MODELING AND SYNTHESIS OF MULTICOMPUTER INTERCONNECTION NETWORKS*

10 P.

Extended Abstract

Hilda M. Standley D. Steve Auxter

Department of Computer Science and Engineering
The University of Toledo
Toledo, Ohio 43606

(419)537-2993

(419)537-2303

(NASA-CR-186619) MODELING AND SYNTHESIS OF MULTICOMPUTER INTERCONNECTION NETWORKS (Toledo Univ.) 10 p CSCL 09B

N90-25604

Unclas
G3/62 0287499

* Funded by NASA Lewis Research Center, Grant #NAG 3-699.

MODELING AND SYNTHESIS OF MULTICOMPUTER INTERCONNECTION NETWORKS

Introduction

The type of interconnection network employed has a profound effect on the performance of a multicomputer and multiprocessor design. Adequate models are needed to aid in the design and development of interconnection networks. A novel modeling approach using statistical and optimization techniques is described. This method represents an attempt to compare diverse interconnection network designs in a way that allows not only the best of existing designs to be identified but to suggest other, perhaps hybrid, networks that may offer better performance.

Stepwise linear regression is used to develop a polynomial surface representation of performance in a (k+1) space with a total of k quantitative and qualitative independent variables describing graph-theoretic characteristics such as size, average degree, diameter, radius, girth, node-connectivity, edge-connectivity, minimum dominating set size, and maximum number of prime node and edge cutsets. Dependent variables used to measure performance are average message delay and the ratio of message completion rate to network connection cost. Response Surface Methodology (RSM) optimizes a response variable from a polynomial function of several independent variables. Steepest ascent path may also be used to approach optimum points.

Comparison to Previous Work

Existing modeling approaches are either too limited (e.g., derived for describing performance of only a certain multiprocessor design or family [Siom83]), or excessively complex (e.g., detailed simulations or analytic models of a large-scale multiprocessor [Mars82]). Some models are based on queueing network (QN) theory, but otherwise there is little usage of applied statistical methods in modeling. It has been noted that even though regression analysis and statistical design of experiments could be put to great use in performance measurement studies, they have rarely been so used [Heid84]. The use of optimization techniques for the purpose of architecture synthesis is an equally unexplored area.

The study described in [Nort85] presents a performance-prediction methodology based on simplified QN models and simulation for estimating the mean performance of MIMD shared-memory multiprocessor systems. The approach used could be generalizable to MIMD architectures other than the one given as an example, but the parameters used to define the model are at a very low (i.e., machine-specific) level. The modeling approach described here attempts to maintain a higher level of abstraction in the independent variables, including the used of some graph-theoretic descriptors. This enables the focus to remain more on structural architectural parameters, rather than on particular implementation details of a given architecture.

Another model focuses on task partitioning, allocation, and subtask size as they affect performance [Cvet87]. This model also may be generalized to other architectures, but it approaches the performance question from a different viewpoint than that which is presented here. Its emphasis is on the effects due to "overhead" phenomena such as the way a problem is subdivided for parallel processing, rather than on the effects of basic architectural configurations.

Statistical Model

A sizable number of independent variables are considered for the model. This number may be substantially reduced by screening out unimportant variables. A number of screening procedures for statistical models exist, including stepwise regression [Drap66] and "group screening" for the design of experiments [Kleij75]. In stepwise regression, an independent variable may be successively added to (or removed from) the model, based upon its contribution to the overall predicting ability. In a group screening procedure, the k factors are grouped into g groups, with each group treated as a factor for a more economical design, such as an incomplete 2^k factorial design. If any group-factor is found to be insignificant, then that group can be ignored thereafter. Any significant group can subsequently be divided for further examination.

Among the independent variables considered are the following: (most have graph-theoretic definitions; see [Busa65],[Deo74],[Boff82])

(i) Size: The number of nodes (processors) in the ICN.

(ii) Average degree per node: The average number of incident edges per node. As used here, this corresponds to the average number of communication links per processor, or more precisely, the number of adjacent processors per processor.

(iii) Diameter: The length of the maximum shortest path in a graph, corresponding to the maximum distance a message may travel in the processor network.

(iv) Weight per node: The is a "contrived" measure, in that it combines the effect of both degree and message distance for nodes. For each ICN, this is computed as $(\text{diameter})^2 \times (\text{average degree per node})$.

(v) Radius: The eccentricity of the center(s) of a graph. The eccentricity of a node is the length of the maximum shortest path from that node to any other node in the graph; a center of a graph is the node (or nodes) having minimum eccentricity.

(vi) Girth: The length of the shortest cycle in a graph; since edges (links) here are considered bidirectional, and by not considering trivial "loops" as cycles, the girth of any graph will usually be at least 3.

(vii) Node-connectivity: The minimum number of nodes such that their removal from a graph will result in an un-connected graph.

(viii) Edge-connectivity: The minimum number of edges such that their removal from a graph will result in an un-connected graph.

(ix) Connection cost: [Witt81] The total number of bus connections to nodes.

(x) Minimum dominating set size: A dominating set for a graph is a set of nodes such that every node in the graph either belongs to the dominating set, or is adjacent to a member of that set.

(xi) X(m): [Wilk72] The network reliability measures of $X^n(m)$ and $X^e(m)$ denote, respectively, the maximum number of prime node and edge cutsets of size m, with respect to any pair of nodes in the network.

Several dependent or response variables are considered. Among these variables are:

(i) Message completion rate: The rate at which a network of processors can route messages from source to destination [Reed87].

(ii) Average message delay: The average number of communication links that must be traversed by a message [Witt81].

(iii) Connection cost: As defined above, the cost variable may be combined with other measures to give a more practical performance metric. If cost is not considered in some way, either as a constrained independent variable, or as part of the optimized performance variable, the solution will tend to be the simplistic result: maximized performance implies maximized cost. The composite performance measure of message completion rate divided by cost, if maximized, can assure that cost will not grow unreasonably with message completion rate. Note that other cost measures may be defined analogously, if desired.

Optimization

Response Surface Methodology (RSM) involves the optimization of a response variable, based on some polynomial function of several independent variables [Myer71]. A stepwise procedure, RSM does not guarantee that the true global optimum will be found, but it will at least find a local optimum [Kleij87]. The optimization technique of steepest ascent path can also be used to approach the local or global optimum point(s). When found, the optimum point or points can be thought of as representative of an "ideal" architectural configuration, based on the values of the various independent variables.

In the absence of discrete or realistically-valued optimum points, the gradient vector may indicate the direction(s) of greatest improvement, i.e. which variable will induce the greatest gain in performance when changed. Figures 1 and 2 show two views of a simple example where the response variable (message completion rate divided by cost) is defined as a polynomial function of two independent variables. For each dimension, the direction of greatest increase in performance is evident.

Architecture Synthesis

Upon ascertaining the optimum point(s) in the k-space, the results must be interpreted as dictating an actual interconnection network. Because of the large problem space and large values for many of the independent variables, the variables are treated as continuous rather than discrete, which indicates that integer programming is not appropriate [Phil76]. Any optimum point is likely not to be integer-valued, so it is necessary to examine integer-valued points neighboring the optimum.

Figure 1.
Message Completion Rate / Cost
as a function of Diameter and
Node Connectivity

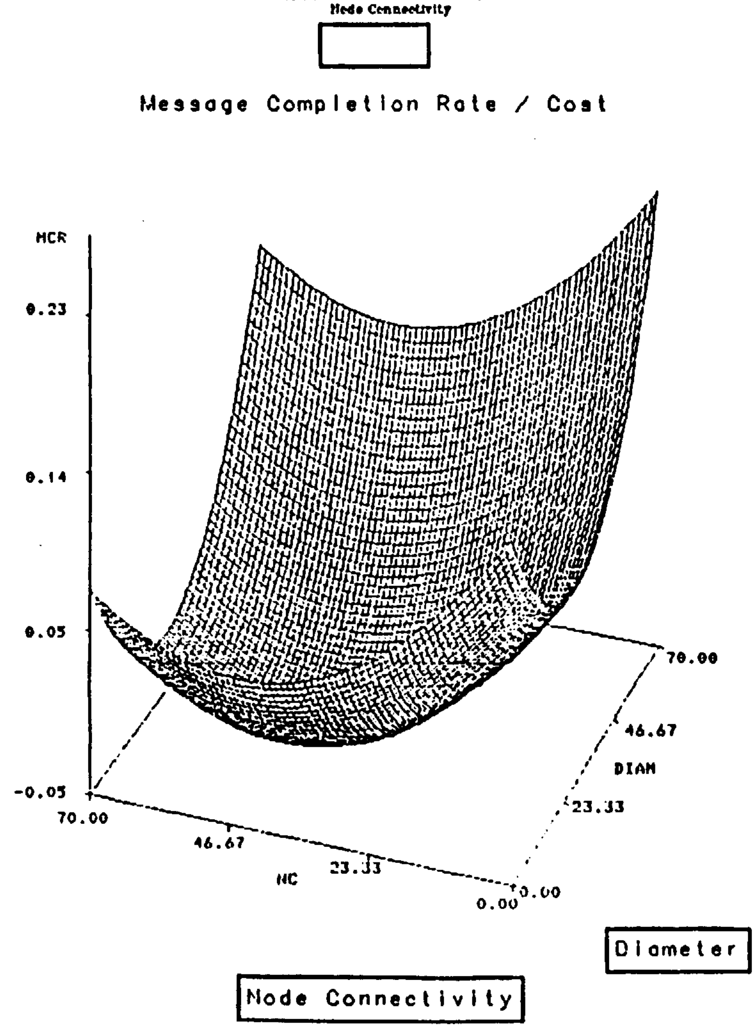
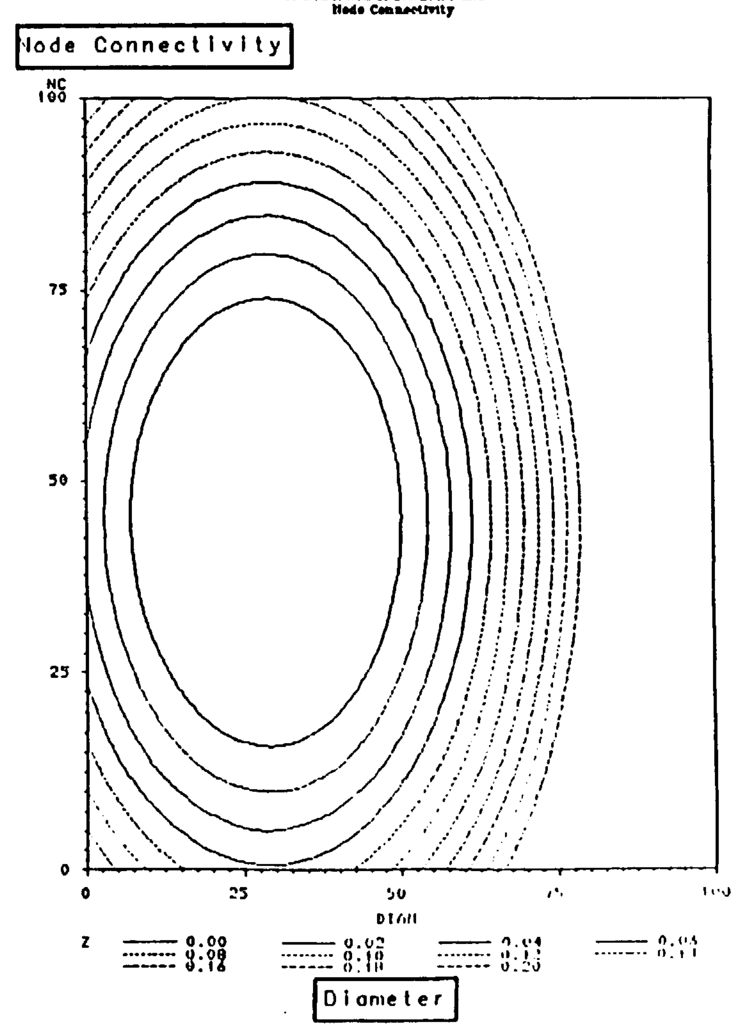


Figure 2.
Message Completion Rate / Cost
as a function of Diameter and
Node Connectivity



Preliminary Results

Some initial results are given using data compiled from five types of interconnection networks with various numbers of processing elements. The architectures used to define the independent variables for the model are: (i) Lens interconnection [Fink81], (ii) 3-dimensional torus [Reed87], (iii) Cube-connected cycles [Reed87] [Witt81], (iv) 3-dimensional spanning-bus hypercube [Witt81] [Reed87], (v) 3-dimensional dual-bus hypercube [Witt81] [Reed87]. The values for N, the number of nodes, and the response variable (performance measures) values come from [Reed87] and [Witt81]. The performance measures (dependent variables) used here are the upper bound on message completion rate, the ratio of message completion rate to network connection cost, and the average message delay. The independent variable values are obtained using graph theoretic or other definitions as stated above.

For examples, two different polynomial regression models (with two different dependent variables) are presented, including variables retained for the model, analysis of variance information, and percent of variation in response accounted for by the model. Up to third-order terms are considered for all variables. Variables are deleted from the model in a backward elimination fashion [Drap66].

Example 1: The remaining independent variables are diameter (D), girth (G), weight (W), node connectivity (NC), minimum dominating set size (MDS), average degree (AD), X(n), X(e), and size (N). The response variable, Y, is message completion rate divided by connection cost.

Regression Equation:

$$Y = 0.1406 - 0.0004745D - 0.00000909W - 0.005764NC + 0.00002943X(n) - 0.00007685\,X(e) + 0.0011752AD^2 + 0.0002126G^2 + 0.00000002X(e)^2.$$

R^2 (percent of variation explained) = 94.7%

Example 2: The remaining independent variables are diameter (D), girth (G), node connectivity (NC), average degree (AD), and X(e). The response variable is average message delay.

Regression equation:

$$Y = -0.1079 + 0.28550D + 0.9720G + 0.5165NC + 0.0007465X(e) - 0.1210AD^2 - 0.002776G^2$$

R^2 (percent of variation explained) = 99.9%

Since the gradient for both of the above regression polynomials does not vanish for any values of the independent variables, there can be no local maxima for the response variables. By examining the gradient, however, it can be seen which variables can effect the greatest improvement in response when changed.

Conclusion

Being able to express performance measures as functions of modifiable design parameters is the most obvious benefit of a modeling approach such as this, along with the ability to decide upon improvements in design for a given application. The success of this method of analysis depends upon the choice of appropriate performance measures and the selection of network parameters that are found useful in determining performance.

Bibliography

[Boff82] T.B. Boffey, "Graph Theory in Operations Research," The MacMillan Press, Ltd., 1982.

[Busa65] R.G. Busacker, T.L. Saaty, "Finite Graphs and Networks: An Introduction with Applications," McGraw-Hill, Inc., 1965.

[Cvet87] Z. Cvetanovic, "The Effects of Problem Partitioning, Allocation, and Granularity on the Performance of Multiple Processor Systems," IEEE Trans. Comp., Vol. C-36, April 1987, pp. 421-432.

[Deo74] N. Deo, "Graph Theory with Applications to Engineering and Computer Science," Prentice-Hall, Inc., 1974.

[Drap66] N.R. Draper, H. Smith, "Applied Regression Analysis," Wiley & Sons, 1966.

[Fink81] R.A. Finkel, M.H. Solomon, "The Lens Interconnection Strategy," IEEE Trans, Comp., Vol. C-30, Dec. 1981, pp. 960-965.

[Heid84] P. Heidelberger, S.S. Lavenberg, "Computer Performance Evaluation Methodology," IEEE Trans. Comp., Vol. C-33, Dec. 1984, pp. 1195-1220.

[Kleij75] J.P.C.Kleijnen, "Screening Designs for Poly-Factor Experimentation," Technometrics, Vol. 17, Nov. 1975, pp. 487-493.

[Kleij87] J.P.C. Kleijnen, "Statistical Tools for Simulation Practitioners," Marcel Dekker, Inc., 1987.

[Mars82] M.A. Marsan, M. Gerla, "Markov Models for Multiple Bus Multiprocessor Systems," IEEE Trans, Comp., Vol. C-31, March 1982, pp. 239-248.

[Myer71] R.H. Myers, "Response Surface Methodology," Allyn and Bacon, 1971.

[Nort85] A. Norton, G.F. Pfister, "A Methodology for Predicting Multiprocessor Performance," Proc. Int'l Conf. Parallel Processing, 1985, pp. 772-778.

[Phil76] D.T. Phillips, A. Ravindran, J.J. Solberg, "Operations Research," Wiley and Sons, 1976.

[Reed87] D.A. Reed, D.C. Grunwald, "The Performance of Multi-Computer Interconnection Networks," Computer, Vol. 20, June 1987, pp. 63-73.

[Siom83] K.O. Siomalas, B.A. Bowen, "performance of Cross-Bar Multiprocessor Systems," IEEE Trans.Comp, Vol. C-32, July 1983, pp. 689-695.

[Wilk72] R.S. Wilkov, "Analysis and Design of Reliable Computer Networks," IEEE Trans. Comm., Vol. COM-20, June 1972, pp. 660-678.

[Witt81] L.D. Wittie, "Communication Structures for Large Networks of Microcomputers," IEEE Trans. Comp., Vol. C-30, April 1981, pp. 264-273.