Search for the Most Reliable Network of Fixed Connectivity Using Genetic Algorithm

Ho Tat Lam
The Hong Kong University of Science and Technology,
Clear Water Bay, Hong Kong
htlam96@hotmail.com

Kwok Yip Szeto
The Hong Kong University of Science and Technology,
Clear Water Bay, Hong Kong
phszeto@ust.hk

ABSTRACT
Reliability is one of the important measures of how well a system meets its design objective, and mathematically is the probability that a system will perform satisfactorily for a given period of time. When the system is described by a network of N components (nodes) and L connections (links), the reliability of the system becomes a network design problem that is an NP-hard combinatorial optimization problem. In this paper, genetic algorithm is applied to find the most reliable connected network with the same connectivity, (i.e. with given N and L). The accuracy and efficiency of genetic algorithm in the search of the most reliable network(s) of same connectivity is verified by exhaustive search. Our results not only demonstrate the efficiency of our algorithm for optimization problem for graphs, but also suggest that the most reliable network will have high symmetry.

Categories and Subject Descriptors
G.2.2 [**Mathematic of Computing**]: Graph Theory– *Graph algorithms, Network design problem.*

General Terms
Algorithms

Keywords
Reliability of networks, Genetic Algorithm, Graph optimization

1. INTRODUCTION
Reliability of a network describing the various components and their connections is an important measure, which gives the probability that the system will perform satisfactorily for at least a given period of time. Reliability of a system can be significantly improved by designing the system. Assuming the issue of reliability of the individual components is standardized, the objective of designing a most reliable system becomes the problem of finding the most reliable network. In this paper, a general method of applying genetic algorithm in graph searching is introduced to find the most reliable network with the same connectivity, (i.e. with given N and L), which can be verified by exhaustive search if the network is not too large. We focus on the design of network assuming that the system consists of N components and there are exactly L links connecting them, with at most one link connecting two nodes. Here we assume every link of the network has same robustness, measured by a parameter p which is the probability that the link is working (and probability $1-p$ that the link is broken). We follow the well-established framework in the theory of signature [1] to analyze network reliability. The success of the system in terms of reliability means that every pair of components (nodes in the network language) has at least one path connecting them. In this context, the optimization problem of the reliability of a network is equivalent to the following design problem: given N nodes and L links, find the most reliable topology. In this paper, we assume our network is the simplest undirected unweighted network. The network design problem is to find among $C_L^{N(N-1)/2}$ possible graphs the most reliable one [1-3]. The solution space is exponentially large and our problem of network reliability is an NP-hard combinatorial optimization problem [4]. In this paper, the robustness parameter p is set to be 0.7 in our computation. Our focus is on the topology of the network.

GECCO'14, July 12–16, 2014, Vancouver, BC, Canada.
ACM 978-1-4503-2881-4/14/07.
http://dx.doi.org/10.1145/2598394.2598412

2. GENETIC ALGORITHM
Our genetic algorithm of searching for the most reliable network is shown in Algorithm. In each generation, we select the most reliable network and replace the other networks with three different operators: creation, mutation and/or crossover with the fittest chromosome.

Algorithm Genetic algorithm

```
n:= number of networks in a generation
generate n networks
evaluate network reliability
while the condition is not satisfied
produce new generation
evaluate network reliability
find the most reliable network
end while
output the most reliable network
```

A network can be described by an adjacency matrix. All the nodes of the network with N nodes are labeled from 1 to N and the adjacency matrix A of the network is a $N\times N$ matrix with entry $A_{ij}=1$ if the node i and node j are connected by a link, and $A_{ij}=0$ otherwise. The adjacency matrix representation is useful in evaluating network reliability. In order to apply genetic algorithm, the network is translated into a string representation. The upper triangle of A, if labeled by row sequentially, corresponds to a string of length $N(N-1)/2$. The entries of this long representation are either 0 or 1. For example, network (a) in Figure 1 can be represented by the long string 1001111101. However, if we only focus on the position of 1 in the long string, a network can be uniquely represented by a string of L numbers corresponding to the position of all the nonzero entries in the string. As an example, we illustrate the string representations and the adjacency matrices of three networks with 5 nodes and 7 links in Figure 1.

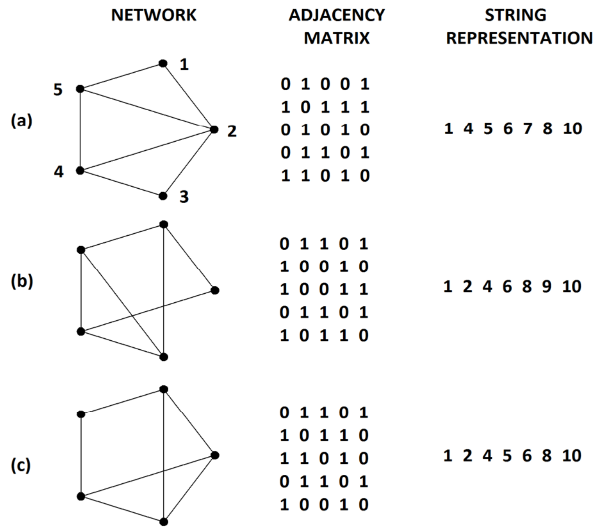

Figure 1. Networks with 5 nodes and 7 links with their adjacency matrices and short string representations. The nodes of all networks are labeled in the clockwise direction, as shown in network (a). The number in the short string representation corresponds to the ordering of the string in the long representation of length N(N-1)/2=10.

The new generation of chromosomes (networks) in our population is produced by three methods: creation, mutation and crossover. Creation creates new networks. It is used to generate the initial population and replace the networks with relatively low reliability. It also provides diversity to the population due to the random generation of chromosomes (networks). Mutation changes one of the entries of the short string. In network language, mutation operator removes one of the links in the network and places it at another empty position, which is similar to rewiring. The new network obtained from mutation has similar reliability as the original one. Thus, mutation operator allows us to perform local searching. Crossover finds the two disjoint sets of two strings, divides the set into two pieces and interchanges the pieces. It can be single or multiple point crossover. In other words, crossover divides the network into two subgraphs, retains their connection but interchanges the subgraphs. Crossover allows those reliable structure interchanges within the population. It moves the searching region from one sector to another of the solution space. It also avoids the searching from getting trapped in a local region.

As an example, let's consider the mutation from link 7 to link 2 in network (a) in the short string representation. This results in the change from the original a=(1,4,5,6,7,8,10) to c=(1,2,4,5,6,8,10). However, we can also obtain (c) by performing crossover of network (a) with network (b). Comparing b=(1,2,4,6,8,9,10) with a=(1,4,5,6,7,8,10), these two strings differ in the links 2,5,7,9. We can select the crossover points at links 2,7, to obtain two offspring networks, (c) and another network (1,4,6,7,8,9,10). Note that there are many ways to obtain (c) with different genetic operators.

3. PERFORMANCE AND ANALYSIS

We perform exhaustive search to justify the efficiency of genetic algorithm. We run our genetic algorithm and stop the search once the fittest chromosome agrees with the result obtained by exhaustive search. The computation effort of the algorithm is measured by computation units. Here each computation unit stands for the computational effort required in generating a new network and evaluating its reliability. The major computational cost arises from the evaluation of the reliability so that the searching time is proportional to the number of networks that the algorithm required to try before reaching the solution. The ratio between the number N(GA) of the networks that genetic algorithm required to reach to the exact solution obtained by exhaustive search with N(EX) number of networks tried is a measure of the efficiency of genetic algorithm compared to exhaustive search. This is listed in Table 1.

Table 1. Ratio = N(GA)/N(EX) for given N and L.

N	6	6	6	6	6	7
L	7	8	9	10	11	10
Ratio	0.019	0.045	0.222	0.221	0.051	0.012

We have performed numerical experiment for larger networks with 10 nodes using Genetic Algorithm and our results support the following conjecture: *for all pairs of N and L, the most reliable network has the most concentrated degree distribution. In other words, the nodes of the most reliable network tend to have the same degree.*

4. CONCLUSION AND DISCUSSION

We have developed a general method to find the most reliable network with N nodes and L links using genetic algorithm. Within the confine of our numerical experiments, we show that the efficiency of genetic algorithm is much higher than exhaustive search. In terms of application, our method should be useful also in other problems on network searching when we change the fitness function, such as traffic, circuit design and communication network [12].

5. ACKNOWLEDGEMENTS

K.Y. Szeto acknowledges the support of grant FS-GRF13SC25 and FS-GRF14SC28. We acknowledge many useful discussions with Degang Wu.

6. REFERENCES

[1] Francisco J. Samaniego,System signatures and their applications in engineering reliability, Boston, MA: Springer Science+Business Media, LLC, 2007

[2] Kuo, W., Prasad, V. R., Tillman, F. A., Hwang, C.: Optimal ReliabilityDesign, Cambridge University Press, Cambridge, MA (2001)

[3] Colbourn, C. J.: The Combinatorics of Network Reliability. Oxford University Press, Oxford (1987)

[4] Garey, M.R. and Johnson, D.S. (1979) Computers and Intractability:A Guide to the Theory of NP-Completeness, W.H. Freemanand Co.. San Francisco, CA.

[5] Aggamal,K.K. andRai,S.(1981)Reliability evaluation in computercommunication networks. IEEE Transactions on Reliability,R-30, 32-35.