Proceedings of the 2007 IEEE Symposium on Foundations of Computational Intelligence (FOCI 2007)

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)

This work has been supported by the Natural Sciences and Engineering Council of Canada (NSERC).

Dr. B. Ombuki-Berman is with Brock University, Computer Science Department, St. Catharines, ONT, CANADA (e-mail: bombuki@brocku.ca).

M. Ventresca is a Ph.D. student member of the Pattern Analysis and Machine Intelligence (PAMI) laboratory in the Systems Design Engineering Department, University of Waterloo, Waterloo, ONT, CANADA (email: mventres@pami.uwaterloo.ca)

1-4244-0703-6/07/$20.00 ©2007 IEEE

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 edges) be denoted by

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

Each subset E_T is associated with an incidence vector $y = (x_e)_{e \in E} E\{0,1\}^{|E|}$ 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 $E_T \subseteq E$

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

For each edge $e \in E$, we can define ξ_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 introduced for each feasible network containing a given edge, for all edges in the network. Hence, this represents a new binary variable Y_e^c, where $c \in \xi_e$, such that

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

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 $(V - w, E \setminus \delta(\{w\}))$, 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]:

$$\min \sum_{e \in E} C_e x_e \tag{4}$$

such that,

$$x(\delta(V')) \geq 2, \; V' \subset V \tag{5}$$

$$x(\delta_G - w(V')) \geq 1, \; w \in V, \; V' \subset V \setminus \{w\} \tag{6}$$

$$\sum_{c \in \xi_e} Y_e^c \geq x_e, \; e \in E \tag{7}$$

$$\sum_{c \in \xi_e, f \in c} Y_e^c \leq x_f, \; e \in E, \; f \in E \setminus \{e\} \tag{8}$$

$$x_e, Y_e^c \in \{0,1\}, \; c \in \xi_e, \; e \in E \tag{9}$$

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

III. Landscape Analysis

In this section we will describe the concept of a fitness landscape, 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 determined 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 $B(s_n) = \{s_0 \in V | \exists s_1, ..., s_n \in V \text{ with } s_{i+1} \in N(s_n)$ and $f(s_{i+1}) > f(v_i) \; \forall i, \; 0 \leq i \leq n\}$ [18]. 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 ε 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

$$H(\varepsilon) = -\sum_{p \neq q} Pr_{[pq]} \log_6 Pr_{[pq]} \tag{10}$$

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 $s_i \in \{\bar{1}, 0, 1\}$, where each s_i is recursively obtained for a particular value of ε based on Equation (11), so $s_i = \Psi_f(i, \varepsilon)$. Thus, ε essentially represents an accuracy or sensitivity parameter of the analysis.

$$\Psi(i,\varepsilon) = \begin{cases} \bar{1}, & \text{if } f_i - f_{i-1} < -\varepsilon \\ 0, & \text{if } |f_i - f_{i-1}| \leq \varepsilon \\ 1, & \text{if } f_i - f_{i-1} > \varepsilon \end{cases} \tag{11}$$

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 μ is the length of the summarized string $S'(\varepsilon)$.

$$M(\varepsilon) = \frac{\mu}{n} \tag{12}$$

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

$$\Phi_s(i,j,k) = \begin{cases} k, & \text{if } i > n \\ \Phi(i+1, i, k+1), & \text{if } j = 0 \text{ and } s_i \neq 0 \\ \Phi(i+1, i, k+1), & \text{if } j > 0,\ s_i \neq 0,\ s_i \neq s_j \\ \Phi(i+1, j, k), & \text{otherwise} \end{cases} . \tag{13}$$

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

$$\mathbb{E}[M(\varepsilon)] = \left\lfloor \frac{nM(\varepsilon)}{2} \right\rfloor \tag{14}$$

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.

$$h(\varepsilon) = -\sum_{p \in \{\bar{1}, 0, 1\}} Pr_{[pp]} \log_3 Pr_{[pp]} \tag{15}$$

Here, $Pr_{[pp]}$ represents the probability of sub-blocks $\bar{1}\bar{1}$, 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 $e_2 = (i_1, j_2) \in E \setminus S$
4:    If possible, randomly select $e_3 = (j_1, j_3) \in E \setminus S$
5:    **if** $S \cup \{e_2\}$ is feasible **then**
6:       Let $S = S \cup \{e_2\}$
7:    **end if**
8:    **if** $S \cup \{e_3\}$ is feasible **then**
9:       Let $S = S \cup \{e_3\}$
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:       Let $S = S \cup \{(j_2, j_1)\}$
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:       Let $S = S \cup \{(j_3, j_1)\}$
16:    **end if**
17: **end if**
18:
19: If possible, probabilistically remove ≤ 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 $\mathbb{E}[M(\varepsilon)]$ 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 μ 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 ($\mathbb{E}[M(\varepsilon)]$) 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 $\mathbb{E}[M(\varepsilon)] \approx 36$ and show a similar decrease to $\mathbb{E}[M(\varepsilon)] \approx 28$ 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 $\{\bar{1}, 0, 1\}$). 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

TABLE I: Summary Results for Euclidean Edge Lengths.

Prob.	K	μ	IC	DBI	PIC	$\mathbb{E}[M(\varepsilon)]$
N10	300	2538.90	0.6941	0.7067	0.4993	36.9405
	400	2213.40	0.7654	0.6634	0.4693	34.6944
	500	1563.88	0.7406	0.5398	0.3774	28.1075
N20	200	8228.83	0.4949	0.6519	0.4949	36.6155
	300	5644.80	0.4967	0.6797	0.4967	36.7488
	400	3514.90	0.4929	0.6903	0.4929	36.4644
	500	2470.80	0.3994	0.6051	0.3994	28.2547
N30	200	15860.90	0.4803	0.6202	0.4900	36.2463
	300	11741.56	0.5115	0.6324	0.4939	36.5458
	400	6746.42	0.6164	0.6685	0.4981	36.8550
	500	4093.32	0.7128	0.6667	0.4762	35.1434
N40	200	24684.95	0.4479	0.6058	0.4877	36.0718
	300	17402.60	0.4867	0.6202	0.4913	36.3398
	400	10986.28	0.5670	0.6539	0.4973	36.7940
	500	4712.54	0.7006	0.6850	0.4959	36.6150
N50	200	21410.40	0.4617	0.6085	0.4897	36.2168
	300	8056.66	0.6235	0.6673	0.5005	36.8579
	400	5409.08	0.6689	0.6915	0.5053	37.3639
	500	3271.86	0.6924	0.6860	0.5142	37.7393

TABLE II: Summary Results for Unit Edge Lengths.

Prob.	K	μ	IC	DBI	PIC	$\mathbb{E}[M(\varepsilon)]$
N10	3	3560.32	0.6122	0.7070	0.4954	36.6522
	5	3741.28	0.6386	0.7156	0.4988	36.9334
	10	3626.88	0.6266	0.7190	0.4961	36.7010
N20	3	10732.04	0.5274	0.6455	0.4959	36.6976
	5	10399.32	0.4863	0.6319	0.4919	36.3861
	10	10541.66	0.5051	0.6414	0.4926	36.4480
	16	10382.68	0.4910	0.6305	0.4909	36.3417
N30	3	21916.92	0.4442	0.6049	0.4866	35.9906
	5	22043.50	0.4487	0.6058	0.4857	35.9996
	10	22007.60	0.4489	0.6067	0.4857	35.9841
	16	22083.02	0.4521	0.6078	0.4879	36.0979
N40	3	40024.38	0.4400	0.6006	0.4884	36.1270
	5	39887.56	0.4272	0.5983	0.4863	35.9636
	10	39910.76	0.4327	0.5978	0.4855	35.9375
	16	39912.78	0.4290	0.5967	0.4866	36.0607
N50	3	56299.32	0.4179	0.5974	0.4851	35.8824
	5	56280.54	0.4157	0.5976	0.4850	35.8660
	10	56502.46	0.4189	0.5976	0.4842	35.7960
	16	56366.84	0.4184	0.5966	0.4860	35.9009

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.

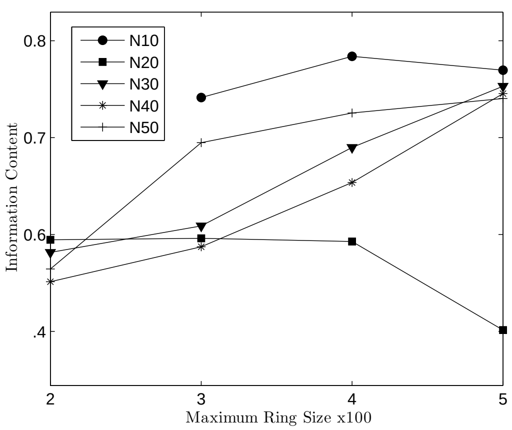

Fig. 1: Information content of each problem type with respect to increasing problem size.

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 it's 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.

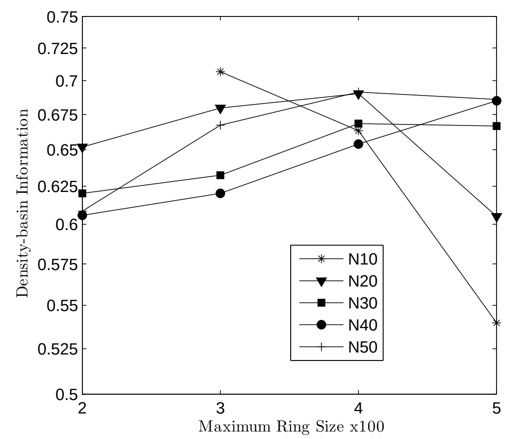

Fig. 2: The change in density-basin information as the ring size bound is relaxed.

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 $\mathbb{E}[M(\varepsilon)] \approx 36$ 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 $IC_{N10}-IC_{N20} \geq IC_{N20} - IC_{N30} ... \geq IC_{N40} - IC_{N50}$. So, as the number of vertices increases the relative difficulty decreases and seems to approach some (unknown) limit.

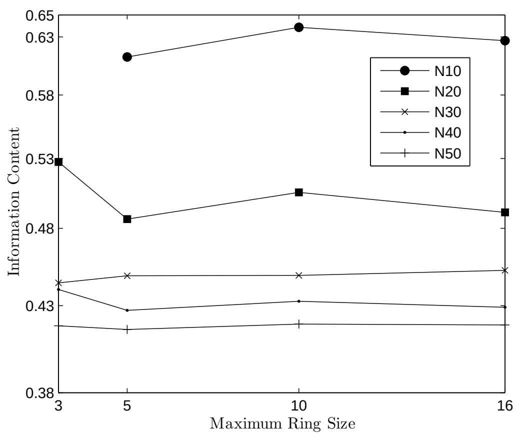

Fig. 3: How the information content changes with respect to increasing ring constraint for each problem.

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.

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.

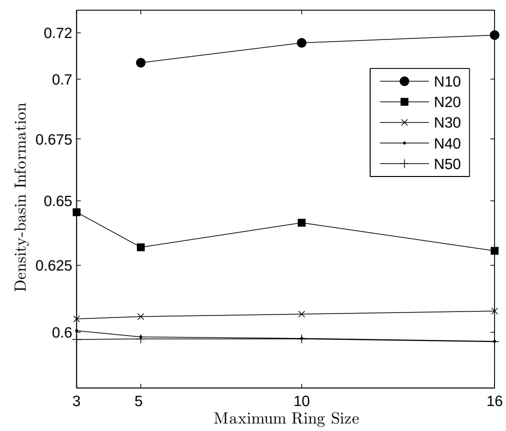

Fig. 4: The change in DBI for each problem as the ring constraint increases.

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), μ (average fitness along a random walk), IC (Information Content), DBI (Density-basin Information), PCI (Partial Information Content) and $\mathbb{E}[M(\varepsilon)]$ (Expected optima along a random walk at sensitivity ε).

A. All Euclidean Edge Results

Prob.	K	μ	IC	DBI	PIC	$\mathbb{E}[M(\varepsilon)]$
N10-1	400	2238.9	0.7354	0.695	0.4763	35.205
N10-1	500	1482.4	0.7619	0.6132	0.4872	35.7541
N10-2	400	2604.5	0.7417	0.698	0.4733	35.006
N10-2	500	1725.7	0.7685	0.6384	0.4411	34.0188
N10-3	300	2386.8	0.6891	0.7108	0.4976	36.816
N10-3	400	1810.1	0.7891	0.6411	0.491	36.329
N10-3	500	1575.4	0.7687	0.4854	0.3374	24.8467
N10-4	300	2691	0.699	0.7025	0.501	37.065
N10-4	400	1958.6	0.7732	0.6312	0.4486	33.134
N10-4	500	1421.4	0.6457	0.3954	0.2458	18.6589
N10-5	400	2454.9	0.7876	0.6518	0.4571	33.798
N10-5	500	1614.5	0.7584	0.5664	0.3753	27.2588
N20-1	200	8516.7	0.4946	0.6591	0.4946	36.59
N20-1	300	4887.1	0.4997	0.6909	0.4997	36.969
N20-1	400	2703.9	0.4845	0.6831	0.4845	35.828
N20-1	500	1873	0.3677	0.5245	0.3677	24.213
N20-2	200	8406.5	0.4935	0.6406	0.4935	36.51
N20-2	300	5798.8	0.4937	0.6712	0.4937	36.523
N20-2	400	3214.7	0.4983	0.7018	0.4983	36.875
N20-2	500	1875.5	0.4058	0.6987	0.4058	27.5893
N20-3	200	8133.8	0.4958	0.6548	0.4958	36.687
N20-3	300	6822.4	0.4957	0.6723	0.4957	36.674
N20-3	400	2846.8	0.4961	0.6923	0.4961	36.71
N20-3	500	2598.2	0.4146	0.6185	0.4146	29.5871
N20-4	200	7858.3	0.4956	0.6531	0.4956	36.675
N20-4	300	5070.9	0.4976	0.6844	0.4976	36.829
N20-4	400	2966.9	0.4854	0.7021	0.4854	35.906
N20-4	500	2248.9	0.3039	0.4857	0.3039	21.8987
N20-5	400	5842.2	0.5	0.672	0.5	37.003
N20-5	500	3758.4	0.505	0.698	0.505	37.9856
N30-1	200	15449.3	0.4862	0.6228	0.4914	36.344
N30-1	300	9252.2	0.547	0.6472	0.4982	36.868
N30-1	400	4667	0.6774	0.6966	0.4999	36.982
N30-1	500	3087.4	0.7284	0.6897	0.4785	35.5124
N30-2	300	16455.3	0.4706	0.6121	0.4896	36.217
N30-2	400	11210	0.5287	0.6286	0.4901	36.261
N30-2	500	6987.5	0.6254	0.6389	0.5001	36.8745
N30-3	300	13570.1	0.4824	0.6189	0.492	36.386
N30-3	400	7421.3	0.5559	0.6519	0.4986	36.901
N30-3	500	4257.8	0.6459	0.6882	0.5124	37.89
N30-4	200	17475.1	0.451	0.6108	0.4879	36.093
N30-4	300	10785.3	0.4953	0.6304	0.4929	36.474
N30-4	400	5622.8	0.6081	0.6836	0.4969	36.763
N30-4	500	2875.5	0.7485	0.7015	0.4687	34.8751
N30-5	200	14658.3	0.5037	0.627	0.4907	36.302
N30-5	300	8644.9	0.5623	0.6536	0.497	36.784
N30-5	400	4811	0.712	0.6817	0.5049	37.368
N30-5	500	3258.4	0.8158	0.6154	0.4215	30.5648
N40-1	200	24426.5	0.4521	0.6057	0.4885	36.118
N40-1	300	13240.8	0.5113	0.63	0.4944	36.574
N40-1	400	6609.5	0.6343	0.6836	0.503	37.227
N40-1	500	4685.5	0.7485	0.6806	0.4886	35.8745
N40-2	300	26685.8	0.4388	0.6038	0.4857	35.898
N40-2	400	19803.7	0.4617	0.6099	0.4902	36.277
N40-2	500	7065.8	0.6845	0.6555	0.5014	37.5946
N40-3	200	22739.6	0.464	0.6101	0.4912	36.338
N40-3	300	11848.2	0.5383	0.6361	0.496	36.703
N40-3	400	6193.7	0.6581	0.6864	0.5055	37.405
N40-3	500	3698.4	0.7156	0.6849	0.4879	35.8467

continued on next page

continued from previous page

N40-4	200	24360.1	0.446	0.606	0.4871	36.041
N40-4	300	14016.6	0.499	0.6242	0.4946	36.598
N40-4	400	7184.7	0.6075	0.6727	0.4983	36.863
N40-4	500	4026.7	0.7158	0.7258	0.5148	37.9948
N40-5	200	27213.6	0.4295	0.6014	0.4838	35.79
N40-5	300	21221.6	0.4459	0.6071	0.4856	35.926
N40-5	400	15139.8	0.4732	0.6169	0.4893	36.198
N40-5	500	4086.3	0.6384	0.6784	0.4867	35.7642
N50-1	200	17434.1	0.4863	0.6177	0.4932	36.486
N50-1	300	5906	0.6595	0.6928	0.5014	37.114
N50-1	400	4587.6	0.6874	0.6877	0.5078	37.5487
N50-1	500	3047.9	0.6687	0.6178	0.5168	37.9841
N50-2	300	8786.8	0.608	0.6621	0.4996	36.975
N50-2	400	6077.6	0.6719	0.6902	0.5088	37.8894
N50-2	500	3188.9	0.6974	0.7154	0.5374	38.6749
N50-3	200	22652.4	0.4592	0.6064	0.4888	36.147
N50-3	300	6684.8	0.6756	0.6793	0.4994	36.967
N50-3	400	4862.9	0.6874	0.6879	0.4875	36.5879
N50-3	500	3249.1	0.7189	0.6846	0.4803	36.4598
N50-4	200	20290.5	0.4561	0.6092	0.4887	36.142
N50-4	300	8246.7	0.5798	0.6457	0.5086	37.0649
N50-4	400	5167.6	0.6538	0.7025	0.5174	37.4682
N50-4	500	3011.8	0.6819	0.7125	0.5275	37.7314
N50-5	200	25264.6	0.4453	0.6008	0.4879	36.092
N50-5	300	10659	0.5948	0.6568	0.4934	36.1684
N50-5	400	6349.7	0.6439	0.6894	0.5049	37.3251
N50-5	500	3861.6	0.6952	0.6998	0.5088	37.8465

B. All Unit Edge Results

Prob.	K	μ	IC	DBI	PIC	$\mathbb{E}[M(\varepsilon)]$
N10-1	3	3329.9	0.6032	0.7061	0.4943	36.582
N10-1	5	3626	0.6497	0.7198	0.4976	36.812
N10-1	10	3457.5	0.6258	0.7232	0.4924	36.422
N10-2	3	3450.3	0.6058	0.7143	0.4943	36.571
N10-2	5	3815.6	0.6511	0.7219	0.4976	36.815
N10-2	10	3610.7	0.6335	0.7235	0.4945	36.593
N10-3	3	3242.4	0.606	0.7079	0.4938	36.535
N10-3	5	3544.6	0.6497	0.7202	0.4984	36.87
N10-3	10	3384.7	0.6315	0.7217	0.4962	36.72
N10-4	3	3591.4	0.5975	0.6971	0.4933	36.485
N10-4	5	3651.8	0.6119	0.7082	0.4936	36.516
N10-4	10	3724.7	0.6245	0.7112	0.4946	36.6
N10-5	3	4187.6	0.6487	0.7096	0.5011	37.088
N10-5	5	4068.4	0.6308	0.7077	0.5068	37.6541
N10-5	10	3956.8	0.6176	0.7155	0.5027	37.1698
N20-1	3	11954.8	0.5302	0.638	0.4986	36.894
N20-1	5	11599	0.483	0.6215	0.4917	36.374
N20-1	10	11657.3	0.5066	0.634	0.4922	36.425
N20-1	16	11450.6	0.4859	0.6252	0.4903	36.269
N20-2	3	9427.8	0.5284	0.6465	0.4942	36.574
N20-2	5	9168.6	0.48	0.6271	0.4895	36.205
N20-2	10	9211.3	0.5038	0.6383	0.4919	36.392
N20-2	16	9182.7	0.483	0.6283	0.4879	36.087
N20-3	3	10623.2	0.5224	0.6467	0.4957	36.683
N20-3	5	10198.1	0.4777	0.6274	0.4882	36.116
N20-3	10	10458.9	0.5052	0.6403	0.4926	36.437
N20-3	16	10382.2	0.483	0.6292	0.4889	36.174
N20-4	3	11005.4	0.5268	0.6429	0.4954	36.656
N20-4	5	10762.3	0.4905	0.6306	0.4937	36.523
N20-4	10	10823.5	0.5033	0.6355	0.494	36.551
N20-4	16	10698.4	0.4987	0.6265	0.4913	36.4782
N20-5	3	10649	0.5293	0.6535	0.4956	36.681

continued on next page

continued from previous page

N20-5	5	10268.6	0.5001	0.6527	0.4965	36.7125
N20-5	10	10557.3	0.5067	0.6588	0.4925	36.4351
N20-5	16	10199.5	0.5046	0.6431	0.4961	36.7005
N30-1	3	22533.4	0.4433	0.607	0.4856	35.914
N30-1	5	22691.8	0.4415	0.606	0.4875	36.061
N30-1	10	22644.8	0.4499	0.6072	0.4877	36.075
N30-1	16	22997.2	0.4549	0.6083	0.4887	36.136
N30-2	3	21807.3	0.4354	0.6024	0.4864	35.988
N30-2	5	21938.5	0.4354	0.6013	0.4854	35.894
N30-2	10	21987.2	0.4408	0.6044	0.4845	35.827
N30-2	16	22185.8	0.4486	0.6064	0.4893	36.197
N30-3	3	20758.3	0.4337	0.6007	0.4848	35.847
N30-3	5	21312.6	0.4587	0.6082	0.4913	36.345
N30-3	10	20912.8	0.4393	0.6048	0.4866	35.99
N30-3	16	20883.5	0.4433	0.6054	0.4869	36.011
N30-4	3	20560.3	0.4371	0.6016	0.4868	36.016
N30-4	5	20798.2	0.4392	0.6028	0.486	35.937
N30-4	10	20909.1	0.4449	0.606	0.4879	36.077
N30-4	16	20857.9	0.4429	0.6105	0.4876	36.1698
N30-5	3	23925.3	0.4715	0.613	0.4893	36.188
N30-5	5	23476.4	0.4685	0.6106	0.4785	35.7612
N30-5	10	23584.1	0.4697	0.6111	0.4816	35.9513
N30-5	16	23490.7	0.4706	0.6083	0.4869	35.9756
N40-1	3	40916.9	0.441	0.6005	0.4903	36.267
N40-1	5	40603.3	0.42	0.598	0.4858	35.915
N40-1	10	40672.1	0.4289	0.6001	0.4853	35.892
N40-1	16	40422.3	0.4196	0.5983	0.4832	35.734
N40-2	3	42194.4	0.4356	0.6005	0.488	36.104
N40-2	5	42077.3	0.4168	0.5986	0.4859	35.938
N40-2	10	41827.3	0.4258	0.5983	0.4865	35.993
N40-2	16	42315.3	0.4174	0.5981	0.4846	35.836
N40-3	3	40712.7	0.441	0.6038	0.4867	35.99
N40-3	5	40485.7	0.4264	0.5995	0.4845	35.823
N40-3	10	40485.7	0.4306	0.6004	0.4847	35.854
N40-3	16	40573.4	0.4268	0.6015	0.4875	35.9576
N40-4	3	39699.8	0.4417	0.5987	0.488	36.101
N40-4	5	39786.4	0.4368	0.5964	0.4877	36.0474
N40-4	10	39884.6	0.4406	0.5973	0.4891	36.2987
N40-4	16	39906.1	0.4386	0.5941	0.4878	36.2012
N40-5	3	36598.1	0.4406	0.5996	0.4891	36.173
N40-5	5	36485.1	0.4358	0.5988	0.4875	36.0945
N40-5	10	36684.1	0.4377	0.5931	0.4821	35.6497
N40-5	16	36346.8	0.4425	0.5913	0.4897	36.5746
N50-1	3	57464.7	0.4151	0.5968	0.4851	35.875
N50-1	5	57497.5	0.4153	0.5982	0.4831	35.727
N50-1	10	58280.6	0.4167	0.5994	0.4846	35.827
N50-1	16	58354.2	0.4162	0.5977	0.4856	35.923
N50-2	3	58676	0.4126	0.5968	0.4856	35.922
N50-2	5	58440.5	0.4128	0.5969	0.4866	35.974
N50-2	10	58767.5	0.4153	0.5973	0.4862	35.961
N50-2	16	58753.4	0.4201	0.597	0.4868	36.007
N50-3	3	54678.7	0.4123	0.5963	0.4844	35.827
N50-3	5	55030.5	0.4141	0.5976	0.4861	35.956
N50-3	10	54721.5	0.4172	0.5964	0.487	35.996
N50-3	16	54824.7	0.4156	0.5978	0.4895	36.0945
N50-4	3	57158	0.4273	0.5982	0.4851	35.888
N50-4	5	56953.1	0.4186	0.5975	0.4852	35.9046
N50-4	10	57166.3	0.4225	0.6004	0.4811	35.5746
N50-4	16	56886.5	0.4287	0.5918	0.4876	36.0018
N50-5	3	53519.2	0.4221	0.5989	0.4855	35.9
N50-5	5	53481.1	0.4175	0.5977	0.4842	35.7684
N50-5	10	53576.4	0.4227	0.5944	0.4823	35.6214
N50-5	16	53015.4	0.4113	0.5987	0.4806	35.4781

continued on next page

continued from previous page

REFERENCES

[1] A. R. P. White, J. W. Mann, and G. D. Smth, "Genetic Algorithms and Network Ring Design," *Annals of Operations Research*, vol. 86, pp. 347–371, 1999.

[2] B. Fortz, *Design of Survivable Networks with Bounded Rings*. PhD thesis, Universite Libre de Bruxelles, 1999.

[3] B. Fortz, M. Labbe, and F. Maffoli, "Solving the Two-connected Network with Bounded Meshes Problem," *Operations Research*, vol. 48, no. 6, pp. 866–877, 2000.

[4] B. Fortz and M. Labbe, "Polyhedral Results for Two-connected Networks with Bounded Rings," *Mathematical Programming*, vol. 93, no. 1, pp. 27–54, 2002.

[5] M. R. Garey and D. S. Johnson, *Computers and Intractability, A Guide to the Theory of NP-Completeness*. W.H. Freeman and Company, 1979.

[6] D. Jungnickel, *Graphs, Networks and Algorithms*. Springer-Verlag, 1999.

[7] A. M. Alvarez, J. L. Gonzalez-Velarde, and K. De-Alba, "Graph Embedded Scatter Search for the Multicommodity Capacitated Network Design Problem," *Journal of Heuristics*, vol. 11, pp. 233–257, 2005.

[8] K. K. Aggarwal, Y. C. Chopra, and J. S. Bajwa, "Topological Layout of Links for Optimizing the Overall Reliability in a Computer Communications System," *Micro-electronics and Reliability*, vol. 22, no. 3, pp. 347–351, 1982.

[9] M. M. Atiqullah and S. S. Rao, "Reliability Optimization of Communication Networks using Simulated Annealing," *Micro-electronics and Reliability*, vol. 33, no. 9, pp. 1303–1319, 1993.

[10] G. Walters and D. K. Smith, "Evolutionary Design Algorithm for Optimal Layout of Tree Networks," *Engineering Optimization*, vol. 24, pp. 261–281, 1995.

[11] F. Altiparmak, B. Dengiz, and A. E. Smith, "Optimal Design of Reliable Computer Networks, A Comparison of Meta-heuristics," *Journal of Heuristics*, vol. 9, no. 6, pp. 471–487, 2003.

[12] T. Magnanti and R. Wong, "Network Design and Transportation Planning: Models and Algorithms," *Transportation Science*, vol. 19, no. 1, pp. 1–55, 1984.

[13] B. Fortz, *Design of Survivable Networks with Bounded Rings*. Kluwer, 2000.

[14] P. Merz and B. Freisleben, "Fitness Landscapes, Memetic Algorithms, and Greedy Operators for Graph Bipartitioning," *Evolutionary Computation*, vol. 8, no. 1, pp. 61–91, 2000.

[15] M. Ventresca and B. Ombuki, "A Genetic Algorithm for the Design of Two-connected Networks with Bounded Rings," *Computational Intelligence and Applications*, vol. 5, no. 2, pp. 267–281, 2005.

[16] S. Wright, "The Role of Mutation, Inbreeding, Crossbreeding and Selection in Evolution," in *Sixth International Conference on Genetics*, pp. 356–366, Oxford University Press, 1932.

[17] V. K. Vassilev, T. C. Fogarty, and J. F. Miller, *Advances in Evolutionary Computation: Theory and Applications*.

[18] T. Jones, *Evolutionary Algorithms, Fitness Landscapes and Search*. PhD thesis, University of New Mexico, 1995.

[19] L. Barnett, "Netcrawling-Optimal Evolutionary Search with Neutral Networks," in *Congress on Evolutionary Computation*, pp. 30–37, 2001.

[20] E. Weinberger, "Correlated and Uncorrelated Landscapes and How to Tell the Difference," *Biological Cybernetics*, vol. 63, 1990.

[21] B. Naudts and L. Kallel, "A Comparison of Predictive Measures of Problem Difficulty in Evolutionary Algorithms," *IEEE Transactions on Evolutionary Computation*, vol. 4, no. 1, pp. 1–16, 2000.

[22] V. K. Vassilev, T. C. Fogarty, and J. F. Miller, "Information Characteristics and the Structure of Landscapes," *Evolutionary Computation*, vol. 8, no. 1, pp. 31–60, 2000.