
Evolving Failure Resilience in Scale-Free Networks

George Leu and Akira Namatame

Dept. of Computer Science, National Defense Academy
leugeorge@yahoo.com , nama@nda.ac.jp

Summary. Today our society tends to become more and more dependent on large scale (global) infrastructure networks. In many cases, attacks on a few important nodes of such systems lead to irreparable local or, worse, global damages. Thus, designing resilient networks rather than reducing the effects of some unexpected attacks becomes a must. As the most resilient network, regarding any kind of attacks, should be a full-connected graph, it is obvious that implementing such a network is a utopia. This paper proposes an original multi-objective method for optimizing complex networks' structure, taking into account the implementation costs. A micro genetic algorithm is used in order to improve networks' resilience to targeted attacks on HUB nodes while keeping the implementation costs as low as possible.

1 Introduction

Most of the existing complex networks, such as internet, power transmission grids, world-wide terrestrial, maritime or air transportation networks, are believed to have a similar statistical characteristic, power law distribution of the nodes degrees; they are so called scale-free networks. From the connectivity point of view, scale-freeness provides a well known tolerance to random failures, but they are susceptible to failures of the high-connected (HUB) nodes. Attacks on these specific nodes maylead to a very fast disintegration of the whole network.

In the last years, a huge effort has been done to analyze this vulnerability, in order to improve networks' structure and their resilience, respective. Usually, the optimized networks have been found by trying to find new analytical representations, with best results obtained after many analysis and great amount of calculation and time [4,5,7,19].

Recently, new heuristic methods based on the Genetic Algorithms (GAs) have been used to optimize networks' structure, taking in account simple attributes, such as number of links/nodes and their connectivity, degree distribution, degree sequence [1,9].

In this paper, an original method for designing low-cost networks resilient to targeted attacks is proposed. The aim is to obtain a network having the lowest implementation cost and the highest resilience to targeted attacks or, depending on the designer's goal, specific weights for the two objectives can be assigned in order to improve the performance in the desired way.

2 Multi-Objective optimization Problem

As the aim is to improve the network's resilience and keep the cost at a low level, it is obvious that a multi-objective problem has to be solved. Thus an objective function has to be found. Apart from the complex and time consuming analytical approach, the objective function shows itself in a very natural way, by simply looking at the network while thinking of the two opposite goals. In other words, it is enough to "picture" the ideas of *resilience* and *low cost* and the desired function yields clearly as sum of the two objective functions.

The first objective function is related to resilience and has to measure and improve the

ability of the network to remain connected if attacked. Maximizing this function will conduct to a very robust graph regarding attacks on high-connected nodes. The strength of the network can be measured using *topological integrity* defined as follows:

$$G = \frac{N'}{N} \quad (1)$$

where N' is the number of nodes in the largest connected network after the attack and N is the initial number of nodes. Assuming that the network is continuously attacked, nodes being removed one by one starting from most to less connected nodes, the picture of this attack and the network's integrity will look like in Fig. 1

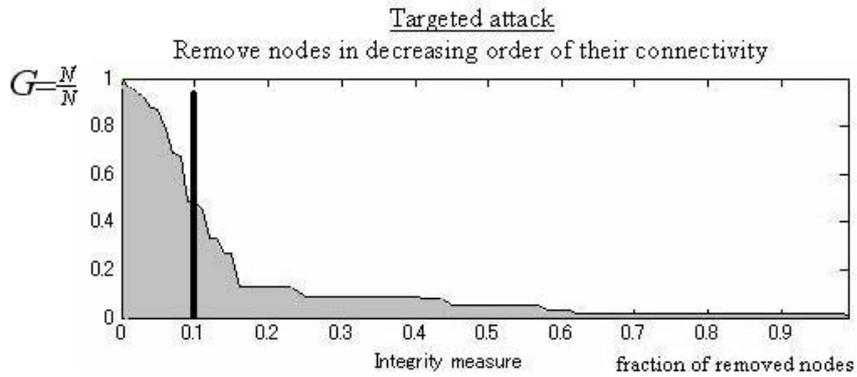


Fig. 1. Topological integrity of the network. For a Scale-Free network only 10% most connected nodes are taken in account. The other nodes are low connected, being irrelevant for the optimization.

Trying to improve the robustness of a network using this picture is the same thing with maximizing the hashed area situated under the graph of G . Still, as the network that is to be optimized is a Scale-Free network, only a few nodes are very high connected, most of them being low and very low connected. Thus, only first 10% important nodes are taken into account for the area maximization, rest of the nodes being irrelevant for the optimization process. According to this way of thinking, the first objective function will be like in Eq. (2)

$$f_{obj1} = \sum_{upto10\%} G = \sum_{upto10\%} \frac{N'}{N} \quad (2)$$

Note that the best network from the first objective point of view will always be a full (or almost full) connected graph impossible to be implemented in the real world, mainly because of the costs, but not only. For this reason, a constraint in the number of links is needed, the second objective that is.

The second objective function is related to implementation costs. "Cost" in general is a very wide notion. It consists of and depends on many other notions, such as traffic, energy, distances, clients, goods etc. If talked about cost in general then it was very difficult to find a way for minimizing it without taking in account everything, an enormous waste of processing power and time. For this reason a simple way of defining costs has been imagined, as follows. If one is trying to find an objective function for minimizing costs by simply looking at the network structure, then a basic definition of cost should only include the number of edges that

graph has. Basically speaking number of edges is indeed proportional with the implementation cost and no other parameter is needed when the topology only is to be optimized. Of course, the subject of the present paper could be anytime enlarged and many other things could be taken into account, but in this scenario only physical connectivity (no traffic, no distance) is relevant for the optimization process. According to this way of thinking the second objective will be like in Eq. (3)

$$f_{obj2} = E/E_{max} \quad (3)$$

where E is number of edges of the obtained graph and E_{max} is the maximum number of edges (of the full-connected graph). The number of edges has been normalized to the number of edges of the full-connected graph for convenience reasons only.

Note that the best network from the second objective point of view will be always a very low connected (tree like) structure, which is indeed the cheapest option but also has the lowest resilience possible.

As the purpose of this study is to find a structure which shows high resilience to targeted attacks and low implementation cost, it is necessary to find an overall objective function which combines the two opposite objectives: robustness and cost. This issue will be discussed below, in the G.A. section.

3 Genetic Algorithm

As it was said above, the whole optimizing process is based on simple observation of network's structure. The genetic algorithm modifies graph's structure trying to find a particular network which provides the best resilience and the lowest implementation cost. The whole optimizing process is described below.

Genetic Algorithms usually uses populations of *individuals*. Each of the individuals has its own *performance level* which depends of how good it is as a potential solution of the given problem. The most effective individuals are allowed to reproduce themselves, usually through the most common genetic operators, such as crossover, mutation, cloning. Thus yield new individuals called *children* who keep inside them issues from their parents. The less effective individuals will die, while the effective ones will forward their capabilities to the next generation.

Genetic Algorithms have basically several specific elements, as follows: parameters encoding for the given problem, solution search limits, objective function used to select the best individuals for the reproduction and the hazard involved in the evolution.

3.1 Parameters Encoding

G.A. starts with an initial population made by n individuals, each individual being a Scale-Free network encoded using its adjacency matrix. Unlike the common genetic algorithms which use a binary array for parameters encoding, in this paper a 2D encoding is proposed. Representing the networks using their adjacency matrix will allow the genetic operators to work directly on the network's structure thus providing fast processing and convergence.

$$A_{ij} = \begin{cases} 1 & \text{edge} \\ 0 & \text{no_edge} \end{cases} \quad (4)$$

The initial Scale-Free networks have been generated using *preferential attachment* method (Barabasi&Albert)[2,3,4].

Thus, the ten Scale-Free networks obtained are tested, the initial population is genetically modified and, through the selection process, new generations yield, better from performance point of view.

3.2 Genetic Operators

The presented algorithm only uses mutation as genetic operator. Instead of using crossover and mutation as most of the Genetic Algorithms, two types of mutation have been chosen, each of them having its specific relevance for the network's evolution.

Mutation1 provides a smooth movement in the space of solutions by making fine adjustments for finding a local optimum point. In order to do this, the operator randomly chooses one node i , take one of its stubs (element of the adjacency matrix) at random and modifies its value, or not, with equal probability.

Mutation2 provides a wide search for the global optimum point by generating large jumps in the space of solutions. In order to do this, the operator randomly chooses one node i , take all of its stubs (each element of the adjacency matrix situated on the row/column i) and modifies their values, or not, with equal probability.

The new individuals obtained through the mutation process are reinserted in the old population using ROULETTE method [1] and based on their performance regarding the objective function. The objective function will be described below in paragraph 3.3.

3.3 Objective Function

As the problem is a multi-objective one, the objective function must be carefully chosen, so the evolutionary process could go into the right direction, "increase the robustness with low cost" that is. For a good fit of the objective function with this goal, the method of "weighting coefficients" has been chosen. This means, if there is a two variables function (this means two objectives), the two objectives can be virtually separated, by giving each of them its specific importance (weight) in the optimization process, like in (5) in general:

$$f(x_1, x_2) = w_1 \cdot f(x_1) + w_2 \cdot f(x_2) \quad (5)$$

or, like in (6) for this application:

$$f_{obj} = w \cdot f_{obj_1} + (1-w) \cdot s \cdot f_{obj_2} \quad (6)$$

If decided to minimize or maximize the overall objective function then both of the terms should be minimized or maximized, respectively.

As discussed above, in paragraph 2, there are two opposite goals to be accomplished in this application. First is to maximize the area (2) and second, to minimize the normalized cost (3). This means that in the overall objective function one term is to be maximized and one to be minimized. This can be easily fixed by using the opposite value of one of the terms, as follows:

$$f_{obj} = w \cdot \frac{1}{f_{obj_1}} + (1-w) \cdot s \cdot f_{obj_2} \quad (7)$$

Now both of the terms have to be minimized, so the overall objective function will be also minimized. Minimizing the overall objective function f_{obj} is the best option for this type of application as it provides in the end a very easy way to understand the results.

In (6,7) 's' is a scaling coefficient introduced there for bringing the two terms in the same variation range. Note that without coefficient 's' the second term would be always much smaller than the first one during the evolutionary process, and thus irrelevant for the optimization.

$$\frac{1}{f_{obj_1}} \cong s \cdot f_{obj_2} \quad (8)$$

As about the weighting coefficients, unlike the general use of them, only one single coefficient has been used, as 1's complement, in order to let one decide the importance of the two objectives for his specific application ($w \in [0,1]$).

3.4 Scenario and Process Flow

Number of networks per population is 10. Each network has Power-Law distribution of degrees, generated using Preferential Attachment. Each network has 500 nodes. The algorithms runs until 50 generations are processed. The genetic algorithm flows as follows:

Compute initial population Pop_{init};

WHILE (number of generations < 50)

- *select individuals for reproduction;*
- *create offsprings using mutation1;*
- *create offsprings using mutation2;*
- *compute new generation (reinsertion);*

END

4 Results

In order to test the strength of the proposed GA, w was set to 1 and 0, thus disabling the second and the first objective respectively.

As it was expected, for $w=1$ only the first objective has been taken into account (Eq. 9), the GA evolving in the direction of improving the resilience of the network to targeted attacks. As a result, the optimized network is very robust but also very high connected, as there was no

restriction for the number of edges of the graph (Fig. 2).

$$f_{obj} = \frac{1}{f_{obj_1}} \quad (9)$$

For $w=0$ only the second objective has been taken into account (Eq. 10), the GA evolving in the direction of reducing the cost, which is in fact the connectivity of the graph. As a result, the optimized network is very cheap but also has low resilience, as there was no restriction for the robustness level (Fig. 3).

$$f_{obj} = s \cdot f_{obj_2} \quad (10)$$

At this point the genetic algorithm has been tested and worked properly for each of the objectives. Still, the purpose of this application was to optimize graph's structure taking in account both constrains. For doing this, w should be chosen between 0 and 1 depending on one's interest in either robustness or cost. Assuming that after the optimizing process the evolved network has to have the same type of structure as the initial one, w has been chosen between 0.3 and 0.5. After several simulations has been demonstrated that by choosing w in this interval, after 50 generation the network evolves in a more robust network, less connected than the initial one and having virtually the same type of degree distribution: Power-Law.

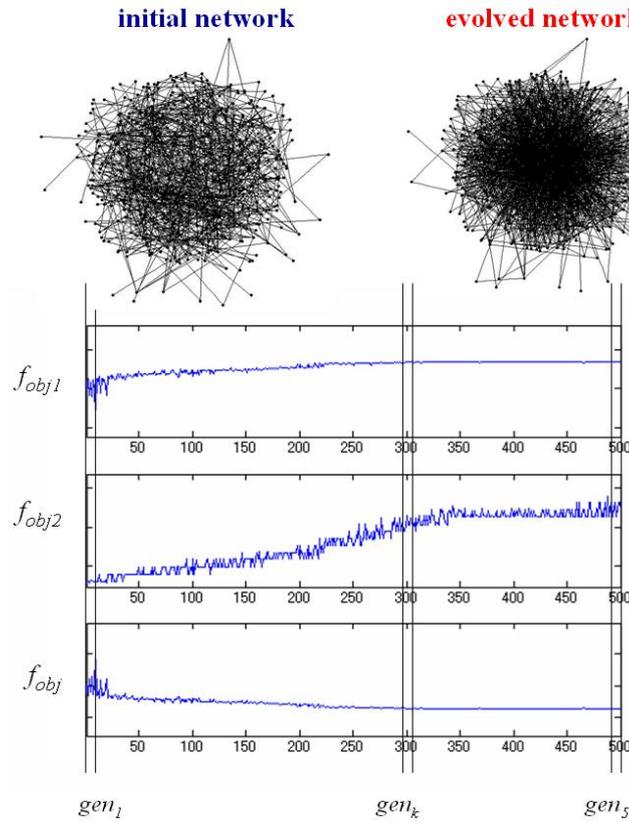


Fig. 2. TOP: Evolved network is very high connected. BOTTOM: first objective (robustness) increases. Second objective is not controlled by the GA and increases dramatically. The overall objective has the first objective's opposite variation.

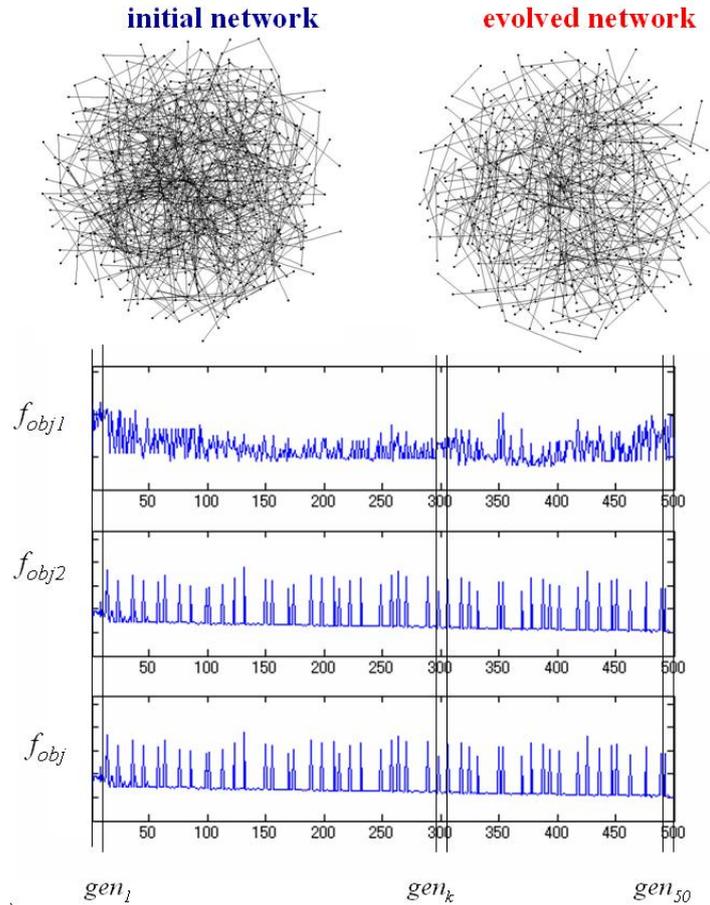


Fig. 3. TOP: Evolved network is very low connected (tree like). BOTTOM: first objective decreases, as it is not controlled by GA. Second objective is minimized. The overall objective is proportional with the second objective.

For $w=0.5$ both objectives have the same importance. As a result, the evolved network has a better resilience while the connectivity is forced to remain almost constant. The overall objective function decreases, showing that the GA evolves in the right direction (Fig. 4 Bottom). In the same time, the cumulative distribution function CDF of the evolved network is almost the same with the initial network's one (Fig. 4 Top).

For $w=0.3$ both objectives are taken into account, but the cost becomes more important than the previous case ($w=0.5$). As a result, the evolved network has a better resilience while the connectivity is forced to decrease. The overall objective function decreases, showing that the GA evolves in the right direction (Fig. 5 Bottom). In the same time, the cumulative distribution function CDF of the evolved network is again almost the same with the initial network's one (Fig. 5 Top).

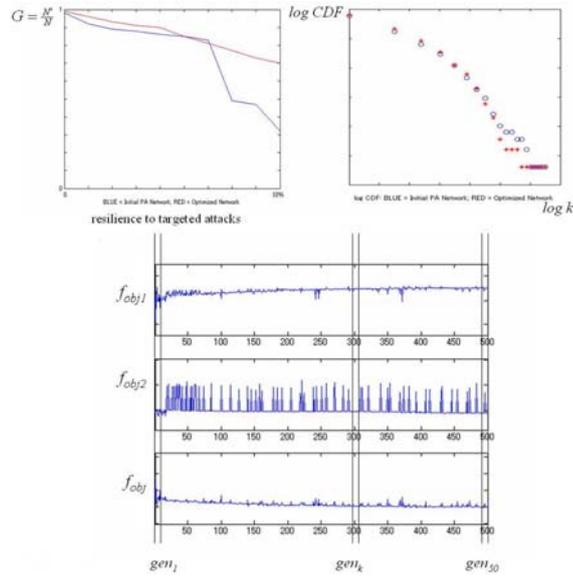


Fig. 4. TOP: circle/blue – initial network; star/red – evolved network. BOTTOM: evolution of the objective functions.

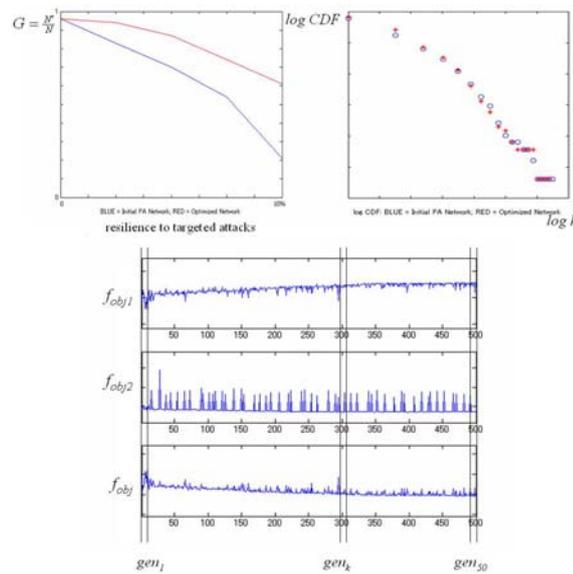


Fig. 5. TOP: circle/blue – initial network; star/red – evolved network. BOTTOM: evolution of the objective functions.

5 Conclusions

First of all, this paper demonstrated that the proposed GA, with its new type of encoding system and genetic operators, could be one viable solution for designing or optimizing large

scale networks. Taking into account both robustness and cost, strong networks can be found, depending on one's interest for either strength or cost.

Though, beyond this simple optimization issue there is more that can be said. It is well known and obvious that if the GA keeps running for a very large number of generations, the evolved network will be a random graph with low connectivity, in other words immune to targeted attacks and very cheap (Fig. 6). This is not such a good situation for the one who has a vulnerable Scale-Free network, and wants a resilient Scale-Free network. In the end he/she will need a more resilient network at the same type with the initial one. As an example, if had an internet type network and wanted to make it more robust, then turning it into a transportation type network wouldn't be such a good solution. For this reason, the most important achievement is that, that for specific values of w and appropriate number of generations the original structure of the network can be preserved in some limits. Of course, structure means in this case degree distribution *only* and this starts a new question. If the evolved network has lower connectivity, higher resilience and virtually the same degree distribution, then other parameters must have been changed during the evolutionary process. If the nodes still have almost the same distribution of degrees then the way in which they are connected to each other has changed. For analyzing this issue, parameters like assortativity, modularity, cluster coefficient or betweenness must be taken in account for the future work.

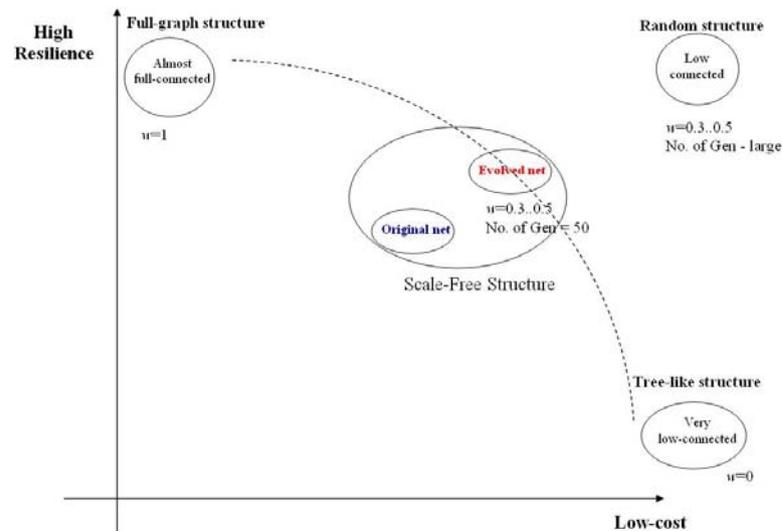


Fig. 6. For specific values of w and appropriate number of generations the original distribution of degrees can be preserved in some limits.

References

1. D.E. Goldberg. Genetic algorithms in search, optimization and machine learning. Addison-Wesley, 1989.
2. A.L. Barabasi and R. Albert. Emergence of Scaling in Random Networks. *Science* 286 (1999), 509.
3. A.L. Barabasi, R. Albert, and H. Jeong. Mean-field theory for scale-free random networks. *Physica A* 272 (1999), 173-187.
4. A.L. Barabasi, R. Albert, and H. Jeong. Scale-free characteristics of random networks: the topology of the world-wide web. *Physica A*, Volume 281, Issues 1-4, 15 June 2000, Pages 69-77

5. L. Li, D. Alderson, W. Willinger, and J. Doyle. A First-Principles Approach to Understanding the Internet's Router-level Topology, *IEEE Transactions on Networking*, 2005, p.1205-1218.
6. J.A. Boyan and M.L. Littman. Packet Routing in dynamically Changing Networks: A Reinforcement Learning Approach, *Advances in Neural Information Processing Systems 6*, 1994.
7. D. Newth and J. Ash. Evolving cascading failure resilience in complex networks, *Proceedings of "The 8th Asia Pacific Symposium on Intelligent and Evolutionary Systems"*, Cairns Australia, 2004.
8. A.E. Motter. Cascade control and defense in complex networks. *Physics Rev. Lett.* 93.098701, 2004.
9. A.E. Motter and Y.C. Lai. Cascade-based attacks on complex networks. *Physics Rev. E*.66.065102, 2002.
10. Y.C. Lai, A.E. Motter, and T. Nishikawa. Attacks and Cascades in Complex Networks. *Physics Lectures Notes*, Vol. 650, p.299, 2004.
11. G. Leu and A. Namatame. Efficient recovery from cascade failures. *Proc. of the 10th Asia Pacific Workshop on Intelligent and Evolutionary Systems*, South-Korea, 2006
12. A. Barrat, M. Barthelemy, R. Pastor-Satorras. and A. Vespignani. *Proc. Natl. Acad. Sci.*, USA 101, 3747 (2004).
13. A. Barrat, M. Barthelemy, and A. Vespignani. *J. Stat. Mech.*, P05003, 2005.
14. L. Dall'Asta, A. Barrat, M. Barthelemy, and A. Vespignani. Vulnerability of weighted networks. *DELIS*, 2006.
15. S. N. Dorogovtsev and J. F. F. Mendes. Evolution of Networks: from biological nets to the Internet and WWW. *Oxford University Press*, Oxford 2003.
16. R. Cohen, K. Erez, D. ben-Avraham, and S. Havlin. *Physics Lett.* 85, 4626, 2000.
17. D. S. Callaway, M. E. J. Newman, S. H. Strogatz, and D. J. Watts. *Physics, Lett.* 85, 5468, 2000.
18. P. Holme, B. J. Kim, C. N. Yoon, and S. K. Han. *Physics*, E 65, 056109, 2002.
19. M.E.J. Newman. The mathematics of networks. *The New Palgrave Encyclopedia of Economics*, 2nd edition.