Topology Generation for Web Communities Modeling*

György Frivolt and Mária Bieliková

Institute of Informatics and Software Engineering
Faculty of Informatics and Information Technologies
Slovak University of Technology in Bratislava
Ilkovičova 3, 842 16 Bratislava, Slovakia
{frivolt,bielik}@fiit.stuba.sk

Abstract. In this paper we present a model of Web communities which constitute a part of the Web structure. The proposed model is aimed at characterization of the topology behind the Web communities. It is inspired by small world graphs that show behaviors similar to many natural networks. We model Web communities as clusters of Web pages using graph grammars. Graph grammars allow us to simulate the structural properties of Web communities including their growth and evolution. An example of a grammar is presented. We discuss possibilities for utilization of the proposed model for research into Web communities, their properties and identification.

1 Introduction

As the Web grows, effective searching for information becomes more and more important. Present Web search engines typically crawl the Web pages in order to build indexes and/or local copies for further analysis. The search is based mainly on analysis of the content gathered. Several search engines use the hyperlink structure to provide additional information regarding the quality of the results (using for example the PageRank algorithm [13]). Knowledge of the structure of the Web graph dramatically improves the search results, in particular ranking of the search results. However, most current search engines consider the Web as a network at a rather low level of abstraction in which the vertices represent Web pages and the edges are associated with hyperlinks that connect the information content of the pages. To capture the features of the Web at a higher level of abstraction, considering a collection of Web pages created by individuals, or any kind of associations that have a common interest on a specific topic (web communities), instead of the Web pages per se, is a challenging task. However, it would enable reasoning at a higher level of abstraction, with the potential for improving the efficiency and accuracy of the information search, and also for improving the search results.

* This work has been partially supported by the Grant Agency of Slovak Republic grant No. VG1/ 0162/03.

Groupings can be observed in various natural networks. People, companies, etc., which form collections (or clusters) represented by vertices, are often denoted as communities. Edges represent interactions such as social relations between people in a social network, or trade relationships in a business network. Communities exist not only in the physical world. Research into the Web showed that they emerge in the virtual world as well [7].

Since it is our intention to model communities on the Web, we concentrate on clusters formed by Web pages. The primary understanding of the Web communities comes from sociology [10]. Similar to the connections found in human society, there exist connections between Web pages created by those who share a common interest (so that the content of the pages is oriented towards a specific topic).

We distinguish two main sources of knowledge that can be extracted from the Web: (i) the Web page content, and (ii) the topology of the network. There is a growing amount of work directed at the identification of Web communities according to the topology of the network based on hyperlink structure, i.e., it is supposed that Web pages which share similar themes, or similar interests of the authors, are interconnected, or that they belong to the same cluster [5,4,12]. Due to the large size of the Web, the topology of the Web network is largely unknown or unexplored. A significant step towards using the topology of the Web for reasoning about its content is the PageRank algorithm [13]. The idea behind PageRank is that it is possible to extract the quality of a Web page based on the references (or hyperlinks) leading to it, i.e., from its position on the network. A different method, but one still relying on topology, is introduced by Jon Kleinberg [5,6]. In [7,10] the authors show structures observable on the Web and explain the motivations for searching among communities on the Web.

Our aim is to define a model which captures the concept of Web communities. The proposed model is tested by constructing an example of a grammar and analyzing selected properties of the graphs generated according to the defined grammar. In a way similar to the approaches mentioned above we rely on the topology of the Web network and assume that the quality of the Web page content is correlated with the incoming/outgoing links of the page.

The rest of the paper is structured as follows. In section 2 we describe small-world graphs, which form a viable alternative for Web modeling. Section 3 discusses the proposed model based on the graph grammar system. In Section 4 we give an example of a grammar together with generated graphs. The properties of the generated graphs are described. The paper concludes with a discussion, a summary, and a description of future directions of our research.

2 Small-world graphs and Web networks

Different types of natural networks share some specific features. Despite their random character the topology of the graphs representing these networks has a number of universal scale-free characteristics and displays a high degree of clustering. The graphs show the so called *small-world effect*, possessing aver-

age vertex-to-vertex distances which increase only logarithmically with the total number of vertices, together with a clustering effect (which is missing in a random graph) [11]. Small-world networks can be observed in many spheres of nature. The networks of neurons in the brain, genetic networks, social networks of people, networks of words in natural languages, the Internet at the router or domain levels, and networks of Web pages, all share the features mentioned [14,9,2].

Ordered and random networks differ in two seemingly opposed ways. Ordered networks exhibit high clustering, i.e., neighboring vertices share several common neighbors. On the other hand, the average distance between any two vertices in an ordered network is high. Random networks show significant differences from ordered networks in these two properties. The growth of a random network with a given coordination number (average number of neighbors of each vertex) results in a decrease of the number of common neighbors. Furthermore, any two vertices can be connected by a relatively short path.

The difference in scale between ordered and random networks is large. Models for scaling the transition from ordered to random networks are studied in [11,15]. Networks called small-world networks share the interesting properties of both random and ordered networks: high levels of clustering and low relative distances between the vertices. These properties for small-world networks are as follows.

Average vertex distance. The average distance ℓ between any two vertices in a small-world network logarithmically depends on the size N of the network:

$$\ell \approx \log(N)$$

Logarithmic dependence allows the average distance between the vertices to be quite small even in very large networks. The precise definition of the average distance between vertices in a small-world network is still a matter of debate, but it is accepted that ℓ should be comparable with the value it would have on a random graph [11].

Clustering. Vertices in the same area are connected to each other. The clustering coefficient C_v for a vertex v with k_v neighbors is

$$C_v = \frac{2E_v}{k_v(k_v - 1)}$$

where E_v is the number of edges between the k_v neighbors of v.

Empirical results indicate that C_v averaged over all nodes is significantly higher for most real networks than for a random network, and the clustering coefficient of real networks is to a high degree independent of the number of nodes in the network [14].

Several authors have studied big portions of the Web network (with vertices representing the Web pages and connections representing hyperlinks pointing from one page to another) and demonstrated its small-world properties. In [11,2] the average diameter for a Web network with $N = 8 * 10^8$ vertices is shown to be $\ell_{web} = 18.59$, i.e., two randomly chosen pages on the Web are on average

19 clicks away from each other. The logarithmic dependence of average distance between the Web pages on the number of the pages is important to the future potential for growth of the Web. For example, the expected 1 000% increase in the size of the Web over the next few years will change ℓ_{web} to only 21 [2].

3 Web topology generation using graph grammars

As already mentioned, the Web graph shows the characteristics of a scale-free network. However, empirical measurements have also shown its hierarchical topology [14]. The modular organization of the Web is related to the high clustering coefficient. The Web model should reflect these characteristics.

We have proposed to model this kind of the pattern using graph generating L-systems. L-systems are a class of string rewriting mechanisms, originally developed by Lindenmayer [8] as a mathematical theory of plant development. With an L-system, a sequence of symbols (string) can be rewritten into another sequence, by replacing all symbols in the string in parallel by other symbols, using so-called rewriting rules (also called production rules).

L-systems are capable of generating fractal-like structures. Self-similarity was observed also in the Web [3]. General properties of the Web topology discussed in Section 2 can also be found in its parts. We expect that the proposed approach is also capable of generating networks that capture the growth of the Web, together with its Web communities large scale topology with the properties of scale-free networks with a high clustering.

Definition 1. *We call a tuple $Gr = (R, \sigma)$ a graph generating L-system, where σ is the initial graph and R is the finite set of production rules written in the form $LHS \rightarrow RHS$.*

The production rules of a graph grammar are mappings of the vertices. The application of a production rule to a vertex of the graph means replacing the vertex with the vertices defined on the right hand side of the rule. We do not distinguish between terminal and non-terminal states.

The LHS of a graph generation rule represents a vertex. The RHS of the rule consists of (a) a list of vertices together with related mappings of edges incident to these vertices and the LHS vertex, and (b) a list of edges joining the mapped vertices defined in the RHS.

Definition 2. *We denote a production rule as:*

$$v \rightarrow \left\{ \begin{array}{l} (v_1, \mu_1, p_1), \\ (v_2, \mu_2, p_2), \\ \ldots \end{array} \right\}, \eta$$

where
$p_i \in [0, 1]$ *is probability of mapping the vertex v to v_i;*
$\mu_i \in [0, 1]$ *is probability of overtaking an incident edge to the vertex v and v_i;*
η *is a subset of edges joining the vertices $v_i \in \{v_1, v_2, \ldots\}$ such that*

$$\eta \subset \{({}_{o}v, {}_{i}v, p)|_{o}v, {}_{i}v \in \{v_1, v_2, \ldots\}, p \in [0,1]\}.$$

where p is probability of generating an incident edge to the vertex ${}_{o}v$ and ${}_{i}v$.

An L-system grows the graph starting with the initial graph by applying production rules. The rule application means a replacement of a vertex with the vertices mapped by the right hand side of the rule. The rule application is called an expansion.

Definition 3. *Let $r \in R$ be a production rule, G a graph, $v \in G(V)$ a vertex, and $e_1, e_2, \ldots \in G(E)$ edges incident to the vertex v. We call an expansion a mapping:*

$$ApplyRule : G \times G(V) \times R \mapsto G^{'}$$

The result of the application of $ApplyRule(G, v, r) = G^{'}$ is:

$$G^{'}(V) = (G(V) \setminus \{v\}) \cup \{p_1(v_1), p_2(v_2), \ldots\}$$
$$G^{'}(E) = \{\mu_1(e_1), \mu_1(e_2), \ldots, \mu_2(e_1, \ldots), \ldots\} \cup \eta(v_1, v_2, \ldots)$$

where
$p_i : \{v_i\} \mapsto \{v_i, \perp\}$, $\mu_i : \{e_i\} \mapsto \{e_i, \perp\}$ is a mapping giving items from the set $\{v_1, \ldots\}$, $\{e_1, \ldots\}$ with probability p_i, resp. μ_i;
η is deduced from p_i as $\eta : 2^{\{p_i(v_1), \ldots\}} \mapsto \{{}_{out}v, {}_{in}v|_{out}v, {}_{in}v \in \{p_1(v_1), \ldots\}\}$.

The graph grows by repeated expansion. The inference step in a grammar is executed by the application of randomly chosen rule on every vertex.

Definition 4. *Let R be a set of rules, and G_1 and G_2 graphs. We say that G_2 is inferred from G_1 if a sequence $s = (v_1, r_1), (v_2, r_1), \ldots (v_n, r_n)$ exists, where*

- $\forall v \in G_1(V) \exists i \leq n \exists r \in R : s_i = (v, r)$ *and* $\forall i, j \leq n, i \neq j : v(s_i) \neq v(s_j)$ [1]
- $\coprod_{i=1}^{n} ApplyRule(G_1, s_i) = G_2$ [2]

$G_1 \dot{\rightsquigarrow}_R G_2$ denotes that G_2 was inferred from G_1 in one inference step using R. If there exists a sequence of inference steps $G_1 \dot{\rightsquigarrow}_R G_2, G_2 \dot{\rightsquigarrow}_R G_2, \ldots G_{n-1} \dot{\rightsquigarrow}_R G_n$, we say that G_n can be inferred from G_1 and denote it as $G_1 \rightsquigarrow_R G_n$.

We note that every vertex is mapped during one inference step once and only once. Finally we define the language generated by a grammar.

Definition 5. *Let $Gr = (R, \sigma)$ be a graph generating L-system. We call set of graphs L a language generated by the grammar Gr if every graph contained in L can be inferred from the initial graph σ using the rules from the finite set R: $L = \{G|\sigma \rightsquigarrow_R G\}$.*

[1] $v(s_i)$ is the first (vertex) item of the tuple.
[2] $\coprod_{i=1}^{3} ApplyRule(G_1, s_i) = ApplyRule(ApplyRule(ApplyRule(G_1, s_1), s_2), s_3)$

4 Graph grammar application

We have used the proposed language in our experiments to generate a topology with properties similar to the Web network. Our approach is demonstrated by a simple grammar containing three rules. We measure two properties of the generated graphs: the clustering coefficient and the graph diameter. We show that the formalism presented in the previous section is strong enough to generate graphs with properties that resemble small-world networks. Although the generated graphs are directed, in our measurements of the clustering coefficient and network diameter we consider them as undirected, which suffices for the purposes of evaluating the characteristics of the generated graphs. We developed a software prototype for graph generation using the specified grammar, in the Python programming language. The visualization of the generated graphs was performed by the BioLayout software[3].

4.1 Definition of the example grammar

The example grammar contains three rules generating three kinds of structures:

- *hierarchies:* a vertex is mapped to one central and several child vertices;
- *bipartite graphs:* generated vertices are divided into two sets such that no edge connects vertices in the same set;
- *cliques:* a vertex is mapped to the graph where a majority of vertices is connected.

Fig. 1 shows examples of the first expansion of each rule: a hierarchy with three child vertices, a bipartite structure with two sets by three vertices and a five clique cluster. Fig. 2 presents graphs generated by several expansions using again each rule. The grammar of every example produces graphs from an initial graph of a single vertex: $Gr = (R, \sigma = G(\{v\}, \emptyset))$. The rules R are defined thereinafter.

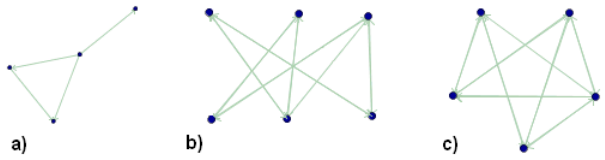

Fig. 1. Illustration of one expansion for (a) hierarchy, (b) bipartite structures and (c) cliques.

Hierarchies. Hierarchical organization can be observed in several real complex networks including the Web. A graph theoretical discussion related to the fact that the hierarchy is a fundamental characteristic of many complex systems can be found in [14].

[3] http://www.ebi.ac.uk/research/cgg/services/layout/

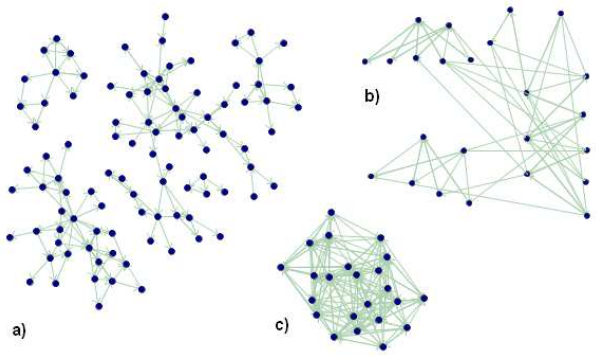

Fig. 2. Structures generated by the hierarchy production rule after 4 inference steps (a), bipartite (b) and clique (c) generation rule after 2 inference steps.

An example of a hierarchy generation rule is defined as follows:

$$v \rightarrow \left\{ \begin{array}{l} (v_{central}, 1.0, 1.0), \\ (v_{child_1}, \ 0.2, 0.8), \\ (v_{child_2}, \ 0.2, 0.8), \\ (v_{child_3}, \ 0.2, 0.8) \end{array} \right\}, \left\{ \begin{array}{l} (v_{central}, v_{child_i}, 0.8), \\ (v_{child_i}, v_{central}, 0.2), \\ (v_{child_i}, v_{child_j}, 0.2) | i, j \leq 3, i \neq j \end{array} \right\}$$

The hierarchy generation rule of our grammar produces a structure containing one central and a maximum of three child vertices. The central vertex is with high probability connected with the child vertices. We set a lower probability for generating connections between the child vertices.

The graph generated by four inference steps has a clustering coefficient of 0.475. The diameter of the largest component is 8, and the total number of vertices and edges is 93 and 169, respectively (see Fig. 2a).

Bipartite graphs. Bipartite structure models service-provider relationships, which occur on the current Web quite often. Web communities in this case are formed implicitly, i.e., the community is formed by unconnected vertices (an actual example of this is where providers' pages on similar topics do not provide links to each other).

The rule defined below generates a bipartite graph $K_{3,3}$. The clustering coefficient of the structure after the first expansion is 0. The clustering remains low after two inference steps. The graph in Fig. 2b has clustering coefficient 0.111.

$$v \rightarrow \left\{ \begin{array}{l} (v_{service_1}, \ 0.3, 0.8), \\ (v_{service_2}, \ 0.3, 0.8), \\ (v_{service_3}, \ 0.3, 0.8), \\ (v_{customer_1}, 0.3, 0.8), \\ \ldots \\ (v_{customer_5}, 0.3, 0.8) \end{array} \right\}, \left\{ \begin{array}{l} (v_{service_i}, v_{customer_j}, 0.8), \\ (v_{customer_j}, v_{service_i}, 0.8) | i \leq 3, j \leq 5 \end{array} \right\}$$

Cliques. The clique structure models mutually interconnected Web pages. This kind of structure can be found, for example, in Web portals such as corporate Web sites or home pages. An example of a clique generation rule is defined as follows:

$$v \rightarrow \left\{ \begin{array}{l} (v_1,\, 0.6,\, 0.8), \\ (v_2,\, 0.6,\, 0.8), \\ \ldots \\ (v_5,\, 0.6,\, 0.8) \end{array} \right\}, \{(v_i, v_j, 0.8) | i, j \leq 5, i \neq j\}$$

The clustering coefficient of the graph in Fig. 2c generated by two inference steps is 0.619.

4.2 Mixed structures

We have generated various mixed structures using a grammar consisting of the three rules defined above. The rules are applied randomly, each vertex is mapped by one of the three rules in every step of inference. Two exampes of graph evolution are illustrated in Fig. 3. The measured values are listed in Tab. 1.

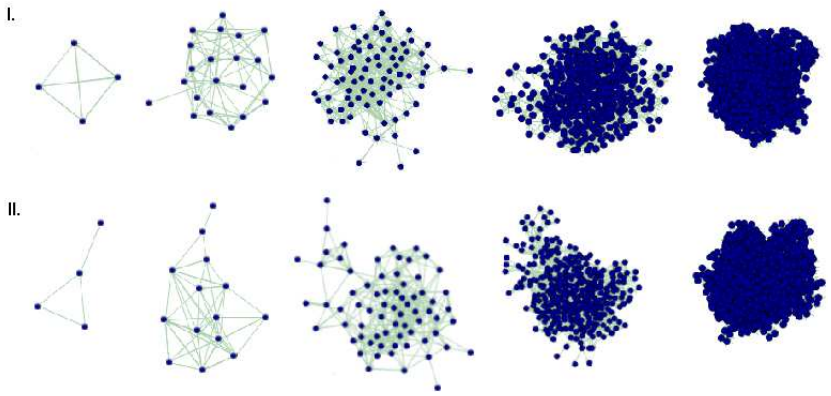

Fig. 3. Illustration of graph evolutions. Evolution of the graph *I* is started by a clique structure, whereas graph evolution *II* starts as a hierarchical structure. The initial shape of the graph persists over the growth, however after several iterations the two graphs become similar in shape and properties (see Tab. 1).

5 Discussion and conclusions

The main contribution of this paper is to propose a formalism capable of modeling the topology of Web communities. The results in Tab. 1 support our aim to generate graphs with small-world effects. The clustering coefficient is much higher than in random graphs. However, more experiments are needed in order to tune the parameters defined within the production rules, or to define new useful production rules that would improve the small-world characteristics of the generated graphs. One such extension is to introduce edges between distant vertices.

inf. steps	$\|G(V)\|$	$\|G(E)\|$	clustering	diameter	avg. out deg.	avg. in deg.
I-1	4	11	1.0	1	2.75	2.75
I-2	21	100	0.3657	3	5.0	4.76
I-3	81	423	0.3542	5	5.42	5.29
I-4	370	2 352	0.3254	7	6.60	6.44
I-5	1 719	11 997	0.2788	10	7.18	7.01
I-6	7 856	60 206	0.2744	14	7.88	7.73
II-1	4	4	0.5833	2	1.33	1.33
II-2	14	56	0.6418	4	4.31	4.31
II-3	71	371	0.2788	7	5.46	5.3
II-4	330	2 063	0.2844	9	6.47	6.29
II-5	1 519	10 852	0.2841	11	7.37	7.2
II-6	7 094	55 272	0.2841	14	8.01	8.85

Table 1. Properties of generated graphs illustrated in Fig. 3/I in the first half and fig. 3/II in the second half of the table.

Naturally several directions for future work emerge. We give a list of possible usages of the formalism presented in this paper.

Analysis of graph properties based on the rules. The results presented in Tab.1 show that although the initial properties of the graphs differ, after several iterations the resulting generated graphs have similar clustering coefficients and diameters. These properties depend on the rules of the grammar. So we assume that the properties can be computed without the inference of the graphs, which can save considerable resources when experimenting with appropriate rules for Web topology generation.

Modeling interactions between web pages. Currently we map only one vertex to a set of the vertices. By mapping more vertices we could model also interactions between Web sites. Such a model requires also modeling of attributes of the Web pages and a definition of strategies for identification of those vertices, which repose in the LHS of the rules. Although our current model produces expanding graphs, a set of rules extended by the possibility of a definition of more vertices on the RHS could also decrease the number of vertices or edges.

Definition of scalable models. Models introduced in [1,11] are scalable. Similar scale parameters can be introduced into the formalism proposed here. Tweaking of these parameters would result in grammars with different properties.

Graph pattern recognition. The proposed formalism can be used for testing or modeling some aspects of natural networks. A tool for generating networks similar to natural ones can be useful for testing algorithms for identification of the structure of the network, which was our main intention. However, another aspect that we also found interesting was the recognition of patterns defined on the RHS of grammar rules of a natural network. We expect to be able to identify a network's structure by working backwards through the inference sequence using recognition of RHS patterns. This process is far from simple. We should at least

ensure the continuous backward chaining and effective recognition of isomorphic graphs.

The proposed model extends classical L-systems by defining probabilities of the mapping of vertices and edges. By not using exact patterns we hope to decrease the complexity of the problem of computation. We believe that the work presented in this paper can be of great help in the analysis of Web communities. The characteristics of generated graphs are promising in the sense that they possess similar properties to those expected of actual Web graphs. Generated graphs could serve as a basis for identification of Web communities and their use in searching for information and recommending of high quality.

References

1. Albert-László Barabási and Réka Albert. Emergence of scaling in random networks. *Science*, 286:509–512, October 1999. www.sciencemag.org.
2. Albert-László Barabási, Réka Albert, and Hawoong Jeong. Scale-free characteristics of random networks: the topology of the World-Wide Web. *Physica A*, 281:69–77, 2000.
3. Stephen Dill, Ravi Kumar, Kevin S. McCurley, Sridhar Rajagopalan, D. Sivakumar, and Andrew Tomkins. Self-similarity in the Web. 2:205–223, August 2002.
4. Gary Flake, Steve Lawrence, and C. Lee Giles. Efficient identification of Web communities. In *Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 150–160, Boston, MA, August 2000.
5. David Gibson, Jon M. Kleinberg, and Prabhakar Raghavan. Inferring Web communities from link topology. In *Proc. of the 9th ACM Conf. on Hypertext and Hypermedia*, pages 225–234, 1998.
6. Jon M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46:604–632, September 1999.
7. Ravi Kumar, Prabhakar Raghavan, Sridhar Rajagpalan, and Andrew Tomkins. Trawling the Web for emerging cyber-communities. In *Proc. of the 8th World Wide Web Conference*, pages 1481–1493, 1999.
8. Aristid Lindenmayer. Mathematical models for cellular interaction in development. 18:280–315, 1968.
9. Mária Markošová. Language as a small world network. In J. Kelemen and V. Kvasnička, editors, *Proc. of Conf. on Cognition and Artificial Life*.
10. Pınar Yolum and Munindar P. Singh. Dynamic communities in referral networks. *Web intelligence and agent systems*, 1:105–116, December 2003.
11. Mark E. J. Newman. Models of small world (a review). *Physical Review Letters*, May 2000. cond-mat/0001118.
12. Mark E. J. Newman and Michelle Girvan. Finding and evaluating community structure in networks. *Physical Review Letters*, 2004. cond-mat/026113.
13. Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The Pagerank citation ranking: Bringing order to the Web. Standford Digital Libraries, Technologies Project, 1998.
14. Erzsébet Ravasz and Albert-László Barabási. Hierarchical organization in complex networks. *Physical Review Letters*, September 2002. cond-mat/0206130.
15. Duncan J. Watts. *Small Worlds: The Dynamics of Networks between Order and Randomness*. Princeton University Press, 2003.