A Message Passing Model for Communication on Random Regular Graphs

Andreas Bieniek, Michael Nölle, Gerald Schreiber

Abstract— **Randomized routing strategies are applied to avoid worst-case communication patterns on communication networks used for parallel computers. Instead of a randomized routing strategy on a regular communication network, the randomization of the network itself is investigated in this paper. A model to describe properties of random regular graphs is presented. With the results of the model it is possible to predict the number of communication cycles necessary for the routing of a given communication load. Each node transmits the data in a buffered store and forward fashion. Different strategies for buffering messages in case of conflicts are incorporated into the stochastic model. The number of communication cycles can be predicted without simulating the communication traffic of the network.**

Keywords: *parallel computing, random graphs, randomized routing, performance analysis*

I. Introduction

Running a distributed algorithm with a regular communication pattern on a system with a regular communication network can produce – in some cases – a large number of communication conflicts. In this case the unevenly distributed communication load causes high congestion on some edges of the communication network. The communication time increases significantly compared to the average time for the same communication load. Looking at all possible permutations the possibility of a worst-case communication pattern is mostly very small, but some commonly used communication patterns produce them on typical communication networks of parallel computers. For example the transposition permutation of data blocks among the processors is a known worst-case communication pattern for butterfly networks [9].

Different analytical and randomized strategies to avoid worst-case communication patterns have been developed. If the communication pattern of the algorithm is known beforehand, the problem is equivalent to find an optimal embedding of the graph describing the communication pattern of the algorithm into the graph described by the communication network of the computer. Applying randomized methods usually guarantees good results for most cases with a small probability of producing worst-case patterns. The exact knowledge of the communication pattern of the algorithm is not necessary.

Valiant [13] published a randomized strategy where each processor chooses a random intermediate address for the packet in addition to the final destination. The packets are routed to their random intermediate address and thereafter to their final destination. Based on this idea many randomized routing methods have been developed for different network types. The most important criteria beside the time to complete the communication is the maximum number of messages buffered at a node in case of conflicts. An overview of randomized routing methods is given in [10], [8]. The drawback of methods based on sending the messages to random directions at some stages is the increased path length for most messages.

A different solution is to distribute the global memory shared by some processors in a pseudo-random fashion. A regular access pattern to the global memory distributed among the processors is converted into irregular communication patterns in the network. The disadvantage is the increased communication load for most algorithms. If the random distribution disturbs the ordering or neighborhood relation of the data items utilized by the algorithm, unnecessary communication load is produced.

Another strategy is to build pseudo random properties into the network structure of the communication hardware as proposed by Leighton et. al for randomly-wired splitter networks [9]. In [9] the randomness is added in such a way that the well known $\log N$ destination tag routing algorithm for butterfly net-

This research was partially supported by the "Deutsche Forschungsgemeinschaft" (DFG) within the project *Paralleles Rechnen in der digitalen Bildverarbeitung und Mustererkennung* and the European Community (EC) within the project *Nonlinear Adaptive Techniques (NAT)* at the Technical University of Hamburg-Harburg.

Andreas Bieniek, Michael Nölle and Gerald Schreiber are with the Technische Universität Hamburg-Harburg, Technische Informatik I, Harburger Schloßstraße 20, D-21071 Hamburg, Germany. email: [bieniek, noelle, g-schreiber]@tu-harburg.d400.de

works can be applied. Compared to a 2-dilated butterfly the fault tolerance is improved using the same amount of hardware. For general permutations it outperforms the 2-dilated butterfly network. The transpose problem, a worst case permutation for dilated butterfly networks, is not a worst case permutation for randomly-wired splitter networks.

Staring from this idea we investigate the properties of random regular graphs as communication networks. In section II graph properties like the average distance of two nodes, the distribution of node distances or bounds for the diameter are estimated for random r-regular graphs of given degree and size. In comparison to our model, the "Theory of random graphs" deals with asymptotic proofs for graph properties, increasing the number of nodes to infinity [5], [4].

In section III the results are connected with a message passing model to show the potential of random graphs as a topology for parallel computers. A stochastic model to describe the communication on random networks is presented. Different strategies for buffering messages in case of conflicts are incorporated. The results can be obtained without performing simulations of the communication network.

The theoretical results are verified on a transputer based Parsytec SC-128 configured to 4-regular random graphs of different size.

II. Random graph model

The communication network of a parallel computer can be described as a graph $G(V, E), V \in \mathbb{Z}_N = \{1, 2, \ldots, N\}$. The processing nodes are represented by the node set V of the graph and the edges E of the graph represent the edges of the communication network. In the paper we consider only bidirectional communication channels. Therefore an edge represents a bidirectional communication facility connecting two nodes. The degree of a node is the number of edges connected with the node. A graph is called regular or r-regular iff all nodes have a constant degree r.

To obtain the desired graph properties the path trees of random regular graphs are considered. A path tree from root node $R \in \mathbb{Z}_N$ includes all cycle free paths from R to the other nodes. The root node itself is on level 0 of the tree and all paths with distance i to R are forming level i of the tree. A walk tree includes all possible walks starting from R. In a walk tree cycles are allowed and can be traversed more than one time.

The goal is to approximate the number of nodes in each level of the path tree. With our approximations made for the stochastic model the direct correspondence of the trees and random regular graphs is lost. Generally it is not possible to construct a regular graph from a tree following our stochastic experiment of the model. With a small probability it is even possible that a node is not in the tree at all. Nevertheless the predicted graph properties are close to the measured values of the tested random regular graphs.

Consider a walk tree starting from an arbitrary root node R. In any r-regular graph a walk tree from root node R can be build in the following way:

Let $S(X)$ denote the successor node within the tree. There are r successor nodes $S_j(R) \in \mathbb{Z}_N \backslash R, j = 1, \ldots, r$ connecting the root node with level 1 of the tree. From any node X^i of level $i = 1, \ldots N-2$ there are $r-1$ successor nodes $S_j(X^i) \in \mathbb{Z}_N \backslash X^i, j = 1, \ldots, r-1$ to level $i+1$ of the tree.

For any r-regular graph the number of nodes of level $i = 0 \ldots N-1$ of a walk tree is:

$$\begin{aligned}
\tilde{K}_0 &= 1 \\
\tilde{K}_1 &= r \\
\tilde{K}_2 &= r(r-1) \\
\tilde{K}_i &= (r-1)\tilde{K}_{i-1} = r(r-1)^{i-1}. \qquad (1)
\end{aligned}$$

The path trees of random regular graphs are approximated with the following experiment:

The tree is created level by level starting from root node R. The successor nodes $S(X_{i-1})$ of level $i-1$ form the nodes of level i. All successor nodes are selected independently with probability $1/N-1$ from $\mathbb{Z}_N \backslash X_{i-1}$. This resembles the idea that for any selected node of the random graph the neighbor nodes are not predictable. Our simplification that the experiment is viewed statistically independent for each occurance makes it impossible to construct a regular graph from such a tree in general [1].

For each $S(X_{i-1})$ we test if the node number is contained in the path from R to X_{i-1}, excluding X_{i-1}. Selecting the new node from $\mathbb{Z}_N \backslash X_{i-1}$ and excluding X_{i-1} takes the fact into account that the graph is loop-less.

In each new level the total number of nodes has to be corrected subtracting the number of nodes creating cycles. On the path from R to X_{i-1}, excluding X_{i-1}, there are $i-1$ distinct nodes from the set $\mathbb{Z}_N \backslash X_{i-1}$. Therefore the probability that a new node $S(X_{i-1})$ on level i creates a cycle is $\frac{i-1}{N-1}$.

[1] Please note that a node can occur multiple time in the path tree, having different neighbors each time

Modeling the path tree in this way the expected number of nodes K_i in level i is given by:

$$\begin{aligned} K_0 &= 1 \\ K_1 &= r \\ K_i &= (r-1)K_{i-1}(1-\frac{i-1}{N-1}) \\ &= r(r-1)^{i-1}\prod_{j=1}^{i-1}(1-\frac{j}{N-1}),\ i>1. \end{aligned} \quad (2)$$

Fig. 1 shows an example of a path tree of a 3-regular graph with node 1 as the root node. The nodes creating a cycle are shaded and have to be removed to get the path tree. To calculate the average and maximum length of the shortest paths, a destination node $D \in \mathbb{Z}_N$ is selected randomly. $P_i(D)$ is the probability to find node D at least one time in level i and not to find D in all levels $< i$. Please note that the average length of the shortest path to D can be determined with $P_i(D)$.

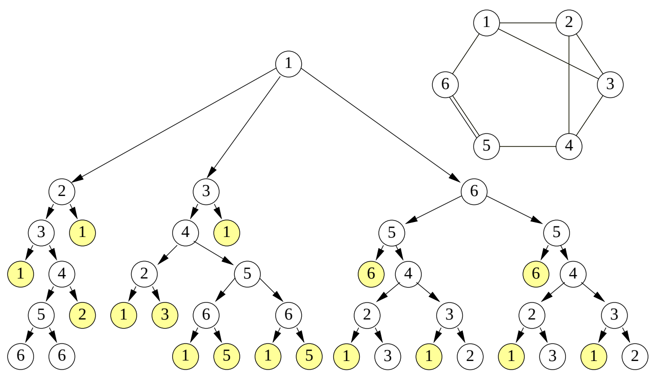

Fig. 1. Paths from node 1 to all other nodes in a 3-regular graph. The shaded nodes form cycles and have to be removed to get the path tree.

The root R is selected from $\mathbb{Z}_N$ while all other nodes of the path tree are selected from $\mathbb{Z}_N \backslash R$. Therefore the probability that D equals R is $\frac{1}{N}$. For the other nodes the probability that D appears x times in level i with K_i nodes in can be viewed as an Bernoulli experiment with probability $\frac{1}{N-1}$ of success in each trial and K_i trials:

$$\begin{aligned} \widetilde{BV}_i(x) &= BV_{(K_i,\frac{1}{N-1})}(x) \\ &= \binom{K_i}{x}(\frac{1}{N-1})^x(1-\frac{1}{N-1})^{K_i-x}. \end{aligned} \quad (3)$$

With equation (3) the probability $\tilde{P}_i(D)$ that there is no path of length i from R to D is equal to the case that level i does not contain D:

$$\begin{aligned} \tilde{P}_0(D) &= 1-\frac{1}{N} \\ \tilde{P}_i(D) &= \widetilde{BV}_i(0) = (1-\frac{1}{N-1})^{K_i},\ i>0. \end{aligned} \quad (4)$$

With equation (4) we derive:

$$\begin{aligned} P_0(D) &= 1-\tilde{P}_0(D) = \frac{1}{N} \\ P_i(D) &= (1-\tilde{P}_i(D))\prod_{j=0}^{i-1}\tilde{P}_j(D) \\ &= (1-\frac{1}{N})(1-(1-\frac{1}{N-1})^{K_i})\cdot \\ &\quad (1-\frac{1}{N-1})^{\sum_{j=1}^{i-1}K_j} \\ &= (1-\frac{1}{N})((1-\frac{1}{N-1})^{\sum_{j=1}^{i-1}K_j} - \\ &\quad (1-\frac{1}{N-1})^{\sum_{j=1}^{i}K_j}),\ i>0. \end{aligned} \quad (5)$$

The average number of nodes with distance i from the root node R is $NP_i(D)$. With this result and the message passing model of section III it is possible to predict the communication cycles on random regular graphs for different communication loads and buffering strategies.

A. *Average path length*

The average path length of random r-regular graphs approximated by the tree model is the expected value E^p of the probability distribution $P_i(D)$ over all possible path lengths $i, i = 0, \dots N-1$:

$$E^p = \sum_{i=0}^{N-1} iP_i(D) = \sum_{i=1}^{N-1} iP_i(D). \quad (6)$$

Fig. 3 compares the expected path length E^p with the measured average path length from implemented graphs. In Fig. 4 the average number of nodes with distance i, i.e. $NP_i(D)$, is compared.

B. *Diameter*

In the "Theory of random graphs" almost every graph has property P if the probability that a graph does not have property P tends to 0 with $n \to \infty$ ([3], [5]). The following asymptotic bounds for the diameter of random r-regular graphs are proven in [4], [5] (cf. Fig 5):

Theorem 1: Let $r \geq 3$ and $\epsilon > 0$ be fixed and let $d = d(n)$ be the least integer satisfying

$$(r-1)^{d-1} \geq (2+\epsilon)rn\log n.$$

Then almost every r-regular graph of order n has diameter at most d.

Theorem 2: For a fixed natural number $r \geq 3$, the diameter of almost every r-regular graph of order n is at least

$$\lfloor \log_{r-1} n \rfloor + \lfloor \log_{r-1} \log n - \log_{r-1}\{6r/(r-2)\} \rfloor + 1.$$

Care has to be taken when results which are proved asymptotically are applied to small or medium sized graphs [5].

In contrast to the bounds given in theorem 1 and 2, the tested random graphs show a small spread in the diameter which is also observed in [5], chapter XVI.

For random regular graphs the estimation of the diameter with the stochastic model developed in this paper is based on the probability $P_{\leq d}$ that *one* shortest path is equal or smaller than d. To get a reliable boundary for the diameter of the graph the fact that there are at least $o(N^2)$ shortest paths in the graph has to be taken into account. There are $o(N^2)$ trials with probability $P_{\leq d}$ that the path length is smaller or equal to d. The probability $1 - P_{\leq d}$ for a path with a distance greater than d has to be adapted to the size of the graph:

$$\begin{aligned} P_{\leq d} &= \sum_{i=0}^{d} P_i(D) = \frac{1}{N} + \sum_{i=1}^{d} P_i(D) \\ &= \frac{1}{N} + (1 - \frac{1}{N})(\sum_{i=1}^{d}(1 - \frac{1}{N-1})^{\sum_{j=1}^{i-1} K_j} \\ &\quad - \sum_{i=2}^{d+1}(1 - \frac{1}{N-1})^{\sum_{j=1}^{i-1} K_j}) \\ &= \frac{1}{N} + (1 - \frac{1}{N})((1 - \frac{1}{N-1})^{\sum_{j=1}^{0} K_j} \\ &\quad -(1 - \frac{1}{N-1})^{\sum_{j=1}^{d} K_j}) \\ &= \frac{1}{N} + (1 - \frac{1}{N})(1 - (1 - \frac{1}{N-1})^{\sum_{j=1}^{d} K_j}) \\ &= 1 - (1 - \frac{1}{N})(1 - \frac{1}{N-1})^{\sum_{j=1}^{d} K_j}. \qquad (7) \end{aligned}$$

For $d = N - 1$ the probability $1 - P_{\leq N-1}$ is the probability that the chosen node is not in the graph at all. It is a result from the simplification that the random choice of the successors $S(X_i)$ for each node X_i is viewed as a statistically independent experiment. The probability $1 - P_{\leq N-1}$ rapidly tends to 0 with increasing size of the graph.

C. Results

The theoretical results are compared with measurements on are transputer based Parsytec SC-128 configured to 4-regular random graphs of different size. For each graph size the average of 10 configured graphs are taken.

Fig. 3 shows that the expected distance E^p is a close estimate for the measured average distance of the implemented random graphs. The average distance of the random graphs is smaller compared to a de Bruijn graph of the same size and degree. The de Bruijn graph belongs to the class of graphs with logarithmic diameter [6], [8].

In Fig. 4 the expected number of nodes with distance i is compared to measured distance distributions of different graph types. The average number of nodes with a shortest path of length l to a selected node is drawn over the path length. The expected value is close to the measured distances distribution of the implemented random graphs. The model gives also a reasonable good estimate for de Bruijn graphs for at least 64 nodes.

The distribution of the node distances is a basis for the communication model presented in section III. Since the distribution of the node distances of random graphs is also an approximation for de Bruijn graphs, the number of cycles necessary for random communication patterns on de Bruijn graphs can be predicted with the random model as well.

The variation of the average node distance with the degree of the random graphs is shown in Fig. 6. With increasing degree the decrease of the average node distance is most significant for sparse graphs and gets less effective for dense graphs.

In Fig. 5 the asymptotic bounds of theorem 1 and 2 are compared with bounds from equation (7) with a limit for $1 - P_{\leq d}$ of 10^{-2} and 10^{-6} and the measured diameter of de Bruijn Graphs and implemented 4-regular random graphs. It shows that the 10^{-2} bound is exceeded by larger implemented random graphs while the 10^{-6} bound still holds. This demonstrates the need to adapt the probability $1 - P_{\leq d}$ to the $o(N^2)$ increase in the number of shortest paths.

III. Message passing model

In this section a communication model for random regular graphs is presented. With the model it is possible to predict the number of communication cycles for different communication loads and buffering strategies.

This model is independent from the model presented in section II except that initially the messages are grouped according to the distance distribution of random regular graphs (5). The results can be used to approximate the number of communication cycles to route random communication patterns on de Bruijn graphs since the difference in the distance distribution is small (Fig. 4).

The main advantage of the model is the quantitative estimation of communication cycles. The buffering of messages in case of conflicts is incorporated into the stochastic model without the need to simulate the communication traffic of the network.

In the model all messages are send to their destination on a shortest path. In each synchronized communication cycle at most one message can be transferred to each neighbor node. For each link a queue is maintained to buffer messages in case of conflicts.

To get the number of communication cycles all messages in the system are grouped according their distance to the destination. The idea of grouping the messages according to their remaining distance was also applied in [7] to show communication properties of single stage interconnection networks. Initially the total number of messages to transmit are grouped according to the distance distribution of random regular graphs (5).

In each synchronized communication step the number of messages in each group are updated according the probability of being transmitted. Messages which are communicated are transferred to the group with distance decreased by one. Messages which are not transmitted are kept in the same group. The probability of being transmitted depends on the number of messages in every group and the strategy of selecting the messages from the local buffers. Fig. 2 shows how the messages are grouped and the possible transitions. Assuming an N node undirected r-regular

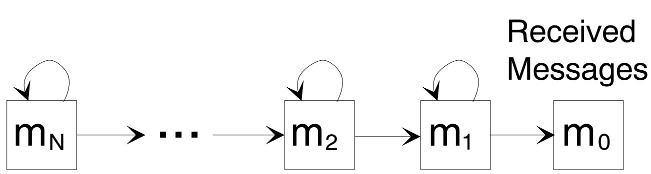

Fig. 2. Messages with the same remaining distance to the destination are one group. In each transmission cycle some messages are transmitted to the group with distance decreased by one. Messages which are not transmitted are kept in the same group.

random graph as a communication network, there are rN communication facilities in the network. At communication cycle $t = 0, 1, \ldots$ there are m_i^t messages with distance $i, i = 0 \ldots N-1$ to the destination. Initially for $t = 0$ the M message to transmit are grouped according the probability of having a shortest paths of length i. Therefore $m_i^0 = MP_i(D)$. The number of communication cycles is the number of updates until the average number of messages in group m_0^t for the received messages is close to M.

The probability that a selected message is in the buffer of a given communication facility is $1/rN$. For m_i messages in the network, the probability that k messages are in the buffer of a communication facility can be described with the binomial distribution:

$$BV_{(m_i^t, \frac{1}{rN})}(k) = \binom{m_i^t}{k} (\frac{1}{rN})^k (1 - \frac{1}{rN})^{m_i^t - k}. \tag{8}$$

The probability P^E that none of the m_i^t messages is in the buffer of a communication facility is:

$$P_i^E = BV_{(m_i^t, \frac{1}{rN})}(0) = (1 - \frac{1}{rN})^{m_i^t}. \tag{9}$$

With a total number of m_i^t messages with remaining distance i to the destination, the probability P_i^T that there is at least one message with remaining distance i in the buffer of a selected communication facility is:

$$P_i^T = 1 - P_i^E = 1 - (1 - \frac{1}{rN})^{m_i^t}. \tag{10}$$

Only one of the messages waiting in the buffer of a communication facility can be transmitted in a communication cycle. Three strategies to select the message to be transmitted are investigated. One is to select the message with the shortest remaining distance to the destination (SRPF). The second strategy is to select the message with the longest remaining distance to the destination (LRPF) and the last strategy we investigate is to use a FIFO buffer at each communication facility (FIFO).

The following equations incorporate the buffering strategies into the model. The updated number of messages in each group after a communication step are given according to the previous number of messages in each group.

A. Shortest remaining path first

With the SRPF strategy a messages with distance i to the destination is transmitted iff there is a message with distance i and no message with distance $< i$ in the buffer of the communication facility. In every transmission cycle the number of messages m_i^t in the network with a remaining distance $i, i = 0 \ldots N-1$ to the destination are modified in the following way:

$$m_0^{t+1} = m_0^t + rNP_1^T$$

$$
\begin{aligned}
m_1^{t+1} &= m_1^t + rN(1-P_1^T)P_2^T - rNP_1^T \\
m_2^{t+1} &= m_2^t + rN \cdot \\
&\quad \left[(1-P_1^T)(1-P_2^T)P_3^T - (1-P_1^T)P_2^T\right] \\
m_i^{t+1} &= m_i^t + rN\left[P_{i+1}^T(1-P_i^T) - P_i^T\right] \cdot \\
&\quad \prod_{k=1}^{i-1}(1-P_k^T). \qquad (11)
\end{aligned}
$$

B. Longest remaining path first

With the LRPF strategy the message with the longest remaining distance to the destination is transmitted first. A message with distance i is selected iff there is no message with distance $> i$ in the buffer. For each transmission cycle the number of messages in each group is modified in the following way:

$$
\begin{aligned}
m_{N-1}^{t+1} &= m_{N-1}^t - rNP_{N-1}^T \\
m_{N-2}^{t+1} &= m_{N-2}^t + rN\left[P_{N-1}^T - (1-P_{N-1}^T)P_{N-2}^T\right] \\
m_i^{t+1} &= m_i^t + rN\left[P_{i+1}^T - (1-P_{i+1}^T)P_i^T\right] \cdot \\
&\quad \prod_{k=i+2}^{N-1}(1-P_k^T) \\
m_0^{t+1} &= m_0^t + rNP_1^T\prod_{k=2}^{N-1}(1-P_k^T). \qquad (12)
\end{aligned}
$$

C. First in first out

To model the FIFO strategy, it is assumed that all $\sum_{k=1}^{N-1} m_k^t$ messages in the system are distributed with equal probability over the rN FIFO's of the system communication facilities. No difference according to the remaining distance is made for the selection of the message to transmit. For a selected FIFO the probability P^E that there is no message for transmission is:

$$
P^E = (1-\frac{1}{rN})^{\sum_{i=1}^{N-1} m_i^t}. \qquad (13)
$$

The probability that a FIFO is not empty and a message can be transmitted in the next transmission cycle is:

$$
P^T = 1 - P^E = 1-(1-\frac{1}{rN})^{\sum_{i=1}^{N-1} m_i^t}. \qquad (14)
$$

We assume that the order of messages in the FIFO is not predictable. The probability to select a message with distance i to the destination is proportional to m_i^t in relation to the total number of messages which have not reached their destination. The number of messages transmitted in each group is modified in the following way:

$$
\begin{aligned}
m_0^{t+1} &= m_0^t + \frac{rNP^T}{\sum_{k=1}^{N-1} m_k^t} m_1^t \\
m_i^{t+1} &= m_i^t + \frac{rNP^T}{\sum_{k=1}^{N-1} m_k^t}\left[m_{i+1}^t - m_i^t\right]. \qquad (15)
\end{aligned}
$$

IV. Experimental results

A comparison of the different strategies shows (Fig. 7, 8 and 9) that in all cases the Longest Remaining Path First strategy (LRP-First) is superior to the FIFO and the Shortest Path First (SRP-First) strategy.

The difference is more significant with increasing size of the network (Fig. 8). Sending an average of 2 messages per node on a random 4-regular graph with less than 100 nodes, the number of transmission cycles can be reduced by no more than 1 cycle with the LRP-First strategy in comparison to the simple FIFO strategy. For medium sized graphs it shows that with a given amount of hardware it is more advantageous to reduce the cycle time instead of spending the hardware implementing the LRP-First strategy.

Fig. 7 shows the case of a constant graph with 64 nodes and increasing number of messages send per node. In the case of low network load the number of transmission cycles is almost constant. With increasing number of messages per node the network saturates with the result of a linear increase of the transmission cycles. The saturation is also the reason for the linear increase of the transmission cycles for the "total exchange" or "multi-node scatter" operation on random graphs with constant degree (Fig. 9). In the "multi-node scatter" operation every processor sends a different message to all other nodes.

The most important result is the following: Considering an algorithm where each communication cycle can be performed in a constant number of steps on a graph G with maximum degree d. If the nodes used by the algorithm are embedded randomly in a r-regular random graph, the average number of transmission cycles to simulate one communication cycle of the algorithm can be predicted with the model assuming d messages are send by each node. Because of the random embedding the possibility of a worst-case communication pattern is very small. If the maximum degree d is a constant with the size of the graph, the number of transmission cycles to simulate one transmission cycle of the algorithm increases logarithmically with the number of nodes (Fig. 8). The buffering of messages and the congestion is taken into account by the model.

In Fig. 10 the communication time on a de Bruijn graph B is compared with the time performing the same communication a random graph G with the same degree and size. Graph B is embedded randomly into graph G. Each node of B sends a message to 2 adjacent nodes. Asynchronous message passing is implemented [11] on the configurable transputer-based system with up to 128 nodes. The messages are send as one packet in a store and forward mode on a shortest path. Before each measurement the system is synchronized. For each size the average and maximum communication time over 10 different random graphs is measured. For the estimation the communication time for the de Bruijn graph, representing on communication cycle, is multiplied with the number of communication cycles predicted by the model. The results show that the predicted average communication time is between the measured average and maximum communication time.

The practical use of random graphs is demonstrated with two sorting algorithms integrated into the parallel image processing system PIPS [12]. The first algorithm corresponds to the Batchers sorting circuit (BS) [1] and runs optimal with constant dilation on a de Bruijn graph B. The second sorting algorithm is called *Shuffle Ring Sort (SRS)* [2]. It communicates with constant dilation on a shuffle ring graph (SRG), a combination of a shuffle exchange and a ring graph. In Fig. 11 the efficiencies of the sorting algorithms performed on random graphs are compared with the efficiencies on the optimal graphs. The results show that additional overhead of the embedding into the random graph is small.

V. Conclusion

In this paper we have shown that random regular graphs are suitable as a communication network for parallel computers. The estimated graph properties for random regular graphs like the average distance between two nodes and the estimated distribution of the node distances are close to the measured values of the implemented graphs. The stochastic communication model makes it possible to predict the average number of synchronized communication steps. In our experiments the synchronized communication model is a good estimate for the communication time of an asynchronous message passing system. Two implemented sorting algorithms demonstrate the practical use of random regular graphs as a communication network. In many cases the additional overhead caused by the random embedding of the communication graph can be hidden.

The approximation made using the path trees as a superset of random regular graphs leaves the question of proved bounds for this approximation open. Future work has to be done in this area. Moreover it will be promising to model and verify the standard deviation from the estimated average case and the memory necessary for buffering messages under different communication loads.

References

[1] K.E. Batcher. Sorting networks and their applications. In *Proc. AFIPS 1968 Spring Jt. Computer Conf.*, volume 32, pages 307–314, Washington, D.C., 1968. Thompson Book,Co.

[2] A. Bieniek, M. Nölle, and G. Schreiber. A regular N-node bounded degree network for sorting N^2 items with optimal speedup. In *Proc. of the International Workshop on Parallel Image Analysis (IWPIA '95)*, Lyon, France, December 1995.

[3] B. Bollobás. The diameter of random graphs. *Transactions of the ameerican mathematical society*, 267(1):41–51, September 1981.

[4] B. Bollobás. *The diameter of random regular graphs*, pages 125–134. 1982.

[5] B. Bollobás. *Random Graphs*. Academic Press, 1985.

[6] N.G. de Bruijn. A combinatorial problem. In *Proceedings of the section of science*, pages 758–764. Appl. Mathematical Science, Koninklije Nederlandse Academie van Wetenshapen, 1946.

[7] J.R. Burke, C. Chen, T.-Y. Lee, and D.P. Agrawal. Performance analysis of single stage interconnection networks. *IEEE Transactions on Computers*, 40(3):357–365, March 1991.

[8] F.T. Leighton. *Introduction to Parallel Algorithms and Architectures*. Morgan Kaufman Publishers, San Matio, California, 1992.

[9] F.T. Leighton and B.M. Maggs. Fast algorithms for routing around faults in multibutterflies and randomly-wired splitter networks. *IEEE Transactions on Computers*, 41(5):578–587, May 1992.

[10] T. Leighton, B. Maggs, and S. Rao. Universal packet routing algorithms. In *29th Annual Symposium on Foundations of Computer Science*, pages 256–269, October 1988.

[11] M. Nölle and J. Kliewer. Meps rev. 5.0 – A Parallel Message Passing System. Interner Bericht 11/94, Technische Informatik I, TU-HH, September 1994.

[12] M. Nölle, G. Schreiber, and H. Schulz-Mirbach. PIPS–a general purpose Parallel Image Processing System. In G. Kropatsch, editor, *16. DAGM - Symposium "Mustererkennung"*, Wien, September 1994. Reihe Informatik XPress, TU-Wien.

[13] L.G. Valiant and G.J. Brebner. Universal schemes for parallel communication. In *Proc. 13'th ACM Symp. on Theory of Computing*, pages 263–277, May 1981.

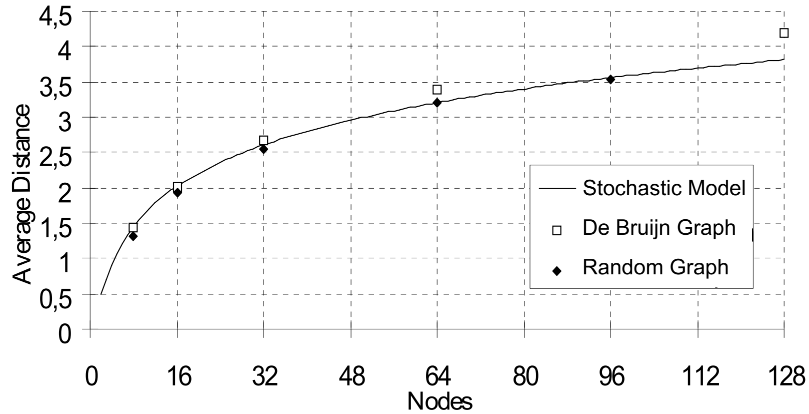

Fig. 3. Measured distance compared with model

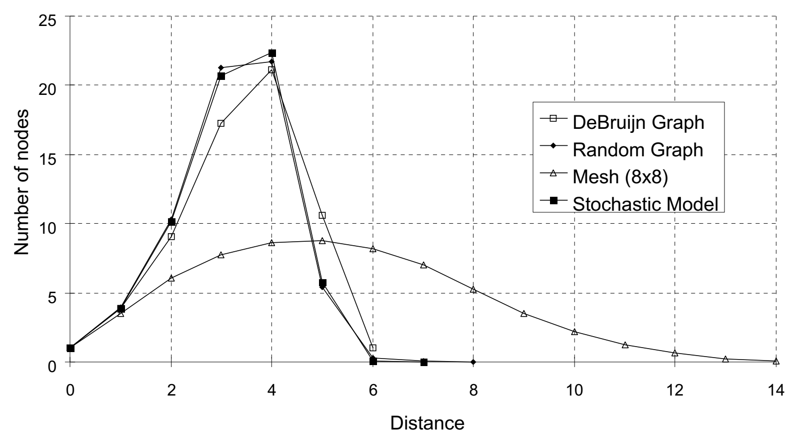

Fig. 4. Number of nodes with distance d, averaged over all nodes of the graph. Overall number of nodes is 64 for all graphs. For the random graph the average over 10 implemented graphs is taken.

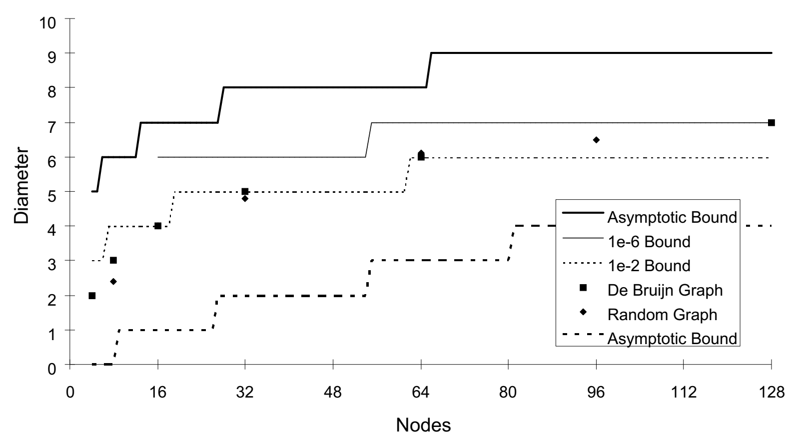

Fig. 5. Measured diameter compared with upper bound and lower bound for random 4-regular graphs.

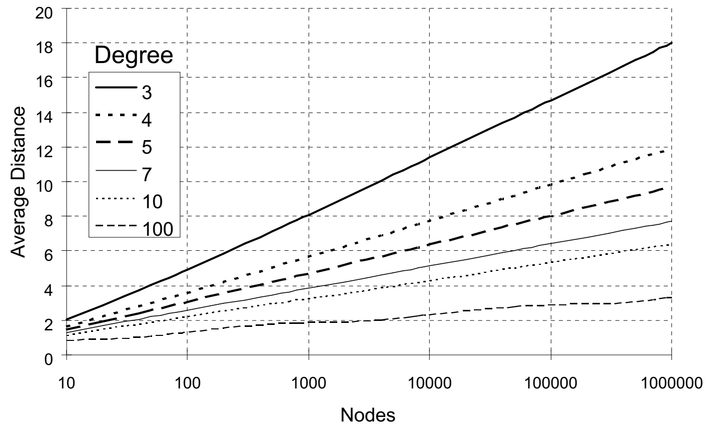

Fig. 6. Modeled distance for random graphs with varying node degree and number of nodes

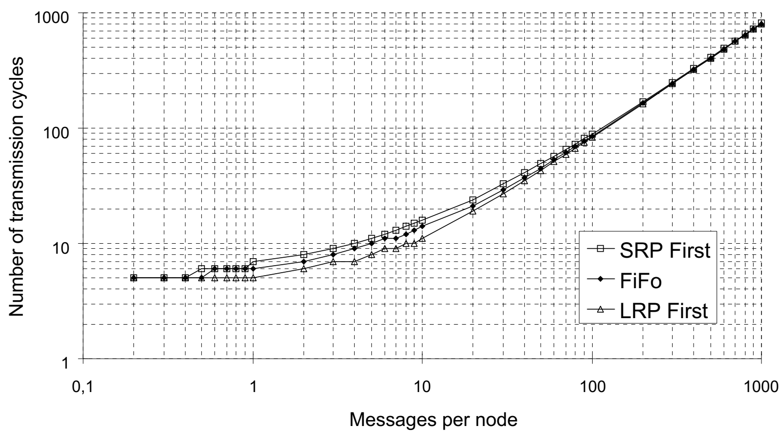

Fig. 7. Number of transmission cycles needed for a 64 nodes random graph with 4 links per node over the average number of messages send from each node to a random destination, using different routing strategies

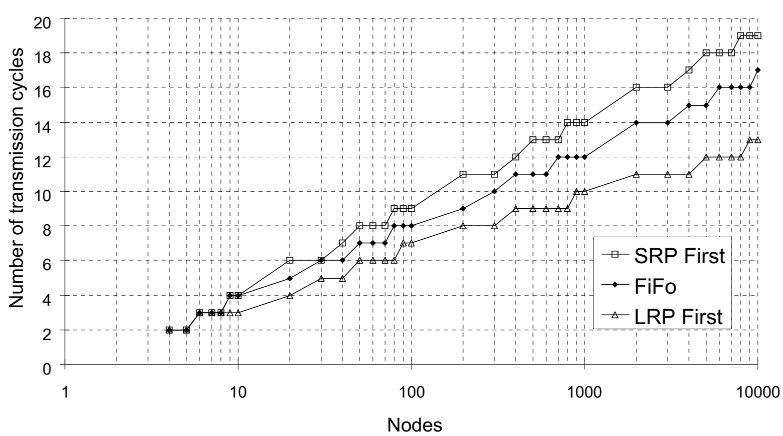

Fig. 8. Number of transmission cycles needed for random graphs with 4 links per node and an average of two messages to transmit per node.

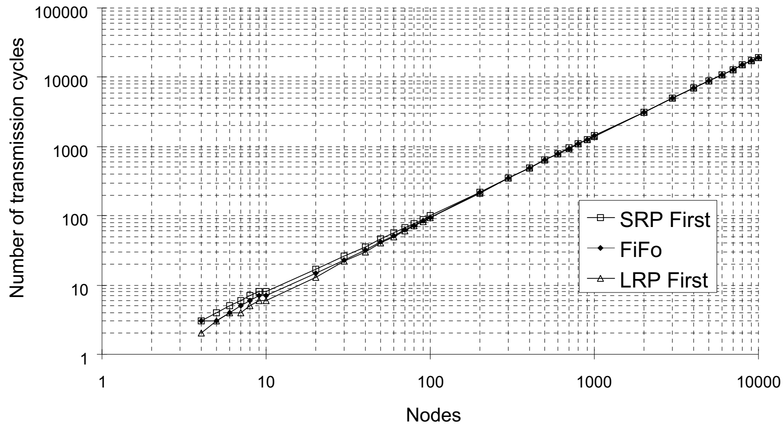

Fig. 9. Number of transmission cycles needed for multi-node scatter operation in random graphs with 4 links per node.

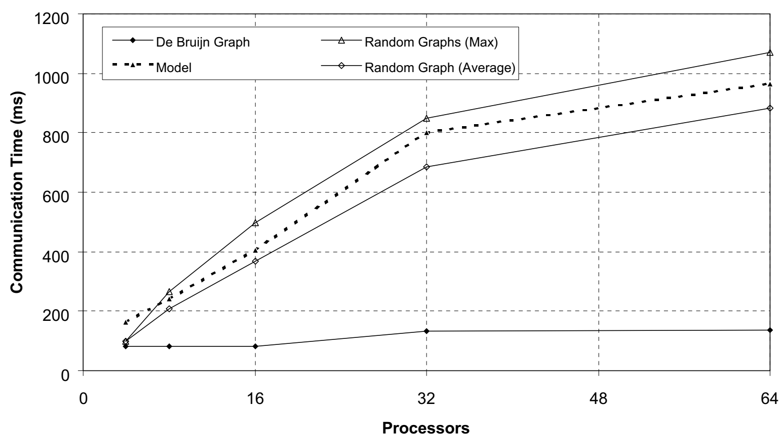

Fig. 10. Predicted transmission time compared with measured time. Two message of 10^5 byte each are send on two edges of the (embedded) de Bruijn graph.

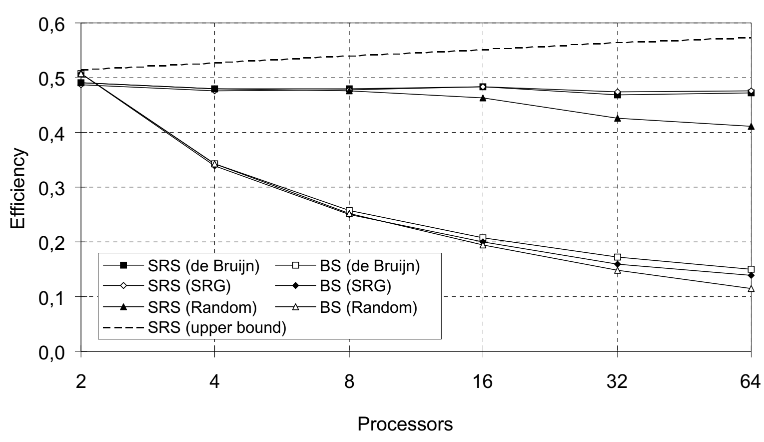

Fig. 11. Biton sort and SRS sort algorithm on dedicated and random graphs, sorting 2^{17} keys per processor.