi_e \quad (9)$$

where $V' \neq V \neq \emptyset$.

### III. Landscape Analysis

In this section we will describe the concept of a fitness land-
scape, and its relation to search difficulty. We will also describe
the information-theoretic measures that were employed in our
analysis.

### A. The Fitness Landscape

Any search problem can be thought of in terms of a search
space representing all the candidate solutions to the problem
at hand. Probably the most popular analogy of the search space is
the fitness landscape, conceptualized by Sewall Wright [16] in
1932. He described the search space as a multi-dimensional
landscape where each solution representation is mapped to a
fitness value. Thus, the fitness landscape is a function of
both the solution representation and the search operators. The
implication being that in order to facilitate efficient searching,
the search operators should be designed in conjunction with
the implicated structure of the fitness landscape. Therefore,
if it is possible to construct landscapes that are easier to
search, it is likely that the search procedure will produce a
higher quality solution than it otherwise would. Furthermore,
it should also be possible to determine the problem difficulty
given the current representation, operators and evaluation
function, making the study of fitness landscapes a vital part
of searching.

The structure of a fitness landscape is completely deter-
mined by the characteristics of smoothness, ruggedness and
neutrality [17], all of which relate to differences in fitness
between neighboring solutions. All three characteristics arise
from the properties of the landscape’s local optima.

Assuming a maximization problem with search space $S$,
a solution $s \in S$ is defined to be a local maximum if its
fitness is greater than or equal to all of its neighbors, i.e.,
$f(s) \geq f(w) \forall w \in N(s)$, where the neighborhood $N(s)$
is defined as the set of solutions reachable from $s$ by a
single application of the search operator being considered. A
landscape is considered rugged if there is a high number of local
optima present in the landscape. In the event that few
optima exist, the landscape may either be smooth or rugged.
If optima are characterized by large basins of attraction
the landscape is considered to be smooth.

A basin of attraction of a solution $s_n$ is defined as the set
of vertices $B(s_n) = \{s_0 = V | s_1, ..., s_n \in V \text{ with } s_{n+1} \in N(s_n) \text{ and } f(s_{n+1}) \geq f(v_i) \forall i, 0 \leq i \leq n\}$. The size of a
basin is generally considered to be defined as the number of
solutions within it. Therefore, those local optima with small
attractive basins can be considered isolated [18]. Hence, larger
basins of attraction imply a smoother landscape.

Landscapes characterized by few local optima generally
contain large amounts of neutrality [19]. That is, their fitness
does not change even though their solution representations are
being altered. Under this model a search process is dominated
by long periods of neutral epochs interspersed with periods
of punctuated equilibrium where fitness will rapidly improve.
During the neutral epochs, the set of current solutions will
randomly drift through the search space. Neutral areas of
a landscape are a result of the presence of plateaus and
ridges [19], where a plateau is a subset $P$ of two or more
solutions such that for every pair of solutions $s_0, s_n \in P$ a
subset $\{s_1, ..., s_n\}$ exists where $f(s_i) = f(s_{i+1})$ and $s_{i+1} \in
N(s_i), 0 \leq i \leq n$.

Based on the above characteristics, we can deduce whether
the search will likely be difficult for an algorithm to discover
a high quality solution not. For example, if it is found that the landscape contains few isolated optima then it is probably going to be difficult for the search to discover one of these optima. Instead, most of the search time will be spent drifting over low quality neutral areas of the search space, and rarely will it discover good solutions.

Various statistical metrics have been proposed to quantify different aspects of the landscape, for example Weinberger [20] proposed the autocorrelation metric used to examine smoothness (a good review of these statistical measures is presented in [21]). However, in this paper we will base our analysis on the information-theoretic measures proposed by Vassilev, Fogarty and Miller [17]. Most landscape analysis techniques approximate characteristics of the landscape via a random walk-based analysis, which we do here as well.

B. Information-Theoretic Metrics

We will now outline the information-theoretic measures we employed in this study (initially proposed in [17] and [22]). These metrics are all based on a random walk of the landscape. That is, we begin at an initially random solution \( s_0 \) and continually apply the search operator, therefore generating a sequence of solutions \( < s_0, ..., s_n > \), where \( n \) is the length of the walk. Associated with this sequence are the corresponding sequence of fitness values \( < f(s_0), ..., f(s_n) > \) which will be used by the following measures.

The Information Content measures the ruggedness with respect to the flat or neutral areas of the landscape. The degree of flatness sensitivity is based on an empirically decided parameter \( \varepsilon \) which is restricted to the range \( [0, ..., L] \), where \( L \) is the maximum fitness difference along the random walk. Consequently, the analysis will be most sensitive when \( \varepsilon = 0 \). This measure is calculated according to:

\[
H(\varepsilon) = - \sum_{p \neq q} Pr[pq] \log_3 Pr[pq]
\]

where probabilities \( Pr[pq] \) represent the frequencies of the possible fitness transitions from solution \( p \) to \( q \) while performing a random walk. Each of the \( [pq] \)'s are elements of the string \( S(\varepsilon) = s_1 s_2 s_3 s_n \), of symbols \( s_i \in \{1, 0, 1\} \), where each \( s_i \) is recursively obtained for a particular value of \( \varepsilon \) based on Equation (11), so \( s_i = \Psi_f(i, \varepsilon) \). Thus, \( \varepsilon \) essentially represents an accuracy or sensitivity parameter of the analysis.

\[
\Psi(i, \varepsilon) = \begin{cases} 
1, & \text{if } f_i - f_{i-1} < -\varepsilon \\
0, & \text{if } |f_i - f_{i-1}| \leq \varepsilon \\
1, & \text{if } f_i - f_{i-1} > \varepsilon 
\end{cases}
\]

The Partial Information Content (PIC) which indicates the modality or number of local optima present on the landscape. The idea behind this measure is to filter out non-essential parts of \( S(\varepsilon) \) in order to acquire an indication of the modality of the random walk and therefore of the landscape. Equation (12) gives the formula to calculate the PIC, where \( n \) is the length of the original walk and \( \mu \) is the length of the summarized string \( S'(\varepsilon) \).

\[
M(\varepsilon) = \frac{\mu}{n}
\]

The value for \( \mu = \Phi(1, 0, 0) \) is determined via the recursive function

\[
\Phi_s(i, j, k) = \begin{cases} 
k, & \text{if } i > n \\
\Phi(i + 1, i, k + 1), & \text{if } j = 0 \text{ and } s_i \neq 0 \\
\Phi(i + 1, i, k + 1), & \text{if } j > 0 \text{ and } s_i \neq 0, s_i \neq s_j \\
\Phi(i + 1, j, k), & \text{otherwise}
\end{cases}
\]

When the value of \( M(\varepsilon) = 0 \) it indicates that no slopes were present on the path of the random walk, meaning that the landscape is rather flat or smooth. Similarly, if \( M(\varepsilon) = 1 \) then the path is maximally multi-modal and likely very rugged. Furthermore, it is possible to calculate the expected number of optima of a random walk of length \( n \) via

\[
E[M(\varepsilon)] = \frac{nM(\varepsilon)}{2}
\]

The Density-Basin Information (DBI) measure is given in Equation (15), and indicates the flat and smooth areas of the landscape as well as the density and isolation of peaks. Thus, it provides an idea of the landscape structure around the optima.

\[
h(\varepsilon) = - \sum_{p \in \{0, 1\}} Pr[pq] \log_3 Pr[pq]
\]

Here, \( Pr[pq] \) represents the probability of sub-blocks \( 11, 00 \) and \( 11 \) of occurring (hence, the logarithm is taken to base 3 which scales the result between 0 and 1). A high number of peaks within a small area would result in a high DBI value. Conversely, if the peak is isolated the measure will yield a low value. Thus, this information gives an idea as to the size and nature of the basins of the landscape. Landscapes with a high DBI content should be easier for an evolutionary algorithm to attract to the area of fitter solutions. In contrast, it is likely that for landscapes with a low DBI value an evolutionary algorithm is less likely to discover regions of high fitness.

IV. EXPERIMENTAL SETUP

In this section we will describe how we encode a solution to the 2CNBR problem. Additionally, we will outline the search operator used by the random walk process to gather the data which we utilize to perform the landscape analysis.

A. Solution Representation

We utilized the same solution representation as our previous work [15] whereby each network is encoded as an edge list. So, an initial feasible network is generated by randomly selecting an edge \( e_i \in E \) from the set of valid edges and adding it to the current solution \( S \). This process is repeated until adding an edge results in a valid 2-connected network. Therefore, \( S \subseteq E \) and the graph \( G \) resulting from \( S \) is valid with respect to the constraints described in Section II.
B. Search Operator

The search operator used here was based on that introduced in our previous work [15]. Briefly, the idea is to select an edge from the current network and add one edge from each vertex to the current solution (if possible), and attempt to form a polygon. In the minimal case both edges share a common point resulting in a triangle. After the new edges are added to the network, up to two redundant edges are removed. It is important to note that the network is feasible at every stage of this process. Pseudocode for this process is presented in Algorithm 1, where \( \mathcal{U}(0, 1) \) generates a uniform random value between 0 and 1 and \( S \) represents the edge list of the current solution, as described above.

Algorithm 1 Polygon-based Search Operator

```
Require: Some solution \( S \subseteq E \)
1: if \( \mathcal{U}(0, 1) < 0.6 \) then
2: randomly select edge \( e_1 = (i_1, j_1) \in S \)
3: if possible, randomly select edge \( e_2 = (i_1, j_2) \in E \setminus S \)
4: if possible, randomly select edge \( e_3 = (j_1, j_3) \in E \setminus S \)
5: if \( S \cup \{e_2\} \) is feasible then
6: \( S = S \cup \{e_2\} \)
7: end if
8: if \( S \cup \{e_3\} \) is feasible then
9: \( S = S \cup \{e_3\} \)
10: end if
11: if edge \( (j_2, j_1) \in E \setminus S \) and \( S \cup \{(j_2, j_1)\} \) is feasible then
12: \( S = S \cup \{(j_2, j_1)\} \)
13: end if
14: if edge \( (j_3, j_1) \in E \setminus S \) and \( S \cup \{(j_3, j_1)\} \) is feasible then
15: \( S = S \cup \{(j_3, j_1)\} \)
16: end if
17: end if
18:
19: if possible, probabilistically remove \( \leq 3 \) edges from \( S \)
```

In lines 3 and 4 we attempt to randomly select 2 unused edges, and if adding \( e_1 \) and/or \( e_2 \) to \( S \) is feasible, then do so in lines 6 and 9, respectively. Additional edges from \( e_1 \) and \( e_2 \) are attempted to be found and added to \( S \) in lines 11-16, therefore forming some polygon. In line 19 we attempt to remove up to 3 edges from \( S \), where larger edges are more likely to be removed. The value of 0.6 in line 1 was empirically decided, but not optimal over all problem instances.

It should also be noted that this operator preserves the validity of \( S \). That is, after it has completed \( S \) remains valid with respect to the constraints in Section II.

V. Results

In this section we will present the results of our analysis which will provide insight into the difficulty/hardness associated with the search of 2CNBR solutions. Our analysis is concerned with test problems proposed by Fortz [2] and [13], for both Euclidean and unit edge length ring size constraints. The problem instances are named according to the number of vertices present in the graph. For example N10-1 represents the first instance of a problem with 10 vertices, N10-2 the second, and so on. The ring constraints for both Euclidean and unit edge problems were based on those outlined in [2], [3], [4], [13].

In order to perform our landscape analysis we conduct 1,000 random walks, each composed of 100 applications of the search operator. We experimentally determined that this gathered a sufficient amount of data to perform a valid analysis. Each random walk begins from a randomly generated network, as described in the previous section. Each of the landscape metrics are calculated after every random walk, and averaged over the 1,000 walks. That is, the results show the average value over the total number of walks as opposed to calculating once after all walks (which most notably impacts the \( \mathbb{E}[M(e)] \) value).

A. Euclidean Edge Length

We first discuss the results obtained for experiments where the edge length is described by a Euclidean distance metric. We present the main findings of the results here, and Appendix A shows all the obtained data.

Table I presents a summary of the results for each problem grouped by the maximum ring size. The first column \( \mu \) corresponds to the average fitness (sum of edge lengths) found along a random walk. The information content (IC), density-basin information (DBI), partial information content (PIC) and expected optima (\( \mathbb{E}[M(e)] \)) are also shown. As the ring size (\( K \)) increases the average fitness tends to decrease. Therefore it seems that a larger maximum ring size implies an easier problem.

With the exception of the N10 and N20 instances the PIC value shows little fluctuation, which means that the N30, N40 and N50 instances are not very sensitive to changing ring size bounds. However, the smaller instances do show a large decrease in expected optima along a given random walk. For both N10 and N20 the \( \mathbb{E}[M(e)] \approx 36 \) and show a similar decrease to \( \mathbb{E}[M(e)] \approx 28 \) which shows that optimal values to these problems should be easier to discover since there are less local optima as the maximum ring size is relaxed. However, in general, the PIC and expected optima are similar for each instance, and thus do not aid much in differentiating which problems are harder to solve.

According to the results in Table I, the IC and DBI values have greater variability, and can then be more beneficial to our examination.

We plot the information content with respect to the maximum ring size for each vertex set size in Figure 1. Instance N20 shows some increase in IC as the maximum ring constraint is relaxed. Thus, the landscape for N20 seems to become easier for search since it exhibits a lower variety of shapes (with respect to combinations of the set \( \{1, 0, 1\} \)). Although to a lesser degree, instances of problem N10 also show a slight decrease in IC, but not likely enough to have a noticeable impact on search difficulty. Problem instances with 50 vertices (N50) shows logarithmic-style increase in IC, implying larger bounds on the ring constraint will have a lower
impact on search difficulty. The remaining two instances, N30 and N40, show a steep increase in IC as the ring size bound increases. Therefore, these two problems become increasingly difficult, and should be the hardest of the 5 problem types to solve.

Fig. 1: Information content of each problem type with respect to increasing problem size.

Figure 2 shows a plot of the summary results of the DBI. The most noticeable change is the steep decline in DBI for problem type N10. Since the IC also showed some decrease it can be inferred that the landscape is mainly smooth-flat and should be easy to search [17]. Similarly, instances of the N20 family of problems also are relatively smooth, but only when the ring size reaches its maximum of 500. As this size increases from 200 to 400 the problem shows an increase in difficulty. As with the IC, the DBI value for N50 problems seems to behave according to a logarithmic-like curve, reinforcing the previous hypothesis stating that instances of this problem become relatively easier to solve as the maximum ring size relaxes. Additionally, these DBI results for N30 and N40 also support the previous claim that the problems rapidly increase in difficulty as the ring size reaches 500.

Fig. 2: The change in density-basin information as the ring size bound is relaxed.

B. Unit Edge Length

We will now present the findings for experiments where the maximum ring size is determined by a unit edge length. The full table of results can be found in Appendix B.

From the summary results presented in Table II we observe very little change in each of the measured variables (at least within groups). This is even the case for the average network cost (μ) which implies that these problems may be noticeably more difficult than the Euclidean edge instances. The expected number of local optima along a random walk $E[M(ε)] \approx 36$ for all instances, and the partial information content is also relatively stable at about 0.49. Although a very slight decline from about 0.50 for N10 instances to 0.485 for N50 problems is observed, this can be considered negligible in practice.

Figure 3 shows a plot of the information content change for each problem, as the ring constraint increases from 3 to 16.
Despite the stagnant intragroup behavior, there is a relatively large difference between groups. Specifically, the IC decreases as the number of vertices increases. Additionally, the relative change between successive increases in problem size (vertices) with respect to the IC also decreases, that is $IC_{N_{10}} - IC_{N_{20}} \geq IC_{N_{20}} - IC_{N_{30}} \geq IC_{N_{30}} - IC_{N_{40}}$. So, as the number of vertices increases the relative difficulty decreases and seems to approach some (unknown) limit.

![Fig. 3: How the information content changes with respect to increasing ring constraint for each problem.](image)

We show a plot of the DBI with respect to an increasing ring constraint for each problem type in Figure 4. As with the IC, there is negligible intragroup change in these values. Although, a similar intergroup behavior is observed whereby the relative changes in DBI as the number of vertices increases becomes increasingly smaller. In conjunction with the other data presented in Table II we can conclude that each of these problems exhibits a relatively equal difficulty. However, since networks with a smaller number of vertices inherently have a smaller search space, the probability of discovering a high quality solution is greater than for problems with a larger number of vertices.

![Fig. 4: The change in DBI for each problem as the ring constraint increases.](image)

**C. Summary**

From the above results we were able to determine the expected search difficulty of both Euclidean and unit edge length versions of the 2CNBR problem. We found that the Euclidean edge length problems were easier to differentiate, which is a direct result of the nature of the distance metric. That is, edges are easier to differentiate if the Euclidean edge weight is utilized. If we employ the unit edge concept, a search algorithm must infer the actual Euclidean cost, which becomes increasingly difficult as the number of edges increases. This supports the hypothesis described by Fortz [2] and [13].

We also discovered that the unit edge problem difficulty is basically invariant under these measures. That is, the intragroup comparison between instances of a problem type yield little or no information regarding its difficulty. We were only able to distinguish intergroup differences according to different levels of IC and DBI, although they did behave in a similar manner.

It should also be noted that most of the above hypotheses regarding the search hardness can been validated for Euclidean edge length results by comparing with those results found in [15]. However, due to space limitations we are not able to present this comparison between the landscape analysis and experimental findings.

**VI. CONCLUSION AND FUTURE WORK**

We have provided an information-theoretic landscape analysis of the instances provided by Fortz [2] and described the expected difficulty associated with each. As a consequence, we were able to provide further support to the claim that unit edge length problems are indeed harder to solve. Our results also enabled us to describe the expected search difficulty for Euclidean edge length problems. The examination considered both an increase in number of vertices (and consequently edges) as well as an increasing maximum ring size.

Possible directions for future work include a comparison of the expected search difficulty with actual results from a search algorithm, including those we obtained in previous work [15]. Additionally, the examination of different search operators and their influence on the expected search difficulty forms another interesting direction. Also extending the analysis to include other metrics, such as those based on statistics, or even the development of new more robust metrics is an interesting possibility.

**APPENDIX**

For both Euclidean and unit edge lengths the columns are labeled according to: Prob. (problem instance), $\mu$ (average fitness along a random walk), IC (Information Content), DBI (Density-basin Information), PCI (Partial Information Content) and $E[M(\varepsilon)]$ (Expected optima along a random walk at sensitivity $\varepsilon$).
### A. All Euclidean Edge Results

<table>
<thead>
<tr>
<th>Prob.</th>
<th>K</th>
<th>(\mu)</th>
<th>IC</th>
<th>DRI</th>
<th>PIC</th>
<th>(z_{\text{M}(\mathcal{E})})</th>
</tr>
</thead>
<tbody>
<tr>
<td>N10-1</td>
<td>400</td>
<td>2238.9</td>
<td>0.7354</td>
<td>0.695</td>
<td>0.4763</td>
<td>35.205</td>
</tr>
<tr>
<td>N10-1</td>
<td>500</td>
<td>1482.4</td>
<td>0.7619</td>
<td>0.6132</td>
<td>0.4872</td>
<td>35.7541</td>
</tr>
<tr>
<td>N10-2</td>
<td>400</td>
<td>2604.5</td>
<td>0.7417</td>
<td>0.698</td>
<td>0.4733</td>
<td>35.006</td>
</tr>
<tr>
<td>N10-2</td>
<td>500</td>
<td>1725.7</td>
<td>0.7685</td>
<td>0.6384</td>
<td>0.4411</td>
<td>30.0188</td>
</tr>
<tr>
<td>N10-3</td>
<td>400</td>
<td>2386.8</td>
<td>0.6891</td>
<td>0.7108</td>
<td>0.4976</td>
<td>36.816</td>
</tr>
<tr>
<td>N10-3</td>
<td>500</td>
<td>1810.1</td>
<td>0.7891</td>
<td>0.6411</td>
<td>0.491</td>
<td>36.329</td>
</tr>
<tr>
<td>N10-5</td>
<td>400</td>
<td>1575.4</td>
<td>0.7687</td>
<td>0.4854</td>
<td>0.3374</td>
<td>24.8467</td>
</tr>
<tr>
<td>N10-5</td>
<td>500</td>
<td>1958.6</td>
<td>0.7732</td>
<td>0.6312</td>
<td>0.4486</td>
<td>33.134</td>
</tr>
<tr>
<td>N10-4</td>
<td>500</td>
<td>1421.4</td>
<td>0.6457</td>
<td>0.3954</td>
<td>0.2458</td>
<td>18.6589</td>
</tr>
<tr>
<td>N10-4</td>
<td>400</td>
<td>2591.7</td>
<td>0.699</td>
<td>0.7025</td>
<td>0.501</td>
<td>37.085</td>
</tr>
<tr>
<td>N10-5</td>
<td>400</td>
<td>2454.9</td>
<td>0.7876</td>
<td>0.6518</td>
<td>0.4571</td>
<td>33.798</td>
</tr>
<tr>
<td>N10-5</td>
<td>500</td>
<td>1614.5</td>
<td>0.7584</td>
<td>0.5664</td>
<td>0.3753</td>
<td>27.2588</td>
</tr>
</tbody>
</table>

### B. All Unit Edge Results

<table>
<thead>
<tr>
<th>Prob.</th>
<th>K</th>
<th>(\mu)</th>
<th>IC</th>
<th>DRI</th>
<th>PIC</th>
<th>(z_{\text{M}(\mathcal{E})})</th>
</tr>
</thead>
<tbody>
<tr>
<td>N10-1</td>
<td>400</td>
<td>3214.7</td>
<td>0.4937</td>
<td>0.7018</td>
<td>0.4983</td>
<td>36.875</td>
</tr>
<tr>
<td>N10-1</td>
<td>500</td>
<td>1875.5</td>
<td>0.4088</td>
<td>0.6987</td>
<td>0.4058</td>
<td>27.5893</td>
</tr>
<tr>
<td>N10-2</td>
<td>300</td>
<td>5798.8</td>
<td>0.4937</td>
<td>0.6712</td>
<td>0.4937</td>
<td>36.523</td>
</tr>
<tr>
<td>N10-2</td>
<td>500</td>
<td>3241.7</td>
<td>0.4937</td>
<td>0.7018</td>
<td>0.4983</td>
<td>36.875</td>
</tr>
<tr>
<td>N10-3</td>
<td>500</td>
<td>2598.2</td>
<td>0.4146</td>
<td>0.6165</td>
<td>0.4146</td>
<td>29.5871</td>
</tr>
<tr>
<td>N10-4</td>
<td>500</td>
<td>7858.3</td>
<td>0.4956</td>
<td>0.6531</td>
<td>0.4956</td>
<td>36.675</td>
</tr>
<tr>
<td>N10-5</td>
<td>300</td>
<td>5079.9</td>
<td>0.4976</td>
<td>0.6844</td>
<td>0.4976</td>
<td>36.829</td>
</tr>
<tr>
<td>N10-5</td>
<td>500</td>
<td>2966.9</td>
<td>0.4854</td>
<td>0.7021</td>
<td>0.4854</td>
<td>35.906</td>
</tr>
<tr>
<td>N10-4</td>
<td>500</td>
<td>2248.9</td>
<td>0.3039</td>
<td>0.4857</td>
<td>0.3039</td>
<td>21.8987</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Prob.</th>
<th>K</th>
<th>(\mu)</th>
<th>IC</th>
<th>DRI</th>
<th>PIC</th>
<th>(z_{\text{M}(\mathcal{E})})</th>
</tr>
</thead>
<tbody>
<tr>
<td>N10-4</td>
<td>500</td>
<td>3584.2</td>
<td>0.3072</td>
<td>0.672</td>
<td>0.3072</td>
<td>37.005</td>
</tr>
<tr>
<td>N10-5</td>
<td>500</td>
<td>3738.5</td>
<td>0.5056</td>
<td>0.6097</td>
<td>0.5056</td>
<td>37.9856</td>
</tr>
</tbody>
</table>

### continued on next page
| N50-4  | 16 | 53015.4 | 0.4115 | 0.5987 | 0.4806 | 35.4781 |
| N50-5  | 10 | 53376.4 | 0.4227 | 0.5944 | 0.4823 | 35.6214 |

REFERENCES


