A Partitioned Approach to Protein Interaction Mapping*

Yanga Byun, Euna Jeong, and Kyungsook Han

Department of Computer Science and Engineering
Inha University, Inchon 402-751, South Korea
quaah@hanmail.net, eajeong@munhak.inha.ac.kr, khan@inha.ac.kr
http://wilab.inha.ac.kr/protein/

Abstract. This is a brief description of a new program for drawing protein interactions in three-dimensional space. Our program divides nodes into three groups: biconnected subgraph in the center, terminal nodes at the outermost region, and the rest in between them. Experimental results show that the program efficiently generates a clear and aesthetically pleasing drawing of large-scale protein interaction networks and that it is an order of magnitude faster than other force-directed layouts.

Protein interactions, when visualized as an undirected graph, often yield a non-planar, disconnected graph with nodes of wide range of degrees. This paper presents a new force-directed program that divides nodes into three groups:

- Group V_1 is a set of terminal nodes, i.e., nodes with degree 1.
- Group V_2 is a set of nodes in the subgraphs separated by cutvertices, except those in the longest subgraph.
- Group V_3 consists of nodes which are members of neither V_1 nor V_2.

While we find groups in the order of V_1, V_2, and V_3, we layout them in reverse order; V_3 is first positioned in the center of the sphere, V_2 in the outer region of V_3, and V_1 in the outer region of V_2 and V_3. Our program was implemented in Microsoft's C♯ and runs on any Windows 2000/XP/Me/98/NT 4.0 system.

Table 1 shows the running times on five cases, Brain (http://www.infosun.fmi.uni-passau.de/GD2001/graphC/brain.gml), Gd29 (http://www.infosun.fmi.uni-passau.de/GD2001/graphA/GD29.gml), Y2H (http://depts.washington.edu/sfields/yplm/data/), MIPS-G and MIPS-P (http://mips.gsf.de/proj/yeast/tables/interaction/), at each stage of partitioning nodes into three groups, finding shortest paths in three groups, and layout and drawing. The test cases of Brain and Gd29 are different from the other data not only in the size of data sets but also in the relative size of their V_3. Brain has 28 nodes in V_3 out of total 33 nodes and Gd29 has 129 nodes in V_3 out of total 178 nodes. However, in test cases Y2H, MIPS-G, and MIPS-P the ratio of V_3 to the total number of nodes is less than 1/2. Comparison of our algorithm with other force-directed layout algorithms shows that ours is more effective for bigger graphs and for graphs that do not have excessively high portion of V_3 (data not shown). If the portion of V_3 is high, the improvement in speed can be marginal. Fig. 1 shows the drawings of the MIPS physical interaction data.

* This work was supported by the advanced backbone IT technology development program of the Ministry of Information and Communication of Korea under grant number IMT2000-C3-4.

M.T. Goodrich and S.G. Kobourov (Eds.): GD 2002, LNCS 2528, pp. 370–371, 2002.

Table 1. Running times at each stage on five cases. P: partitioning nodes into V_1, V_2 and V_3, SP: finding shortest paths in all groups, LD: layout and drawing.

Data	Edges	Nodes			Running times			
		V_1	V_2	V_3	P	SP	LD	total ($= P + SP + LD$)
Brain	135	4	1	28	0.08s	0.02s	0.15s	0.25s
Gd29	344	40	10	128	0.84s	0.90s	2.06s	3.80s
Y2H	542	255	100	118	1.41s	0.87s	3.49s	5.77s
MIPS-G	805	198	102	231	3.24s	5.16s	8.52s	16.92s
MIPS-P	2372	665	289	572	56.39s	1m 18.82s	56.20s	3m 11.41s

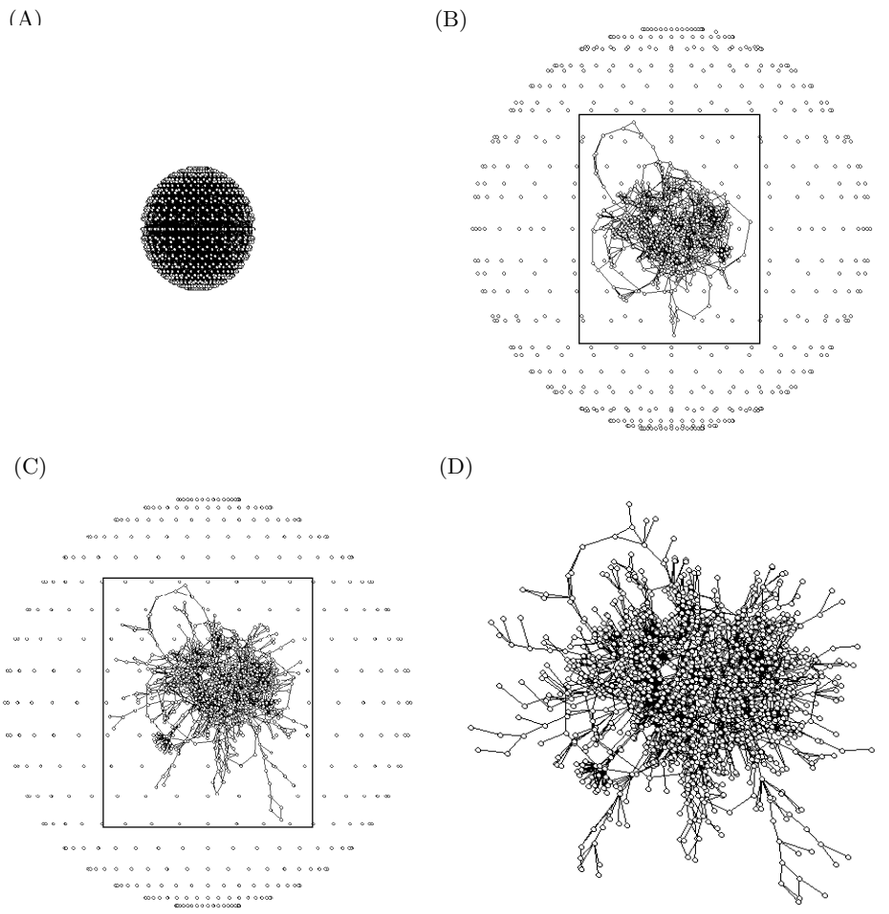

Fig. 1. Drawings of the largest connected component of the MIPS physical interaction data with 1526 nodes and 2372 edges. (A) Initial layout, (B) After drawing the nodes in V_3 shown in the rectangle, (C) After drawing the nodes in V_3 and V_2 shown in the rectangle, (D) Final drawing.