Stability in the Self-Organized Evolution of Networks

Thomas Jansen
Univ. Dortmund
FB Informatik, LS 2
44221 Dortmund, Germany
Thomas.Jansen@udo.edu

Madeleine Theile
Univ. Dortmund
FB Mathematik, LS V
44221 Dortmund, Germany
Madeleine.Theile@udo.edu

ABSTRACT

The modeling and analysis of large networks of autonomous agents is an important topic with applications in many different disciplines. One way of modeling the development of such networks is by means of an evolutionary process. The autonomous agents are randomly chosen to become active, may apply some kind of local mutation operators to the network and decide about accepting these changes via some fitness-based selection whereas the fitness models the agent's preferences. This general framework for the self-organized evolution of networks can be instantiated in many different ways. For interesting instances, one would like to know whether stable topologies eventually evolve and how long this process may take. Here, known results for one instantiation are improved. Moreover, a more natural and local instantiation is presented and analyzed with respect to the expected time needed to reach a stable state.

Categories and Subject Descriptors

F.2.2 [**Analysis of Algorithms and Problem Complexity**]: Nonnumerical Algorithms and Problems—*Computations on Discrete Structures*; G.2.2 [**Discrete Mathematics**]: Graph Theory—*Network Problems*; I.6.5 [**Simulation and Modeling**]: Model Development

General Terms

Theory, Algorithms, Performance, Design

Keywords

evolutionary algorithms, self-organization, stability, run time analysis

1. INTRODUCTION

Large networks of autonomous agents that evolve over time can be found in many different systems, either natural (like social networks) or technical (like the Internet). In such networks it is often the case that each agent benefits from having a sufficiently central position in the network while on the other hand each direct connection draws from its limited resources. If agents are able to change their direct connections the network topology evolves over time driven by each of the agent's desire to improve its position in the network. Assuming that the network contains rational agents it is reasonable to ask whether there are stable topologies where the network's evolution comes to an halt. Clearly, this depends on the initial network as well as on the precise rules and circumstances governing the behavior of the agents. If a stable topology is eventually reached, then it is near at hand to ask how long it takes to reach such a stable network topology.

Such ensembles of interacting agents have been investigated under many different perspectives. In mechanism design [14] one is concerned with the design of protocols that ensure maximum overall profit in spite of the agents selfishness. When analyzing questions of network design with selfish agents one is interested in finding out whether a specific setting that may involve costs for connections and individual cost functions for each agent allows for stable solutions which can be characterized as Nash equilibria. Depending on the specific setting, such questions may be difficult to answer [2, 3, 7].

A quite different approach is to model the development of such a graph as an evolutionary process that is controlled by the random application of local mutation operators where the outcome is subject to a selection that is governed by the preferences of the acting agent. The main advantage of such a model is that it does not assume the availability of global information for the agents. Restricting the agent's access to local information is obviously appropriate when modeling huge networks without central control. Such an approach is taken by Lehmann and Kaufmann [12]. They present a quite general model for the self-evolution of a graph, instantiate this model, observe that for their instantiations unique stable topologies exist and provide some upper and lower bounds for the expected time the evolutionary process needs to reach such a stable topology. We improve on these results in several ways. On the one hand, we reconsider the model discussed by Lehmann and Kaufmann [12] and improve some of their bounds significantly. On the other hand, and more importantly, we present a modification of this model that is in much better accordance with the "spirit" and intentions connected to the investigation of such evolving networks. For this model, we discuss stable topologies and prove upper and lower bounds on the expected time needed to reach a stable topology.

GECCO'07 July 7-11, 2007, London, England, United Kingdom.

In the following section we give precise definitions of the considered models and the type of questions we investigate. Section 3 starts with the consideration of a simple fitness function. The rather disappointing results motivate the consideration of less obvious fitness functions. For one such function we derive results for the underlying model (Sec. 4). In both sections we start with a recapitulation of known results [12] before presenting our improved bounds. A novel model that is completely local is discussed in Section 5. Throughout, we present empirical results that help to build up a reasonable intuition for questions that our analytical results do not answer. Finally, in Section 6, we conclude and discuss possible directions of future research.

2. DEFINITIONS

The model we consider consists of three main components: the network that evolves over time, the local mutation operator, and the fitness function that describes the preferences for each agent. The network consists of the set of agents that is assumed to be constant over time and the connections between these agents that evolve over time. A formal definition follows.

Definition 1. A *network evolving over time* is defined by
- a finite set of agents V and
- for each time step $t \in \mathbb{N}_0$, an undirected graph $G_t = (V, E_t)$ with $E_t \subseteq \{\{u, v\} \mid u \neq v \in V\}$.

Furthermore, we define
- the set $M(v)$ of potential edges adjacent to $v \in V$, $M(v) = \{\{u, v\} \mid u \in V \setminus \{v\}\}$,
- $\alpha(v)$, the power set of $M(v)$, $\alpha(v) = \mathcal{P}(M(v))$,
- $\mathbb{G}$, the set of all graphs G_t, $\mathbb{G} = \{G_t\}$.

The fitness function expresses the preferences for each agent, thus, each agent $v \in V$ may have a different fitness function. Since we are interested in the development of the complete network, we consider the overall fitness, too.

Definition 2. Fitness is described by a *fitness function* $f\colon \mathbb{G} \times V \to \mathbb{R}$, with $f(\cdot, v)$ expressing the preferences of agent $v \in V$. The *overall fitness* $F\colon \mathbb{G} \to \mathbb{R}$ is given by $F(G_t) := \sum_{v \in V} f(G_t, v)$.

The mutation operator defines the kind of changes the agents are allowed to make. The idea is that an agent can decide about its own connections and no others. We assign no costs to connections.

Definition 3. For any time step $t \in \mathbb{N}_0$, let $N_t(v) \in \alpha(v)$ denote the set of all edges adjacent to v, i. e., $N_t(v) = M(v) \cap E_t$. A mutation operator is defined as a randomized function $C\colon \mathbb{G} \times V \to \alpha(v)$ replacing the current neighborhood $N_t(v)$ by a new neighborhood $N_{t+1}(v) = C(G_t, v)$ chosen randomly according to some distribution.

We embed these ingredients in an algorithm that determines how the agents are allowed to apply mutation and fitness-based selection. Together with concrete fitness functions and mutation operators, this defines the complete model.

Definition 4. **Algorithmic Frame**
1. $t := 0$; Randomly choose $G_t = (V, E_t)$ according to some fixed distribution.
2. Select $v \in V$ uniformly at random.
3. Create E^* from E_t by applying $C(G_t, v)$.
4. If $f((V, E^*), v) \succ f(G_t, v)$ then $G_{t+1} := (V, E^*)$ else $G_{t+1} := G_t$.
5. $t := t + 1$. Continue at line 2.

The selection in line 4 is based on a relation $\succ$ defined on fitness values. Depending on the fitness function either minimization or maximization may be appropriate. Moreover, we may implement either strict selection (using either $<$ or $>$) or a selection allowing for random walks on plateaus of equal fitness (using either $\leq$ or $\geq$).

Formally this algorithm consists of an infinite loop but we already remarked that we are interested in the expected number of iterations until the graph reaches a stable topology, which is true when no node can improve its current fitness with respect to the chosen fitness function and selection.

For the definition of concrete fitness functions it is useful to define some notions for the networks we are dealing with. Since we consider the evolution of networks, we define all terms with respect to some concrete current network G_t.

Definition 5.
- For $v \in V$, the *degree* $\deg_t(v)$ is the number of nodes v is directly connected to, $\deg_t(v) := |\{u \mid \{u, v\} \in E_t\}|$.
- For $u, v \in V$, a *path* $P_t(u, v)$ is a sequence of nodes v_1, v_2, …, v_k with $u = v_1$, $v = v_k$ and $\{v_i, v_{i+1}\} \in E_t$ for all $i \in \{1, 2, \ldots, k-1\}$.
- We call $k-1 = |P_t(u, v)|$ the *length* of the path $P_t(u, v)$.
- For $u, v \in V$, the *distance* $d_t(u, v)$ between u and v is defined as the minimal length of a path between u and v.
- For $v \in V$, the *eccentricity* $e(G_t, v)$ of v in G_t is defined by $e(G_t, v) := \max\{d_t(u, v) \mid u \in V\}$.
- The *diameter* $D(G_t)$ is defined by $D(G_t) := \max\{e(G_t, v) \mid v \in V\}$.
- For $v \in V$, the *closeness centrality* $c(G_t, v)$ of v in G_t is defined by $c(G_t, v) := \sum_{u \in V} d_t(u, v)$.
- The *Wiener Index* is defined by $\sum_{v \in V} c(G_t, v)$.

In practice, it may be desirable for an agent to have a central position in a network. Therefore, we consider fitness functions that aim at expressing the centrality of an agent's position. While there are several reasonable ways defining the centrality of an agent's position within a network, it is not at all clear how this notion can be defined with respect to unconnected networks. Thus we restrict our attention to the case of networks that form one single connected component. Since we do not impose any costs on the connections of an agent, obviously, the clique becomes the unique solution where all agents have central positions. As this trivial solution is not acceptable with respect to costs, we choose another way of restricting the number of connections. We concentrate our attention to connected networks with a minimal number of connections, i. e., spanning trees. This does not imply a bound on the number of connections a single agent may have while minimizing the total number of connections. This may be seen as a compromise between introducing costs for connections (complicating the analysis considerably) and neglecting connection costs completely (leading to the clique as trivial and undesirable "solution").

Our focus on spanning trees implies that we have to consider mutation operators that transform a spanning tree into another spanning tree. The following mutation operator was

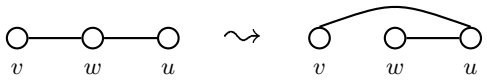

Figure 1: Application of inner mutation.

introduced by Lehmann and Kaufmann [12]. Remember that $v \in V$ is an agent selected uniformly at random in line 2 of the main algorithm.

Definition 6. **Inner Mutation**
$N := \{u \in V \mid d_t(u,v) = 2, \deg_t(u) > 1\}$
If $N \neq \emptyset$ then
  Select $u \in N$ uniformly at random.
  Let w be the unique $w \in V$ with $\{\{u,w\},\{v,w\}\} \subseteq E_t$.
  Replace $\{v,w\}$ by $\{v,u\}$.

We call this mutation operator inner mutation since it only takes agents into account that are not leaves of the spanning tree, i. e. agents that are inner nodes. In Figure 1, the application of this mutation is depicted. It is easy to see that one mutation changes a spanning tree into a different spanning tree if $N \neq \emptyset$. Otherwise, the network remains unchanged.

While it is intuitively clear to restrict the mutation to inner nodes (such nodes have the potential of moving the active agent v closer to the center), it is not strictly necessary in order to mutate spanning trees into spanning trees. Therefore, we propose a different mutation operator, which we will use throughout section 4, that is less restrictive and that we consider to be more natural.

Definition 7. **Mutation**
$N := \{u \in V \mid d_t(u,v) = 2\}$
If $N \neq \emptyset$ then
  Select $u \in N$ uniformly at random.
  Let w be the unique $w \in V$ with $\{\{u,w\},\{v,w\}\} \subseteq E_t$.
  Replace $\{v,w\}$ by $\{v,u\}$.

We close this section by remarking that we will use the terms with high probability and with overwhelming probability which we define as random events that happen with probability $(1 - O(1/n))$ and with probability $(1 - e^{\Omega(n^{\varepsilon})})$ for a constant $\varepsilon > 0$, respectively.

3. RESULTS FOR A SIMPLE FITNESS FUNCTION

We are interested in modeling selfish agents that aim at obtaining a central position in the network. Remember that we restricted our interest to networks forming spanning trees on the set of agents V. Clearly, we may measure how central an agent's position is by the distances to the other agents in the network. This leads to eccentricity as fitness function that needs to be minimized. It is easy to see that there is a unique optimal topology, the graph with diameter 2, i. e., the star. As long as the diameter of the graph is greater than 2 the leaves defining a longest path within the tree can still reduce their eccentricity. Using $\leq$ for comparison in selection together with the inner mutation leads to the algorithm investigated by Lehmann and Kaufmann [12]. For this model they proved that with probability 1 the optimal topology will be found but that there exists a family of networks where this takes $\Omega(2^n/n)$ steps. We improve this lower bound to $\Omega(2^n)$ for any network different from the optimal solution.

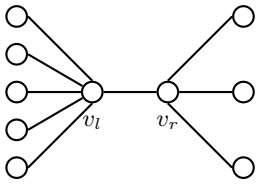

Figure 2: Network with diameter 3 with $k = 5$, $n = 10$.

THEOREM 1. *The expected number of steps needed to reach a stable topology for a network with $|V| = n$ agents different from the star using inner mutations, eccentricity as fitness function and $\leq$ for comparison is $\Omega(2^n)$.*

PROOF. The crucial observation is that the mutation can change the diameter of a network by at most 1. This is easy to see. There is at least one path in the network with length $D(G_t)$. If in a mutation the nodes u, v, w are not on such a path, obviously, the diameter cannot change. Otherwise, the length of any path including u, v, or w can change in length by at most 1. We conclude that the unique stable topology (the star with diameter 2) can only be reached via some network with diameter 3. Such a network can be seen in Figure 2. It has two central nodes, v_l and v_r, and is characterized by the number k of nodes on the left side. Each mutation leading to a change in the topology can either increase or decrease k by 1. The star is reached if k is either decreased to 0 or increased to $n-2$. It is easy to see that the probability to increase k by 1 is given by $(n-k-2)/n$, the probability to decrease k by 1 is k/n and with probability $2/n$ k is neither increased nor decreased. We recognize that this coincides with a pure random walk on the Boolean hypercube of dimension $N = n-2$. For this random walk with the all 0-string as only absorbing state the expected time to absorption is $2^N \cdot (1+o(1))$ [9]. Here we have two absorbing states, the all 0-string and the all 1-string. Clearly, the situation is symmetric in the sense that states i and $N-i$ have equivalent transition probabilities. By combining states i and $N-i$ we get a grouped Markov chain without absorbing states but rather reflecting states. We can verify that $(\pi)_i$ with $\pi_i = \binom{N}{i} \cdot c$ with $c = 2/\left(2^N + \binom{N}{N/2}\right)$ is the unique stationary distribution. We compare this to the stationary distribution of the Markov chain describing the original random walk that is given by $\pi'_i = 2^{-N} \cdot \binom{N}{i}$. We see that $\pi_i/\pi'_i = 2 - O(1/N^{1/2})$ holds. Since the stationary distribution of our Markov chain deviates from the stationary distribution of the random walk with one absorbing state only by $\Theta(1)$-factor we obtain, using the fundamental theorem of finite Markov chains [13], $\Omega(2^n)$ as lower bound on the number of steps needed to reach an absorbing state. □

We see that in the final phase when the diameter is already decreased to 3, an exponential number of steps is needed to reach the optimal topology. Lehmann and Kaufmann introduce inner mutations arguing that mutations selecting a leaf cannot lead to changes in the topology due to the selection employed. While this observation is correct, allowing leaves to be subject of mutation changes the transition probabilities in the random process. Surprisingly, this is sufficient to reduce the expected length of this final phase to $O(n^3)$.

THEOREM 2. *The expected number of steps needed to reach a stable topology for a network with $|V| = n$ agents and di-*

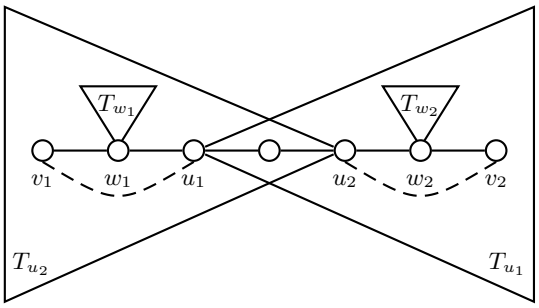

Figure 3: Mutations decreasing F by 1.

ameter 3 using mutations, eccentricity as fitness function and $\leq$ for comparison is $O(n^3)$.

PROOF. Again, we consider a network as shown in Figure 2. We see that the probability to decrease k by 1 equals $1/n$, because with probability $(k/n) \cdot (1/k)$ one of the k nodes is chosen, which then chooses the right inner node for a mutation. By the same line of argumentation the probability to increase k by 1 equals $1/n$ and, thus, the probability not to change the topology equals $(n-2)/n$. We can model the situation as a random walk on $0, 1, \ldots, n-2$ with 0 and $n-2$ as the only absorbing states. We neglect steps not changing the current state and compensate for this by taking an additional factor of $\Theta(n)$ into account. Since the probability for changing the current state is $\Theta(1/n)$, this does not affect the order of growth of the expected number of steps. Then we consider a fair random walk with transition probabilities 1/2, 1/2 for any non-absorbing state. It is well known that the expected number of steps equals $i(n-i)$ when started in state i [18]. Thus, we obtain $O(n \cdot \max\{i(n-i) \mid 0 < i < n\}) = O(n^3)$ as upper bound as claimed. □

4. RESULTS FOR A MORE COMPLEX FITNESS FUNCTION

Eccentricity as fitness function leads to a model where an exponential number of steps is needed to come to a stable state if inner mutations are employed. Thus it comes as no surprise that Lehmann and Kaufmann [12] also consider another fitness function. They prove that using closeness centrality $c(G_t, v)$ instead of the eccentricity $e(G_t, v)$ for all agents $v \in V$ reduces the time needed to reach a stable state on average from exponential to some polynomial. They prove that the expected number of steps needed to reach a stable topology for a network with $|V| = n$ agents using inner mutations, closeness centrality as fitness function and $<$ for comparison is $O(n^5)$. They do not prove a lower bound, though. One way of finding out whether this upper bound draws a realistic picture is trying to come up with lower bounds. Before we present a rather simple lower bound of order $\Omega(n \log n)$, we need to show that the unique stable topology is again the star, and we need to define a distribution for choosing an initial network. In evolutionary algorithms, using the uniform distribution is most common. It is well known that spanning trees can easily be selected uniformly at random using Prüfer numbers [4, 15]. For random spanning trees, the expected diameter is $\Theta(n^{1/2})$ and the variance is known [1, 16]. Thus it is easy to bound the deviation from the expected diameter with high probability by a small additive constant by using of Chebyshev's inequality (compare [13]).

We measure the progress of the network's evolution by the overall fitness $F\colon \mathbb{G} \to \mathbb{R}$ with $F(G_t) = \sum_{v \in V} c(G_t, v)$. Clearly, F can only decrease during a run. Again the unique network with diameter 2, the star, is the optimal and only stable network. For the star the F-value is

$$\underbrace{n-1}_{\text{center node}} + \underbrace{(n-1)}_{\#\text{other nodes}} \cdot \Bigg(\underbrace{1}_{\text{to center node}} + \underbrace{(n-2)}_{\#\text{other nodes}} \cdot \underbrace{2}_{\text{to other nodes}} \Bigg) = 2n^2 - 4n + 2 = \Theta(n^2).$$

As long as the diameter of the network is at least 3, there is always a mutation decreasing F by 1. This can be seen as follows. Consider some G_t with $D(G_t) \geq 3$ and two agents $v_1, v_2 \in V$ with $d_t(v_1, v_2) = D(G_t)$. Clearly, v_1 and v_2 are leaves. We are in a situation as in Figure 3 where $u_1, u_2 \in V$ are the two nodes subject to mutation and $w_1, w_2 \in V$ the two "middle nodes" involved (compare Figure 1). We claim that either selecting v_1 as active agent and u_1 as node with distance 2 or v_2 as active agent and u_2 as node with distance 2 leads to a mutation causing a decrease of the F-value. If v_1 and u_1 are selected, $\{v_1, w_1\}$ is replaced by $\{v_1, u_1\}$. The distances to all nodes in the sub-tree T_{w_1} increases by 1, the distances to all nodes in the sub-tree T_{u_1} decreases by 1. All other distances remain unchanged. Let $|T|$ denote the number of nodes in sub-tree T. If $|T_{u_1}| > |T_{w_1}|$ holds, the F-value is decreased. Otherwise we assume that v_2 and u_2 are selected. Due to symmetry, our observations now hold with respect to u_2, v_2, w_2 and their sub-trees. We have more nodes in T_{u_1} than in T_{w_1} since this is the case considered. Observe that v_1 and v_2 are connected via a path $v_1, w_1, u_1, \ldots, u_2, w_2, v_2$. Thus, T_{w_1} is included in T_{u_2} and T_{w_2} is included in T_{u_1}. Using this to relate the sizes of the sub-trees we see that the F-value is decreased by at least 1. Knowing the unique stable topology and the diameter after initialization we can now prove a lower bound.

THEOREM 3. *The expected number of steps needed to reach a stable topology for a network with $|V| = n$ agents chosen uniformly at random using either mutations or inner mutations, closeness centrality as fitness function and either $\leq$ or $<$ for comparison is $\Omega(n \log n)$.*

PROOF. We consider the current diameter $D(G_t)$. Clearly, there is at least one pair of nodes $u, v \in V$ with $d_t(u, v) = D(G_t)$. The diameter can only decrease if a node on such a longest path between such nodes u and v in G_t is selected. But there may be more than just one longest path and more such pairs of nodes. In this case, the diameter can only decrease if all longest paths are decreased in length by a mutation. This can only occur if nodes on edges common to all longest paths are selected. We see that in any case the probability to decrease the diameter by 1 is bounded above by $D(G_t)/n$. The expected waiting time for such a step is therefore bounded below by $n/D(G_t)$. Since the diameter can only decrease by at most 1 in a single mutation, we obtain

$$\sum_{i=3}^{D_0} \frac{n}{i} = \Omega(n \log D_0)$$

as lower bound on the expected number of steps if D_0 denotes the initial diameter. As we know, $\mathrm{E}(D_0) = \Theta(n^{1/2})$ holds and we have $\mathrm{Prob}(D_0 \geq cn^{1/2}) \geq 1 - \varepsilon$ for some constant $c > 0$ and some constant ε (depending on c) with $0 < \varepsilon < 1$. Thus, we have $\Omega(n \log n^{1/2}) = \Omega(n \log n)$ as lower bound as claimed. □

Clearly, there is a huge gap between this simple lower bound $\Omega(n \log n)$ and the upper bound $O(n^5)$ due to Lehmann and Kaufmann [12]. It is, however, not difficult to see that Lehmann's and Kaufmann's bound of $O(n^5)$ is far from being tight. In fact, it can easily be improved to $O(n^{7/2})$.

We call a successful mutation of the type needed to show how to come to a stable topology *essential* when the neighbor w_1 of leaf v_1 to be activated would itself become a leaf after removing all of its leaf-nodes. An essential mutation has probability $\Omega(1/n)$ which is established by considering the two events in the mutation step. Choosing v_1 as active has probability $(\deg(w_1)-1)/n$ and selecting node u_1 for the edge exchange has probability $1/(\deg(w_1) - 1)$. Therefore, we get an upper bound on the expected number of steps of $O(n \cdot (F_0 - 2n^2 + 4n - 2))$ if F_0 is the initial F-value. Clearly, $F(G_t) \leq n^2 \cdot D(G_t)$ holds. Since we have $D(G_t) \leq n$, an upper bound of $O(n^4)$ follows. This holds independent of the initial distribution of the network. As discussed above, using the uniform distribution we are confronted with random spanning trees having expected diameter $\Theta(n^{1/2})$ and bounded variance [1, 16]. This leads to an upper bound of $O(n^{7/2})$. By proving that the F-value can be decreased by far more than just 1 in a single mutation this bound can be improved even further.

THEOREM 4. *The expected number of steps needed to reach a stable topology for a network with $|V| = n$ agents using mutations, closeness centrality as fitness function and $<$ for comparison is $O(n^{5/2})$.*

PROOF. For the proof we need the notion of a centroid. For a spanning tree on $|V| = n$ nodes let for $v \in V$ the $\deg(v)$ different sub-trees that are obtained by removing all but one edge from v be denoted $T_i(v)$ with $i \in \{1, 2, \ldots, \deg(v)\}$. For $v \in V$ we define its weight $w(v) := \max\{|T_i(v)| \mid i \in \{1, 2, \ldots, \deg(v)\}$. A node $v \in V$ with minimal weight is called centroid. Already Jordan [11] proved that each tree has one or two unique centroids. We make use of the following description of centroids [8]. Either there is some $v \in V$ with $w(v) \leq (n-1)/2$ and this is the unique centroid or there are two adjacent nodes $v_1, v_2 \in V$ with $w(v_1) = w(v_2) = n/2$ and these are the only centroids. After the initialization of the graph it is a priori not known which node is the centroid but it is obvious that the center node of the star is the unique centroid. This characterization allows us to improve the analysis of the F-value-decrease for an essential, successful mutation to $\Omega(n)$. The active leaf decreases its distance to the centroid and thus the fitness of at least $(n-1)/2$ nodes behind the centroid is decreased by at least 1. The F-value of any spanning tree is at most $O(n^3)$ (the path) which would be reduced by an essential mutation by at least $\Omega(n)$ in expected waiting time $O(n)$.
This result would already reduce the expected number of steps to reach a stable topology to $O(n^3)$ without any further considerations of a probability distribution of spanning trees. But using the uniform distribution we know that the diameter of a uniform random labeled spanning tree is $\Theta(n^{1/2})$ with high probability, leading to an initial F-value of $O(n^{5/2})$ with high probability. By the proven F-value decrease $O(n^{3/2})$ essential mutations are enough to produce a stable topology, leading to the expected runtime of $(1 - 1/n) \cdot O(n^{5/2}) + 1/n \cdot O(n^3) = O(n^{5/2})$. □

We believe that this upper bound can still be improved, if we make use of even more structural properties of uniform random labeled spanning trees. We present the ideas here, as we can only show this improved upper bound of $O(n^2)$ for a family of spanning trees. Since we make use of properties of spanning trees chosen uniformly at random, we present the precise algorithm used to define a bijection between spanning trees on n nodes and strings of length $n-2$ of $\{1, 2, \ldots, n\}$. Note that this algorithm (due to Agnarsson, Deo, and Micikevicis [1]) defines a bijection different from the original Prüfer numbers [15].

Given a spanning tree $T = (V, E)$ with $V = \{1, 2, \ldots, n\}$, the Prüfer number is computed in the following way starting with an empty string P using a queue Q as data structure.

1. Insert the leaves of T in Q in ascending order.
2. While $|V| > 2$ do
3. Remove v from the queue Q and from V.
4. Append the number of the neighbor u of v to the current string P.
5. If u is a leaf now, append u to the queue Q.
6. Output P.

We observe that it is easy to derive the degree of a node in the spanning tree from the corresponding Prüfer number. Clearly, its degree is exactly by 1 greater than the number of times it appears in this Prüfer number. It is easy to see that the degree of a node is binomially distributed such that $\mathrm{Prob}(\deg(v) = d) = \binom{n-2}{d-1} n^{-d+1}(1 - 1/n)^{n-2-d+1}$ holds. This allows to derive upper and lower bounds on the number of nodes of a given degree where upper and lower bounds fall together for nodes of degree $d = o(n^{1/2})$. As the maximal degree of a uniform spanning tree is $\Theta(\log(n)/\log\log n)$ (compare [13]) with high probability we can use $n/(e \cdot (d-1)!)$ as the expected number of nodes of degree d, i.e. the expected number of nodes of degree two is $\Theta(n)$. The scenario is also known as the Balls-and-Bins model which is defined by a number of balls that are thrown to a number of bins with the location of each ball chosen independently and at random (compare [13]). We profit from a rich set of methods which deal with the dependencies of the associated random variables that naturally arise. Dubashi and Ranjan [5] for example show that Chernoff bounds can be applied when we are interested in the deviations from the expected number of bins with a certain load. By actually applying Chernoff bounds we bound the deviation of the expected value of the random variable X_d, which counts the number of nodes of degree $d = O(\log n/\log\log n)$, with overwhelming probability by a multiplicative constant c, that is $\Pr(X_d \leq c \cdot E[X_d]) \leq (1 - e^{-\Omega(n^\epsilon)})$ and $\Pr(X_d \geq c \cdot E[X_d]) \leq (1 - e^{-\Omega(n^\epsilon)})$, i.e. there are $\Theta(n)$ nodes of degree two with overwhelming probability.

The family we have in mind follows from the fact that Entringer et al. [6] prove a value of $\Theta(n^{5/2})$ for the expected Wiener Index of uniform random labeled spanning trees, that is the average distance between two nodes in an average random tree is $\Theta(n^{1/2})$. The family we consider consists of $\Omega(n^{1/2})$ many disjoint sub-trees of depth $\Theta(n^{1/2})$

which is the only possibility to actually construct a labeled spanning tree with Wiener Index $\Theta(n^{5/2})$. Such a spanning tree can be constructed by joining all the nodes of degree two to $\Omega(n^{1/2})$ many paths of length $\Theta(n^{1/2})$ and using all the other nodes of degree $O(\log n/\log\log n)$ (which is the maximal degree with high probability) to join those paths together to form a spanning tree.
As an essential mutation can occur in each of the disjoint sub-trees the probability for an essential mutation increases to $\Omega(1/n^{1/2})$ thus reducing the expected runtime for a spanning tree from this family to $O(n^2)$.

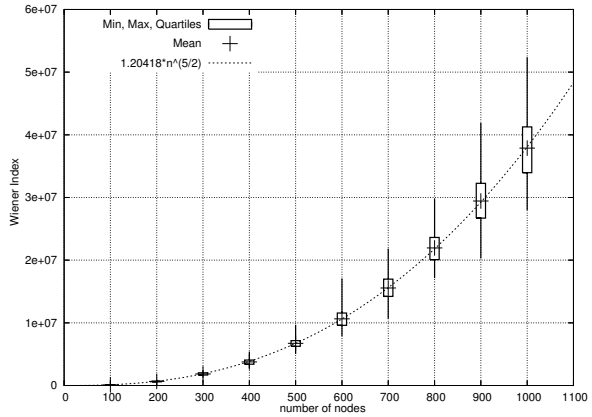

Figure 4: Mean Wiener Index of spanning trees generated uniformly at random.

We believe that it is possible to generalize the result to improve the expected runtime. Janson [10] proves a value of $\Theta(n^5)$ for the variance of the expected Wiener Index of uniform random labeled trees. This implies that it is not possible to use standard tools like Chebyshev's inequality to bound the deviance from the expected Wiener Index. However, we have already shown by means of the known first and second moment of the diameter of a uniform random labeled tree that the Wiener Index is $O(n^{5/2})$ with high probability. This as well as experimental data supports our believe that it should be possible to find a suitable lower bound in order to show that the Wiener Index of a uniform random labeled tree is $\Theta(n^{5/2})$ with high probability. The experimental data was obtained for $n \in \{100, 200, \ldots, 1000\}$ nodes in 100 independent runs for each value of n. In each run, a spanning tree of n nodes was generated uniformly at random and the Wiener Index computed. We plot the mean values, the first and third quantiles as well as the minimum and maximum in Figure 4. We fit $c \cdot n^{5/2}$ to this data (using gnuplot's fit, obtaining an error of 1.07595e+11 with the standard deviation of the fit of 109339 and $c = 1.20418$).

Clearly, our upper bound of $O(n^{5/2})$ is much closer to the truth than the $O(n^5)$ upper bound due to Lehmann and Kaufmann [12]. There is, however, still a gap of order $\Theta(n^{3/2}/\log n)$ between this upper bound and the lower bound of order $\Omega(n \log n)$. In order to build up an intuition about the expected number of steps we perform experiments for not too large values of n. While the results of such experiments cannot deliver any asymptotic results, they enable us to judge the quality of our bounds for rather small values of n. We present the results of 100 independent runs where for each run a new spanning tree is generated uniformly at random. We consider different values of n (the number of vertices), namely $n \in \{100, 200, \ldots, 1000\}$. We plot the mean value, the first and third quantile as well as the minimum and maximum of the 100 runs for each value of n. To get an impression of the quality of our lower bound $\Omega(n \log n)$ we plot $cn \log n$ where the constant $c \in \mathbb{R}^+$ is obtained by a simple regression analysis (using gnuplot's fit). The error obtained in this regression analysis is 9.3258e+09 with a standard deviation of 32190.1. Moreover, we fit the function cn^2 to this data. Here, gnuplot's fit obtained an error of 2.47344e+08 with a standard deviation of 5242.39 when using 0.323626 as multiplicative constant. The results can be seen in Figure 5.

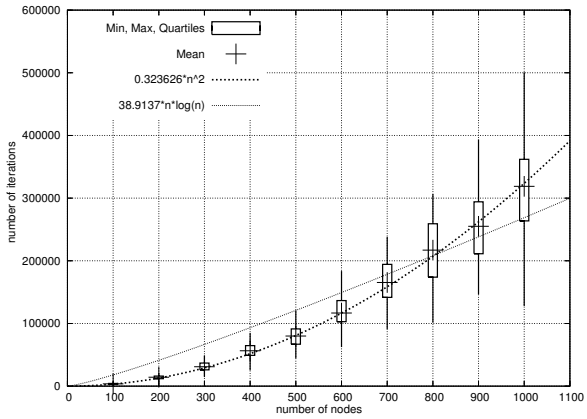

Figure 5: Experimental data using closeness centrality as fitness.

5. A LOCAL FITNESS FUNCTION

Closeness centrality is clearly a much better fitness function than eccentricity with respect to the time needed on average to find an optimal topology. But it has the disadvantage of being less well motivated. While eccentricity measures the centrality of an agent's position quite accurately and is in good agreement with our understanding of selfish agents, closeness centrality is a much more global measure. But both fitness functions are not at all in accordance with our idea of agents acting locally leading to a globally optimal topology as emergent phenomenon. Eccentricity and closeness centrality both require that each agents knows the distances to all agents in the complete network exactly. This is not very different from knowing the complete network. Such global knowledge is usually not available in practical settings. It may therefore be doubted that the results obtained so far are of much use for a practitioner. Therefore, we propose a different fitness function that can be computed by each agent using only local knowledge. We consider the design and analysis of this fitness function to be our main contribution and the most important improvement over the work due to Lehmann and Kaufmann [12].

Clearly, the most direct and most local measure of the connectedness of a single agent $v \in V$ is its current degree $\deg_t(v)$. Clearly, maximization of the current degree cannot lead to a stable state since in a spanning tree at most one node may be connected to all other nodes. Considering the unique globally optimal topology, the star, we see that the connectedness of each node is very good even though all nodes that are different from the center node have only degree 1, since they are connected to the center node, only. We still consider them to be well connected

since the distance to any other node is just 2. This motivates the definition of a fitness function that counts for some agent $v \in V$ the number of nodes in distance at most 2. If $N_t(v) := \{u \in V \mid \{u, v\} \in E_t\}$ denotes the set of direct neighbors of v, we can express this fitness $f\colon \mathbb{G} \times V \to \mathbb{R}$ as

$$f(G_t, v) = |\{u \in V \mid d_t(u, v) \in \{1, 2\}\}| \\ = \deg(v) + \sum_{u \in N_t(v)} (\deg(u) - 1) = \sum_{u \in N_t(v)} \deg(u).$$

Clearly, each agent $v \in V$ aims at maximizing $f(\cdot, v)$. While in the extreme case of the star this fitness function still implies that each agent $v \in V$ knows about the distances to all other agents in the network, it is in general a fitness function that requires only quite restricted local knowledge to be computed by any agent.

We consider the algorithmic frame (Definition 4) and this local fitness function together with our mutation operator (Definition 7) and $>$ for comparisons in the selection step (line 4). It is easy to see that again the star is the unique stable and globally optimal topology. The first question to be answered is whether this topology can be reached using an arbitrary network as starting point at all. We give an affirmative answer by providing an upper bound on the expected number of steps this takes.

THEOREM 5. *The expected number of steps needed to reach a stable topology for a network with $|V| = n$ agents chosen uniformly at random using mutations, f as fitness function and $>$ for comparison is $O(n^3)$.*

PROOF. Clearly, the process has reached a stable topology when an agent $v^* \in V$ with degree $\deg_t(v^*) = n - 1$ exists. For any mutation using some node $v \in V$ as active node and some node $u \in V$ with $d_t(v, u) = 2$ as "partner" we have a trivial lower bound of $\Omega(1/n^2)$ for the probability that it occurs: Since there are only n nodes, any node is selected with probability $1/n$ as active. Furthermore, each node has at most $n-2$ other nodes with distance exactly 2, so the probability to select u is bounded below by $1/(n-2)$ leading to $1/(n(n-2)) = \Omega(1/n^2)$ as lower bound as claimed. Since we have $\deg_t(v) \geq 1$ for any node $v \in V$ at any time step $t \in \mathbb{N}_0$ (since the network is connected), it suffices to prove that (1) the maximal degree in the network cannot decrease over time and (2) at any time step there exists at least one node with maximal degree such that a mutation can increase its degree by 1.

Clearly, the degree of a node $z \in V$ can only change if it participates in a mutation as node u, v, w as seen in Figure 1. We see that the degree of v does not change, the degree of u is increased and the degree of w is decreased by this mutation. Thus, for (1) we only have to consider the case $z = w$. Let $G^* := (V, E^*)$ denote the graph that is created by mutation prior to selection. Whether the mutation leads to a change in the network is determined in the selection step by a comparison of $f(G_t, v)$ and $f(G^*, v)$. Remember that $f(G_t, v) = \sum_{u \in N_t(v)} \deg(u)$ holds if $N_t(v)$ denotes the set of agents with direct connections to v in G_t. We see that we have $f(G^*, v) = f(G_t, v) - (\deg_t(w) - 2) + (\deg_t(u) - 1) = f(G_t, v) - \deg_t(w) + \deg_t(u) + 1$ by taking into account that v loses the neighbors of w except for u and v itself and gains the neighbors of u except for w. Thus, the mutation changes the network iff $\deg_t(u) \geq \deg_t(w)$ holds. As a consequence the degree of a node z can only decrease over time if it either does not have maximal degree or another node with maximal degree exists. In the latter case, the degree of this node is increased so that the maximal degree in the network is increased. So, the maximal degree in the network cannot decrease over time as claimed.

For the proof of (2), we reconsider the three nodes u, v, w involved in a mutation as seen in Figure 1. As we already discussed the only node with increased degree is u. If the mutation actually changes the network, however, depends on the function value of f. As we discussed for (1), the topology is changed iff $\deg_t(u) \geq \deg_t(w)$ holds. Thus, selecting a node v as active node such that a node u with maximal degree can be selected as node with distance 2 leads to a mutation that increases the maximal degree. This completes the proof (2) and thus of the upper bound $O(n^3)$. □

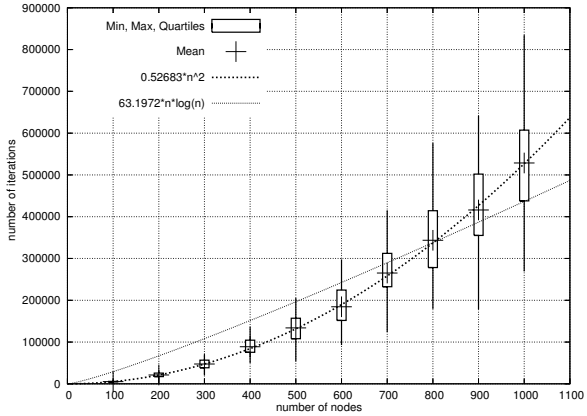

Figure 6: Experimental data using the number of neighbors with distance ≤ 2 as fitness.

We remark that the lower bound from Theorem 3 carries over to this fitness function. Therefore, we have a gap of order $\Theta(n^2/\log n)$ between upper and lower bound. Again, we conjecture that the lower bound can be improved. Nevertheless, we do not believe the upper bound $O(n^3)$ to be tight. A more careful analysis may lead to an improved upper bound that may even reveal that our local fitness function is with respect to the average performance not inferior to closeness centrality as fitness functions in spite of its limited use of information on the current network's topology. To support this point of view we present the results of some experiments (see Figure 6) consisting of 100 independent runs with a randomly chosen spanning tree for each run. We consider the same values of the number of nodes n as above and, again, plot the mean value, the first and third quantile as well as the minimum and maximum. Moreover, we fit the same curves to this data, namely $c_1 n \log n$ for a lower and $c_2 n^2$ for an upper bound. The errors obtained by gnuplot's fit are 2.75368e+10 with a standard deviation of 55314.1 and constant $c_1 = 63.1972$ and 2.5771e+08 with a standard deviation of 5351.11 and constant $c_2 = 0.52683$, respectively. The results can be seen in Figure 6.

6. CONCLUSIONS AND OUTLOOK

The development of networks of autonomous, selfish agents is a process that takes place in different areas. It can be modeled in quite different ways, here the modeling as an self-evolutionary process has been discussed. This perspectives allows for the formal analysis of emergent behavior like

the finding of optimal stable topologies and, moreover, for the analysis of the average time needed to find such a topology. Building on results due to Lehmann and Kaufmann [12] we improved lower and upper bounds on the expected time needed to find an optimal stable topology. For eccentricity as fitness function, we proposed a slightly modified mutation operator that reduces the expected time needed to decrease the diameter from 3 to 2 from exponential to a polynomial of small degree. For closeness centrality as fitness function, we could improve the upper bound from $O(n^5)$ to $O(n^{5/2})$ and present a lower bound of order $\Omega(n \log n)$. The main improvement presented is the design and analysis of a different fitness function that fits with the original ideas of the analysis of such networks of autonomous agents. It allows for a purely local computation of the fitness value by each of the agents and does not require any global knowledge. For this fitness function an upper bound of $O(n^3)$ on the expected time needed to find the unique globally optimal topology was shown.

Clearly, many questions are still open. For eccentricity as fitness function and our mutation operator an upper bound on the expected time to reach a stable topology is needed. Our upper bound takes into account the final phase (reducing the diameter from 3 to 2), only. It is an interesting question whether our mutation operator leads to an overall polynomial expected run time even for arbitrary initial distributions. For closeness centrality, it would be nice to close the $\Theta(n^{3/2}/\log n)$ gap between the upper and lower bound. We conjecture that the upper bound for a family of spanning trees can be generalized and the lower bound can be improved significantly to $\Omega(n^2)$. This would also help with respect to our local fitness function. The lower bound $\Omega(n \log n)$ carries over and we speculate that improved lower bounds would carry over, as well. Moreover, we speculate that the upper bound $O(n^3)$ is not tight and may be further reduced, perhaps even to $O(n^2)$. The results of experiments reported make both conjectures seem plausible.

Reconsidering the complete model, the algorithmic frame is quite restrictive. It is assumed that at each time step exactly one agent is selected to become active. Clearly, again, this requires some kind of global information exchange or global control. It would be much nicer to have the agents decide (randomly) themselves when to become active. It is not difficult to see, however, that the mutation operators considered here are no longer able to guarantee that the network remains connected if more than one agent may become active in each time step. Consider for example two mutations where two nodes which have a distance of two to each other become active and choose the respectively other node for an edge exchange. Then the edges to the middle node will be deleted, leaving a forest of trees. The interesting and non-trivial task which arises here is to find ways of guaranteeing connectedness on a local level. Connectedness of a graph seems to be a global property of the network. This can be made plausible by comparing the difficulty of deciding connectedness for single perceptrons with bounded diameter [17]. Design and analysis of appropriate models and appropriate mutation operators are subject of future research.

7. ACKNOWLEDGMENTS

The first author was supported by the DFG (Deutsche Forschungsgemeinschaft) as part of the collaborative research center "computational intelligence" (SFB 531).

8. REFERENCES

[1] G. Agnarsson, N. Deo, and P. Micikevicis. On the expected number of level-i-nodes in a random labeled tree. *Bulletin of the ICA*, 41:51–60, 2004.

[2] E. Anshelevich, A. Dasgupta, E. Tardos, and T. Wexler. Near-optimal network design with selfish agents. In *Proceedings of the 35th Annual ACM Symposium on Theory of Computing (STOC '03)*, pages 511–520, 2003.

[3] V. Bala and S. Goyal. A noncooperative model of network formation. *Econometrica*, 68:1181–1230, 2000.

[4] N. Deo and P. Micikevicis. A new encoding for labeled trees employing a stack and a queue. *Bulletin of the ICA*, 34:77–85, 2002.

[5] D. Dubashi and D. Ranjan. Balls and bins: a study in negative dependence. *Random Structures and Algorithms*, 12(2):99–124, 1998.

[6] R. C. Entringer, A. Meir, J. W. Moon, and L. Székely. On the wiender index of trees from certain families. *Australasian Journal of Combinatorics*, 10:211–224, 1994.

[7] A. Fabrikant, A. Luthra, E. Maneva, and C. Papadimitriou. On a network creation game. In *Proceedings of the 22nd Annual Symposium on Principles of Distributed Computing (PODC '03)*, pages 347–351, 2003.

[8] M. Fischermann, A. Hoffmann, D. Rautenbach, L. Szekely, and L. Volkmann. Wiener index versus maximum degree in trees. *Discrete Applied Mathematics*, 122:127–137, 2002.

[9] J. Garnier, L. Kallel, and M. Schoenauer. Rigorous hitting times for binary mutations. *Evolutionary Computation*, 7(2):173–203, 1999.

[10] S. Janson. The wiener index of simply generated random trees. *Random Structures and Algorithms*, 22(4):337–358, 2003.

[11] M. E. C. Jordan. Sur les assemblages des lignes. *Journal für die Reine und Angewandte Mathematik*, 70:185–190, 1869.

[12] K. Lehmann and M. Kaufmann. Evolutionary algorithms for the self-organized evolution of networks. In *Proceedings of the Conference on Genetic and Evolutionary Computation (GECCO '05)*, pages 563–570, 2005.

[13] M. Mitzenmacher and E. Upfal. *Probability and Computing*. Cambridge University Press, 2005.

[14] N. Nisam. Algorithms for selfish agents. In *Proceedings of the 16th Annual Symposium on Theoretical Aspects of Computer Science (STACS '99)*, pages 1–15, 1999.

[15] H. Prüfer. Neuer Beweis eines Satzes über Permutationen. *Archiv für Mathematik und Physik*, pages 142–144, 1918.

[16] A. Renyi and G. Szekeres. On the height of trees. *Journal of the Australian Mathematical Society*, 7:497–507, 1967.

[17] R. Rojas. *Neural Networks*. Springer, 1996.

[18] D. Williams. *Probability with Martingales*. Cambridge Mathematical Textbooks, 1991.