Why simulating evolutionary processes is just as interesting as applying them *

Katharina A. Lehmann
Wilhelm-Schickard-Institute, University Tübingen
Sand 14
72076 Tübingen, Germany
lehmannk@informatik.uni-tuebingen.de

ABSTRACT

Evolutionary algorithms are very efficient tools to find a near-optimum solution in many cases. Until now they have been mostly used to find results but in this article we argue that evolutionary algorithms can also be used to simulate the evolution of complex systems. We model complex systems as networks in which agents are connected by edges if they interact with each other. It is known that many networks of this kind exhibit stable properties despite the dynamic processes they are subject to. We show here how evolutionary processes on complex systems can be modeled with a new kind of evolutionary algorithm which we have presented in [8]. We will show that some evolutionary processes within this framework yield networks with stable properties in reasonable time. An understanding of what kind of evolutionary processes will produce what kind of network properties in what time is vital to transfer evolutionary processes to technical ad-hoc networks in order to improve their flexibility and stability in quickly changing environments.

Categories and Subject Descriptors

F2.2 [**Analysis of Algorithms and Problem Complexity**]: Nonnumerical Algorithms and Problems—*Computations on discrete structures*; G2.2 [**Discrete Mathematics**]: Graph Theory—*Network Problems*; I6.5 [**Simulation and Modeling**]: Model Development

General Terms

Algorithms

Keywords

Evolutionary Algorithms, Evolution of Networks, Self-Organization of Complex Systems

*Supported by DFG-Grant Ka812/11-1

GECCO June 25–29, 2005 Washington, DC, USA

1. INTRODUCTION

One of the most influential scientific revolutions was started by the book "On the Origin of Species by Means of Natural Selection" by Charles Darwin in 1859 where he laid the foundations for the theory of evolution. Approximately 100 years later, some groups of scientists independently developed computational models that were based on the principles given by Darwin [5, 7, 10]. These models were named "Genetic algorithms" or "Evolutionary Strategies" to reveal their relation to evolution in biological systems. Today, the notion 'Evolutionary Algorithm' captures a huge number of methods which commonly use the idea of non-directed mutation and selection to optimize a given object. These methods have been successful in praxis as a black-box optimization tool: Whenever the number of possible instances of an object -the so called search space- is too big to search it exhaustively for the optimal candidate, a method based on evolutionary principles may find a way to a reasonably good instance without looking at all possible instances.

Summarizing, these methods have mainly been developed to produce a good and in best cases even optimal **result**. *Our aim is to understand how self-organization of networks can be understood as a* **process** *that yields stable and often near-optimal* **results**. In the following we will develop a philosophical viewpoint why simulating the evolution of networks can give us very interesting insights into the self-organization of complex systems in Sec. 2. We will show in Sec. 3 on very simple examples that different evolutionary processes can yield the same result but within totally different timescales. In Sec. 4 we will discuss how the understanding of evolutionary processes in social and other real-world networks may be helpful to design self-organization rules for, e.g., ad-hoc communication networks and other technical networks that need to be adaptable to changing environmental demands.

2. A PHILOSOPHICAL VIEW ON THE EVOLUTION OF COMPLEX SYSTEMS

Complex systems are systems of interacting agents where the behaviour of the whole system is not deductable from the behaviour of single agents. An agent is any entity that has access to a limited resource like money, time or energy, and that can interact with other entities.

Although complex systems evolve in many different aspects we want to concentrate on the evolution of the relations between agents that is often driven by the agents

themselves.We want to understand how this evolutionary process works such that it results in stable properties that emerge on the global system-level. Our approach is based on some assumptions: The first is that all relations in complex systems come with a cost. This is clear for all kind of communication or transportation systems but also most social systems can only be maintained if time and energy is invested into the social relationships constituting it. If relations come with a cost we can further assume that the system has to support some function which is needed by its participants. Without any gain for the agents we would not expect the agents to invest into the building of such a system. For example, the functionality provided by communication networks is the connection between numerous devices, whereas transport networks give the opportunity to ship goods and persons, and social networks seem to be an essential basis for all kind of cultural achievements in humanity. We further assume that the cost of any edge is paid by those participants of the network that are directly attached to it. Further, every participant is assumed to be able to evaluate the network's functionality from its own viewpoint, despite the fact that the participant will in most cases not be able to overview the whole network. Again, in communication networks this could be done by evaluating the connection provided to other participants, in transport networks it may be measured by the time it takes to transport some good from A to B and in social networks it may be evaluated by a lot of fuzzy feelings that describe the homophily among the participants on many dimensions. Certainly, the action of an agent within the system is depending on its evaluation of the current functionality of the system with respect to herself. We can assume that every agent requires a minimal functionality of the system: If she is content with the current functionality she will not invest more into the building of new relations but maybe she will try to remove some of the edges to minimize her cost. On the other hand, if an agent is not content with the functionality of the current system she may try to change some of her relations or to add some relations in the hope that this increases the functionality experienced by herself in subsequent timesteps. This last assumption implicitly contains that agents act selfishly, i.e., they will not invest in relations without a gain for themselves.
In summary we assume:

1. Relations between agents in complex systems come with a cost.

2. The emerging complex system provides some functionality that is dependent on the actual topology, i.e., the question of who is connected to whom.

3. Every agent can measure the functionality the systems provides to her, i.e., the evaluation function of the functionality of a network and the corresponding complex system is depending on the evaluating agent. It may thus very well be that the same topology is evaluated very differently from different agents.

4. Agents can only change their own edges, i.e., they can decide which relations to build and to remove.

5. Agents act selfishly.

In the following, we will model complex systems as networks where the participants are represented by nodes, and edges between nodes represent relations between the corresponding agents. Note that we will only represent one relation in one network despite the fact that normally agents can be related in various aspects. The evolution of complex systems is reduced to the evolution of the topology of the network, i.e., the dynamic changes of edges within the network.

Combining the above given assumptions on complex systems and agents in complex systems, it is most surprising that many dynamic - and thus evolving - networks display some very stable, global properties. We want to illustrate this statement with two examples.

2.1 Small Worlds in Social Networks

A social network is defined as a set of persons who are connected by social relations. Looking at diverse social networks like coauthorship networks, email contacts or film actor networks [1, 11, 2, 4] one can always find that they constitute so-called 'small worlds', i.e., the network is small in the sense that only a small number of edges is needed to walk from one node to another. This alone is not very interesting since any balanced tree or random graph will also have a small diameter which is scaling with $O(\log n)$ where n is the number of nodes in the network. But if we look a bit closer, social networks are more or less like grids, i.e., most of the relations emerge in local clusters. A pure grid network would show a much higher diameter than all of the social networks do, e.g., the diameter of a two-dimensional grid scales with $O(\sqrt{n})$. Thus, small-worlds are special because they combine two properties that were thought to be contradicting each other: localness of edges and a small diameter. Both characteristics are only functions of the current topology of the network and are important, e.g., for the velocity with which a virus spreads [11, 9]. The question is now how these systems can develop and maintain these characteristic, global properties despite the dynamic and indepedent decisions of people within these social networks.

2.2 Power-Law Networks

Another example for a stable property of many evolving networks is that the degree distribution is power-law shaped: The degree of a node is defined as the number of relations it participates in. The probability $P(k)$ to find a node with degree k is proportional to $k^{-\gamma}$ in a power-law, where γ is some constant. This distribution says that most nodes have a very small degree, while little of them have a huge degree. This property influences heavily the robustness and attack-sensitivity of a network and can be seen, e.g., in the Internet or the WWW [3]. How can this property evolve on a global scale if every decision of homepage designers and router engineers is made independently?

In [8] we have presented a formal framework for the evolution of networks that is based on the theory of evolution as given by Darwin and on the assumptions given above. It is a kind of specialized evolutionary algorithm that describes the self-organized and selfish evolution of networks in such a way that it is still amenable to mathematical analysis. We will show that with this framework we can try to describe what kind of rules will help to evolve a network's topology with stable properties in reasonable time and which will not. Of course, we are just at the beginning of this kind of research, so the example on which we base our discussion is simple.

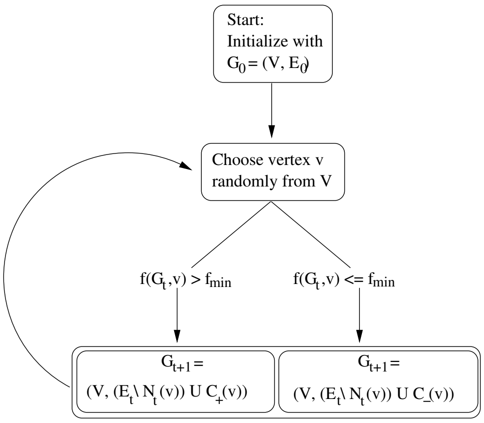

Figure 1: The framework for the evolution of networks. The model begins with a graph $G_0 = (V, E_0)$. In each time step t one of the nodes evaluates the fitness of the network G_t with respect to itself, given by the value of $f(G_t, v)$. If this is higher than a given minimal value f_{min}, the node will change its current neighborhood $N_t(v)$ to a neighborhood given by the changing rule $C_+(v)$. If the evaluation yields a value smaller or equal than the minimal value f_{min} the node changes its current neighborhood according to $C_-(v)$.

3. A FORMAL FRAMEWORK FOR THE EVOLUTION OF NETWORKS

The above given assumptions lead us to the following formal framework (s. Fig. 3): Let $G = (V, E)$ be a graph with a set of nodes V and a set of edges $E \subseteq V \times V$ where any single edge $e = (v, w) \in E$ describes a certain relation between two nodes v and w. G_t describes the graph at time step t.

In this first approach we will use a fixed number of nodes, dynamic aspects on the set of nodes are discussed in [8]. Let $f(G, v)$ be an evaluating function on G, dependent on v, that describes the functionality of G to node v. Let f_{min} describe the minimal functionality that node v wants to get from G. Further, we will assign two adaptation rules to any node v, denoted by C_+ and C_-. Let G_0 be the graph before the evolution starts and t be set to 0. The evolution of a network is then given by the following iterative procedure:

1. Choose one of the nodes $v \in V$ uniformly at random.
2. Evaluate the functionality of G_t with respect to node v by $f(G_t, v)$. If this value is lower than the minimal functionality $f_{min}(v)$ required by v, apply C_-. Otherwise apply C_+. G_{t+1} is then built from G_t with the changes implied by either of these rules.
3. Set t to $t + 1$

3.1 Is this an evolutionary algorithm?

We claim that the given framework is part of the huge family of evolutionary algorithms: The process is an iteration of evaluation of the functionality of a system and change (*mutation*) of the system. On this abstract level, the framework certainly looks like an evolutionary algorithm, e.g., a (1,1) evolutionary algorithm. Of course, the evaluation function builds the strongest deviation from other evoluationary algorithms since it is depending on the evaluating node. Thus, the system is not evaluated from the outside but from the inside and this can lead to very opposing values of the objective function on the same instance of the network. Why should one use such a personalized evaluating function instead of one global objective function? It is clear that we can always build a global objective function that is, e.g., given by the sum, minimum or average over all values $f(G, v)$ of all nodes v in the network. If we were only interested in the **result** of the optimization process, i.e., the one topology that optimizes the objective function for the system on a whole, one should certainly use a normal evolutionary algorithm with a big population size and the objective function should be given by any meaningful combination of the single values of the evaluation function of all nodes. But since we are interested in the **process** that yields stable and near-optimal results, we also have to simulate the mechanism. That is why we propose the above given framework to model the evolution of complex systems and claim at the same time that it preserves the most prevalent principles that build the basis of other evolutionary algorithms.

3.2 An illustrating example

The following simple example will illustrate the procedure: Let T be a connected tree, i.e., a graph without cycles where we can walk from every node to every other node. The distance $d(v, w)$ of two nodes in a tree is given by the minimal number of edges to be traversed to get from v to w. The eccentricity $ecc(v)$ of a node v is defined as the maximal distance it has to any other node:

$$ecc(v) = max_{w \in V} d(v, w) \tag{1}$$

The diameter $D(G)$ of a graph G is defined as the maximal distance within the graph. Let now the tree be, e.g., some overlay graph in a P2P-system where it is vital that no two nodes have a distance greater than 2, i.e., the diameter of the emerging tree should be at most 2. Then, we can formulate the functionality of any given tree with respect to some given node v to be the node's current eccentricity. The minimal functionality is then given by the requirement that the node's current eccentricity is not greater than 2. Thus, as long as the eccentricity of a node is greater than 2, some rule C_+ for the improvement of the current topology has to be applied. Rule C_- will be omitted such that a node with an eccentricity of at most 2 will change nothing. We will now show that the design details of the improvement rule C_+ are vital for the evolution of the system.

3.3 The first rule: 2nd neighbors that are non-leaves

Let v be a node with an eccentricity greater than 2. Second neighbors of v are those nodes with distance 2 to v. We will call this latter set the second non-leaf neighbors of v. The goal is to decrease the eccentricity of v in the long run and to maintain the connectedness of the tree. A simple rule to maintain the connectedness is to choose one of the second neighbors w, build an edge between v and w and remove the edge between v and that first neighbor z that is connected to both, v, and z.
To decrease the eccentricity of v in the long run we should not accept any new edge that increases the eccentricity of v. It is clear that any second neighbor of v that happens

to be a leaf (a node with only one edge) will increase the eccentricity of v. Thus, the first rule $C_+ = R_1$ is: If your eccentricity is greater than f_{min} then first choose uniformly at random one of those second neighbors w that are non-leaves. Let z be the node that is connected to both, v and w. Build an edge between v and w and remove the edge betweeen v and z if the eccentricity of v in this case is not increased compared to the eccentricity in the unchanged graph.

As we could show in [8] this rule will certainly lead to a graph with the wanted diameter of 2, but expectedly it will take exponential time to accomplish this:

LEMMA 1. *Rule R_1 will build a tree with a diameter of no more than 2 within expectedly $\Omega(\frac{1}{n}\,2^n)$ time steps.*

3.4 The second rule: Regard all distances

As we could also show in [8], a slightly changed improvement rule C_+ will reduce the number of time steps needed significantly. In this rule, denoted by R_2, an edge (v, z) is only replaced by an edge to a second neighbor w if the distance from v to **all** other nodes is strictly decreased in the new tree. We can show that the expected runtime to evolve a tree with the wanted diameter of 2 is then bounded by $O(n^5)$.

LEMMA 2. *Rule R_2 will build a tree with diameter of no more than 2 within expectedly $O(n^5)$ time steps.*

Note, that both rules will stop the evolution of the network when the diameter of the emerging network has reached a value of at most 2: If no node has an eccentricity of more than 2 - which is certainly true if the diameter is at most 2 - than no node will try to change its local neighborhood anymore.

4. DISCUSSION

These two examples have shown that the careful design of evoluationary rules within the formal framework given above can lead to the emergence of a topology with a wanted property, in this case a low diameter. Of course, this example is very simple but it shows that a network can evolve to a state with a global and stable property without the guidance of any outstanding supervisor. We have also shown that the details of a rule are extremely influential on the expected runtime of the system until the stable property emerges.

Our goal for the future is to find out what kind of simple rules will evolve what kind of network structures in what time. In this paper, the functionality of the network was chosen to be the short paths it provides between all pairs of nodes. Other possible functionalities provided by different kind of networks are: Attack resistance, redundancy of pathways to avoid congestion, or simply providing a communication backbone with minimal costs. To build new rules for evolving networks with these functionalities, we will first have to create appropriate evaluating functions that measure the effectiveness of a network's structure with respect to these functionalities. This sketches one field of further research for us. In this article, we have also simplified the model by using a uniform cost function for the edges and a fixed number of nodes in the network. Further research will show whether we can introduce more realistic edge costs, e.g., the geometrical distances of agents as an approximation for edge costs, and what happens if the node set changes dynamically.

We hope that a simulation of the evolution of networks with the rules deducted from experiments and analysis will tell us how self-organization works in real-world networks and complex systems. What could it help to understand these mechanisms? We have two visions what could be done with this knowledge:
The first is that a thorough understanding of these mechanisms makes it possible to construct small self-organizing communication devices that build their own ad-hoc network such that certain wanted properties of the topology will arise.
Whereas the first vision is technical and may be fulfilled in some years, the second is more visionary: Gleich anticipates in [6] that a new profession should arise, the 'networker'. The networker is a person that is familiar with the emergent properties of networks and their underlying complex systems and is able to recognize processes that will lead to wanted and unwanted events. He or she is the coordinator that supports the building of new edges where needed and will guide information flow through the network. Such a person will need to learn more about the mechanisms that are underlying network evolution. As we said above, the change of relations in complex systems is only one part of the whole evolution of complex systems. But since all processes on the system make use of these relations, the network's topology and its evolution build the basis that needs to be understood first and we hope that we can make some first progresses here.

5. REFERENCES

[1] L. Adamic. The small world web. In *Proceedings of ECDL'99 - Lecture Notes on Computer Science*, pages 443–452, 1999.

[2] L. Adamic and E. Adar. Friends and neighbors on the web, 2001.

[3] R. Albert, H. Jeong, and A.-L. Barabási. Error and attack tolerance of complex networks. *Nature*, 406:378–382, 2000.

[4] K. Börner, J.T. Maru, and R.L. Goldstone. The simultaneous evolution of author and paper networks. *PNAS*, 101, 2004.

[5] G.E.P Box. Evolutionary operation: a method for increasing industrial productivity. *Appl. Stat.*, 6(2):81–101, 1957.

[6] M. Gleich. *Web of Life - Die Kunst vernetzt zu leben.* Hoffman und Campe, 1st edition, 2002.

[7] J.H. Holland. *Adaptation in natural and artificial systems.* Ann Arbor: The University of Michigan Press, 1975.

[8] K.A. Lehmann and M. Kaufmann. Evolutionary algorithms for the self-organized evolution of networks. In *Proc. of GECCO 2005 (accepted)*, 2005.

[9] C. Moore and M.E.J. Newman. Epidemics and percolation in small-world networks. *Phys. Rev. E*, 61:5678–5682, 2000.

[10] I. Rechenberg. *Evolutionsstrategie: Optimierung technischer Systeme nach Prinzipien der biologischen Evolution.* Frommann-Holzoog, 1973.

[11] D. J. Watts and S. Strogatz. Collective dynamics of small world networks. *Nature*, 393:440–442, 1998.