May, 1996

LIDS-P 2338

Research Supported By:

NSF ECS 85-52419
Belllcore, Inc.
Draper Laboratory
ARO DAAL03-86-K-0171and DAAL03-92-G-0115

On the Average Communication Complexity of Asynchronous Distributed Algorithms

Tsitsiklis, J.N.
Stamoulis, G.D.

On the Average Communication Complexity of Asynchronous Distributed Algorithms

JOHN N. TSITSIKLIS AND GEORGE D. STAMOULIS

Massachusetts Institute of Technology, Cambridge, Massachusetts

Abstract. We study the communication complexity of asynchronous distributed algorithms. Such algorithms can generate excessively many messages in the worst case. Nevertheless, we show that, under certain probablistic assumptions, the expected number of messages generated per time unit is bounded by a polynomial function of the number of processors under a very general model of distributed computation. Furthermore, for constant-degree processor graphs, the expected number of generated messages is only $O(nT)$, where n is the number of processors and T is the running time. We conclude that (under our model) any asynchronous algorithm with good time complexity will also have good communication complexity, on the average.

Categories and Subject Descriptors: C.2.1 [**Computer-communication networks**] Network Architecture and Design—*network communications*; G.1.0 [**Numerical analysis**]: General—*parallel algorithms*; G.m [**Miscellaneous**]: Queueing theory

General Terms: Algorithms, performance

Additional Key Words and Phrases: asynchronous distributed algorithms

1. *Introduction*

In recent years, there has been considerable research on the subject of asynchronous distributed algorithms. Such algorithms have been explored both in the context of distributed numerical computation, as well as for the purpose of controlling the operation of a distributed computing system (e.g., finding shortest paths, keeping track of the system's topology, etc. [Bertsekas and Gallager 1987]). Some of their potential advantages are faster convergence, absence of any synchronization overhead, graceful degradation in the face of bottlenecks or long communication delays, and easy adaptation to topological changes such as link failures.

In the simplest version of an asynchronous distributed algorithm, each processor i maintains in its memory a vector y^i consisting of a variable x_i.

The authors' research was supported by the National Science Foundation (NSF) under grant ECS 85-52419, with matching funds from Bellcore, Inc. and the Draper Laboratory and by the ARO under grants DAAL03-86-K-0171 and DAAL03-92-G-0115.

Authors' current address: Laboratory for information and Decision Systems, Massachusetts Institute of Technology, Cambridge, MA 02139; e-mail:jnt@mit.edu; gstamoul@theseas.ntua.gr.

together with an estimate x_j^i of the variable x_j maintained by every neighboring processor j. Every processor i updates once in a while its own variable x_i on the basis of the information available to it, according to some mapping f_i. In particular, x_i is replaced by $f_i(y^i)$. Furthermore, if the new value of x_i is different from the old one, processor i eventually transmits a message containing the new value to all of its neighbors. When a neighbor j receives (in general, with some delay) the new values of x_i, it can use it to update its own estimate x_j^i of x_i.

A standard example is the asynchronous Bellman–Ford algorithm for the shortest path problem. Here, there is a special processor designated by 0, and for each pair (i, j) of processors, we are given a scalar c_{ij} describing the length of a link joining i to j. One version of the algorithm is initialized with $x_i = c_{i0}$, $i \neq 0$, and is described by the update rule

$$x_i := \min\left\{x_i, \min_{j \in \{i,0\}} \left\{c_{ij} + x_j^i\right\}\right\}, \qquad i \neq 0.$$

Under reasonable assumptions, the distributed asynchronous implementation of this algorithm terminates in finite time and the final value of each x_i is equal to the length of a shortest path from i to 0 [Bertsekas 1982].

In general, whenever some processor i receives a message from another processor j, there is a change in the vector y^i and, consequently, a subsequent update by processor i may lead to a new value for x_i that has to be eventually transmitted to the neighbors of processor i. Thus, if each processor has d neighbors, each message reception can trigger the transmission of d messages, and there is a clear potential for an exponential explosion of the messages being transmitted. Indeed, there are simple examples, due to Gafni and Gallager (see [Bertsekas and Tsitsiklis 1989, p. 450]), showing that the asynchronous Bellman–Ford algorithm for an n-node shortest path problem is capable of generating $\Omega(2^n)$ messages, in the worst case. These examples, however, rely on a large number of "unhappy coincidences": the communication delays of the different messages have to be chosen in a very special way. It is then reasonable to inquire whether excessive amounts of communication are to be expected under a probabilistic model in which the communication delays are modeled as random variables.

In the main model studied in this paper, we assume that the communication delays of the transmitted messages are independent and identically distributed random variables, and show that the expected number of messages transmitted during a time interval of duration T is at most of the order of $nd^{2+1/m}(\ln d)^{1+1/m}T$, where n is the number of processors, d is a bound on the number of neighbors of each processor, and m is a positive integer depending on some qualitative properties of the delay distribution; in particular, $m = 1$ for an exponential or a uniform distribution, while for a Gamma distribution, m equals the corresponding number of degrees of freedom.[1] Note that this estimate corresponds to $O(d^{1+1/m}(\ln d)^{1+1/m})$ messages per unit time on each link, which is quite favorable if d is constant (i.e., when the interprocessor connections are very sparse). Our result is derived under practically no assumptions on the detailed operation of the asynchronous algorithm, with one

[1] In fact, it will be seen that, for $m = 1$, the logarithmic factor in the upper bound can be removed.

exception discussed in the next paragraph. Furthermore, the result is valid for a very broad class of probability distributions for the message delays, including the Gamma distributions as special cases.

Since we are assuming that the delays of different messages are independent, messages can arrive out of order. Suppose that a message l carrying a value x_j is transmitted (by processor j) before but is received (by processor i) later than another message l' carrying a value x'_j. Suppose that l is the last message to be every received by i. Then, processor i could be left believing that x_j is the result of the final update by processor j (instead of the correct x'_j). Under such circumstances, it is possible that the algorithm terminate at an inconsistent state, producing incorrect results. To avoid such a situation, it is essential that a receiving processor be able to recognize whether a message just received was transmitted earlier than any other already received messages and, if so, discard the newly arrived message. This can be accomplished by adding a timestamp to each message, on the basis of which old messages are discarded. There are also special classes of algorithms in which timestamps are unnecessary. For example, in the Bellman–Ford algorithm described earlier, the value of x_j is nonincreasing with time, for every j. Thus, a receiving processor i need only check that the value x in a newly received message is smaller than the previously stored value x, and discard the message if this is not the case.

The above-described process of discarding "outdated" messages turns out to be a very effective mechanism for controlling the number of messages generated by an asynchronous algorithm. In particular, whenever the number of messages in transit tends to increase, then there are many messages that are overtaken by others, and therefore discarded. On the other hand, our "post office" model of independent and identical distributed message delays is unlikely to be satisfied in many parallel processing systems. It is more likely to hold in loosely coupled distributed systems in which processors communicate by means of some general communication facility.

1.1. Outline of the Paper. In Section 2, we present our model and assumptions and state the main results, which are then proved in Section 3. In Section 4, we discuss issues related to the average time complexity of an asynchronous algorithm under the same probabilistic model. Finally, in Section 5, we provide a brief discussion of alternative (possibly, more realistic) probabilistic models of interprocessor communication, and argue that under reasonable models, there will exist some mechanism that can keep the number of transmitted messages under control.

2. *The Model and the Main Results*

There are n processors, numbered $1, \ldots, n$, and each processor i has a set $A(i)$ of neighboring processors.[2] Let $d = \max_i |A(i)|$. The process starts at time $t = 0$, with processor 1 transmitting a message to its neighbors.

Whenever processor i receives a message, it can either ignore it, or it can (possibly, after some waiting time) transmit a message to some of its neighbors. Suppose that a message l is transmitted from i to j and, at some later time, another message l' is transmitted from i to j. If l' is received by j before l, we

[2] To simplify language, we make the assumption that $i \in A(j)$ if and only if $j \in A(i)$. Our subsequent results remain valid in the absence of this assumption.

say that l has been *overtaken* by l', and that l is ***discardable***. We will be assuming that discardable messages have essentially no effects at the receiving processor. In addition, we will allow processors to send messages that are self-triggered, that is, not caused by a message reception. However, a bound will be assumed on the frequency of self-triggered message transmissions. Our main assumption is:

Assumption 2.1

(a) Every discardable message is *ignored* by the receiving processor.
(b) Every nondiscardable message can trigger *at most* one transmission to each one of the neighbors of the receiving processor.
(c) During any time interval of length T, a processor may send at most T messages that have not been triggered by a received message, on any outgoing link.

Assumption 2.1(b) allows a processor to ignore messages that are not discardable. In practical terms, this could correspond to a situation where a processor i receives a message, updates its value of y^i, evaluates $x_i = f_i(y^i)$ and finds that the new value of x_i is the same as the old one, in which case there is nothing to be communicated to the neighbors of i.

We will be assuming that the communication delays of the different messages are independent and identically distributed, with a common cumulative probability distribution function F; that is, if D is the delay of a message, then $\Pr(D \le t) = F(t)$.

Simply assuming that message delays are independent and identically distributed, is actually insufficient for our purposes and does not fully capture the intuitive notion of "completely random and independent" communication delays. For example, even with independent and identically distributed message delays it is still possible that a processor "knows" ahead of time the communication delay of each one of the messages to be transmitted, and then acts maliciously: choose the waiting time before sending each message so as to ensure that as few messages are discarded as possible. Such malicious behavior is more difficult to analyze, and also very unnatural. Our next assumption essentially states that as long as a message is in transit, there is no available information on the remaining delay of that message, beyond the prior information captured by F.

Note that if a message has been in the air for some time $s > 0$, and only the prior information is available on the remaining delay of that message, then its total delay D is a random variable with cumulative distribution function

$$G(r|s) = \Pr[D \le r | D > s] = \frac{F(r) - F(s)}{1 - F(s)}, \qquad r \ge s. \tag{2.1}$$

[Of course, $G(r|s) = 0$ if $r < s$.]

Assumption 2.2

(a) The communication delays of the different messages are positive, independent and identically distributed random variables, with a common cumulative distribution function F.
(b) For every $s > 0, t \ge 0$, and every i, j, k, the following holds. The conditional distribution of the delay of the kth message transmitted from i to j,

conditioned on this message having being sent at time t and not being received within s time units, and also conditioned on any other events that have occurred up to time $t + s$, has the cumulative probability distribution function $G(\cdot|s)$.

Finally, we will be using the following technical assumption on F:

Assumption 2.3. There exists some positive integer m and some $\epsilon_0 > 0$ [with $F(\epsilon_0) < 1$] such that F is m times continuously differentiable in the interval $(0, 2\epsilon_0]$ and satisfies

$$\lim_{t \downarrow 0} F(t) = \lim_{t \downarrow 0} \frac{dF}{dt}(t) = \cdots = \lim_{t \downarrow 0} \frac{d^{m-1}F}{dt^{m-1}}(t) = 0 \qquad \text{and}$$

$$\lim_{t \downarrow 0} \frac{d^m F}{dt^m}(t) > 0;$$

moreover, there exist $c_1, c_2 > 0$ such that the mth derivative of F satisfies

$$c_1 \leq \frac{d^m F}{dt^m}(t) \leq c_2, \qquad \forall t \in (0, 2\epsilon_0].$$

Our assumption on the distribution of the delays is satisfied, in particular, in the case of a probability density function f that is right-continuous and infinitely differentiable at 0. Of course, the assumption also holds under milder conditions, such as right-continuity of f at 0 together with $\lim_{t \downarrow 0} f(t) > 0$; in this case, we have $m = 1$. (Various important distributions satisfy these properties; e.g., the exponential and the uniform distributions.) Assumption 2.3 is also satisfied by the Gamma distribution with m degrees of freedom. Roughly speaking, Assumption 2.3 requires that $F(t) = \Theta(t^m)$ for $t \in (0, 2\epsilon_0]$.

Our main results are given by the following two theorems. In particular, Theorem 2.4 corresponds to the case where Assumption 2.3 is satisfied with $m = 1$, while Theorem 2.5 corresponds to $m > 1$.

THEOREM 2.4. *Assume that $T \geq 1$ and that $m = 1$. Then, there exists a constant A (depending only on the constants c_1, c_2 and ϵ_0 of Assumption 2.3), such that the expected total number of messages transmitted during the time interval $[0, T]$ is bounded by And^3T.*

THEOREM 2.5. *Assume that $T \geq 1$ and that $m > 1$. Then, there exists a constant A' (depending only on the constants m, c_1, c_2 and ϵ_0 of Assumption 2.3), such that the expected total number of messages transmitted during the time interval $[0, T]$ is bounded by $A'nd^{2+1/m}(\ln d)^{1+1/m}T$.*

Notice that the difference between Theorems 2.4 and 2.5 lies on the logarithmic factor; a short discussion of this point is provided in Subsection 3.5.

3. *Proofs of the Results*

3.1. AN EASY SPECIAL CASE. In this subsection, we motivate Theorem 2.4 by considering the the following special case:

(i) The message delays have exponential probability distributions, with mean 1.

(ii) Each processor transmits a message to every other processor, immediately upon receipt of a nondiscardable message. (That is, the underlying graph is assumed to be complete.)

(iii) There are no self-triggered messages except for one message that starts the computation.

Let $m_{ij}(t)$ be the number of messages in transit from i to j at time t, that have not been overtaken; that is, no later transmitted message from i to j has already reached its destination. [The notation $m_{ij}(t)$ should not be confused with the constant m involved in Assumption 2.3.] Every message that is in transit has probability Δ of being received within the next Δ time units. Thus, at time t, the rate at which messages arrive to j along the link (i, j) is $m_{ij}(t)$. Since any such arrival triggers a message transmission by j, the rate of increase of $m_{jk}(t)$ is $\sum_{i \neq j} m_{ij}(t)$. On the other hand, an arrival of a message traveling along the link (i, j) overtakes (on the average) half of the other messages in transit across that link. Thus,

$$\frac{d}{dt}E\left[m_{jk}(t)\right] = \sum_{i \neq j} E\left[m_{ij}(t)\right] - E\left[m_{jk}(t)\right] - \frac{E\left[\left(m_{jk}(t) - 1\right)m_{jk}(t)\right]}{2}$$

$$= \sum_{i \neq j} E\left[m_{ij}(t)\right] - \frac{1}{2}E\left[m_{jk}(t)\right]^2. \tag{3.1}$$

Let $M(t) = \sum_{j=1}^{n} \sum_{k \neq j} E[m_{jk}(t)]$. The Schwartz inequality gives

$$\frac{1}{n^2}M^2(t) \leq \sum_{j=1}^{n} \sum_{k \neq j} E\left[m_{jk}(t)\right]^2$$

and eq. (3.1) becomes

$$\frac{d}{dt}M(t) \leq nM(t) - \frac{1}{2n^2}M^2(t).$$

Note that whenever $M(t) \geq 2n^3$, we have $(dM/dt)(t) \leq 0$ and this implies that $M(t) \leq 2n^3$, for all $t > 0$. Thus, the rate of reception of nondiscardable messages, summed over all links, is $O(n^3)$. Since each such message reception generates $O(n)$ message transmissions, messages are generated at a rate of $O(n^4)$. We conclude that the expected number of messages generated during a time interval $[0, T]$ is $O(n^4T)$, which agrees with Theorem 2.4 for the case $d = O(n)$.

We can now provide some intuition for the validity of Theorem 2.4 for the case $m = 1$: messages with communication delay above ϵ_0 will be overtaken with high probability and can be ignored; messages with communication delay below ϵ_0 have approximately uniform distribution (cf. Assumption 2.3 with $m = 1$), which is approximately the same as the lower tail of an exponential distribution, for ϵ_0 small. Thus, we expect that the analysis for the case of exponential distributions should be representative of any distribution satisfying Assumption 2.3 with $m = 1$. In fact, the proof of Theorem 2.4 is based on the argument outlined above. The proof of Theorem 2.5 is based on a somewhat different idea and is more involved.

3.2. SOME NOTATION AND TERMINOLOGY. We start by considering the transmissions along a particular link, say the link from i to j. Let M_{ij} be the (random) number of messages transmitted by processor i along that link during the time interval $[0, T]$. Any such message is called *successful* if it arrives at j no later than time T and if it is not discarded upon arrival, that is, if that message has not been overtaken by a later transmitted message along the same link. Let S be the number of successful messages sent from i to j. With the exception of T self-triggered messages, only successful messages can trigger a transmission by the receiving processor. Therefore,

$$M_{jk} \le T + \sum_{i \in A(j)} S_{ij}, \qquad \forall k \in A(j),$$

which leads to

$$E[M_{jk}] \le T + \sum_{i \in A(j)} E[S_{ij}], \qquad \forall k \in A(j). \tag{3.2}$$

In order to establish Theorems 2.4 and 2.5, we upper bound $E[S_{ij}]$ by an appropriate function of $E[M_{ij}]$. This is done in a different way for each of the two theorems.

3.3. THE PROOF OF THEOREM 2.4.

THEOREM 2.4. *Assume that $T \ge 1$ and that $m = 1$. Then, there exists a constant A (depending only on the constants c_1, c_2 and ϵ_0 of Assumption 2.3), such that the expected total number of messages transmitted during the time interval $[0, T]$ is bounded by And^3T.*

The proof of Theorem 2.4, rests on the following result:

LEMMA 3.3.1. *There exist constants B, B', depending only on the constants c_1, c_2 and ϵ_0 of Assumption 2.3, such that*

$$E[S_{ij}] \le B\sqrt{TE[M_{ij}]} + B'T. \tag{3.3}$$

PROOF OF THEOREM 2.4. Let $Q = \max_{i,j} E[M_{ij}]$. Then, Eq. (3.3) yields $E[S_{ij}] \le B\sqrt{TQ} + B'T$. Using Eq. (3.2), we obtain $E[M_{jk}] \le T + dB\sqrt{TQ} + dB'T$. Taking the maximum over all j, k, and using the fact $d \ge 1$, we obtain $Q \le dB\sqrt{TQ} + d(B' + 1)T$. Suppose that $Q \ge T$. Then $Q \le d(B + B' + 1)\sqrt{TQ}$, which yields $Q \le (B + B' + 1)^2 d^2 T$. If $Q < T$, this last inequality is again valid. We conclude that there exists a constant A such that $Q \le Ad^2T$. Thus, $E[M_{ij}] \le Ad^2T$ for every link (i, j) and since there are at most nd links, the expected value of the total number of transmitted messages is bounded above by And^3T, which is the desired result. □

It now remains to prove Lemma 3.3.1.

PROOF OF LEMMA 3.3.1. For the purposes of the lemma, we only need to consider a fixed pair of processors i and j. We may thus simplify notation and use M and S instead of M_{ij} and S_{ij}, respectively.

Note that if $E[M] \le T/\epsilon_0^2$, then $E[S] \le T/\epsilon_0^2$ (because $S \le M$) and Eq. (3.3) holds, as long as B' is chosen larger than $1/\epsilon_0^2$. Thus, we only need to consider the case $E[M] > T/\epsilon_0^2$, which we henceforth assume.

Successful messages can be of two types:

(i) Those that reach their destination with a delay of at least ϵ_0; we call them *slow* messages.
(ii) Those that reach their destination with a delay smaller than ϵ_0; we call them *fast* messages.

Let S_s and S_f be the number of slow and fast successful messages, respectively. We will bound their respective expectations using two somewhat different arguments, starting with $E[S_f]$.

3.3.1. *A Bound on the Expected Number of Fast Successful Messages.* We split $[0, T]$ into disjoint time intervals of length

$$\delta \stackrel{\text{def}}{=} \sqrt{\frac{T}{E[M]}}.$$

To simplify notation, we assume that $\sqrt{TE[M]}$ is an integer. (Without this assumption, only some very minor modifications would be needed in the argument that follows.) Thus, the number of intervals in $T/\delta = \sqrt{TE[M]}$. Note also that $\delta < \epsilon_0$, due to our assumption $E[M] > T/\epsilon_0^2$.

Let $t_k = (k-1)\delta$ be the starting time of the kth interval. Let $\mathcal{S}_k$ be the set of messages transmitted during the kth interval, and let I_k be the cardinality of $\mathcal{S}_k$. Let $\mathcal{M}_k$ be the set of messages with the following properties:

(a) The time t at which the message was transmitted satisfies $t_k - \epsilon_0 < t \le t_k$.
(b) At time t_k, the message has not yet reached its destination.
(c) The message has not been overtaken by another message that has reached its destination by time t_k.

Thus, the set $\mathcal{M}_k$ contains the messages that are in transit at time t_k, that still have a hope of being successful (not yet overtaken), and that have not been in the air for "too long". Let N_k be the cardinality of $\mathcal{M}_k$.

Consider now a message in the set $\mathcal{M}_k$ and suppose that it was transmitted at time $t_k - s$, where $0 \le s < \epsilon_0$. Such a message reaches its destination during the time interval $(t_k, t_{k+1}]$ with probability

$$G(\delta + s|s) = \frac{F(\delta + s) - F(s)}{1 - F(s)}.$$

[See eq. (2.1) and Assumption 2.2.] Furthermore, Assumption 2.3 (which was taken to hold with $m = 1$) implies that

$$c_1\delta \le F(\delta + s) - F(s) \le c_2\delta, \qquad \forall \delta, s \in [0, \epsilon_0];$$

also, for $s \in [0, \epsilon_0]$, we have $0 < 1 - F(\epsilon_0) \le 1 - F(s) \le 1$. [Recall that $F(\epsilon_0) < 1$ by Assumption 2.3.] Thus, it follows that

$$c_1\delta < G(\delta + s|s) \le \alpha_2\delta, \qquad \forall \delta, s \in [0, \epsilon_0], \tag{3.4}$$

where $\alpha_2 = c_2/[1 - F(\epsilon_0)]$. Therefore, the probability that a message in the set $\mathcal{M}_k$ reaches its destination during $(t_k, t_{k+1}]$ lies between $c_1\delta$ and $\alpha_2\delta$. Similarly, for any message in the set $\mathcal{S}_k$, the probability that it reaches its destination during the time interval $(t_k, t_{k+1}]$ is at most $F(\delta)$, which does not exceed $\alpha_2\delta$. [To see this, apply eq. (3.4) with $s = 0$.]

For a message to be received during the time interval $[t_k, t_{k+1}]$ and for it to be successful and fast, it is necessary that it belong to the set $\mathcal{A}_k \cup \mathcal{S}_k$. Using the bounds of the preceding paragraph, the expected number of such successful fast messages is bounded above by $\alpha_2\delta(E[N_k + I_k])$. Adding over all k, we see that the expected number of successful fast messages satisfies

$$E[S_f] \le \alpha_2\delta \sum_{k=1}^{T/\delta} E[N_k + I_k]. \tag{3.5}$$

Next, we estimate the number of messages in the set $\mathcal{A}_k$ that also belong to $\mathcal{A}_{k+1}$. (Notice that these two sets may possibly intersect, because $t_{k+1} - \epsilon_0 < t_k$ due to the assumption $\delta < \epsilon_0$.) Let us number the messages in the set $\mathcal{A}_k$ according to the times that they were transmitted, with later transmitted messages being assigned a smaller number. Note that the lth message in $\mathcal{A}_k$ belongs to $\mathcal{A}_{k+1}$ only if none of the messages $1, \ldots, l$ has been received during the time interval $(t_k, t_{k+1}]$. Using our earlier calculations, each message in $\mathcal{A}_k$ has a probability of at least $c_1\delta$ of being received during $(t_k, t_{k+1}]$. Using the independence of the delays of different messages (Assumption 2.4), the lth message in $\mathcal{A}_k$ makes it into $\mathcal{A}_{k+1}$ with probability no larger than $(1 - c_1\delta)^l$. Summing over all l, the expected number of elements of $\mathcal{A}_k$ that make it into $\mathcal{A}_{k+1}$ is bounded above by $1/(c_1\delta)$. The set $\mathcal{A}_{k+1}$ consists of such messages together possibly with some of the elements of $\mathcal{S}_k$. We thus have

$$E[N_{k+1}] \le \frac{1}{c_1\delta} + E[I_k]. \tag{3.6}$$

Combining eqs. (3.5) and (3.6), and using the property $\sum_{k=1}^{T/\delta} E[I_k] = E[M]$, we obtain

$$\begin{aligned} E[S_f] &\le \frac{\alpha_2 T}{c_1\delta} + \alpha_2\delta \sum_{k=1}^{T/\delta} E[I_{k-1} + I_k] \\ &\le \frac{\alpha_2 T}{c_1\delta} + 2\alpha_2\delta E[M] \\ &= \left(\frac{\alpha_2}{c_1} + 2\alpha_2\right)\sqrt{TE[M]}. \end{aligned} \tag{3.7}$$

3.3.2. *A Bound on the Expected Number of Slow Successful Messages.* We now derive an upper bound for the expected number of successful "slow" messages. For the purposes of this argument, we split $[0, T]$ into intervals of length $\epsilon_0/2$. (The last such interval might have length smaller than $\epsilon_0/2$ if $2T/\epsilon_0$ is not an integer.) The total number of such intervals is $\lceil 2T/\epsilon_0 \rceil$. Let $t_k = (k-1)\epsilon_0/2$. Let us number the messages transmitted during $[t_k, t_{k+1}]$, with later transmitted messages being assigned a smaller number. Clearly, a message generated at time $t_{k+1} - s$, with $0 \le s \le \epsilon_0/2$, is received during the time interval $[t_{k+1}, t_{k+2}]$ with probability $F(s + \epsilon_0/2) - F(s)$; reasoning similarly as in previous cases, it is seen that this probability is at least $c_1(\epsilon_0/2)$. Notice now that for the lth message transmitted during $[t_k, t_{k+1}]$ to be a slow and successful message, it is necessary that none of the messages $1, \ldots, l$ transmitted during that same interval is received during the time interval

$[t_{k-1}, t_{k-1}]$: the probability of this event is at most $(1 - c_1(\epsilon_0/2))^l$. Thus, the expected number of messages that are transmitted during $[t_k, t_{k-1}]$ and are slow and successful is bounded above by $2/c_1\epsilon_0$. Adding over all k, we obtain

$$E[S_s] \le \left\lceil \frac{2T}{\epsilon_0} \right\rceil \cdot \frac{2}{c_1\epsilon_0} \le B'T, \tag{3.8}$$

where B' is a suitable constant.

Since $E[S] = E[S_f] + E[S_s]$, eqs. (3.7) and (3.8) complete the proof of the lemma. □

3.4. The Proof of Theorem 2.5

THEOREM 2.5. *Assume that $T \ge 1$ and that $m > 1$. Then, there exists a constant A' (depending only on the constants m, c_1, c_2 and ϵ_0 of Assumption 2.3), such that the expected total number of messages transmitted during the time interval $[0, T]$ is bounded by $A' n d^{(m+1)/m} (\ln d)^{1+1/m} T$.*

The proof of Theorem 2.5 rests on the following result:

LEMMA 3.4.1. *There exists a constant $\hat{B}$, depending only on the constants m, c_1, c_2 and ϵ_0 of Assumption 2.3, such that*

$$E[S_f] \le \hat{B} T^{m/(m+1)} E[M_j]^{1/(m+1)} \max\left\{1, \ln\left(\frac{E[M_j]}{T}\right)\right\}. \tag{3.9}$$

PROOF OF THEOREM 2.5. Let $Q = \max_{j,k} E[M_{jk}]$. Then, eq. (3.9) yields

$$E[S_f] \le \hat{B} T^{m/(m+1)} Q^{1/(m+1)} \max\left\{1, \ln\left(\frac{Q}{T}\right)\right\}.$$

Using eq. (3.2), we obtain

$$E[M_k] \le T + d\hat{B} T^{m/(m+1)} Q^{1/(m+1)} \max\left\{1, \ln\left(\frac{Q}{T}\right)\right\}.$$

Taking the maximum over all j, k, and using the fact $d \ge 1$, we obtain

$$Q \le dT + d\hat{B} T^{m/(m+1)} Q^{1/(m+1)} \max\left\{1, \ln\left(\frac{Q}{T}\right)\right\}.$$

Suppose that

$$Q > x^{(m+1)/m} T.$$

Then,

$$Q \le d(\hat{B} + 1) T^{m/(m+1)} Q^{1/(m+1)} \ln\left(\frac{Q}{T}\right),$$

which yields

$$\frac{(Q/T)^{m/(m+1)}}{\ln\left[(Q/T)^{m/(m+1)}\right]} \le \bar{B}d, \tag{3.10}$$

where $\bar{B} = ((m+1)/m)(\hat{B} + 1)$.

Next, we prove the following auxiliary result: If $x > e$ and $x/\ln x \le y$, then $x \le 2y \ln y$. Indeed, since $x/\ln x$ is an increasing function of x for $x > e$, it is sufficient to show that if $x/\ln x = y$ then $x \le 2y \ln y$. Thus, it is enough to show that $x \le 2(x/\ln x)\ln(x/\ln x)$ or $x \le 2x - 2x(\ln\ln x/\ln x)$; equivalently $2\ln\ln x \le \ln x$ or $\ln x \le \sqrt{x}$, which is true for all $x > e$.

Due to eq. (3.10) and the assumption $Q > \exp((m+1)/m)T$, we can apply the above result with $x = (Q/T)^{m/(m+1)}$ and $y = \bar{B}d$; thus, it follows that

$$\left(\frac{Q}{T}\right)^{m/(m+1)} \le 2\bar{B}d\ln(\bar{B}d),$$

which gives

$$Q \le A' d^{1+(1/m)}(\ln d)^{1+(1/m)}T,$$

where A' is a suitable constant. If $Q \le \exp((m+1)/m)T$, this last inequality is again valid. We conclude that there exists a constant A' such that $Q \le A' d^{1+1/m}(\ln d)^{1+1/m}T$. Each processor sends M_{ij} messages along every link (i, j). Since $E[M_{ij}] \le A' d^{1+1/m}(\ln d)^{1+1/m}T$ and since there are at most nd links, the expected value of the total number of transmitted messages is bounded above by $A' nd^{2+1/m}(\ln d)^{1+1/m}T$, which is the desired result. □

It now remains to prove Lemma 3.4.1.

PROOF OF LEMMA 3.4.1. For the purposes of the lemma, we only need to consider a fixed pair of processors i and j. We may thus simplify notation and use M and S instead of M_{ij} and S_{ij}, respectively.

Let δ be defined as follows:

$$\delta \overset{\text{def}}{=} \left(\frac{T}{E[M]}\right)^{1/(m+1)}. \tag{3.11}$$

Note that if $\delta \ge \epsilon_0$, then $E[M] \le T/\epsilon_0^{m+1}$, which implies that

$$E[M] \le \left(\frac{1}{\epsilon_0}\right)^m T^{m/(m+1)}(E[M])^{1/(m+1)};$$

therefore, eq. (3.9) holds as long as $\hat{B}$ is chosen larger than $1/\epsilon_0^m$. Thus, we only need to consider the case $\delta < \epsilon_0$, which we henceforth assume.

We split the interval $[0, T]$ into disjoint intervals of length δ. To simplify notations, we assume that T/δ is an integer. (Without this assumption, only some very minor modifications would be needed in the arguments to follow.) For definiteness, let the qth interval be $\mathcal{I}_q = [(q-1)\delta, q\delta)$, with the exception of $\mathcal{I}_{T/\delta} = [T - \delta, T]$. Let M_q denote the number of messages generated during $\mathcal{I}_q$. Clearly, we have

$$\sum_{q=1}^{T/\delta} E[M_q] = E[M]. \tag{3.12}$$

Let S_q be the number of nondiscardable messages generated during $\mathcal{I}_q$. We have

$$\sum_{q=1}^{T/\delta} E[S_q] = E[S]. \tag{3.13}$$

Henceforth, we fix some $q \in \{1, \ldots, T/\delta\}$ and we concentrate on bounding $E[S_q]$.

Let $\hat{N}_q$ be the number of messages that are generated during the interval $\mathcal{I}_q$ and arrive *no later* than time $q\delta$.

LEMMA 3.4.2. *$E[\hat{N}_q] \le \alpha_2 \delta^m E[M_q]$, where $\alpha_2 = c_2/m!$, where c_2 and m are the constants of Assumption 2.3.*

PROOF OF LEMMA 3.4.2. Let $t_1, \ldots, t_{M_q}$ be the times in $\mathcal{I}_q$, in increasing order, at which messages are generated. Let $D_1, \ldots, D_{M_q}$ be the respective delays of these messages. We have

$$E\left[\hat{N}_q\right] = \sum_{k=1}^{\infty} \Pr[M_q \ge k] \Pr\left[D_k \le q\delta - t_k \mid M_q \ge k\right]$$

$$\le \sum_{k=1}^{\infty} \Pr[M_q \ge k] \Pr\left[D_k \le \delta \mid M_q \ge k\right], \tag{3.14}$$

where the last inequality follows from the fact $t_k \ge (q-1)\delta$. By Assumption 2.2, the delay of a message is independent of all events that occurred until the time of its generation; hence, we have

$$\Pr\left[D_k \le \delta \mid M_q \ge k\right] = F(\delta), \tag{3.15}$$

because, at time t_k, the event $M_q \ge k$ is *known* to have occurred. Furthermore, using Assumption 2.3 and some elementary calculus, we see that there exist constants $\alpha_1, \alpha_2 > 0$ such that

$$\alpha_1(x^m - y^m) \le F(x) - F(y) \le \alpha_2(x^m - y^m), \qquad \text{for } 0 \le y \le x \le 2\epsilon_0. \tag{3.16}$$

(In particular, $\alpha_1 = c_1/m!$ and $\alpha_2 = c_2/m!$.) Applying eq. (3.16) with $x = \delta$ and $y = 0$, we have $F(\delta) \le \alpha_2 \delta^m$; combining this with eqs. (3.14) and (3.15), we obtain

$$E\left[\hat{N}_q\right] \le \alpha_2 \delta^m \sum_{k=1}^{\infty} \Pr[M_q \ge k] = \alpha_2 \delta^m E[M_q]. \tag{3.17}$$

□

Let $\tilde{S}_q$ be the number of nondiscardable messages that are generated during $\mathcal{I}_q$ and arrive *after* time $q\delta$. Recalling that $\hat{N}_q$ is the number of messages that are generated during $\mathcal{I}_q$ and arrive no later than $q\delta$, we have

$$E[S_q] \le E\left[\hat{N}_q\right] + E\left[\tilde{S}_q\right]. \tag{3.18}$$

Lemma 3.4.2 provides a bound for $E[\hat{N}_q]$; thus, it only remains to upper bound $\tilde{S}_q$.

LEMMA 3.4.3. *We have*

$$E\left[\tilde{S}_q\right] \le \beta_1 \delta^m N_q + \frac{\beta_2}{\beta_3} \ln(N_q + 1) + \frac{1}{1-\gamma},$$

where $\alpha_1, \beta_1, \beta_2, \beta_3, \gamma$ are constants that depend only on the constants introduced in Assumption 2.3.

PROOF OF LEMMA 3.4.3. Let $\mathcal{F}$ stand for the history of the process up to and including time $q\delta$. Let $\hat{N}_q$ be the number of messages that were transmitted during $\mathcal{F}$ and have not been received by time $q\delta$; note that $\hat{N}_q = M_q - \hat{N}_q$. We will be referring to the aforementioned $\hat{N}_q$ messages as $P_1, \ldots, P_{\hat{N}_q}$. In particular, message P_k is taken to be generated at time t_k, where $(q-1)\delta \le t_1 \le t_2 \le \cdots \le t_{\hat{N}_q} < q\delta$. The delay of P_k is denoted by D_k; there holds $D_k \ge q\delta - t_k$, by assumption. Note that $\hat{N}_q$ and $(t_1, \ldots, t_{\hat{N}_q})$ are $\mathcal{F}$-measurable; that is, their values are known at time $q\delta$. Also, Assumption 2.2 implies that, conditioned on $\mathcal{F}$, the random variables $D_1, \ldots, D_{\hat{N}_q}$ are independent, with the conditional cumulative distribution of D_k being $G(\cdot | q\delta - t_k)$.

In the analysis to follow, we assume that $\hat{N}_q \ge 2$; the trivial cases $\hat{N}_q = 0$ and $\hat{N}_q = 1$ will be considered at the end. At time $q\delta$, message P_k has been in the air for $s_k \overset{\text{def}}{=} q\delta - t_k$ time units; notice that $s_k \le \delta$. Let R_k denote the random variable $D_k - s_k$; that is, R_k is the residual time (after $q\delta$) for which message P_k will remain in the air. As argued above, conditioned on $\mathcal{F}$, the random variables $R_1, \ldots, R_{\hat{N}_q}$ are independent; moreover, the conditional cumulative distribution function of R_k is given by

$$H_k(r) \overset{\text{def}}{=} \Pr[R_k \le r | \mathcal{F}] = G(r + s_k | s_k) = \frac{F(r + s_k) - F(s_k)}{1 - F(s_k)}. \quad (3.19)$$

Let $f(r) = (dF/dr)(r)$ and $h_k(r) = (dH_k/dr)(r)$; both derivatives are guaranteed to exist in the interval $(0, \epsilon_0]$ due to Assumption 2.3 and the fact $s_k < \delta < \epsilon_0$. Clearly, if $k \ne \hat{N}_q$, then for P_k not to be discardable it is necessary that messages $P_{k+1}, \ldots, P_{\hat{N}_q}$ arrive later than P_k. Therefore, we have

$$\begin{aligned}
\Pr[P_k \text{ is nondiscardable} | \mathcal{F}] &\le \Pr\left[R_k \le R_l \text{ for } l = k+1, \ldots, \hat{N}_q | \mathcal{F}\right] \\
&= \int_0^\infty \Pr\left[r \le R_l \text{ for } l = k+1, \ldots, \hat{N}_q | \mathcal{F}\right] dH_k(r) \\
&= \int_0^\infty \left(\prod_{l=k+1}^{\hat{N}_q} \Pr[R_l \ge r | \mathcal{F}] \right) dH_k(r) \\
&= \int_0^\infty \left(\prod_{l=k+1}^{\hat{N}_q} [1 - H_l(r)] \right) dH_k(r).
\end{aligned}$$

We split this integral into three parts and for each part, we use a different bound for the integrand: for $r \in [0, \delta]$, we use the bound $1 - H_l(r) \le 1$; for $r \in [\epsilon_0, \infty)$, we use the bound $1 - H_l(r) \le 1 - H_l(\epsilon_0)$. We therefore obtain

$$\begin{aligned}
\Pr[P_k \text{ is nondiscardable} | \mathcal{F}] \le H_k(\delta) &+ \int_\delta^{\epsilon_0} \left(\prod_{l=k+1}^{\hat{N}_q} [1 - H_l(r)] \right) dH_k(r) \\
&+ \prod_{l=k+1}^{\hat{N}_q} [1 - H_l(\epsilon_0)]. \quad (3.20)
\end{aligned}$$

In what follows, we derive an upper bound for each of the three terms in eq. (3.20).

Starting with $H_i(\delta)$, we have

$$H_i(\delta) \le \frac{\alpha_2\left[(\delta + s_k)^m - s_i^m\right]}{1 - F(s_k)}.$$

due to eqs. (3.19) and (3.16). Since $s_k \le \delta$, we have $(s_k + \delta)^m - \delta^m \le (2^m - 1)\delta^m$; moreover, there holds $0 < 1 - F(\epsilon_0) \le 1 - F(s_i)$, because $s_i \le \delta < \epsilon_0$ and $F(\epsilon_0) < 1$ (see Assumption 2.3). Combining these facts, it follows that

$$H_i(\delta) \le \frac{\alpha_2(2^m - 1)}{1 - F(\epsilon_0)}\delta^m = \beta_1\delta^m. \tag{3.21}$$

Furthermore, let Δ be a small positive real number; by eq. (3.19), we have

$$H_i(r + \Delta) - H_i(r) = \frac{F(r + s_i + \Delta) - F(r + s_i)}{1 - F(s_i)}.$$

Since $s_i \le \delta < \epsilon_0$, it follows from eq. (3.16) that

$$\begin{aligned}\frac{\alpha_1}{1 - F(s_i)}&\left[(r + s_i + \Delta)^m - (r + s_i)^m\right] \\ &\le H_i(r + \Delta) - H_i(r) \\ &\le \frac{\alpha_2}{1 - F(s_i)}\left[(r + s_i + \Delta)^m - (r + s_i)^m\right], \qquad \forall r \in [0, \epsilon_0].\end{aligned}$$

Reasoning similarly as in the case of eq. (3.21), it follows (after some algebra) that

$$\begin{aligned}\alpha_1[(r + \Delta)^m - r^m] &\le H_i(r + \Delta) - H_i(r) \\ &\le \frac{\alpha_2(2^m - 1)}{1 - F(\epsilon_0)}[(r + \Delta)^m - r^m], \qquad \forall r \in [0, \epsilon_0].\end{aligned} \tag{3.22}$$

On the other hand, using eq. (3.16), we have

$$\alpha_1[(r + \Delta)^m - r^m] \le F(r + \Delta) - F(r) \le \alpha_2[(r + \Delta)^m - r^m], \qquad \forall r \in [0, \epsilon_0];$$

this together with eq. (3.22) implies that there exist constants $\beta_2, \beta_3 > 0$, which do not depend on i, such that

$$\beta_3[F(r + \Delta) - F(r)] \le H_i(r + \Delta) - H_i(r) \le \beta_2[F(r + \Delta) - F(r)], \qquad \forall r \in [0, \epsilon_0].$$

Using this, it follows easily that

$$h_i(r) \le \beta_2 f(r), \qquad \forall r \in [0, \epsilon_0], \tag{3.23}$$

and

$$H_i(r) \ge \beta_3 F(r), \qquad \forall r \in [0, \epsilon_0]. \tag{3.24}$$

Combining eqs. (3.23) and (3.24), we have

$$\int_{\delta}^{\epsilon_0} \left(\prod_{l=k+1}^{N_q} [1 - H_l(r)] \right) dH_k(r) \le \beta_2 \int_{\delta}^{\epsilon_0} [1 - \beta_3 F(r)]^{N_q - k} f(r)\, dr$$

$$= \frac{\beta_2}{\beta_3} \int_{\delta}^{\epsilon_0} [1 - \beta_3 F(r)]^{N_q - k} d(\beta_3 F(r))$$

$$\le \frac{\beta_2}{\beta_3} \int_0^1 (1 - y)^{N_q - k} dy$$

$$= \frac{\beta_2}{\beta_3} \frac{1}{N_q - k + 1}, \tag{3.25}$$

where we have also used the fact $\beta_3 F(\epsilon_0) \le H_k(\epsilon_0) \le 1$ [see eq. (3.22) with $r = \epsilon_0$ and $l = k$]. Similarly, by eq. (3.24), we have

$$\prod_{l=k+1}^{N_q} [1 - H_l(\epsilon_0)] \le [1 - \beta_3 F(\epsilon_0)]^{N_q - k} = \gamma^{N_q - k}, \tag{3.26}$$

where γ is constant and satisfies $0 \le \gamma < 1$.

Combining eqs. (3.20), (3.21), (3.25), and (3.26), we obtain

$$\Pr[P_k \text{ is nondiscardable} \mid \mathcal{F}] \le \beta_1 \delta^m + \frac{\beta_2}{\beta_3} \frac{1}{N_q - k + 1} + \gamma^{N_q - k}.$$

The above result holds for $k = 1, \ldots, N_q - 1$; adding over all those k, we have

$$\sum_{k=1}^{N_q - 1} \Pr[P_k \text{ is nondiscardable} \mid \mathcal{F}] \le \beta_1 \delta^m (N_q - 1)$$

$$+ \frac{\beta_2}{\beta_3} \sum_{k=1}^{N_q - 1} \frac{1}{N_q - k + 1} + \sum_{k=1}^{N_q - 1} \gamma^{N_q - k}. \tag{3.27}$$

Notice that

$$\sum_{k=1}^{N_q - 1} \frac{1}{N_q - k + 1} = \sum_{k=2}^{N_q} \frac{1}{k} \le \ln(N_q + 1),$$

and

$$\sum_{k=1}^{N_q - 1} \gamma^{N_q - k} \le \sum_{k=1}^{\infty} \gamma^k = \frac{\gamma}{1 - \gamma},$$

because $0 \le \gamma < 1$. Thus, it follows from eq. (3.27) that

$$E[\tilde{S}_q \mid \mathcal{F}] = \sum_{k=1}^{N_q} \Pr[P_k \text{ is nondiscardable} \mid \mathcal{F}]$$

$$\le \beta_1 \delta^m N_q + \frac{\beta_2}{\beta_3} \ln(N_q + 1) + \frac{\gamma}{1 - \gamma} + 1.$$

where the term "+ 1" bounds the probability that P_{N_q} is nondiscardable. The above result was established for all $N_q \geq 2$; it is straightforward to see that it also holds for $N_q = 1$ and for $N_q = 0$. We now take expectations, to remove the conditioning on $\mathcal{F}_q$, and the desired result is obtained. □

We now combine Lemmas 3.4.2 and 3.4.3, together with eq. (3.18), and obtain

$$E[S_q] \leq (\alpha_2 + \beta_1)\delta^m E[M_q] + \frac{\beta_2}{\beta_3} E\left[\ln(M_q + 1)\right] + \frac{1}{1-\gamma},$$

where we have also used the fact $N_q \leq M_q$. The above inequality holds for all $q \in \{1, \ldots, T/\delta\}$; adding over all q, and using eq. (3.13), we obtain

$$\begin{aligned} E[S] &= \sum_{q=1}^{T/\delta} E[S_q] \\ &\leq (\alpha_2 + \beta_1)\delta^m \sum_{q=1}^{T/\delta} E[M_q] + \frac{\beta_2}{\beta_3} \sum_{q=1}^{T/\delta} E\left[\ln(M_q + 1)\right] + \frac{1}{1-\gamma}\frac{T}{\delta}. \end{aligned} \tag{3.28}$$

Since the logarithmic function is concave, Jensen's inequality yields

$$\sum_{q=1}^{T/\delta} E\left[\ln(M_q + 1)\right] \leq \sum_{q=1}^{T/\delta} \ln(E[M_q] + 1) \leq \frac{T}{\delta} \ln\left(\frac{\delta}{T} \sum_{q=1}^{T/\delta} E[M_q] + 1\right).$$

This together with eqs. (3.12) and (3.28) implies that

$$E[S] \leq (\alpha_2 + \beta_1)\delta^m E[M] + \frac{\beta_2}{\beta_3}\frac{T}{\delta} \ln\left(\frac{\delta}{T} E[M] + 1\right) + \frac{1}{1-\gamma}\frac{T}{\delta}. \tag{3.29}$$

By eq. (3.11), we have $\delta^m E[M] = T/\delta = T^{m/(m+1)}(E[M])^{1/(m+1)}$ and $(\delta/T)E[M] = 1/\delta^m = (E[M]/T)^{m/(m+1)}$; since $\delta < \epsilon_0$, we have $(\delta/T)E[M] > 1/\epsilon_0^m$, which gives (after some algebra) that

$$\ln\left(\frac{\delta}{T} E[M] + 1\right) \leq \ln\left(\frac{\delta}{T} E[M]\right) + \ln(\epsilon_0^m + 1).$$

Using these facts, it follows from eq. (3.29) that

$$\begin{aligned} E[S] \leq &\left[\alpha_2 + \beta_1 + \frac{1}{1-\gamma} + \ln(\epsilon_0^m + 1)\right] T^{m/(m+1)}(E[M])^{1/(m+1)} \\ &+ \frac{m\beta_2}{(m+1)\beta_3} T^{m/(m+1)}(E[M])^{1/(m+1)} \ln(E[M]/T); \end{aligned}$$

this proves the lemma for the case $\delta < \epsilon_0$. □

3.5. Discussion. First, we discuss a generalization of Theorems 2.4 and 2.5. Let us suppose that the distribution of the delays is as described by Assumption 2.3, except that it is shifted to the right by a positive amount. (For example, the delay could be the sum of a positive constant and an exponentially distributed random variable.) As far as a particular link is concerned, this

change of the probability distribution is equivalent to delaying the time that each message is transmitted by a positive constant. Such a change does not affect the number of overtakings that occur on any given link. Thus, Lemmas 3.3.1 and 3.4.1 remain valid, and Theorems 2.4 and 2.5 still hold.

Next, we discuss the tightness of the bounds in Theorems 2.4 and 2.5. These bounds are obviously tight if $d = O(1)$, that is, for sparse processor graphs. In general, we are not able to establish that our upper bounds are tight. However, it can be shown that the bound in Lemma 3.3.1 is tight and the bound in Lemma 3.4.1 is tight within a logarithmic factor [Tsitsiklis and Stamoulis 1990]. Since these lemmas are the key to our proofs, we are led to conjecture that the upper bound of Theorem 2.4 is tight and that the upper bound of Theorem 2.5 is tight within a logarithmic factor.

In our results, we have assumed that the delay of all messages are independent and identically distributed, even for messages on different links. If we assume that message delays are independent but that the mean delay is different on different links, then our results are no more valid. In fact, under those circumstances, one can construct examples in which the number of transmitted messages over a given time interval increases exponentially with the number of processors.

4. *Some Remarks on the Time Complexity*

In this section, we still assume that the model of Section 2 is in effect. Furthermore, to simplify the discussion, let us assume that if a message reception triggers the transmission of messages by the receiving processor, these latter messages are transmitted without any waiting time.

Consider the asynchronous Bellman–Ford algorithm and consider a path $(i_k, i_{k-1}, \ldots, i_1, 0)$ from a node i_k to the destination node 0. We say that this path has been *traced* by the algorithm if there exist times $t_1, t_2, \ldots, t_k$ such that a message is transmitted by processor i_j at time t_j and this message is received by processor i_{j+1} at time t_{j+1}, $j = 1, \ldots, k-1$. Under the initial conditions introduced in Section 1, it is easily shown [Bertsekas and Tsitsiklis 1984] that the shortest distance estimate x_{i_k} of processor i_k becomes equal to the true shortest distance as soon as there exists a shortest path from i_k to 0 that has been traced by the algorithm.

It is easily seen that under the model of Section 2, the time until a path is traced is bounded by the sum of (at most n) independent and identically distributed random variables. Assuming that the delay distribution has an exponentially decreasing tail, we can apply large deviations bounds on sums of independent random variables (e.g., the Chernoff bound [Chernoff 1952]). We then see that the time until the termination of the asynchronous Bellman–Ford algorithm is $O(n)$, with overwhelming probability. Furthermore, the expected duration of the algorithm is also $O(n)$.

From the above discussion and Theorem 2.4, we can conclude that, for $m = 1$, the number of messages until termination of the asynchronous Bellman–Ford is $O(n^2d^3)$, with overwhelming probability.[3] Similarly, for $m > 1$, the corresponding upper bound is $O(n^2d^{2+(1/m)}(\ln d)^{1+(1/m)})$. We note that for sparse graphs [i.e., when $d = O(1)$], the asynchronous Bellman–Ford has very good communications complexity, equal to the communication complexity of its synchronous counterpart.

It should be clear at this point that the above argument is not specific to the Bellman–Ford algorithm. In particular, any asynchronous algorithm with polynomial average time complexity will also have polynomial communication complexity, on the average.

5. *Different Models*

We have established so far that (under the assumption of independent and identically distributed message delays) the average communication complexity of asynchronous distributed algorithms is quite reasonable. In particular, discarding messages that are overtaken by others is a very effective mechanism for keeping the number of messages under control.

Modeling message delays as independent and identically distributed random variables seems reasonable when a "general mail facility" is used for message transmissions, and the messages corresponding to the algorithm are only a small part of the facility's load. On the other hand, for many realistic multiprocessor systems, the independent and identically distributed assumption could be unrealistic. For example, any system that is guaranteed to deliver messages in the order that they are transmitted (FIFO links) will violate the independent and identically distributed assumption (unless the delays have zero variance). This raises the issue of constructing a meaningful probabilistic model of FIFO links. In our opinion, in any such model (and, furthermore, in any physical implementation of such a model) the links have to be modeled by servers preceded by buffers, in the usual queuing-theoretic fashion. We discuss such a model below.

Let us assume, for concreteness, that each link consists of an infinite buffer followed by a server with independent and identically distributed, exponentially distributed, service times. In this setup, the following modification to the algorithm makes the most sense: Whenever there is a new arrival to a buffer, every message that has been placed earlier in that same buffer, but has not yet been "served" by the server, should be deleted. This modification has no negative effects on the correctness and termination of an asynchronous distributed algorithm. Furthermore, the rate at which a processor receives messages from is neighbors is $O(d)$. This is because there are at most d incoming links and the arrival rate along each link is constrained by the service rate of the server corresponding to each link. Each message arrival triggers $O(d)$ message transmissions. We conclude that the expected communication complexity of the algorithm will be $O(nd^2T)$, where T is the running time of the algorithm.

We have once more reached the conclusion that asynchronous algorithms with good time complexity T will also have a good communication complexity.

Let us conclude by mentioning that an alternative mechanism for reducing the communication complexity of an asynchronous algorithm is obtained by introducing a "synchronizer" [Awerbuch 1985]. A synchronizer could result in a

[1] For $m = 1$, the formal argument goes as follows. If T is the random time until termination and $C(t)$ is the number of messages transmitted until time t, then

$$\Pr[C(\infty) \geq A_1A_2n^2d^3] \leq \Pr[T \geq A_1n] + \Pr[C(A_1n) \geq A_1A_2n^2d^3].$$

We bound $\Pr[T \geq A_1n]$ using the Chernoff bound, and we bound $\Pr[C(A_1n) \geq A_1A_2n^2d^3]$ using Theorem 2.4 and the Markov inequality.

communication complexity that is even better than the one predicted by Theorem 2.4 or by the calculation in this section. On the other hand, our results indicate that acceptable communication complexity is possible even without a synchronizer.

ACKNOWLEDGMENT. We are grateful to David Aldous for carrying out the calculation in Subsection 3.1, which suggested that a nice result should be possible for the general case. Furthermore, using another heuristic calculation, he suggested that the correct power of d in Theorem 2.5 is $d^{2+(1/m)}$.

REFERENCES

AWERBUCH, B. 1985. Complexity of network synchronization. *J. ACM, 32*, 4 (Oct.), 804–823.

BERTSEKAS, D. P. 1982. Distributed dynamic programming. *IEEE Trans. Automat. Control AC-27*, 610–616.

BERTSEKAS, D. P., AND GALLAGER, R. G. 1987. *Data Networks*. Prentice-Hall, Englewood Cliffs, N.J.

BERTSEKAS, D. P. AND TSITSIKLIS, J. N. 1989. *Parallel and Distributed Computation: Numerical Methods*. Prentice-Hall, Englewood Cliffs, N.J.

CHERNOFF, H. 1952. A measure of asymptotic efficiency for tests of a hypothesis based on a sum of observations. *Ann. Math. Stat. 23*, 493–507.

TSITSIKLIS, J. N., AND STAMOULIS, G. D. 1990. On the average communication complexity of asynchronous distributed algorithms. Tech. Rep. LIDS-P-1986, Laboratory for Information and Decision Systems, MIT, Cambridge Mass.

RECEIVED JUNE 1990; REVISED DECEMBER 1993; ACCEPTED JULY 1994