A Method for Drawing Graphs

R. J. Lipton†

Princeton University
Princeton, N.J. 08544

S. C. North

AT&T Bell Laboratories
Murray Hill, New Jersey 07974

J. S. Sandberg

AT&T Bell Laboratories
Murray Hill, New Jersey 07974

1. Introduction

We are developing programs that draw pictures of graphs in the plane. Since graphs are a very powerful way to capture information such programs have many applications. However, there is a fundamental problem with creating such programs: graphs are abstract objects, and do not include any information about how they are to be displayed. There are an infinite number of pictures that represent a given graph. How are we to decide which is best?

Our main contribution is observing that a drawing of a graph is "good" when it displays as many symmetries as possible. If a picture of a graph is symmetric, then the symmetry must be a property of the abstract graph. On the other hand, a picture does not necessarily exhibit all the symmetries of the underlying abstract graph. So a picture is "good" if it captures as many symmetries of the abstract graph as possible. Since we are drawing pictures in the plane we are interested only in Euclidean symmetries, that is, rotations and reflections.

Consider two drawings of the same graph in figure 1. The one that has more symmetry is usually considered more pleasing and more easily remembered. In fact, the pictures chosen for most graphs in texts such as [4], [20], and [44] are highly symmetric unless some particular property (planarity, cutsets, matchings, etc.) is illustrated.

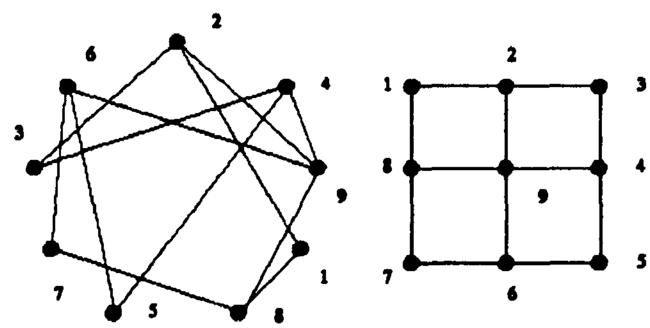


Figure 1

There is a class of graphs whose pictures can be completely determined by their symmetries. We call such graphs *perfectly drawable*. Not every graph is perfectly drawable. For instance, a rigid graph (a graph with a trivial automorphism group) is not perfectly drawable. In this case we draw the graph as if it were a subgraph of a graph having a symmetric drawing. Clearly the complete graph is a possible candidate for this symmetry extension method;

Permission to copy without fee all or part of this material is granted provided that the copies are not made or distributed for direct commercial advantage, the ACM copyright notice and the title of the publication and its date appear, and notice is given that copying is by permission of the Association for Computing Machinery. To copy otherwise, or to republish, requires a fee and/or specific permission.

© 1985 ACM 0-89791-163-6/85/006/0153 $00.75

† The first author was supported in part by NSF grant #N00014-82-K-0549

however, it is rather uninteresting since it is not dependent on the particular rigid graph. Instead we propose finding a graph with few additional edges that contains the given graph and has a non-trivial symmetry.

Since we wish to actually generate drawings of graphs on a bitmapped screen or in typesetter output, we need to compute such pictures in a reasonable amount of time. We can do this for graphs with less than 50 vertices. We have used these programs to generate the pictures found at the end of this paper.

Instead of symmetry as our measure of the quality of the drawing, we could have used dispersion of vertices, minimization of average edge length, or number of edge crossings as our criterion. However when applied in a general setting such techniques may not result in good drawings even in elementary cases. For example, using dispersion, a simple cycle may not be drawn as a circle. Furthermore, minimization of number of edge crossings is computationally intractable [16]. We expect that the symmetry metric for graph drawing can be extended to trees with very satisfactory results.

Previous results in this area include: drawing trees [37] [31], flowcharts [26] [38], and drawing 3-connected graphs [41]. These do not generalize well to making good drawings of arbitrary graphs. In the case of drawing binary trees, hierarchy information and spatial distribution determine placement of vertices in the plane, whereas there is no hierarchical information in abstract graphs. Knuth [26] places vertices linearly and then draws the edges as necessary. Trickey [38] draws directed graphs where edge intersections with vertices are constrained. Tutte [41] deals only with drawing planar graphs using straight edges in the plane. Coxeter [6] [7] considers the symmetries in regular graphs and produces good drawings of complex graphs, as well as characterizes the structure of symmetry groups in three dimensions. Uhr [45] mentions in passing that graphs may be drawn displaying certain symmetries.

2. Definitions

A graph G consists of a set of vertices V and a set of edges E.

An automorphism of G is a permutation π on V with the property that (v_1,v_2) is an edge of G if and only if $(\pi(v_1),\pi(v_2))$ is an edge of G. The automorphisms of G form a group which we denote as $Aut(G)$. Any subgroup of $Aut(G)$ defines a set of equivalence classes over the vertices in G.

To clearly distinguish between a graph G and its picture, we use $\hat{G}$ to denote the picture.

The centroid of a picture of a graph is the unique point that is defined as the average position of all the points assigned to its vertices.

We denote the group of symmetries of $\hat{G}$ as $Sym(\hat{G})$. $Sym(\hat{G})$ is isomorphic to a subgroup of $Aut(G)$. Since we are concerned with Euclidean (distance-preserving in the plane) symmetries of a finite object, $Sym(\hat{G})$ must be generated entirely by permutations associated with reflections and rotations. These allow invariant vertices to be assigned to a line or circle.

3. The Model

Our hypothesis is that we measure the quality of a drawing $\hat{G}$ by maximizing the order of $Sym(\hat{G})$ relative to the order of $Aut(G)$. The pair $<G,\tau>$ where G is a graph and τ is a subgroup of $Aut(G)$, is a set of drawings, possibly empty, such that for all $\hat{G}$ in $<G,\tau>$, τ is isomorphic to a subgroup of $Sym(\hat{G})$.

We say that $<G,\tau>$ is a single drawing only if τ determines a position in the plane for every vertex. For instance a node that is mapped into itself under a reflection has its position determined by the permutation corresponding to the reflection. We consider the node to be placed even though the position may vary on the line of reflection. Any node not mapped into itself under reflection is not fixed by the corresponding permutation. If the set of nodes placed by elements of τ are all of the nodes in V then $<G,\tau>$ is a single drawing.

A graph G is *strongly perfectly drawable* if and only if there is exactly one drawing in $<G, Aut(G)>$. A graph G is *perfectly drawable* if and only if there is exactly one drawing in $<G, \tau>$ for some τ. If G is perfectly drawable then there exists a τ of maximal order, which we call a Maximal Symmetric Subgroup (*MSS*).

4. The Maximal Symmetric Subgroup

Lemma 1: All lines of reflection pass through the origin (centroid under translation).

Lemma 2: All rotations are concentric about the origin (centroid under translation).

Proof: Assume there is a rotation S, not concentric about the centroid of the graph G. Since a symmetry completely partitions the plane by definition, S must move the centroid within the plane. But S is an automorphism with the property that it preserves the $\hat{G}$ that displays S. Thus there are two centroids of $\hat{G}$. But this is impossible since the centroid is unique by definition. Consequently, all rotations are concentric about the centroid of G. □

Note that a similar proof works for Lemma 1.

Lemma 3: Every rotation must form a dihedral group with any line of reflection.

Proof: Follows immediately from the previous two lemmas.

Theorem 1: In any Maximal Symmetric Subgroup there is at most one line of reflection generator.

Proof: Assume there is more than one line of reflection generator. Let l_1 and l_2 be two such generators.

Case 1: There is a rotation in the *MSS*. Then the rotation forms a dihedral group with each of the generators l_1 and l_2. If these are not the same group then there is a rotation of higher order such that l_2 is expressible as a product of this rotation and l_1.

Case 2: There is no rotation in the *MSS*. If l_1 and l_2 are orthogonal then $l_1 l_2$ is a rotation of order 2. If they are not then one can verify by reflection of l_1 through l_2 that there is a rotation of higher order. In either case we obtain a contradiction. Thus the theorem follows. □

Corollary: Given k lines of symmetry, they must equi-partition the plane.

Proof: Follows directly from Theorem 1 and Lemma 3.

Theorem 2: Given a graph G with no homeomorphically expanded edges, and any graph G' that is a homeomorphic expansion of G, then $Aut(G') \subseteq Aut(G)$.

Proof: We proceed by induction on the order of the automorphism group. Furthermore, we assume that we are given the smallest graph with respect to the number of homeomorphic expansions such that a new automorphism element not in $Aut(G)$ is present.

Assume there are two nodes of degree two adjacent to one another. If this path is invariant under an automorphism then one fewer homeomorphic expansion would have sufficed to obtain the new automorphism. Otherwise the image of this path must be a path containing two adjacent vertices of degree two. But again homeomorphic contraction preserves the automorphism group.

Assume there is a single homeomorphically expanded edge. Consider its image under the new automorphism. If its image is another homeomorphically expanded edge then there is a contraction that preserves the automorphism group. If its image is a node of degree two from the original graph then the original graph must have been a multigraph. Since we have exhausted all the possibilities there cannot exist any automorphism introduced by homeomorphic expansion. So we have $Aut(G') \subseteq Aut(G)$. □

We can interpret this theorem as telling us that graph layout is, essentially, independent of homeomorphic expansion. This greatly simplifies the work needed to draw a graph.

Theorem 3: *MSS* is a dihedral group.

Proof: From Theorem 1 we know there is at most one involution necessary to generate all the others. From Lemma 3 we know that if there is a cyclic subgroup in *MSS* then it forms a dihedral subgroup with any involution. Thus if there is at most one cyclic subgroup (of order >2) generator in *MSS* then *MSS* must be dihedral. The fact that there is at most one rotation generator follows from the fact that we are considering Euclidean symmetries. □

5. Drawing Algorithm

Given that we know the structure of *MSS* we would like to draw the graph based on this information. The algorithm for graph drawing determines: the automorphism group, a dihedral symmetry subgroup, and a drawing corresponding to the calculated symmetry subgroup. Since we examine only Euclidean symmetries, Theorem 3 allows us to focus on recursively laying out dihedral symmetry subgroups. We carry out the following six steps in drawing a graph.

1. Calculate $Aut(G)$.
2. Choose generators for *MSS*.
3. If there is a line of reflection find the invariant nodes, edges, and edge midpoints. Assign these fixed points to the line.
4. If there is a rotation then rotate any already placed nodes (thus placing their images).
5. Determine the plane partition given by *MSS*.
6. Assign nodes to plane partition elements. Recursively place the nodes of the subgraph induced above until all nodes are placed.

Theorem 4: Given an *MSS* of a perfectly drawable graph, G, then G can be drawn in polynomial time in the number of vertices.

Proof: By Theorem 3 we know that there are at most two generators for an *MSS* of a perfectly drawable graph. Examining all pairs of generators takes only polynomial time in the number of nodes. Since G is perfectly drawable we know that all nodes are placed by *MSS*. This follows from the condition requiring that $<G,MSS>$ consist of a single drawing. □

6. Conclusion

We have described an algorithm that draws pictures of graphs in the plane. Our main contribution is that a drawing of a graph is "good" when it displays as many symmetries as possible. There is a class of graphs whose pictures can be completely determined by their symmetries. Our criterion provides a common framework for drawing all types of graphs. We provide examples of our drawings after the references.

Further work remains to be done in extending the types of symmetries that are displayed. We are thinking in particular of using multiple independent cycles of rotation.

7. References

1. Biggs, N., Algebraic Graph Theory, Cambridge University Press, 1974.
2. Cameron, P.J., "Automorphism Groups of Graphs," in Selected Topics in Graph Theory, ed. Beineke Wilson, Academic Press, 1983.
3. Cannon, J.J., "Computers in Group Theory: A Survey," CACM, vol. 12, pp. 3-12, 1969.
4. Capobianco, M. and J.C. Molluzzo, Examples and Counterexamples in Graph Theory, North-Holland, 1978.
5. Cayley, A., "The Theory of Groups Graphical Representation," Mathematical Papers, vol. 10, pp. 26-28, Cambridge, 1895.
6. Coxeter, H.S.M., "Self-Dual Configurations and Regular Graphs," Bull. Amer. Math. Soc. , vol. 56, pp. 413-455, 1950.
7. Coxeter, H.S.M. and W.O.J. Moser, Generators and Relations for Discrete Groups, Springer-Verlag, 1980.
8. Coxeter, H.S.M., R. Frucht, and D.L. Powers, Zero Symmetric Graphs,

Academic Press, 1981.

9. Erdos, P. and A. Renyi, "Asymmetric Graphs," Acta Math. Acad. Sci. Hungar., vol. 14, pp. 295-315, 1963.

10. Foster, R.M., "Geometrical Circuits of Electrical Networks," Trans. Amer. Inst. Elec. Engrs., vol. 51, pp. 309-317, 1932.

11. Fowler, G., "Efficient Graph Automorphism by Vertex Partitioning," Artificial Intelligence, vol. 21, pp. 245-269, 1983.

12. Frucht, R., "On the Groups of Repeated Graphs," Bull. Amer. Math. Soc., vol. 55, pp. 418-420, 1949.

13. Frucht, R., "Graphs of Degree Three with a given Abstract Group," Canad. J. Math., vol. 1, pp. 365-378, 1949.

14. Frucht, R., "How to Describe a Graph," Ann. N. Y. A. S., vol. 175, pp. 159-167, 1970.

15. Furst, M., J. Hopcroft, and E. Luks, "Polynomial Time Algorithm for Permutation Groups," 21st FOCS, pp. 36-41, 1980.

16. Garey, M. and D. Johnson, Computers and Intractability, W.H. Freeman, 1979.

17. Graver, J.E. and M.E. Watkins, Combinatorics with Emphasis on the Theory of Graphs, GTM Springer-Verlag, 1977.

18. Grossman, I. and W. Magnus, Groups and Their Graphs, Random House, 1964.

19. Harary, F. and W.T. Tutte, "On the Order of the Group of a Planar Map," J. Comb. Theory, vol. 1, pp. 394-395, 1966.

20. Harary, F., Graph Theory, Addison-Wesley, 1972.

21. Hausner, M., A Vector Space Approach to Geometry, Prentice-Hall, 1965.

22. Jerrum, M., "A Compact Representation for Permutation Groups," 23rd FOCS, pp. 126-133, 1982.

23. Kagno, I.N, "Corrections," Amer. J. Math., vol. 69, p. 872, 1947.

24. Kagno, I.N., "Linear Graphs of Degree <= 6 and Their Groups," Amer. J. Math., vol. 68, pp. 505-520, 1946.

25. Kagno, I.N., "Desargue's and Pappus' graphs and their Groups," Amer. J. Math., vol. 69, pp. 859-862, 1947.

26. Knuth, "Computer Drawn Flowcharts," CACM vol. 6, no. 9, 1963.

27. Lovasz, L., Combinatorial Problems and Exercises, North- Holland, 1979.

28. McKay, B.D., "Computing Automorphisms and Canonical Labelings of Graphs," Lecture Notes in Mathematics, vol. 686, pp. 223-232, Springer-Verlag, 1977.

29. Miller, W., Symmetry Groups and Their Application, Academic Press, 1972.

30. Polya, G., "Kombinatorische Anzhalbestimmugen fur Guppen, Graphen, und Chemische Verbindungen," Acta Math., vol. 68, pp. 145-254, 1937.

31. Reingold, E.M. and J.S. Tilford, "Tidier Drawing of Trees," IEEE Trans. Software Eng., vol. 7, pp. 223-228, 1981.

32. Sabidussi, G., "Graphs with a Given Group and Given Graph Theoretical Properties," Canad. J. Math., vol. 9, pp. 515- 525, 1957.

33. Sabidussi, G., "The Composition of Graphs," Duke Math J., vol. 26, pp. 693-696, 1959.

34. Sabidussi, G., "On the Minimum Order of Graphs with a Given Automorphism Group," Monatsh. Math., vol. 63, pp. 124-127, 1959.

35. Sabidussi, G., "Graph Multiplication," Math. Z., vol. 72, pp. 446-457, 1960.

36. Sims, C., "Computational Methods in the Study of Permutation Groups," in Computational Problems in Abstract Algebra, ed. J. Leech, Pergamon Press, 1970.

37. Supowit, K.J. and E.M. Reingold, "The Complexity of Drawing Trees Nicely," Acta Informatica, vol. 18, pp. 377-392, Jan. 1983.

38. Trickey, H. Private communication.

39. Tutte, W.T., "On the Symmetry of Cubic Graphs," Canad. J. Math., vol. 11, pp. 621-624, 1959.

40. Tutte, W.T., "Convex Representations of Graphs," Proc. London Math. Soc., vol. 10, pp. 304-320, 1960.

41. Tutte, W.T., "How to Draw a Graph," Proc. London Math. Soc., vol. 52, pp. 743-767, 1963.

42. Tutte, W.T., Connectivity in Graphs, University of Toronto Press, 1966.

43. Tutte, W.T., "What is a Map?," in New Directions in the Theory of Graphs, Academic Press, 1973.

44. Tutte, W.T., Graph Theory, Addison Wesley, 1984.

45. Uhr, L. Algorithms— Structured Computer Arrays and Networks, Academic Press, 1984.

46. Watkins, M.E., "Graphical Regular Representations of Alt., Symm., and Misc. Small Groups," Aequationes Math., vol. 11, pp. 40-50.

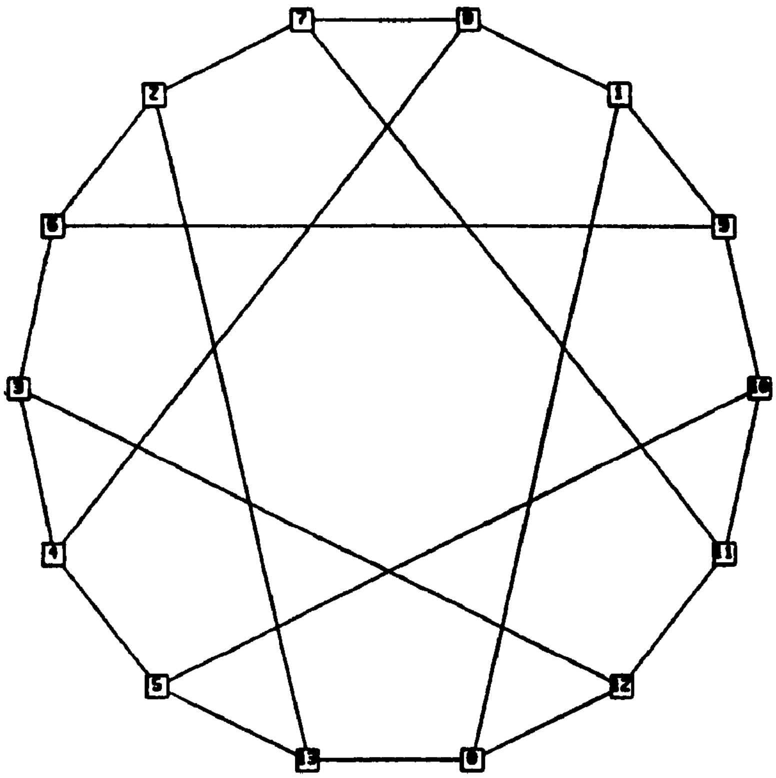
47. Watkins, M.E., "On the Action of Non-abelian Groups on Graphs," J. Combin. Theory Ser. B, vol. 1, pp. 95-104, 1971.

48. Weinberg, L., "On the Maximum Order of the Automorphism Group of a Planar Triply Connected Graph," SIAM J., vol. 14, pp. 729-738, 1966.

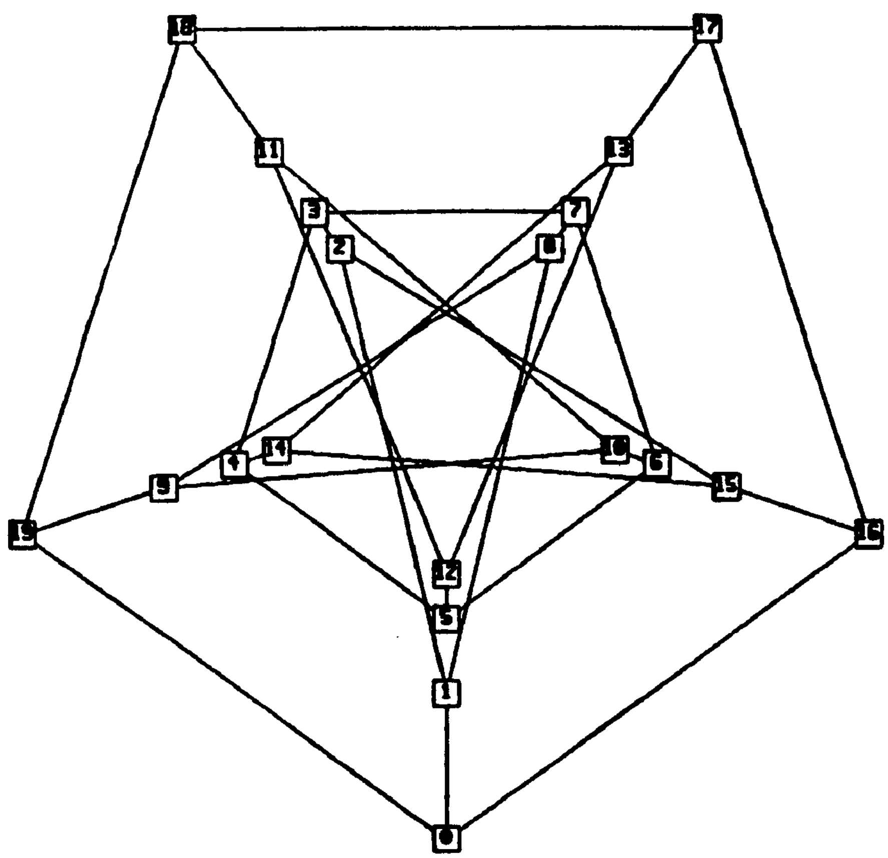
49. Wetherell, C. and A. Shannon, "Tidy Drawings of Trees," IEEE Trans. Software Eng., vol. 5, pp. 514-520, 1979.

50. Weyl, H., Symmetry, Princeton, 1951.

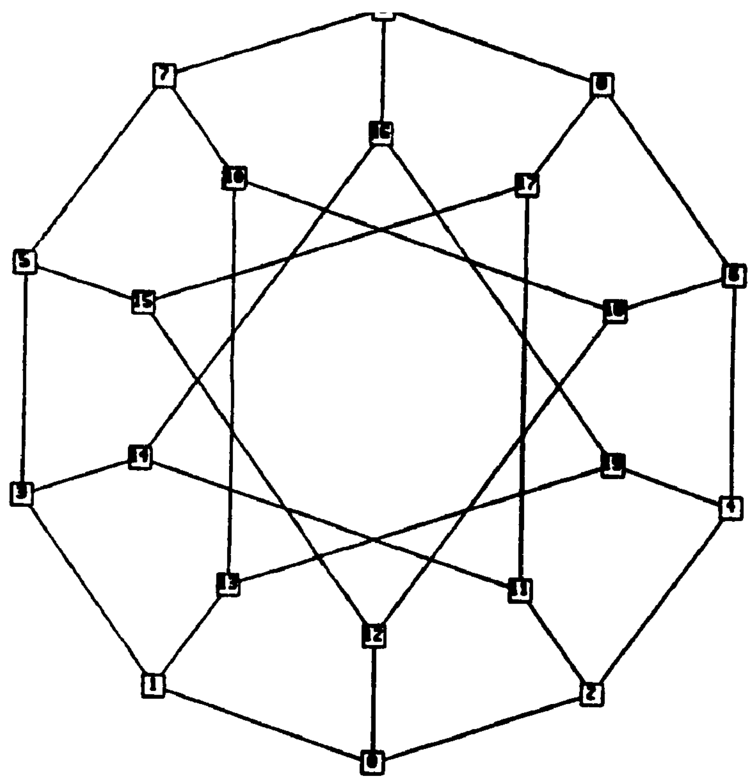
51. Wielandt, Finite Permutation Groups, Academic Press, 1964.



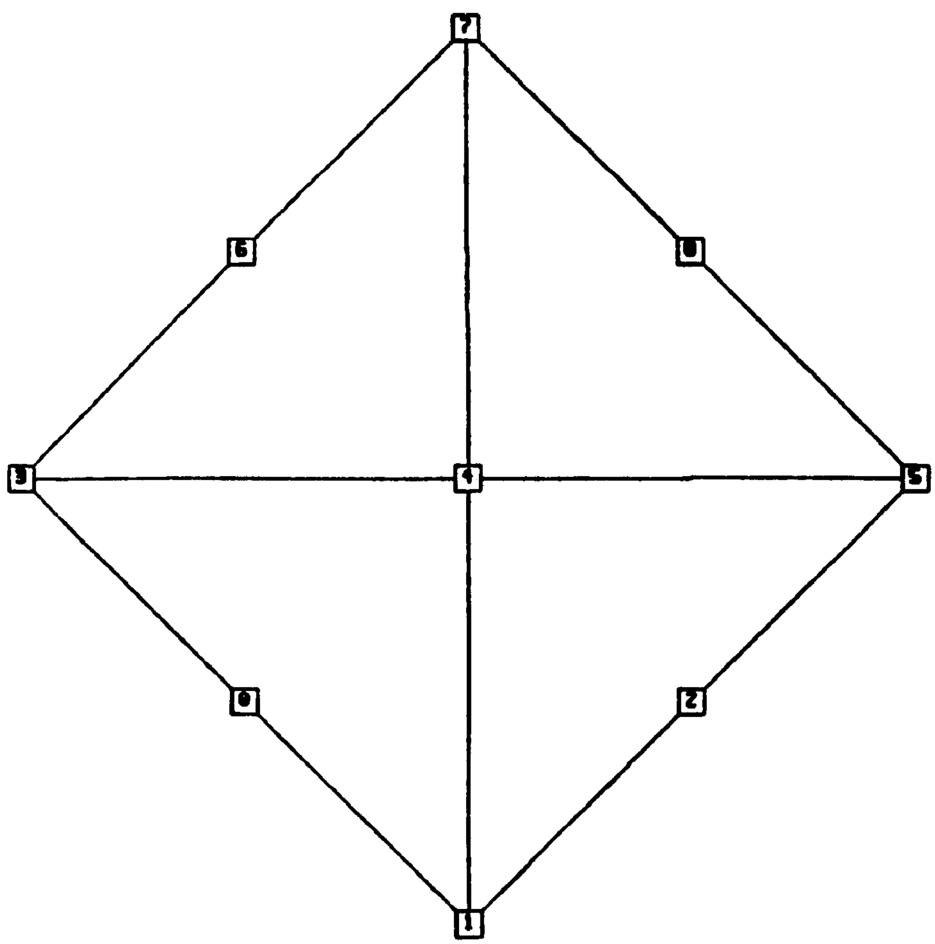
HEAWOOD GRAPH



DODECAHEDRON



UNITRANSITIVE GRAPH



GRID GRAPH