N89-29786

MULTIPROCESSOR ARCHITECTURE: SYNTHESIS AND EVALUATION

Hilda M. Standley

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

Performance of a multiprocessor is determined by

- the algorithms
- the programming language
- the program
- the language support environment and operating system
- number of processing elements
- characteristics of the processing elements
- interconnection network
- shared memory organization

The difficulty in the analysis of multiprocessor performance may be attributed to the large number of factors that may affect performance both independently and through interactions. Such factors may be roughly divided into software and hardware categories: software--the applications algorithm, the nature of the programming language, the efficiency of the program, and the language support environment and operating system; hardware--the number of processing elements, the capabilities of the processing elements, the interconnection network, and the organization of memory.

Goals:

- Ignore the algorithm effect
- Remove the language/programming effect
- Study only those characteristics of the structure of the architecture.

The goal of this study is to remove the influence of the choice of algorithm used for a particular application and to remove the effects of the high-level language and the efficiency of the program. The study concentrates on only those characteristics of the structure of the architecture. The "structure of the architecture" is defined to include those parameters that distinguish an architectural design at the diagram level. For example, the interconnection network plays an integral part in such a description while the capabilities of the individual processing elements, while crucial to the execution of the program, are not represented in the diagram.

Removing the language/programming effect:

- Express maximum amount of parallelism
- Data Flow Diagrams (operation level)
- Data Flow Diagrams (program module level)
- Partitioning and mapping of data flow diagrams

A high-level language notation to express the maximum amount of parallelism is required to assist in removing the language/programming effect. The EASY-FLOW language, based on the data flow paradigm, offers a mechanism for expressing the data dependencies between program modules, down to the level specified by the programmer. These data dependencies are obstacles to parallel execution. Modules which are not related by data dependencies may be executed in parallel. The execution environment must include a mechanism for the partitioning and mapping of the resulting data flow diagrams.

Study the impact of the memory organization and the interconnection network

A queuing network mathematical model is developed for representing the effect of expanding separate shared memories into a system of memory hierarchies.

The two elements of the architectural structure selected for initial study are the memory organization and the interconnection network. A queuing network statistical model for a multiprocessor with shared memory is expanded to include a hierarchy of memory modules at each shared memory cluster.

Model is based on an expanded GMI
(General Model for Memory Interference)

Performance is measured as the expected number of busy memories.

The shared memory hierarchy model is based on the General Model for Memory Interference (GMI) suggested by Hoogendoorn. Each processor cycles between a random access to a particular level within a memory cluster and a time interval in which internal computation is performed. Requests to the same memory cluster are queued at the cluster. Performance is measured by the expected number of busy memories.

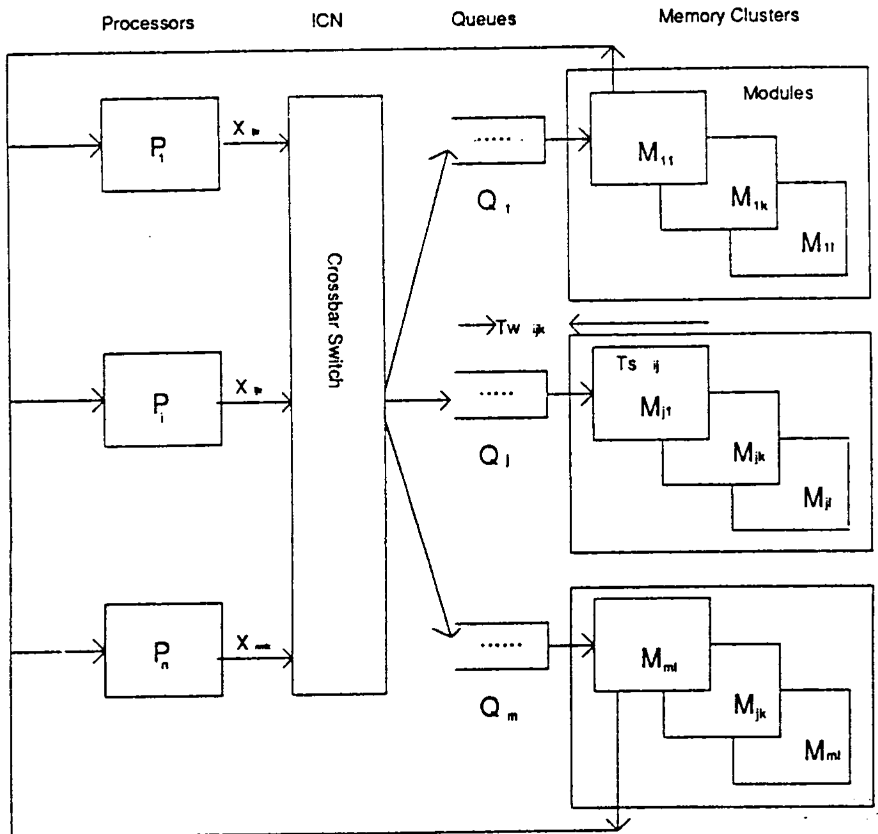


Figure 1. A Multiprocessor System: n PE's, m Memory Clusters, and l Levels

In the shared memory cluster multiprocessor model, the processors are connected to the memory clusters via a crossbar switch. It is assumed that this switch introduces no delay in accessing memory. Requests to memory are queued at each memory cluster. Delays in memory access time may be introduced by interference from other processors accessing memory levels within this same cluster.

A Network II.5 (CACI, Inc.) simulation has been developed in order to evaluate the analytic model. An eight-processor/eight-memory cluster system is evaluated under a variety of access distributions and intervals of computation time between requests to memory. The data collected from 63 simulation runs correlates with the results of the analytic model at 0.9950, overall.

Modeling the effects of the interconnection network

A polynomial surface representation of performance is developed in a (k + 1) space.

Independent variables may be quantitative and/or qualitative:

- size
- average degree (per node)
- diameter
- radius
- girth
- node-connectivity
- edge-connectivity
- connection cost
- minimum dominating set size

For the analysis of the effect of the interconnection network on performance, a polynomial surface representation of performance is developed. Variables thought to influence the performance of a network are: size, average degree (per node), diameter, radius, girth, node-connectivity, edge-connectivity, connection cost, and minimum dominating set size.

Performance measures:

- message completion rate
- average message delay
- connection cost

Dependent measurements are used to gauge performance. Typical performance measures are message completion rate, average message delay, and connection cost. Although the nature of the problem is for the different levels of the independent variables to determine a very much discrete set of performance points, the problem is viewed as being continuous in the performance variable.

Optimization:

Response surface methodology (RSM) optimizes a response variable, based on some polynomial function of several independent variables.

Gradient vector may indicate direction of steepest ascent.

A polynomial function of several independent variables is used to estimate the performance surface. This function is estimated through curve fitting techniques. Response surface methodology (RSM) optimizes the response (performance) variable, working from this estimated polynomial function. In the situation where an optimum is not indicated, gradient vector methods may detect the direction of steepest ascent.

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

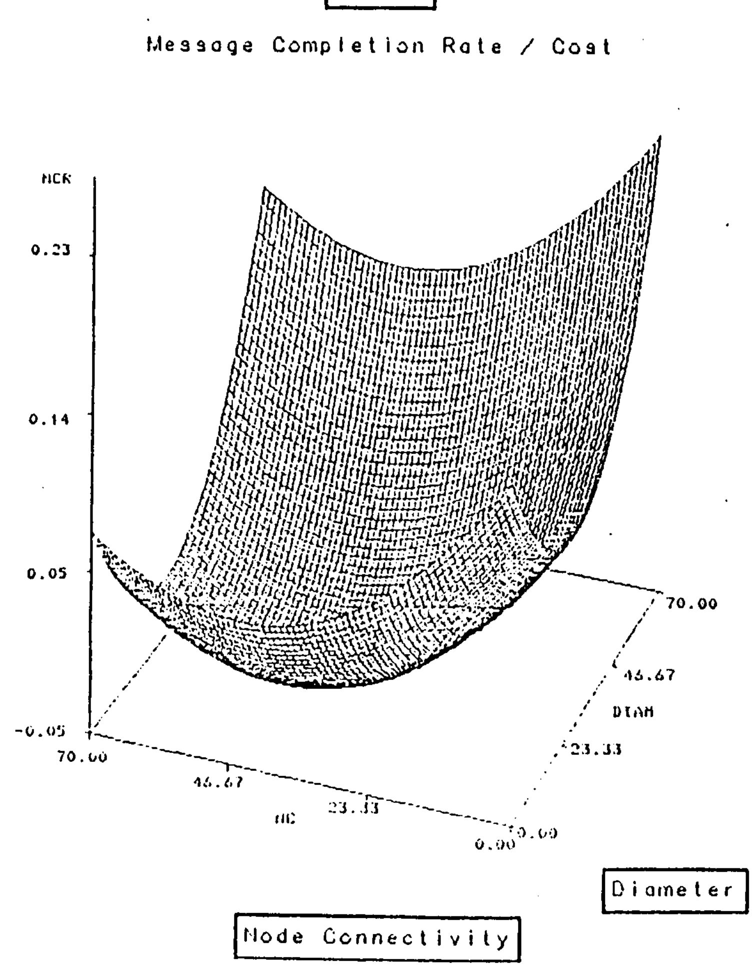
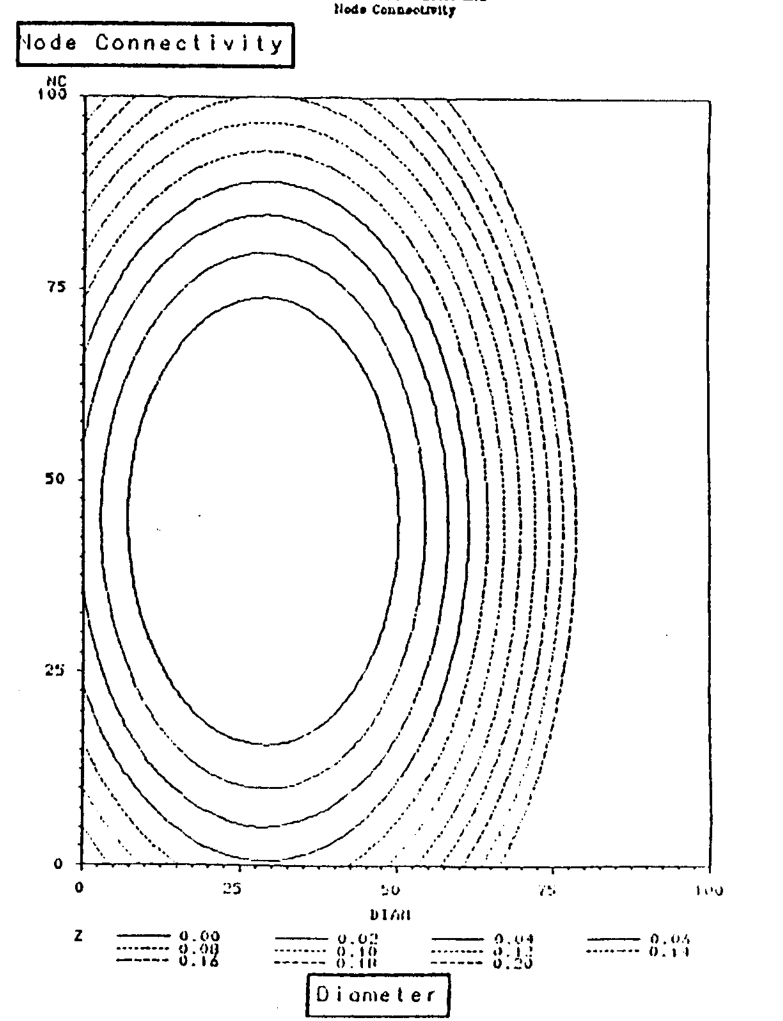


Figure 2.
Message Completion Rate / Cost as a function of Diameter and Node Connectivity



ORIGINAL PAGE IS
OF POOR QUALITY

An application of this analysis uses independent variables of node-connectivity and network diameter and the performance measure of message completion rate/cost. It may be seen from the diagram that better network performance levels occur at the "corners" of the graph, for example when both diameter and node-connectivity are high.

Network Synthesis

The results of this analysis may be used to identify appropriate levels of independent variables to indicate optimum or near-optimum performance networks. Existing, well-studied, networks; networks that are hybrids of existing networks; or completely novel networks may be suggested.