Contention and The Star Graph as a Network Topology

Douglas M. Pase

Oregon Graduate Center
Department of Computer Science
and Engineering
19600 N.W. von Neumann Drive
Beaverton, OR 97006-1999 USA

Technical Report No. CS/E 88-023

Abstract

An interesting network topology, called the ***star graph,*** has been recently proposed, that interconnects more processors with fewer connections and smaller communication delay than the popular n-cube. In this paper it is shown that although the message latency is lower for the star graph, messages require a larger proportion of the network links, which could increase contention within the network.

Contention and The Star Graph
as a Network Topology

Douglas M. Pase

Abstract

An interesting network topology, called the ***star graph***, has been recently proposed, that interconnects more processors with fewer connections and smaller communication delay than the popular n-cube. In this paper it is shown that although the message latency is lower for the star graph, messages require a larger proportion of the network links, which could increase contention within the network.

1. Introduction

A number of different network topologies have been proposed for distributed, message passing computing systems [1,6,7]. The most popular topology has been the binary n-cube, or hypercube. Another topology, the star graph, has been proposed [2,4] which offers a smaller degree, diameter, and average diameter [3]. The star graph is a sub-family of a larger class of graphs known as *Cayley graphs.* This paper proposes a new criterion of topology evaluation, and compares the star graph with the binary n-cube. Section 2 discusses performance issues of a network topology, in particular, latency and contention. A general discussion of Cayley graphs and the definition of a star graph may be found in section 3. Section 4 compares the hypercube and star graph topologies.

2. Network Topology and Performance

The processing power of parallel systems is determined almost exclusively by power of the computing elements and the interconnection network. Much work has already been done on improving the performance of the computing elements and the problems are fairly well understood. A communication network consists of a network topology, and hardware which actually sends and receives data. Here we consider only the effect of topology. Initially only problems with relatively simple communication patterns were attempted, so only simple topologies, such as a ring, mesh, or torus, were needed. However, progressively more complex problems are being solved on distributed processors — problems which demand higher performance from the communication network.

The two major costs associated with communication are message latency and contention. Network topology plays a relatively minor role in message latency. Topology determines the distance a message must travel to reach one node from another. Larger distances imply longer message latencies. Message pipelining is a simple technique that reduces the effect of distance on message latency. If messages are sent as complete indivisible units from one neighbor to another, the latency is

$$l = d \times t$$

or the total distance d times the time it takes to send the message one hop. If the message is divided into packets, and each packet is sent independently the latency becomes

$$l = d \times t' + (p-1) \times t'$$

or the time it takes to ship one packet d hops plus the time it takes to ship each of the remaining packets one hop. Since $t' \propto t/p$, the latency of the second case is generally much smaller. The importance of this is that if p is large latency grows very slowly with the distance a message must travel.

Contention occurs when two different messages compete for the same link between two nodes. It may occur if both messages are intended for the same node, the source of one message is the destination for the other and *vice versa*, or if the two routing paths otherwise happen to share some of the same links. The amount of contention is a function of the network resources in use, which is a function of the network topology and message traffic. As messages travel farther they use more links — links which cannot be used for other transmissions until that packet or message has been transmitted. As more links are used, fewer links are available, and the likelyhood that a message will be delayed increases.

More specifically, the contention increases with increasing distance a message must travel, and decreases with an increase in the number of links in the system. In other words, it increases with the fraction of resources used by each message. If we assume the message destinations have a random distribution with the source node in the center, we can model the fraction of links used per message by

$$\textit{fraction of links used per message} = \frac{\textit{average distance}}{\textit{number of links}}$$

If we assume the whole network is included in the distribution, that is, every node is equally likely to receive a message from every other node, the equation becomes

$$\textit{fraction of links used per message} = \frac{\textit{average diameter}}{\textit{number of links}}$$

The average diameter of a graph is the average distance a message must travel. The number of links in a regular network is *network degree*$\times$*network order*/2 so by substituting this value into the previous equation it becomes

$$\textit{fraction of links used per message} = \frac{2\times\textit{average diameter}}{\textit{network degree}\times\textit{network order}}$$

Now, if we assume that each node is contributing the same number of messages to the system as every other node, we can derive the communication resources in use within the system†. We multiply the previous formula by the network order to get

$$\textit{fraction of links used} = \frac{2\times\textit{average diameter}}{\textit{network degree}}$$

Since what we wanted was a relative indicator of contention, we may drop the constant factor 2. Here we have a rough estimate of performance for a topology. This assumes

(1) Every node sends messages to every other node with equal probability. Each node does not have a preferred node or neighborhood to whom it sends messages.

(2) The network topology is regular — that is, each node has the same number of connections to other nodes.

(3) Network saturation is not significant.

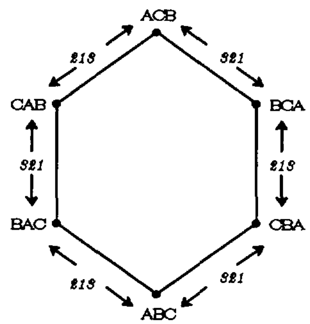
This measure indicates the relative performance required of the links for two networks to yield approximately the same performance. Thus if one network has a value of ½, and another, 1, the second network has twice the contention of the first, and the second network would need to have links which were double the speed of the first to match its level of contention. Note that the first network could have a longer latency and still have less contention.

The network degree has another effect on the system — *cost*. As the degree increases both the number of links and the board complexity of each node increase. This in turn pushes up the dollar cost of the system.

† If no assumptions were made about the message traffic one could still derive upper and lower bounds for the available communication resources. One could conceivably have an application which excercised the network in such a way that the the full bandwidth were used, or one so perverse that every node in the system had a message scheduled for every link within its routing map. If the message traffic is so heavy that saturation occurs, other parameters ignored by this model become important, such as how the conflicts are resolved, whether circuit or packet switching is used, the exact distribution of the messages, the routing method, etc.

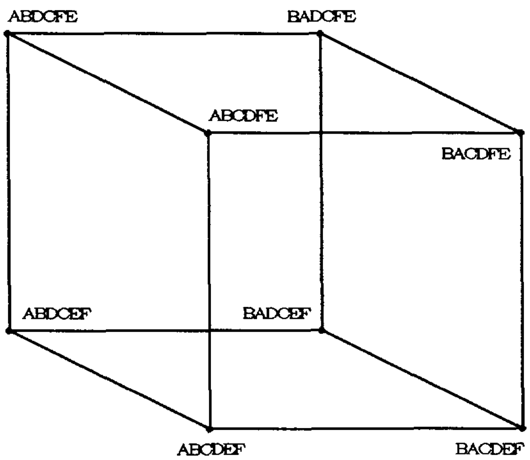
3. Cayley Graphs

Now we turn to a specific collection of network topologies. Given a set of generators for a finite group G, one can draw a graph, called the *Cayley graph*, in which the nodes of the graph correspond to the elements of the group, and the edges correspond to the actions of the generators. Cayley graphs offer many advantages over other topologies which have been proposed for interconnection networks. For example, they are *vertex symmetric*. From any node the rest of the network appears the same, regardless of the node from which the view is taken. The graphs are regular, and routing within a Cayley graph is relatively simple. Many families of Cayley graphs grow very rapidly with small degree.



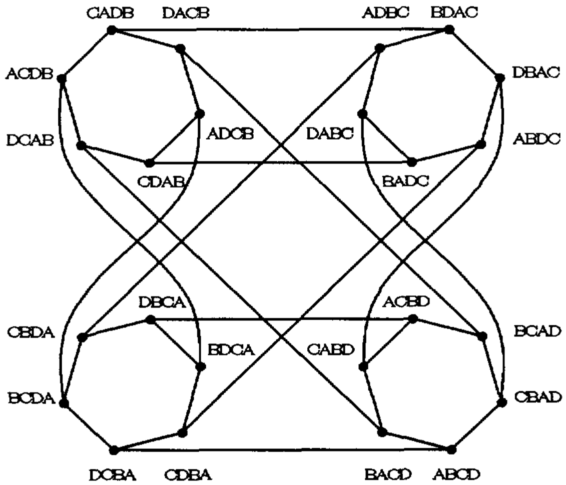
A Simple Cayley Graph

From the definition it is obvious that only the elements and generators of the group are important to the graph, not the group operations themselves. For example, consider the group whose elements are {*ABC, ACB, BAC, BCA, CAB, CBA*}, and whose generators are {*213, 321*}. The first generator transposes the first and second letters of the element. The second generator transposes the first and last letters. This creates the simple graph shown in the figure. In all cases we specify that the set of generators must be closed under inverses so we may view the graph as being undirected. It is obvious that the number of generators determines the degree of the graph.



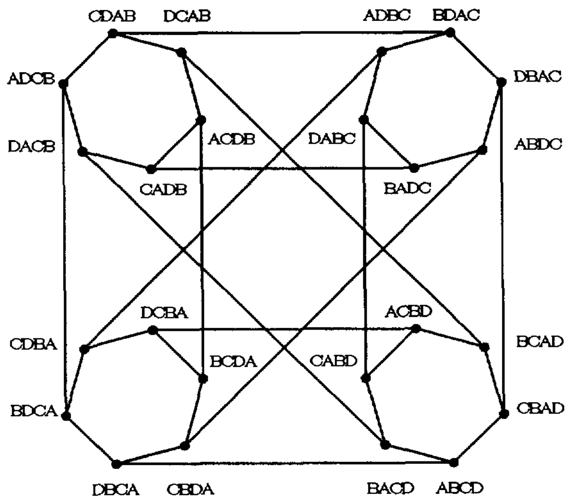
A 3-Dimension Cube

The boolean hypercube is another Cayley graph. Addresses in a hypercube are usually presented as bit vectors, where each bit represents a dimension in the cube, and the values taken as a binary integer represent the node address. One may form a hypercube as a Cayley graph by representing each dimension of the cube as a pair of unique letters. In any such pair there are two orderings of the letters. One ordering represents the bit value '1', the other represents '0'. The generators connect single dimensions by transposing the letters which represent that dimension. A three-cube might be represented by {*ABCDEF, BACDEF, ABDCEF, BADCEF, ABCDFE, BACDFE, ABDCFE, BADCFE*} and the generators {*213456, 124356, 123465*}. Other generators exist which are isomorphic to those given here.



A 24-Node Pancake Graph (Degree 3)

Dense vertex symmetric graphs may also be generated in this fashion. Two such graphs are the pancake graph and the star graph. Both graphs grow as the factorial of the degree (i.e. $(d+1)!$), as opposed to the hypercube which only grows exponentially (i.e. 2^d). Both graphs also have the advantage that they may be constructed recursively — a graph of degree d may be constructed from $d+1$ graphs of degree $d-1$. (A hypercube of degree d may also be formed recursively from 2 graphs of degree $d-1$.) The pancake graph is obtained by viewing the string of letters as a stack of pancakes to be flipped by a spatula. The spatula is inserted into the stack, and the pancakes above the spatula are inverted together. The inverted pancakes are placed once again on top of the original stack. The generators of a pancake graph of degree three would be {*2134, 3214, 4321*}. An exact formula for the diameter and average diameter of this graph not yet been determined, but a more complete discussion, with references, of the problem and its solution may be found in [4].



A 24-Node Star Graph (Degree 3)

The star graph of degree d may be created by selecting $d+1$ unique letters. Each node would be given a name which is a unique permutation of the letters. This yields a graph of $(d+1)!$ nodes. The d generators are formed by transposing the 1^{st} and i^{th} letters. Thus the generators of a star graph of degree three would be {*2134, 3214, 4231*}. The diameter and average diameter of this graph are known to be $\lfloor 3d/2 \rfloor$ and $d+H_{d+1}-3+2/(d+1)$, respectively, where H_n is the n^{th} harmonic number [5].

4. The Hypercube and The Star-Graph

Star graphs, because of their smaller degree, diameter, and average diameter, are an improvement in some respects over the more widely accepted binary n-cube. The message latency will often be better for the star graph because of its smaller average diameter. If a large number of packets is used in message transmission the difference in latency may be small.

A more significant difference is the communication resources available to each node. The ratio for the hypercube is

$$\frac{d/2}{d} = \frac{1}{2}$$

The ratio for the star graph is

$$\frac{d+H_{d+1}-3+\frac{2}{d+1}}{d} \text{ which simplifies to } 1+\frac{H_{d+1}-3+\frac{2}{d+1}}{d}$$

This value is somewhat more difficult to analyze than for the hypercube, but it is sufficient to note that the factor $H_{d+1}-3+2/d+1$ starts small and grows much more slowly than d. In fact, the whole formula climbs quickly to just above 1 then slowly converges back on 1. It first exceeds 1 for the 8-star (degree 8), and reaches its maximum value (~1.037) with the 23-star. The following table gives the formulas and their values for several moderately large networks.

Network	*Degree*	*Order*	*Average Diameter*	*Diameter*	*Avg. Diam. / Degree*
Hypercube	d	2^d	$\frac{d}{2}$	d	½
	7	128	3.5	7	½
	9	512	4.5	9	½
	12	4096	6.0	12	½
Star-Graph	d	$(d+1)!$	$d+H_{d+1}-3+\frac{2}{d+1}$	$\left\lfloor 3\frac{d}{2} \right\rfloor$	$1+\frac{H_{d+1}-3+\frac{2}{d+1}}{d}$
	4	120	~3.68	6	~0.920
	5	720	~4.78	7	~0.957
	6	5040	~5.88	9	~0.980

5. Conclusions

A measure has been presented here which appears to be a reasonable relative indicator of contention for network topologies. It does not attempt to predict the time lost to contention, nor the number of messages which will collide in a particular system. It does provide a basis for comparing topologies by comparing the number of links required by an "average" message. No empirical data has yet been accumulated which would verify its suitability as this type of measure.

According to this measure the hypercube topology has a relatively small amount of contention in message passing, primarily because of its large number of links. The star graph has a much smaller degree and as such would be less expensive to build than a hypercube of equal size. The message latency would also be lower. However, an average message in the star would use about twice the percentage of system links as it would on a hypercube, and that could increase the contention. This imbalance between the two topologies could be rectified by faster links on the star, but it is not clear if such a solution would be worth the additional cost. If the message contention of the intended application set were low, it certainly would not be worth reducing it by half by moving to the more expensive hypercube topology.

The star graph's small ratio makes it a much better contender for general purpose computing (non-topology specific applications) than, say, a grid, whose average diameter grows rapidly and degree remains constant. But whether a star graph or a hypercube is less expensive for the performance depends on whether it is less expensive to have many channels or faster links, which is a function of current technology.

6. References

[1] J.-C. Bermond, C. Delorme, and J.-J. Quisquater, "Strategies for Interconnection Networks: Some Methods from Graph Theory", *Journal of Parallel and Distributed Computing*, pp 433-449, Volume 3, Number 4, December 1986

[2] Sheldon B. Akers and Balakrishnan Krishnamurthy, "A Group Theoretic Model for Symmetric Interconnection Networks", *Proceedings of the 1986 International Conference on Parallel Processing*, pp 216-223, August 1986

[3] Steven P. Levitan, "Evaluation Criteria for Communication Structures in Parallel Architectures", *Proceedings of the 1985 International Conference on Parallel Processing*, pp 147-154, August 1985

[4] Sheldon B. Akers and Balakrishnan Krishnamutrhy, "Group Graphs as Interconnection Networks", *14th International Conference on Fault Tolerant Computing*, pp 422-427, Kissimmee, Florida, June 1984

[5] Donald E. Knuth, *The Art of Computer Programming, Volume 1*, Addison Wesley, 1973

[6] Tse-yun Feng, "A Survey of Interconnection Networks", *IEEE Computer*, pp 12-27, Volume 14, Number 12, December 1981

[7] Kai Hwang and Faye A. Briggs, "Computer Architecture and Parallel Processing", McGraw-Hill, New York, 1984