How to Measure Network Creation Games’ Naturality?

Hedyeh Beyhaghi
Sharif University of Technology
Tehran, Iran
beyhaghi@ce.sharif.edu

Zahra Fahmi
Sharif University of Technology
Tehran, Iran
fahmi@ce.sharif.edu

MohammadAmin Fazli
Sharif University of Technology
Tehran, Iran
fazli@ce.sharif.edu

Jafar Habibi
Sharif University of Technology
Tehran, Iran
jhabibi@sharif.edu

Pooya Jalaly
Sharif University of Technology
Tehran, Iran
jalay@sharif.edu

MohammadAli Safari
Sharif University of Technology
Tehran, Iran
msafari@sharif.edu

ABSTRACT
Modelling is one of major research areas in social network analysis whose goal is to study networks structure and its evolution. Motivated by the intuition that members in social networks behave selfishly, network creation games have been introduced for modelling social networks.

In a typical network creation game, every node is allowed to only change its local structure so as to maximize his own benefit (utility), where utility functions depend on the graph structure and are, in fact, mappings from graph structures to real numbers. If this game converges to an equilibrium graph \( G \) (which we call the game graph), we use this graph as a model for social networks.

In this paper, our aim is to measure how much the output graphs of a given network creation game \( G \) are compatible with a social network \( N \). We first show that the precise measurement is not possible in polynomial time. That is, to decide whether \( N \) is a possible output graph of \( G \) is NP-Complete. Then we propose a method for its approximation; finally, we show the usability of our method by conducting experiments on real network data.

Categories and Subject Descriptors
H.4 [Information Systems Applications]: Miscellaneous;
D.2.8 [Software Engineering]: Metrics—complexity measures, performance measures

General Terms
Measurement

Keywords
Social Network Modeling, Network Creation Games, Measurement, Naturality

1. INTRODUCTION
Modelling of social networks is an important research direction in the study of social networks. Its main goal is to study the structure of networks and their evolution in an attempt to generate graphs (models) with similar structural properties to the real networks (such as logarithmic diameter or power-law degree distribution). There are several benefits enumerated for this research[24]:

○ They give insight of the structure of the networks.
○ They can be used to evaluate the effect of various structural properties of networks on the performance of algorithms.
○ They help us to predict the evolution of networks in future.
○ Could be used to generate smaller graphs for the purpose of simulation experiments.
○ They can be used to compare the similarity of the structures of different networks.
○ They allow succinct representation by keeping only the model parameters.

From our point of view, there are two general approaches for modelling social networks: centralized and decentralized. In the centralized approach, a centralized manager does every thing from adding nodes to adding/removing edges and builds a graph that is aimed to satisfy certain structural properties[1, 4, 6, 15, 17, 20, 21, 23, 24, 27, 33, 34]. The most important problem with this approach is that it completely ignores the nature of various situations in which players/nodes have choice and decide how to get involved in the network. Social settings, for example, involve sentient actors who have discretion in which relationships they form and maintain and how much effort, time, or resources they want to devote to different relationships. Examples of this include trading relationships, political alliances, employer-employee relationships, marriages, professional collaborations, citations, emails, friendships, and so forth.
In the decentralized approach, however, the focus is on how network structure evolves and how it is affected by individuals’ decisions when network nodes are selfish agents. The evolution is modeled by a repeated network game which is called Network Creation (Formation) Game. There are several network creation games which take different networks’ parameters into account\cite{2, 3, 5, 7, 10, 12, 13, 14, 16, 18, 19, 30, 32}. Lets first define these kind of games formally.

**Network Creation Game.** A network creation game is a repeated game defined by a tuple $< n, G, S, U, D, E >$ where

- $n$ is the number of players which are the vertices of graph $G$.
- $G$ is the initial game graph. In this paper, we show vertices of $G$ by $\{v_1, v_2, ..., v_n\}$. This graph will evolve over time by the strategies chosen by players.
- $S = \{S_1, S_2, ..., S_n\}$ where $S_i$ is the set of possible strategies for vertex $v_i$.
- $U = \{u_{v_i} : S_i \rightarrow \mathbb{R}|1 \leq i \leq n\}$ is the set of vertices’ utility functions. $u_{v_i}(s)$ is the utility of $v_i$ when its chosen strategy is $s$. In a best-response game, each vertex $v_i$, in its turn, chooses one a strategy which maximize its utility and change the structure of the graph according to this strategy. These set of strategies is called best response of vertices $v_i$ or $BR_i$.
- $D$ is the dynamic of the game. Dynamic of the game defines the mechanism for choosing the players who can play in each turn. Among different dynamics an important one is Sequential dynamic in which $v_i$ is permitted to play in steps $kn + i$ and in other steps it must be silent. In this paper we focus only on this dynamic.
- $E$ is a set of conditions for game termination which are defined by some functions. The most important and well-studied termination condition is the concept of equilibria and the most famous type of equilibria is Nash-equilibria\cite{31}. Nash equilibrium is the point where each vertex is already playing its best response. In this paper our focus is only on Nash-equilibria.

If all the conditions of $E$ are satisfied, the game finishes and the evolved graph $G$ works as the model.

**An Example.** Consider the basic game defined in\cite{16}. The dynamic of the game is sequential. The strategy of each player is to purchase a set of edges to other vertices so as to minimize their total distance to all vertices. So, $S_i$ is $2^{V(G) – \{v_i\}}$ which is equal to all subsets of vertices (except $v_i$) which $v_i$ can draw edges to their members. Each edge has a constant price $\alpha$ and vertices may not afford connecting themselves to as many as they want. The utility of choosing the strategy $s \subseteq V \setminus \{v_i\}$ is defined as:

$$u_{v_i}(s) = -\sum_{j \neq i} dist_G(v_i, v_j) - \alpha . |s|$$

where $G$ is the evolved game graph after adding edges $\{v_i, v_j | v_j \in s\}$ and $dist_G(v_i, v_j)$ is the distance between $v_i$ and $v_j$ in $G$.

Game terminates if it has converged to a Nash-equilibrium, where no one wants to change its strategy.

Network Creation Games presented up to now, fail to satisfy real networks’ properties desirably. One important structural property of a social network is its $O(log n)$ diameter, which is called small-world property. For the game mentioned by the above example\cite{16}, it is proved that it has a diameter of $2$ for tree equilibria\cite{2}, an upperbound of $2^O(\sqrt{n})$ for diameter of general equilibria\cite{2} and a constant diameter for $\alpha > 273n$\cite{29}. Later on, Demiane et al.\cite{11} introduced a new version of this game in which nodes try to minimize their maximum distance to other nodes. The equilibrium graphs of this game have constant diameter for $\alpha > 129$. Alon et al.\cite{2} introduced two basic network creation games. For the diameters of equilibria in the sum version they proved an upper bound of $2^O(\sqrt{n})$, a lower bound of $3$, and a tight bound of exactly $2$ for trees. For the max version, they found a lower bound of $\Omega(\sqrt{n})$, and a tight upper bound of $3$ for trees. Ehsani et al.\cite{14} proved that the diameters of equilibria in the sum version of bounded budget games are $O(lg n)$ for trees and $2^O(\sqrt{n})$ for general graphs. They also proved diameter of $\Omega(\sqrt{lg n})$ in the max version. Kleinberg et al.\cite{22} studied another network creation game with structural holes. They showed that all Nash equilibria are limited to dense graphs with $\Omega(n^2)$ edges where $n$ is the number of vertices\cite{5}; this is far from basic properties of social networks as social networks have power-law degree distribution and, therefore, have $O(n)$ edges. In addition, most of network formation games yield only symmetric equilibrium networks\cite{7}. In reality, however, asymmetry is noticed in social networks.

The game-theoretic models to date have primarily technological, rather than sociological and are motivated by technical observations such as efficient routing concerns in communication networks\cite{7}. On the other hand, Nash equilibria in most network creation games, are literally f-fetched. As Alon et al.\cite{2} noticed, network agents are computationally bounded. For games presented in \cite{7, 11, 14, 16} computing even best response is NP-hard, so agents cannot even tell whether they are playing best response or not. As a result, Alon et al.\cite{2} suggest proposing more reasonable models, in which we assume that each agent can only draw a constant or sublogarithmic number of edges against others.

The main idea of this paper is to introduce a general framework for measuring the naturality of network creation games, i.e., how much their output graphs match structurally to real social networks. Such measurements might be be used for comparing two network creation models. The inputs of this problem are two network creation games and a social network $\mathcal{N}$ and the output is a boolean showing which of them can simulate $\mathcal{N}$ better. A naive approach is to compute the amount of all structural properties (parameters) for the graphs generated from $\mathcal{G}_1$, $\mathcal{G}_2$ and $\mathcal{N}$ and compare the results; for example we can compare the diameter (as a structural property) of the generated graphs of $\mathcal{G}_1$ and $\mathcal{G}_2$ with the diameter of $\mathcal{N}$ and choose the one which is closer to $\mathcal{N}$. The problem is that this type of measurement is highly dependent on the chosen property and can not give an overall decision about the models. The power of our framework
is its satiety from computing any structural property.

2. NCG DECIDABILITY

Before proposing a framework for comparing the compatibility of network creation games with a given network \( \mathcal{N} \), we first study a fundamental relevant problem regarding the complexity of such models. In the NCG compatibility problems, a network creation game \( \mathcal{G} \) is given and our concern is to decide whether \( \mathcal{N} \) can be generated by \( \mathcal{G} \) or not. We only focus on our fixed configuration (sequential dynamic + Nash equilibria). More formally speaking,

**Definition 1.** In NCG compatibility problem (NCG-Compat), a network creation game, \( \mathcal{G} \), and two networks, \( N_0 \) and \( N_1 \), are given. Game \( \mathcal{G} \) is started from \( N_0 \). Is there any sequential dynamic of the game which converges to \( N_1 \)?

The main result of this section is theorem 2 which shows that NCG-Compat problem cannot be decided in polynomial-time even if we find a Nash-equilibrium in polynomial time. Before declaration of our main result, we present a theorem about hardness of finding a Nash-equilibrium of a network creation even in situations where computing players’ best response is possible in polynomial-time which has not proved yet in the literature. In this section, we use notation \( N(S), S \subseteq V(G) \), for the set of adjacent vertices to the vertices of \( v \in S \) in the graph.

**Theorem 1.** Finding a Nash-equilibrium of a network creation game, even if computing players’ best response can be accomplished in polynomial-time, is NP-Hard.

**Proof.** We reduce Subset-Sum to our problem. Assume that \( S = \{x_1, x_2, x_3, ..., x_n\} \) and \( T \) is a positive integer. We want to decide whether there is a subset of \( S \) whose sum is \( T \). Consider a weighted graph \( G \) with \( n + 2 \) nodes \( \{v_1, v_2, ..., v_n, a, b\} \). For \( i \in \{1, 2, ..., n\} \), \( v_i \) weights \( x_i \) and \( a \) weight 0 and \( T \), respectively. We choose \( N_0 \) to be an empty graph and \( N_1 \) to be one of the Nash equilibrium graphs. In game \( \mathcal{G} \), utility functions are defined as follows.

Node \( a \)'s utility is always constant no matter how he plays. For \( v_i \) \((1 \leq i \leq n)\) there are different circumstances. Assume \( s \) is the action of \( v_i \), i.e., the set of vertices that \( v_i \) connects to. If \( a \in s \) and the total weights of \( a \) in \( s \) is less than or equal to \( T - x_i \), then \( v_i \)'s utility is \(+\infty\). If \( a \) is empty then \( v_i \)'s utility is \(-\infty\). Otherwise, it is \(-\infty\). So, \( v_i \)'s utility is \(+\infty\) only when \( v_i \in N\{v_i\} \) forms a clique with total weight of less than or equal to \( T \).

The role of node \( b \) is to corrupt cliques with total weight which are strictly less than \( T \). The utility of \( b \) is defined as follows. Assume he chooses \( s \). If \( a \) has no neighbours, \( b \) gains \(+\infty\) when \( s \) is empty and \(-\infty\) otherwise. If \( a \) and its neighbours form a clique with a total weight of \( T \), \( u_b(s) = +\infty \) if \( s = \emptyset \) and \(-\infty\) otherwise. In other situations \( u_b(s) = +\infty \) if \( s = \{a\} \cup N\{a\} \) and \(-\infty\) otherwise. More formally: \( u_b(s) = \begin{cases} +\infty & \text{if } s = \{a\} \cup N\{a\} \\ 0 & \text{if } s = \emptyset \\ -\infty & \text{otherwise} \end{cases} \)

The role of node \( v_i \) in the set of adjacent vertices to \( v_i \) is to decide whether there is a subset of \( S \) with total weight equal to \( T \). This results in utilities equal to \(-\infty\) for \( a \)'s neighbors. Thus this structure is not a Nash equilibrium. If \( a \) and its neighbors form a clique with total weight less than \( T \), \( b \) is not playing best response. If \( a \) and its neighbors form a clique with total weight equal to \( T \), other nodes play best response if and only if their degree is zero. Having neighbors cause them to gain \(-\infty\) utility.

**Theorem 2.** NCG-Compat problem is NP-Complete.

**Proof.** We use reduction from our problem to 3-SAT problem. Assume that inputs of 3-SAT are \( m \in \mathbb{N} \) number of clauses like \( C_i = x_{a_i} \lor x_{b_i} \lor \neg(x_{c_i}) \) in which \( 1 \leq a_i < b_i < c_i \leq n \), and \( n \in \mathbb{N} \) number of boolean variables like \( x_i \) in which \( 1 \leq i \leq n \). Now, we make a graph \( G_0 \) with these variables and clauses. Assume that \( U = \{u_1, u_2, ..., u_n\} \) and \( V = \{v_1, v_2, ..., v_n\} \) are two sets of vertices, and \( V(G_0) = U \cup V \). Assign to each vertex \( u_i \), variable \( x_i \) and to each vertex \( v_j \), clause \( C_j \). Assume that

\[
E(G_0) = \{u_i, v_j | u_i \text{ is one of } C_j \text{'s negative literals}\}.
\]
Define graph $G_{eq}$ such that $V(G_{eq}) = V(G_0)$ and

$$E(G_{eq}) = \{u_i, v_j | 1 \leq i \leq n, 1 \leq j \leq m\}.$$  

Now, we define a sequential network formation game and show that $G_{eq}$ is a Nash equilibrium of this game. Then, by giving graph $G_0$ as initial state of our game, we show that finding the sequence of plays which leads to $G_{eq}$ is equal to finding a solution for our 3-SAT. Assume that for each vertex $v \in V$, utility of $v$ is always equal to 0 and utility of $u \in U$ is defined as follow

- If $d(v) \geq 1$ for each vertex $v \in V$ then
  - If $N(u) = V$ then $u(v) = +\infty$.
  - If $N(u) \neq V$ then $u(v) = -\infty$.

- If there exists at least one vertex $v \in V$ for which $d(v) = 0$ then
  - If for each vertex $v_i \in V$ for which $u$ is one of $C_j$ is a non-negative literal, $v_j \in N(u)$ and for each vertex $v_j \in V$ for which $u$ is one of $C_j$’s negative literals, $v_j \notin N(u)$, then $u(v) = +\infty$.
  - otherwise, $u(v) = -\infty$.

First, we prove that $G_{eq}$ is an equilibrium of this game. In $G_{eq}$, for each vertex $v \in V$, $u(v)$ is constant so it will not change its edges, furthermore, for each vertex $v \in V$, $d(v) = n \geq 1$ and for each vertex $u \in U$, $N(u) = V$ so $u(v) = +\infty$ and it will not change its edges as well. Therefore, $G_{eq}$ is a Nash equilibrium of this game.

Now, we prove that by knowing each playing sequence of the vertices which leads to $G_{eq}$, we can find a solution for 3-SAT. In $G_0$, each vertex $u$ is connected to each vertex $v$ for which $x_i$ is a negative literal of $C_j$. So, we can assume that at the initial state of this game, every boolean variable is false. Furthermore, during this game, if there exists at least one vertex $v \in V$ for which $d(v) = 0$, when each vertex $u$ plays in its turn, it connect itself to each vertex $v_j$ for which $x_i$ is a non-negative literal of $C_j$. Thus we can assume that when these vertices play, they change their boolean variables from false to true. Moreover, we can assume that each edge $u, v_j$ in the game’s graph $G_t$ at time $t > 0$, represents that $x_i$’s literal is true in clause $C_j$. When we reach a state of this game, in which each vertex $v \in V$ has at least one adjacent vertex $u \in U$, it means that each clause of the 3-SAT, has at least one literal which is true. From now on, each vertex $u \in U$ that plays, connects itself to every vertex in $V$. So, if our game reaches a state in which the 3-SAT is solved, in at most $n + m$ rounds, the game’s graph converges to $G_{eq}$ and otherwise it will not. If one finds a solution for the 3-SAT, in which set of variables $X = \{x_1, x_2, \ldots, x_n\}$ are true, it is easy to show that if he chooses vertices from $V_i = \{u_{t_1}, u_{t_2}, \ldots, u_{t_k}\}$ to play first, then the game’s graph will converge to $G_{eq}$ and otherwise it will not. We know that finding this solution is NP-Complete and the proof is complete. ∎

3. FRAMEWORK

Assume that $G = < n, G, U, D, E >$ is a network creation game and $N$ is a network with $n$ nodes. We use notation $comp(G, N)$ for showing the amount of compatibility. And we define it as follows:

$$comp(G, N) = \frac{1}{n} \sum_{v \in V} \frac{current\_utility(v)}{best\_utility(v)}$$

While smartness factor is mentioned in the context of Social Balance and Signed Network Formation Games and is measured for each node, compatibility factor is measured for the whole network in a general context. Compatibility of a game can be interpreted as its naturality; therefore, a high compatibility factor shows that the given network creation game $G$ is suitable and natural for network $N$. If $comp(G_1, N) > comp(G_2, N)$, it can be concluded that $G_1$ is more compatible with $N$ than $G_2$.

4. EXPERIMENTAL RESULTS

4.1 Case Studies

In this section we use our framework to illustrate the compatibility of games with networks. We compare different models in this part. First we use a random graph and then a real social network.

We consider Watts-Strogatz random graph[33]. This kind of random graph was the first to model small-world property (as we mentioned in introduction, small-world property is the logarithmic diameter of the social networks).

**Definition 2.** Watts-Strogatz random graphs are the result of a random rewiring procedure. Starting from a ring lattice with $n$ vertices and $k$ edges per vertex, each edge is rewired at random with probability $p$. Parameters $k$ and $p$ satisfy $0 \leq p \leq 1$ and $N \gg K \gg ln(N) \gg 1$. The procedure constructs an undirected graph with $N$ nodes and average degree $k$ with small-world property.

For simulation, we take a 20-vertex graph with $k = 4$ and $p = 0.5$. The game used for the simulation is Fabrikant et al.’s game[16] which we introduced as an example in Section 1. As computing the best response in this game is NP-hard, we do exhaustive search for finding best responses. For different $\alpha$ values we compute the current utility and the best response utility for each node.

The utility function of vertices in this game is

$$u_v(s) = -\sum_{j \neq 1} dist_G(v_i, v_j) - \alpha \cdot |s|.$$

For $\alpha = 1$, people should edges to all other vertices to play best response. Thus, their current action is far from
Average compatibility factor vs $\alpha$ is illustrated in Figure 1. Observed results confirm our expectations discussed above.

In this case the undirected Epinion network[26] and two different games are considered. In order to find an approximate natural game for this network, we firstly investigate the degree distribution of its nodes. Based on previous studies, the degree distribution of social networks follows power law property[9]; thus, most of the nodes have low degrees while there are few ones with high degrees. Secondly, we choose two games $G_1$ and $G_2$. In the first game, $G_1$, the utility function is an exponential function which is defined as follows.

$$u_{v_i}(s) = \frac{1}{\mu} e^{-|s|/\mu}$$

The utility function of the second game, $G_2$, is a normal function that is defined below.

$$u_{v_i}(s) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(|s|-\mu)^2}{2\sigma^2}}$$

where $\mu$ is the mean of both functions, and $|s|$ is the degree of $v_i$, and $\sigma$ is set to be 1. The normal utility function assigns its highest values to the vertices with degrees around $\mu$ and assigns low values to to vertices with high and low degrees while the exponential utility function assigns low values to high degree vertices and vice versa. So, the utility function of the first game acts more like a power law and the first game seems more natural game than the second one. Experimental results, as depicted in Figure 2, show that our

Figure 1: Compatibility factor of Watts-Strogatz graphs[33] with Fabrikant et al.’s game[16] with different values of $\alpha$. The best fit occurs for the game with $\alpha = 2$.

Figure 2: Comparison between compatibility factor of two NCGs defined by exponential and normal utility functions with Epinion network. As expected, exponential functions are more compatible with social networks.
The dynamic of the game is sequential that is player \( v_i \) is just permitted to play its best response at times \( t = kn + i \) for \( 0 \leq k \).

We consider a scenario to measure smartness factor in real social networks. As discussed in [28] computing best response in signed network formation game is NP-hard. In order to calculate smartness factor, we need a method for approximating nodes’ best responses. By relaxing edges values to real values in the interval \([-1, 1]\) we can compute the best response of every player by a quadratic programming.

\[
\begin{align*}
\text{minimize} & -X_i A X_i^T \\
\text{subject to} & -1 \leq x_{ij} \leq 1, \quad j = 1, \cdots, n - 1
\end{align*}
\]

where \( n \) is the number of nodes in the network, the \((n - 1) \times (n - 1)\) matrix \( A \) is the weighted adjacency matrix of \( G - \{v_i\} \), and \( x \) is the best-response of node \( i \) to the remainder of the network. We know that a triangle containing an even number of negative edges is defined as balanced while a triangle with an odd number of negative edges is unbalanced. So, any integer solution to the above formulation computes the number of balanced triangle minus the number of unbalanced triangles that contain \( u \). We solve this QCQP and round its solution (values near \( \zeta \in (-1, 0, +1) \) are rounded to 0). Even though we round the obtained solution to the nearest integer value, this process has not considerable effect on computation of the best response. Our experiments on some real subnetworks as well as random graphs show that almost all solutions to the QCQP problem are very close to +1 or -1.

We consider three on-line social networks used by Leskovec [25], the trust network of the Epinions, the social network of the blog Slashdot, and the voting network of Wikipedia. For each network, we randomly extract an induced subnetwork which contains 100 nodes. As compatibility factor deals with undirected networks, we make them undirected with procedure described in Table 1. Our undirecting procedure is useful because of two reasons: (i) These datasets have very small fraction of reciprocated edges that have different signs (%0.0032 for Epinions, %0.0037 for Slashdot, and %0.0273 for Wikipedia), so the number of deleted edges is too small. (ii) In the balance theory (as opposed to status theory [26]) it is reasonable to expect that the reciprocated edges have same signs since the excessive edges should not have conflict with structural balance property of signed networks. For each node \( v_i \), we calculate \( \text{current utility}(v_i) \) and \( \text{best utility}(v_i) \). Our results show that the former has a wide variety in its value, but the latter has an almost constant value. As shown in Figure 4, there is a big difference between these two values and thus the naturality of this game is very low. This can be explained by the assumption that nodes have limited computational power and also inertia to compute their best response.

### 5. CONCLUDING REMARKS

Network creation games, in spite of their effort to represent a natural model for social networks, commonly result in models which significantly differ from real networks. There are also no accurate and practical measures for the compatibility of network creation games with a given social network; in many cases, precise measurement of compatibility cannot be achieved in polynomial time.
Figure 3: Possible undirected triad in a signed network: triad (a) and (c) are balanced and relatively stable, but triad (b) and (d) are unbalanced and susceptible to break apart. Full and dashed lines represent positive and negative relations respectively [28].

Figure 4: Nodes’ utility, below lines show current utility and above lines show best-response utility [28].

Hence, it would be beneficial to have a suitable and applicable framework to reliably estimate and compare the naturality of network creation games. This paper attempts to introduce one such framework. We have tested the proposed method to measure the compatibility of a network creation game previously introduced in the literature, and have also applied the framework as a comparison measure between several games.

Our ongoing work includes the following:

- Purifying the measurement framework to improve its usability. Our model still has some problems, for example it needs to compute nodes’ best-responses and some times it is NP-hard or time consuming.

- Proposing measures for other types of dynamics. There are many other types of dynamics defined in this context. Some of them are also more natural.

- Studying other types of NCG compatibility problems. These problems are defined formally in Section 2.

6. REFERENCES


[6] A. Blum, T. Chan, and M. Rwebangira. A


