Network Analysis of the SourceForge.net Community

Yongqin Gao and Greg Madey

Department of Computer Science and Engineering
University of Notre Dame
{ygao1,gmadey}@nd.edu

Abstract. Software is central to the functioning of modern computer-based society. The OSS (Open Source Software) phenomenon is a novel, widely growing approach to develop both applications and infrastructure software. In this research, we studied the community network of the SourceForge.net, especially the structure and evolution of the community network, to understand the Open Source Software movement. We applied three different analyses on the network, including structure analysis, centrality analysis and path analysis. By applying these analyses, we are able to gain insights of the network development and its influence to individual developments.

1 Introduction

In recent research, network characteristics have received more and more attention, especially in evolving networks like the Internet, social networks and communication networks [22, 24, 18]. Analyzing these characteristics can reveal interesting information. In this study, we used network analysis to investigate the network characteristics in the evolution of the community network in SourceForge.net.

2 Related Work

Topology analysis is a method that can be used to understand the evolving complex networks [19, 3, 12]. It can also be used to understand the OSS phenomenon. This study also tried to understand the OSS phenomenon by studying the community as a collaboration network where every user and project can be a single node in the network.

Gao et al. [14] analyzed the empirical data they collected from SourceForge to obtain statistics and topological information of the Open Source Software developer collaboration network. They extracted the parameters and generated a model that depicts the evolution of this collaboration network. They also used these parameters to characterize the empirical data they collected from SourceForge, while other research tended to look at the network as a single snapshot in its evolution, which means they all based their observations on network, without respect to time. They were able to inspect the network with consideration of time, using the empirical data collected over more than two years.

Xu, Madey and Gao [25] presented the results of docking [9] a Repast [17] simulation and a Java/Swarm [16] simulation of four social network models of the Open Source Software community. The simulations grew "artificial societies" representing the SourceForge developer/project community. As a byproduct of the docking experiment, they provided observations on the advantages and disadvantages of the two toolkits for modeling such systems.

These previous analyses studied the OSS community based on the global topology of the collaboration network. These methods were not capable of revealing behaviors of a single object such as a user or a project. Our study extended the understanding of OSS to the study of individual behaviors and introduced a new measure set in the study of the OSS community.

3 Our Approach

The analysis we used includes structure analysis, centrality analysis and path analysis. We conducted the analyses in the following manner. First, we conducted the structural analysis, including the following measures: diameter, clustering coefficient and component distribution. Then we conducted the centrality analysis, including the following measures: average degree, degree distribution, average betweenness and average closeness. Finally, we conducted the path analysis on most of the previous measures.

3.1 Structure Analysis

The first analysis is the structure analysis [20, 10]. Structure analysis is used to inspect the macro-measures of the network structure. The measures inspected in the structure analysis describe the network structure in a global view. Study of these measures helps us understanding the influence of network structure to individual nodes in the network.

The *diameter* of a network is the maximum distance (number of hops or edges) between any pair of nodes. The diameter can also be defined as the average length of the shortest paths between any pair of nodes in the network. In our research, we are more interested in the measures that can describe the efficiency of information propagation. So the average value is more suitable for our purpose, and we used the second definition in our research. Strictly speaking, the diameter of a disconnected graph (i.e., one containing isolated components) is infinite, but it is normally defined as the maximum diameter of its sub-clusters or other approximate values. Random graphs and other complex networks all tend to have small diameters. This is the phenomenon scientists referred to as the "small world phenomenon" [23]. The smaller the diameter of a network is, the better the network is connected. The diameter is one of the important attributes in complex network research, especially since the small world phenomenon[1] was popularized. We calculated the diameter measures using approximate method, which can generate fairly accurate results, especially when the network size is huge ($N > 10,000$). More detailed explanation and discussion can be found in [13].

[1] "Six degrees of separation" is a famous claim by Ouisa, a popular character in John Guare's play (1990)

The equation we used to calculate the approximate diameter D is

$$D = \frac{log(N/z_1)}{log(z_2/z_1)} + 1 \tag{1}$$

where N is the number of nodes in the network, z_1 is the average degree of nodes in the network, and z_2 is the average number of nodes two steps away from a given node as defined in [13].

The next measure is *clustering coefficient*. The neighborhood of a node consists of the set of nodes to which it is connected. The clustering coefficient of a node is the ratio of the number of links to the total possible number of links among the nodes in its neighborhood. The clustering coefficient of a graph is the average of the clustering coefficients of all the nodes. Recent research has found that real complex networks typically have a high clustering coefficient, which means that they exhibit a large degree of clustering [5]. Clustering coefficients of some real networks, such as the network we studied in SourceForge, can be calculated more easily from related bipartite graphs [21] by using the generating function method for bipartite graphs. More detailed explanation of this method can be found in [13].

Using this method, the clustering coefficients of these kinds of bipartite structures result in a non-vanishing value,

$$C = \frac{1}{1 + \frac{(\mu_2 - \mu_1)(\nu_2 - \nu_1)^2}{\mu_1 \nu_1 (2\nu_1 - 3\nu_2 + \nu_3)}} \tag{2}$$

where $\mu_n = \sum_k k^n P_d(k)$ and $\nu_n = \sum_k k^n P_p(k)$. In the developer-project bipartite network, $P_d(k)$ represents the fraction of developers who joined k projects, while $P_p(k)$ means the fraction of projects that have k developers.

The last measure in the structure analysis is the *component distribution*. A component of a network is defined as the maximal subset of connected nodes. To formalize the definition of a component, first we define a path in a network as:

- A path $v_1 e_1 v_2 ... e_{n-1} v_n$ is a sequence of nodes such that from each of its nodes v_i there is an edge e_i to the next node v_{i+1} in the sequence. Normally, the first node v_1 is called the start node and the last node v_n is called the end node.

Then the component C of a network can be defined as:

- Component C is a subset of (V,E) of a network. For any pair of nodes v_i and v_j, where $v_i, v_j \in C$, there exists a path $v_i e_i ... e_{j-1} v_j$ between these two nodes. And for any any pair of nodes v_k and v_l, where $v_k \in C$ and $v_l \notin C$, there doesn't exist a path $v_k e_i ... e_{j-1} v_l$ between these two nodes.

3.2 Centrality Analysis

The second analysis is the centrality analysis. Centrality analysis is used to inspect the micro-measures of the network structure or the relative importance of a node within a network. Study of these measures helped us understand the influence of individual nodes to the global network structure.

The first measure is *degree*. The degree of a node, k, equals the total number of other nodes to which it is connected, while $P(k)$ is the distribution of the degree k throughout the network. Degree distribution in real networks was believed to be a normal distribution (when $N \to \infty$), but recently, Albert and Barabási and others found it fit a power law distribution in many real networks [7]. The other measure related to degree is the average degree as $\sum P(k)/N$, which is the average of the node degrees in the network.

The next measure is *betweenness*. Betweenness is a centrality measure of a node within a network. Nodes that occur on many shortest paths between other nodes have higher betweenness than those that do not. For a graph $G(V, E)$ with n nodes, the betweenness $B(v)$ for node v is

$$B(v) = \sum_{s \neq v \neq t \in V} \frac{\sigma_{st}(v)}{\sigma_{st}} \tag{3}$$

where σ_{st} is the number of shortest geodesic paths from s to t, and $\sigma_{st}(v)$ the number of shortest geodesic paths from s to t that pass through the vertex v. This may be normalized by dividing through by the number of pairs of vertices not including v, which is $(n-1)(n-2)$.

The last measure is *closeness*. Closeness is also a centrality measure of a node within a network. Nodes that are “shallow” to the other nodes (that is, those that tend to have short geodesic distances to other nodes within the network) have higher closeness. Closeness is preferred in centrality analysis to mean shortest-path length, as it gives higher values to more central nodes, and so is usually positively associated with other measures such as degree.

The closeness $C(v)$ for a vertex v is the reciprocal of the sum of geodesic distances to all other vertices in the graph:

$$C(v) = \frac{1}{\sum_{t \in V} d_G(v, t)}. \tag{4}$$

3.3 Path Analysis

All the previous analyses (structure analysis and centrality analysis) are based on network snapshot topology. They depict the characteristics of a static network at a given point of time. But these are not the only important analyses in a network, especially an evolving network. With sequence of network snapshots instead of just single snapshot of the networks, we are able to inspect not only the measures (diameter, clustering coefficient, component, degree, betweenness and closeness) in the previous analysis, but also the developing trends of these measures.

We conducted the path analysis on the diameter, clustering coefficient, average degree, betweenness and closeness. By inspecting these developing measures, we are looking forward to understanding more about the life cycle of the network and individual nodes in the network.

4 Results and Discussion

Before discussing these analyses and measures, we need to explain the collaboration network that we studied. From SourceForge.net, we got data on two major entities – developers and projects [2]. In this data, only one relationship existed – the participation between developer and project. There were no direct links between developers and between projects. So, we looked at this network as a bipartite network, where projects and developers were the two kinds of nodes, and edges could only connect different kinds of nodes. There are two transformations from this network – the developer network and the project network. In the developer network, there is only one type of node representing the developer in the collaboration network, and the edge in the network represents the relationship of collaboration. For every pair of nodes i and j, there is an edge connecting i and j only if i and j are collaborating on at least one project. We also generated the project network, where a node represents a project in the collaboration network and the edge in the network represents the relationship of sharing the same developer(s). In the following discussion, we will abbreviate these three networks as P-NET(project network), D-NET(developer network) and C-NET(collaboration network).

4.1 Structure Analysis

The first analysis we applied is the structure analysis, including measures such as diameter, clustering coefficient and component distribution. As discussed in the previous section, the diameter is approximate on the whole network by the equation 1. The resulting approximate diameters for the D-NET are between 5 and 7, while the number of developers in the D-NET ranged from 97,705 to 123,968. Thus, the diameter of the network is quite small with regard to the overall network size (the number of developers in the network). On the other hand, the approximate diameters for the P-NET are between 6 and 8, while the number of projects in the P-NET ranged from 70,089 to 91,713. So the diameter is relatively stable compared to the significant increase of the network size.

The next measure is the clustering coefficient. We also used the approximate clustering coefficient by applying the equation 2. The resulting approximate clustering coefficients for the D-NET are between 0.85 and 0.95. On the other hand, the approximate clustering coefficients for the P-NET are between 0.65 and 0.75. High clustering coefficients reveal the highly clustered property of both the D-NET and the P-NET, which is similar to the results we got from our previous study conducted in, although the networks have been expanded significantly.

Both diameter and clustering coefficient are popular and efficient measures to describe the structure property of a network, especially the cluster property. Highly clustered networks are normally favored in real evolving complex networks like communication networks or social networks for better information propagation.

From the previous measures, we understand that both D-NET and P-NET are highly clustered networks. But these measures do not mean the networks are fully connected. Actually, most of the real networks are not fully connected. There will be

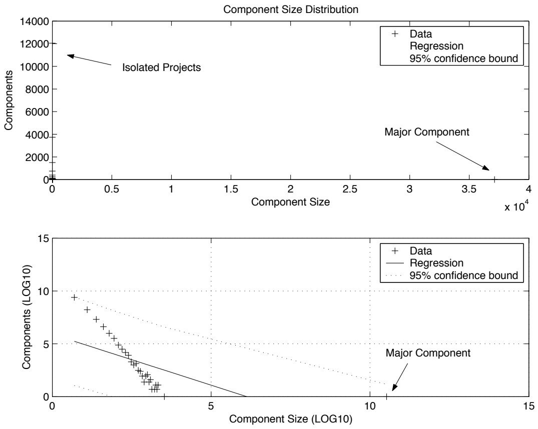

Fig. 1. Project Network Component Distribution

connected parts in the network, which we called as "component". The next measure we inspected is the component distribution. In the SourceForge community, power law exists in the component distribution of the networks. In Figure 1, the component distribution for the P-NET for June 2006 is shown. There are two figures. In the lower figure, after applying log transformation on both coordinates, we found that the component distribution fits a straight line quite well without considering the biggest component (which will be called the major component in latter discussion). The R^2 [11] of linear regression with the major component is 0.4023 and the R^2 of linear regression without the major component is 0.9886. Also, in the lower figure, we illustrated the 95% confident boundary for the linear regression as the dot line. In the upper figure, where the coordinates are in normal scale, we made another interesting discovery: almost all the components are quite close to each other, except the two extremes. One extreme is the major component and the other is the isolated components, which include only isolated developers.

4.2 Centrality Analysis

The second analysis we conducted is the centrality analysis, which focus on the following measures – degree, betweenness and closeness.

Degree is the simplest measure of the connectivity of a node in the network. We also used the C-NET, D-NET and P-NET for June 2006 as examples in this section. There are totally four degrees of developer and project in these three networks:

- *Degree of developer in the C-NET* is the number of projects in which a developer participated in the community.
- *Degree of developer in the D-NET* is the number of developers who have at least one collaboration with the given developer in the community.

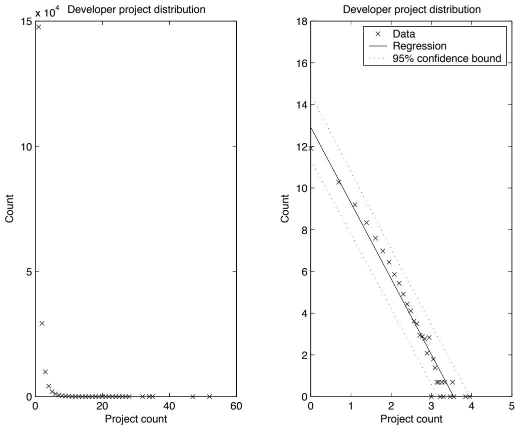

Fig. 2. Developer Size Distribution

- *Degree of project in the C-NET* is the number of developers who participated in the given project in the community.
- *Degree of project in the P-NET* is the number of projects share by at least one common developer with the given project in the community.

In the June 2006 dataset, the average degree of developer in the C-NET is 1.4525; the average degree of developer in the D-NET is 12.31; the average degree of project in the C-NET is 1.7572; the average degree of project in the P-NET is 3.8059, while the sizes of the C-NET, the D-NET and the P-NET are 215,681, 123,968 and 91,713. The average degree of developer and project in the C-NET is relatively low since the isolated developer (developer with single project) and the isolated project (project with only one developer) are big parts of the community.

Then we investigated the degree distributions of the SourceForge.net community. Degree distribution is proven to have a normal distribution in the ER model when $N \to \infty$. This was believed to be a good model for the real complex network before the power law was reported for many real network systems by Barabási et. al [6]. In the SourceForge community, we found that the degree distributions (distributions for all four degrees) also followed power law. Figure 2 and Figure 3 show two of the degree distributions.

These figures are based on the dataset from SourceForge.net for June 2006. The left figures are the degree distributions in normal coordinates. To verify the existence of power law in these distributions, we applied log-log transformations on the data to generate the right figures, which are the degree distributions on log-log coordinates. The 95% confident boundaries for the linear regression also are provided on the figures for all the log-log transformed degree distributions. The R^2 of linear regression for the developer degree distribution in the C-NET is 0.9577. The R^2 of linear regression for

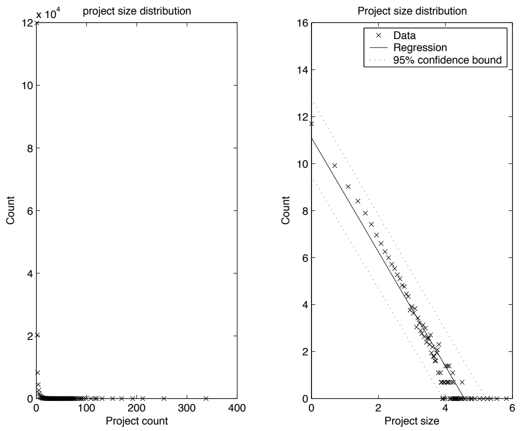

Fig. 3. Project Size Distribution

the project degree distribution in the C-NET is 0.9173. Thus, these distributions fit power law distributions very well.

Betweenness and closeness are also the common measures for centrality analysis. Betweenness is a measure to describe the importance of the node in the network according to shortest path, and closeness is a measure to describe how close the node is to other nodes. Betweenness is a normalized value in $[0, 1]$. The higher the measure is, the more central the node is to the network. Closeness is also bounded by $[0, 1]$, but it is not normalized. So closeness tends to decrease when the network size is increasing. Normally, these measures yield very small value in large networks ($N > 10,000$), so comparison of these measures only makes sense when comparing networks of similar size. Also, using the dataset from SourceForge.net for June 2006, the average betweenness for the P-NET is 0.2669e-003 and the average closeness for the P-NET is 0.4143e-005, which are relatively large for a network this size.

4.3 Path Analysis

All the measures in the previous sections (diameter, clustering coefficient, component, degree, betweenness and closeness) are about the topology of the networks. They depict the characteristics of a static network at a given point of time. These are not the only important attributes in a network, especially an evolving network [15, 4]. Since we had multiple monthly database dumps from SourceForge.net, we were able to investigate the development patterns of these measures of the networks. By applying the path analysis, we can study the life cycle and the evolving patterns of the network and individuals in the network.

The first path analysis is on the average degrees. Average degree $< k >$, which gives the average number of links per node, is a good quantitative measurement for the connectivity of a graph. Two of the developing pattern of the average degrees are

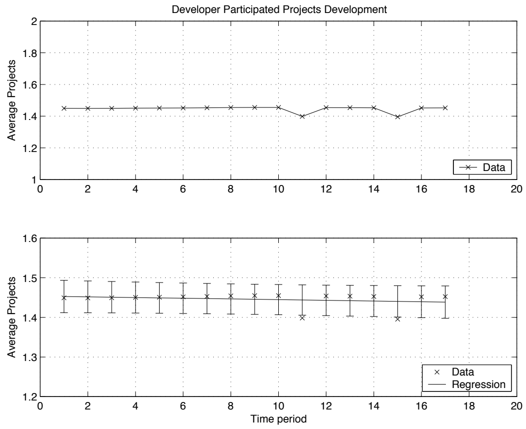

Fig. 4. Average Developer Degree in the C-NET

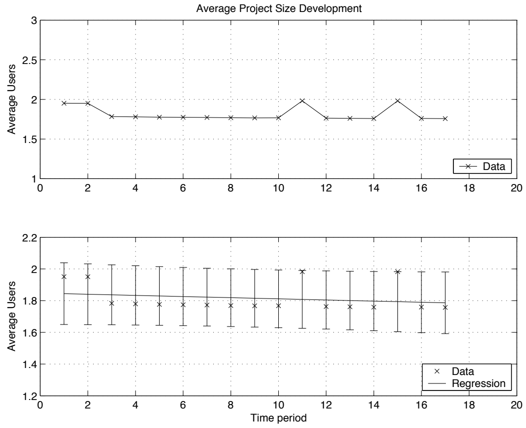

Fig. 5. Average Project Degree in the C-NET

shown in Figure 4, Figure 5. The X coordinate in the figure is the number of months that passed after February 2005.

For the average degrees, we show the linear regression in the lower figures with a 95% error bar for every data point. The slope of the regression for developer degree is -0.0009 and the slope of the regression for project degree is -0.0036. The average degrees are actually decreasing, which means the average project size and average number of projects a single developer participated in are decreasing.

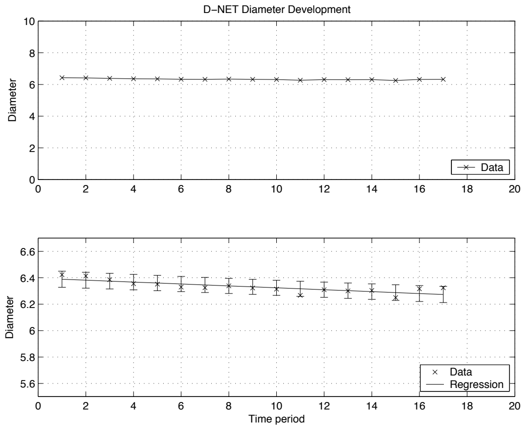

Fig. 6. Developer Network Diameter

Diameter and clustering coefficient is closely related to average degree. We will conduct path analysis on these two measures next. In this paper, we discuss only the D-NET, detailed discussion about other networks can be found in [13].

The diameter of the D-NET is a good measure of network communication ability. A shorter diameter results in fewer average steps needed for one developer to spread a message to another developer and less time needed for an idea to spread through the network. The D-NET has a small diameter, which was calculated in a previous section. Also, we investigated the evolution of the diameter of the D-NET, as shown in Figure 6.

The figure indicates that D decreases with time, which is different from the previous research [8] on random networks that reports that diameter increases with network size. The lower figure shows the linear regression with 95% error bar for the developing trend of diameter for the D-NET. The slope of the regresion is -0.0072.

Clustering coefficient is another important measures of the topology of real networks. So the clustering coefficient, a quantitative measure of clustering, CC, is also a measure we investigated. The approximate clustering coefficient for the D-NET as a function of time is shown in Figure 7.

In the figure, we can observe the increasing trend of the clustering coefficient. The lower figure shows the linear regression with 95% error bar for the developing trend of clustering coefficient for the D-NET. The slope of the regression is 0.0011. The clustering coefficient for the D-NET tells us how much a node's co-developers are willing to collaborate with each other, and it represents the probability that two of its developers are collaborating on a project. Thus, with the evolution of the D-NET, more edges (collaboration relations) are formed. This will lead to an increase in the connectivity of the developer with the neighboring developers. Furthermore, this leads to the increase in the clustering coefficient.

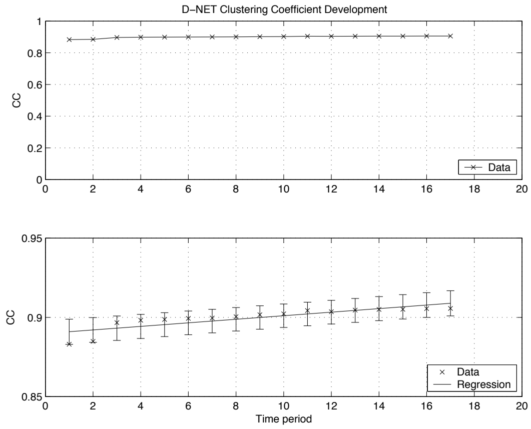

Fig. 7. Developer Network Clustering Coefficient

Increases on clustering coefficients in D-NET suggest the network is evolving to improve its cluster property. The higher the clustering coefficient is, the more connected the network is.

Now we will apply the path analysis on average betweenness and average closeness. Figure 8 shows the developing trend of average betweenness in the P-NET. In the upper figure, we can observe the almost flat developing trend of the betweenness, although the overall size of the network has increased significantly during the same time period. This can suggest that the network is in a stable topology in its own evolution. In the lower figure, we magnify the Y coordinate to have a close look at the developing trend of the average betweenness. The average betweenness has a slightly increasing trend. This observation can be explained by the "rich get richer" phenomenon discovered in other complex networks such as the Internet. Although the network is constantly expanding, the hub (the node with most links) will keep gaining more connections than the others. Also, alternative hubs or regional hubs will also emerge from the network, and these hubs will increase the average betweenness of the network.

Average closeness is another measure of centrality. Figure 9 shows the developing trend of the average closeness in the P-NET. The upper figure is the developing trend in normal coordinates and the lower figure is the developing trend in magnified Y coordinates. We can observe the similar developing behavior of the average closeness to the average betweenness. But average closeness is more flat than the average betweenness. This is because closeness for individual node is not normalized like the betweenness for individual node. Thus, the significant increase in the overall network size will have more influence on the closeness than on the betweenness. Therefore the developing trend of the average closeness will be more flat than the developing trend of the average betweenness.

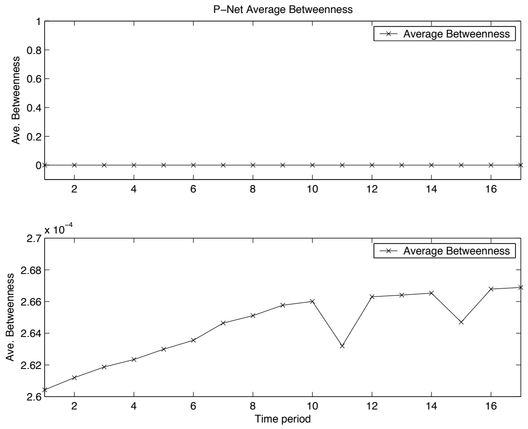

Fig. 8. Project Network Betweenness

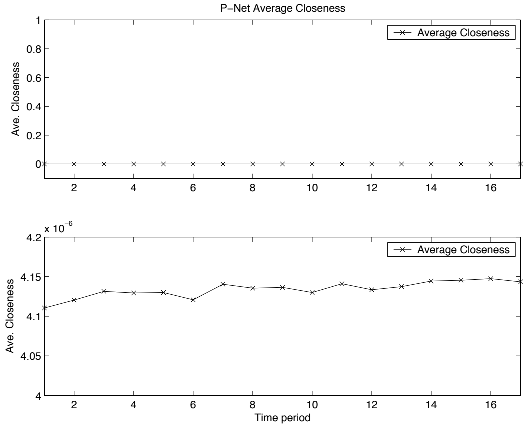

Fig. 9. Project Network Closeness

By using path analysis, we are able to look at not only the topology of the network at a given time, but also at the evolution of the network topology, and the mutual inferences between a single entity in the network and the whole network.

One of the common discoveries in all the path analyses is life-cycle like behaviors. Most of the measures have sustained a stable level throughout the inspected period of time in this study and also have increased/decreased from the previous study carried out in [13]. This phenomenon suggests that we have witnessed the beginning of the evolution of the SourceForge.net community and possibly the mature (stable) era of the SourceForge.net community. By closely watching the SourceForge.net community, we may have deeper insight into the evolution of the OSS community. Also, similar analyses can be applied to other evolving complex networks, such as communication networks or social networks to study the life cycle of those networks.

We also have made another discovery about the network measures. We observed a strong tie between the network measures and the evolution of the community networks. This discovery suggests that the network measures can be used not only as a predictor of the future of an individual entity in the network or the whole network.

5 Conclusion

In this study, we studied multiple network measures of the SourceForge.net community network and their evolution patterns by applying multiple analyses, including structure analysis, centrality analysis and path analysis. In the structure analysis, we calculated the diameter, clustering coefficient and component distribution. The two approximate methods used to calculate the approximate diameter D and approximate clustering coefficient CC are

$$D = \frac{log(N/z_1)}{log(z_2/z_1)} + 1$$

$$CC = \frac{1}{1 + \frac{(\mu_2 - \mu_1)(\nu_2 - \nu_1)^2}{\mu_1 \nu_1 (2\nu_1 - 3\nu_2 + \nu_3)}}.$$

In the centrality analysis, we calculated four different average degrees and four different degree distributions. Also, we calculated average betweenness and average closeness. The equations to calculate the betweenness $B(v)$ and closeness $C(v)$ for individual nodes are

$$B(v) = \sum_{s \neq v \neq t \in V} \frac{\sigma_{st}(v)}{\sigma_{st}}$$

$$C(v) = \frac{1}{\sum_{t \in V} d_G(v,t)}.$$

In the path analysis, we investigated developments of the multiple measures, including average degrees, diameter, clustering coefficient, average betweenness and average closeness.

Acknowledgment. This work was supported in part by the National Science Foundation, CISE/IIS-Digital Society & Technology, under Grant No. 0222829 [1].

References

1. OSS research (http://www.nd.edu/˜oss). 2006.
2. OSS research portal (http://zerlot.cse.nd.edu). 2006.

3. L.A. Adamic and B.A. Huberman. Scaling behavior of the world wide web. *Science*, 287(2115), 2000.
4. J. Ahola. Mining sequential patterns. *VTT research report TTE1-2001-10*, 2001.
5. R. Albert and A.L. Barabási. Dynamics of complex systems: Scaling laws for the period of boolean networks. *Physics Review*, 85(5234), 2000.
6. R. Albert and A.L. Barabási. Statistical mechanics of compex networks. *Reviews of Modern Physics*, 74(47), 2002.
7. R. Albert, H. Jeong and A.L. Barabási. Diameter of world-wide web. *Nature*, 401(130), 1999.
8. B. Bollobás. Random graphs. *London:Academic*, 1985.
9. R. Burton. *Simulating Organizations: Computational Models of Institutions a nd Groups*, chapter Aligning Simulation Models: A Case Study and Results. AAAI/MIT Press, Cambridge, Massachusetts, 1998.
10. J. Camacho, R. Guimerà and L.A.N. Amaral. Preprint cond-mat/0102127. 2001.
11. Online document. Pearson's r. http : //www.analytics.washington.edu/ rossini/courses/intro − nonpart/text/Pearson_s_l_r_l_.html.
12. M. Faloutsos, P. Faloutsos and C. Faloutsos. On power-law relationships of the internet topology. *Computer Communication Review*, 29(251), 1999.
13. Y. Gao. Topology and evolution of the open source software community. *Master Thesis*, 2003.
14. Y. Gao, V. Freeh and G. Madey. Analysis and modeling of the open source software community. *NAACSOS, Pittsburgh*, 2003.
15. J. Hamilton. Time series analysis. *Princeton University Press, Princeton, NJ*, 1994.
16. D. Hiebeler. The swarm simulation system and individual-based modeling. *Advanced technology for natural resource management*, 1994.
17. Repast Information. http://repast.sourceforge.net/. 2002.
18. F. Liljeros, C. Edling, A. Lan, S. He and Y. Aberg. Hub caps could squash STDs. *Nature*, 411(907), 2001.
19. A. Border, R. Kumar, F. Maghoul, P. Raghavan, S. Rajalopagan, R. Stata, A. Tomkins and J. Wiener. Graph structure in the web: experiments and models. *Computer Networks*, 33(309), 2000.
20. J.M. Montoya and R.V. Solé. Preprint cond-mat/0011195. 2000.
21. M.E.J. Newman and D.J. Watts. Random graph models of social networks. *Physics Review*, 64(026118), 2001.
22. S.L. Pimm. *The Balance of Nature*. University of Chicago Press, Chicago, 1991.
23. D.J. Watts. *Small world*. Princenton university press, NJ, 1999.
24. R.J. Williams, N.D. Martinez, E.L. Berlow, J.A. Dunne and A.L. Barabási. Two degrees of separation in complex food webs. *Santa Fe Institute Working Paper Series*, (01-07-036), 2001.
25. J. Xu, Y. Gao, J. Goett, and G. Madey. A multi-model docking experiment of dynamic social network simulations. *Agent conference, Chicago, IL*, 2003.