Routing on Asyncronous Processor Networks *

Efstratios Karaivazoglou and Friedhelm Meyer auf der Heide **

Department of Mathematics and Computer Science and Heinz Nixdorf Institute,
University of Paderborn, 33102 Paderborn, Germany

Abstract. In this work we present models and runtime measures for routing in asynchronous networks. We try to construct them in a way that they can be both realistic and easy to work with. For some of the models presented here variants of techniques used in the analysis of synchronous routing, like the *delay sequence argument*, can be adapted. On the other hand, for others we can only prove large upper bounds for any routing protocol. However, we present a model for which it seems possible to get better than trivial upper bounds, although known proof techniques (like the delay sequence argument) cannot be applied.

1 Introduction

Most theoretical analysis of routing protocols focusses on the synchronous network model. Under this model, all nodes and links of the network see a common clock, while the time is measured in discrete steps. An atomic piece of a message (called packet or flit depending on the routing policy) can be transmitted over a link in one time step. Messages are transmitted synchronously.

The assumptions made under the synchronous network model obviously simplify the efforts for theoretical analysis. For real networks, however, these assumptions do not hold. On large networks it is technologically impossible to provide synchronization via a global hardware clock. Furthermore, routing switches may work at different speeds due to the load that passes through them. Additionally, not all links may support the same bandwidth and transmission latency.

In this paper we propose several models for asynchronous routing, together with measures for routing time. For some of them, variants of known protocols and their time analyses can be used. For other models we only present weak upper bounds. It seems that such models require new types of protocols and new methods for their analysis.

1.1 Known Results (Synchronous network model)

Let G denote a network with N processors. The links are directed, each containing a buffer to store packets. In this paper we focus on *oblivious routing*

[1] *Partially supported by DFG-Graduiertenkolleg "Parallele Rechnernetzwerke in der Produktionstechnik", ME 872/4-1, by the EC Esprit Long Term Research Project 20244 (ALCOM-IT), and by DFG-Sonderforschungsbereich 376 "Massive Parallelitaet: Algorithmen, Entwurfsmethoden, Anwendungen".*

[2] karaiw@hni.uni-paderborn.de, fmadh@uni-paderborn.de

strategies. This means that, for each pair i, j of nodes of G, a shortest path $p_{i,j}$ is predefined along which each message from i to j has to travel. A routing problem is described by a (multi-) set $\mathcal{R}$ of source-destination-pairs (i, j) of nodes of $\mathcal{G}$. For fixed $\mathcal{R}$, the *congestion*, i.e., the maximum number of paths $p_{i,j}, (i, j) \in \mathcal{R}$, passing through the same edge, and the *Dilation*, i.e., the maximum length of the $p_{i,j}$'s, $(i, j) \in \mathcal{R}$, are well defined. $\mathcal{G}$ is *levelled*, if its nodes are partitioned into disjoint sets $V_o, \ldots, V_D$, such that edges only exist between V_i and $V_{i+1}, i = 0, \ldots, D-1$. In this case, packets are only routed from V_o to V_D. (Thus the dilation is D.)

Many theoretical results exist for the synchronous network model. Leighton, Maggs and Rao [LMR88] show that any oblivious routing problem can be routed off-line in time $O(C+D)$, using constant-size link buffers. Their proof shows only the existence of the optimal schedule. In [LM95], Leighton and Maggs present an algorithm for finding the optimal schedule. Still the running time of the algorithm is polynomial in the number of packets and links, so it can not be applied to turn the off-line protocol into an efficient on-line protocol.

Our results for asynchronous routing should be compared to the following results on (randomized) oblivious synchronous routing:

Random rank protocol [Lei92]. This protocol works in levelled networks with unbounded buffers.
Ranade's protocol [Ran91]. It can be applied on levelled networks with bounded buffers of size at least one.
Growing rank protocol [MV95]. This protocol works in arbitrary networks with unbounded buffers.

All protocols route messages according to the routing problem R in time not exceeding O(C+D+logN), with high probability. For their analysis, variations of the *delay sequence* argument, developed by [Ale82] and [Upf84], are used.

It is known (compare e.g. [Lei92]) that the congestion can be very large in the worst case ($\Omega(\sqrt{N})$) for permutation routing in any bounded degree network). On the other hand, for many important networks, the congestion is small for almost all routing problems (e.g., $O(h \cdot D)$ for almost all h-functions in symmetric bounded degree networks, see [MV95]). Using a trick proposed by Valiant (see [Lei92]), the above bounds on routing random routing problems can be turned into routing arbitrary routing problems, e.g., every h-relation can be routed on a bounded degree symmetric network in time not exceeding $O(h \cdot D)$, with high probability.

1.2 Known results (Asynchronous network model)

Most of the research in this area is focused on developing deadlock-free robust protocols, like the ones presented in [Dua93], [DS87]. The only time-complexity results that we were able to find are due to Mansour and Patt-Shamir [MP91]. Their notion of asynchrony is as follows: A link has an arbitrary transmission latency t, where $0 < t \leq 1$. They present time complexity results and estimations about the throughput of the network for the synchronous network model

and show that the proofs can be adapted for their asynchronous model. Several models have been proposed however for asynchronous shared memory machines, like the APRAM model [CZ89] or the Asynchronous Shared Memory Model [Lyn96]. Lynch, in [Lyn96], also proposes a model for asynchronous networks. The author proves the fairness and correctness of this model and presents also some time complexity results for several well-known algorithms adapted to work on an asynchronous network.

2 Models and results

2.1 Random latency model

We model the latency of nodes by independent, identically distributed random variables with a known distribution. The intuition behind this is that the nodes are identical. Thus their latency may vary, but in a simple way.

In order to formalize this model we assume synchronous time steps and an *idle probability* p. This means:

For each processor i and each time step t, i is inactive at time step t with probability p. If it is active, it performs one atomic communication action.

For the above model, we can apply modifications of the delay sequence argument to prove the following theorem.

Theorem 1. *Assume the random latency model with idle probability p. The random rank protocol, Ranade's protocol and the growing rank protocol need routing time*

$$O((1 + \frac{1}{\log(\frac{1}{p})}) \cdot (C + D + log(N))),$$

with high probability.

The proofs are extensions of those for the respective synchronous protocols. The growing rank protocol and its analysis are presented in Section 3. All other proofs are omitted due to space limitations.

2.2 Adversarial models

Usually, in the setting of deterministic asynchronous modelling, we assume that no two actions take place exactly at the same time. Thus, for asynchronous networks, we may assume that processors are activated one after the other, in an order prescribed by an *adversary*. This adversary is considered to act as maliciously as possible, since it has full knowledge of the current configuration of the routing algorithm.

We consider the following notions of rounds, defining new time measures.

- the *global model*: A round is over as soon as every processor is activated at least once.

- the *path oriented model*: A round for packet p is over as soon as every processor on p's path is activated at least once.
- the *message oriented model*: A round for packet p is over when the node where p currently resides is activated.

As far as the global round model is concerned, we can apply the delay sequence argument, with slight modifications. This yields bounds for the running time similar to those for synchronous networks, for the random rank, growing rank and Ranade's protocol.

For the path oriented model, the delay sequence argument does not work, because the delay path constructed is not a routing path for any of the packets. This suggests that designing and analyzing routing protocols w.r.t. the path oriented model is much harder than for all previously mentioned models.

In Section 4 we present the proof for the following result:

Theorem 2. *For a routing problem R, we define the path congestion C(p) of a packet p as the number of different routing paths from R sharing an edge with the routing path of p. D(p) denotes the length of p's routing path. There is a protocol for the linear array such that any packet p is delivered within O(D(p)+C(p)) rounds, w.r.t. the path oriented model.*

Finally, in Section 5, we present an example showing a lower bound of $\Omega(C \cdot D)$ for routing w.r.t. the message oriented model.

Theorem 3. *Routing m packets from the first to the last processor of a linear array of size N needs time $\Omega(m \cdot N)$ w.r.t. the message oriented model.*

Note that Theorems 1 and 2 show an $O(m+N)$ time bound for this routing problem under all other models. Especially this shows a seperation between the path oriented and the message oriented model.

3 Growing rank protocol analysis

In this section we sketch a proof for Theorem 1 for the growing rank protocol (presented in [MV95] for arbitrary synchronous networks with unbounded buffers) w.r.t. the random latency model with idle probability p.

Suppose we are given a shortest paths routing problem with dilation D, congestion C, and size N on an arbitrary network G. Suppose R and m $:= \frac{R}{D}$ are suitably large integers. Initially each packet is assigned an integer rank chosen randomly, indepedently, and uniformly from the set $\{0, 1, \cdots, R-1\}$. Whenever a packet traverses a link its rank is increased by m. If two or more packets are contending to move forward along a link, then one of those with minimum rank is chosen. In order to break ties among packets with the same rank each packet p has a unique *ident-number* denoted by id(p). If there are several packets with the same minimum rank, then the one with the smallest ident-number is chosen. These ident numbers can be easily generated as follows: The ith packet starting

at the jth processor gets the ident-number $i \cdot n + j$ with n denoting the total number of processors.

We extend the notion of delay sequence to include delays caused by inactive processors. A (s+d,l)-delay sequence consists of:

1. s+1 nodes $u_0, u_1, \ldots, u_s$ (not necessarily distinct) .
2. s delay packets $p_1, p_2, \ldots, p_s$ such that the path for p_i crosses node u_i and the node u_{i-1} in that order, for $1 \leq i \leq s$, and the path of p_i leaves the node u_i along the same edges as the path of p_{i-1}, for $2 \leq i \leq s$.
3. s integers $l_1, l_2, \ldots, l_s$ such that l_i is the number of edges on the routing path of packet p_i from node u_i to node u_{i-1} for $1 \leq i \leq s$, and $\sum_{i=1}^{s} l_i \leq l$.
4. s integer keys $r_1, \ldots, r_s$ such that $0 \leq r_s \leq r_{s-1} \leq \ldots \leq r_1 \leq 2R - 1$.
5. d inactive steps out of $T = s + d + l$ total time steps.

The proofs for the following three lemmata are similar to the proofs of Lemmata 4, 5, 6 found in [MV95].

Lemma 4. *Suppose the routing takes $T \geq 2D^*$ or more rounds. Then a $(T - 2D^*, 2D^*)$-delay sequence is active.*

Lemma 5. *If the routing paths of the packets are shortest paths, then the delay packets in the above construction are pairwise distinct.*

Lemma 6. *The probability that an active delay sequence with s distinct delay packets exists is at most*

$$N 2^l \left(\frac{2eC(s+R)}{s}\right)^s R^{-s}$$

Theorem 7. *The growing rank protocol delivers every packet in time*

$$(1 + \frac{1}{\log(\frac{1}{p})}) O(C + \log N + D^*),$$

with high probability, i.e. with probability $\geq 1 - \frac{1}{N}$.

Proof: From Lemmata 4,5 we know that:

$$P(\text{routing takes more than } T = s + d + 2D^* \text{ rounds}) \leq$$
$$\leq P(\text{a } (s + d, 2D^*) - \text{delay sequence with distinct delay packets is active})$$

By assuming that $x \geq 12eC$, $R \geq s$ and $x \geq l + 2logN + 1$ we get from Lemma 6 :

$$P(\text{an active delay sequence with x or more delay packets exists}) \leq \frac{1}{N}$$

Thus we proved that, with high probability, there can be at most $max(12eC, 2D^* + 2logN + 1)$ delay packets in an active delay sequence. We will

now try to bound the expected number of inactive steps in any active delay sequence with a total of $max(12eC, 2D^* + 2logN + 1) + 2D^*$ active steps:

$$P(\text{a (T,l) delay sequence contains d or more inactive steps}) \leq \binom{T}{d} p^d \leq 2^T p^d$$

If we assume that $p \leq \frac{1}{4}$ and set $d = \frac{logN+T}{2}$ we get:

$$P(\text{a (s+d,l) delay seq. contains } s + l + logN \text{ or more inactive steps}) \leq \frac{1}{N}$$

We have shown that, with probability $\geq 1 - \frac{1}{N}$, any active delay sequence containts at most $max(12eC, 2D^* + 2logN + 1)$ delay packets, and that any delay sequence contains $O(s + l + logN)$ inactive steps. Therefore the growing rank protocol delivers every packet with probability, $\geq 1 - \frac{1}{N}$ within time:

$$T = O(C + D + logN), \text{ if } p \leq \frac{1}{4}.$$

If $p \geq \frac{1}{4}$, we can argue like in the analysis of the summation algorithm presented in [CZ95] to prove a delivery time of $T = (\frac{1}{log(\frac{1}{p})} + 1)O(C + D + logN)$. ∎

4 Routing under the path oriented model

Consider the following protocol on a linear array of N processors with unbounded buffers and bidirectional links : We fix an ordering on the packets, such that packets starting in node i get smaller ranks than those in node i+1. Among packets that are competing for the traversal of an edge, the one with highest rank advances. Packets use shortest paths and start moving at round 0.

Protocol analysis We only consider packets that move from left to right. The other packets can be treated seperately in the same way.

For ease of description we will identify packets with their ranks. Consider some packet $u \in \{y, \cdots, y + l\}$ and some processors with indexes i and i+j, $j \in \{0, \cdots, D-1\}$. We denote the set of processors $\{i, \cdots, i+j\}$ by I_j. Consider that packet u started in I_j. We say that u leaves I_j, if it has started in I_j and either reaches its destination in I_j or reaches processor i+j. Let $d_{j,u}$ denote the number of packets starting in I_j, with rank larger than u.

Lemma 8. *For each $u \in \{y, \cdots, y+l\}$ and each $j \in \{0, \cdots, D-1\}$, u leaves I_j after at most $j + d_{i,u}$ steps.*

Proof: We proceed by induction on $u = y + l, y + l - 1, \cdots, y$.
$u = y + l$: Consider any j such that u starts in I_j. Packet u is never delayed because it has the highest rank. Thus it leaves I_j after at most j rounds.

$u < y + l$: Consider any j such that u starts in I_j. Let u' be the last packet that delays u, befores it leaves I_j. This happens in some round t at some node $j' < j$. Furthermore $u' > u$. Thus u' leaves $I_{j'}$ in step t. By induction hypothesis, $t \leq j' + d_{j',u'}$ rounds. Note that $d_{j',u} \geq d_{j',u'} + 1$. Thus $t \leq j' + d_{j',u} - 1$.

As u is never delayed between round t+1 and leaving I_j, it leaves I_j after at most $(t+1) + j - j' \leq j + d_{j',u} \leq j + d_{j,u}$ rounds.

■

Theorem 9. *Consider a linear array with unbounded buffers and a routing problem R, where packet p has path congestion $C'(p)$ and dilation $D(p)$. Then, w.r.t. the path oriented model, the above protocol has delivered p after at most $D(p) + C'(p)$ rounds.*

Proof: Let p be a packet with rank u, that has to travel from j to j+l. Setting $d_{i,u} = C'(p)$ in Lemma 8 implies the theorem.

■

5 Message oriented model

We present the proof of Theorem 3 for the message oriented model.

Suppose we have the following routing problem on a linear array of size N, with unbounded buffers: The leftmost node initially holds $m \geq N$ packets, all destined to the rightmost node. Obviously the congestion of our problem is $C = m$ and the dilation $D = N$.

Now we assume that the adversary activates the processors as follows: First only the leftmost node is activated m times. This means that all packets will move to its right neighbour. Then we activate only this neighbour m consecutive times, and so on, until all packets reach the rightmost node of the array.

Let us label the packets with numbers $1, \cdots, m$. We can formalize the routing process by constructing a $m \times N$ matrix. Every column i of the matrix contains a permutation of the numbers $1, \cdots, m$. For every contention resolution protocol that we may use, such a matrix exists indicating the order in which the routing protocol allows packet to leave node i .

$$\begin{vmatrix} 1 & 2 & 2 \\ 2 & 1 & 1 \end{vmatrix}$$

The above 3×2 matrix corresponds to an array of size 3, m=2. First packet 2 gets priority over packet 1, while at the next two nodes the opposite happens.

It is easy to see that, no matter which routing protocol is used, i.e., what the above matrix looks like, some packet is delayed at least $\frac{m}{2}$ times in each of at least $\frac{N}{2}$ nodes. This yields a lower bound of $\frac{1}{4}m \cdot N = \frac{1}{4}C \cdot D$.

6 Acknowledgement

Thanks to Bob Cypher who contributed to initial discussions on modelling asynchronous routing.

References

[Ale82] M.Aleiunas. Randomized parallel communication. In *Proc. of the Symp. on Princ. of Distrib. Computing*, pp. 60-72, 1982.

[CMSV96] R. Cypher, F. Meyer auf der Heide, C. Scheideler, B. Voecking. Universal Algorithms for Store-and-Forward and Wormhole Routing. In *Proc. of the 28th Symp. on Theory of Computing*, pp. 356-365, 1996

[CZ89] R. Cole and O. Zajicek. The APRAM: Incorporating Asynchrony into the PRAM model. In *Proc. of the 1st Annual ACM Symp. on Paral. Alg. and Arch.*, pp. 169-178, June 1989.

[CZ95] R. Cole and O. Zajicek. The Expected Advantage of Asynchrony. In *Journal of Computer and Systems Sciences* , 51, pp. 286-300, 1995.

[DS87] W. Dally and C. L. Seitz. Deadlock free message routing in multiprocessor interconnection networks. In *IEEE Trans. on Computers*, 63,pages 547-553, 1987

[Dua93] J. Duato. A new theory of deadlock-free adaptive routing in wormhole networks. In *IEEE Trans. on Parallel and Distr. Systems*,pages 1320-1331

[Lei92] F.T.Leighton. Introduction to parallel algorithms and architectures: arrays-trees-hypercubes, *Morgan Kaufmann Publishers* (San Mateo, CA 1992)

[LM95] F.T.Leighton, B.M. Maggs. Fast algorithms for finding $O(congestion + dilation)$ packet routing schedules. In *Proc. of the 28th Int. Conf. on System Sciences*, pp.555-563, 1995

[LMR88] F.T. Leighton, B.M. Maggs, and S.B. Rao. Universal packet routing algorithms (Extended Abstract). In *Proc. of the 29th Annual Symp. on Found. of Comp. Science*, pp. 256-271, 1988.

[Lyn96] N.A. Lynch. Distributed Algorithms, *Morgan Kaufmann Publishers* (San Francisco, CA 1996).

[MP91] Y. Mansour, B. Patt-Shamir. Greedy packet scheduling on shortest paths. In *Proc. of the 10th Annual ACM Symp. on Princ. of Distrib. Computing*, 1991.

[MV95] F.Meyer auf der Heide, and B. Voecking. A Packet Routing Protocol for Arbitrary Networks. In *Proc. of the 12th Symp. on Theor. Aspects of Comp. Science*, pp. 291-302, 1995.

[Ran91] A.G. Ranade. How to emulate shared memory. In *Journal of Computer and System Sciences 42*, pp. 307-326, 1991.

[Upf84] E.Upfal. Efficient schemes for parallel communication. In *Journal of ACM Vol.31*, No.3, pp. 507-517, 1984.