A MULTIPLE COPY APPROACH FOR DELIVERING MESSAGES UNDER DEADLINE CONSTRAINTS

Parameswaran Ramanathan
Dept. of Elec. & Comp. Eng.
University of Wisconsin
Madison, WI 53706-1691
parmesh@engr.wisc.edu

Kang G. Shin
Dept. of Elec. Eng. & Comp. Sc.
The University of Michigan
Ann Arbor, MI 48109-2122
kgshin@eecs.umich.edu

Abstract

This paper proposes a scheme to minimize the expected recovery cost incurred by a distributed real-time system as a result of messages failing to meet their deadline. The scheme is intended for distributed systems with point-to-point interconnection topology. The goal of minimizing the expected cost is achieved by sending multiple copies of a message through disjoint routes and thus increasing the probability of successful message delivery within the deadline. The number of copies of each message to be sent is determined by optimizing the tradeoff between the increase in the message traffic due to additional copies and the decrease in the probability of a message missing its deadline.

1 Introduction

Due mainly to their potential for high performance and high reliability, distributed systems with point-to-point interconnection networks are increasingly being used in the control of critical real-time applications such as avionics, life-support systems, nuclear power plants, drive-by-wire applications and computer-integrated manufacturing systems. A common feature of all these applications is that they can fail not only due to loss of components but also due to time-critical tasks failing to complete their execution within the assigned deadline. Since tasks have to often exchange messages with other tasks in order to accomplish their common goal, it is important to ensure a reliable and timely delivery of all messages sent by time-critical tasks. By "timely" we mean each message must delivered within the deadline assigned to it.

When a message misses its assigned deadline, the system incurs a cost to recover the information contained in that message. This cost will be small if the system can recover by simply ignoring the late message. For example, if the late message contained a sensor value, then the related tasks may be able to estimate that value at the risk of introducing some error in the current computation. On the other hand, if the recovery action involves execution of some additional tasks, then the cost depends on the nature of the additional tasks. In the worst case, the cost will be extremely large, especially if the recovery action fails either due to lack of resources or time[1]. The primary objective of the scheme proposed in this paper is to minimize the expected cost incurred by the system as a result of messages failing to meet their deadlines.

Most existing schemes for message passing under deadline constraints are mainly intended for distributed systems in which the nodes are interconnected through a multiple access network such as a broadcast bus or a token ring [5, 9, 10]. However, due to the inherent susceptibility of multiple access networks to single-point failures, distributed systems with a point-to-point interconnection topology such as meshes and hypercubes are increasingly being used in critical real-time applications. Therefore, in this paper, we propose a scheme specifically intended for a distributed system with a point-to-point interconnection topology. Further, instead of proposing a policy for selecting the next message for transmission as in other schemes, we propose a scheme to reduce the delay in the delivery of the selected message. Each node uses an existing policy like Minimum-Deadline-First to select the next message for transmission and then uses the proposed scheme to reduce delivery delays.

Smaller delivery times are achieved by making use of the multiple disjoint routes that exist in a point-to-point interconnection topology. Depending on the

---

The work reported here is supported in part by the National Science Foundation under Grant MIP-9009154 and the Office of Naval Research under Contract No. N00014-K-85-00122. Any opinions, findings, and conclusions or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the NSF or ONR.

[1] Technically, the cost may be infinite if the system fails as a result of a message failing to meet its deadline. However, for analytic tractability, we will assume that the cost is finite, albeit very large as compared to other costs in the system.

CH2985-0/91/0000/0300/$01.00 © 1991 IEEE

deadline and the criticality of the message, the originating/source node chooses to send one or more copies of the message through disjoint routes. A message that is sent through more than one disjoint route has a higher probability of meeting the deadline because at least one of the copies is likely to traverse a less congested route and hence get delivered before the deadline. However, by sending more copies, the average message traffic on the network increases, thereby possibly increasing the time required to deliver each copy. In the proposed scheme, the number of copies to be sent is chosen in such a way that the expected cost incurred as a result of messages failing to meet their deadlines is minimized.

Numerical experiments were carried out to gauge the benefits of the proposed approach. In all the cases considered, the reduction in the expected cost were in the order 70–90% for realistic network loads. In addition, the proposed approach has an advantage that it is less likely to be susceptible to node/link failures. The likelihood of losing a critical message due to a faulty node/link is reduced because the proposed approach will send more than one copy of a very critical message through disjoint routes. The use of multiple copies to ensure reliable delivery in the presence of component failures has been addressed by others [2, 6]. Our work is unique in the sense that it deals with the problem of failures due to congestion. Although one could integrate our approach with other schemes for ensuring reliable delivery, description of such an integrated scheme is beyond the scope of this paper.

This paper is organized as follows. Section 2 describes the target distributed system. Section 3 describes the proposed scheme. The objective used to determine the optimal number of copies is formalized in Section 4.2 and a numerical example is presented in Section 4.3. The paper concludes with Section 5.

2 System Model

The target system is comprised of a set of processing nodes communicating through a point–to–point interconnection topology. Each node executes a set of real-time tasks, each with an assigned deadline. Based on the *task deadline*, a deadline is assigned to every message the task sends (i.e., *message deadline*). Some cost will be incurred whenever a message fails to meet its assigned deadline. We want to develop a message passing scheme that minimizes the expected cost incurred as a result of messages missing their deadlines.

The messages generated in the system and the message passing scheme are assumed to have the following characteristics:

M1. Messages are generated at each node according to a Poisson process with rate λ_g.

M2. The message lengths are exponentially distributed with mean $\bar{\ell}$. This in turn implies an exponentially distributed message service time at each node with rate μ, where $\mu = \alpha/\bar{\ell}$ for some constant α that represents the time required to transmit one byte if $\bar{\ell}$ is expressed in bytes.

M3. Messages belong to one of n criticality classes. With each class i, we associate a cost K_i that represents the average cost incurred as a result of any one of the messages in that class missing its deadline. The probability of a generated message belonging to class i is c_i.

M4. Nodes have no preferential direction for communication. The probability of a node sending a message to another node that is h hops away is q_h. This probability is assumed to be independent of the criticality class of the message.

M5. The deadline of a message that has to traverse h hops has a known probability density function, $f_{Dh}(u)$, with mean η_h.

M6. Messages with higher criticality[2] are selected (for transmission) ahead of messages with lower criticality. Among messages with same criticality, the selection is based on a first-come-first-served basis.

Modeling assumptions M1–M5 define the characteristics of a message in our system while M6 specifies the message scheduling algorithm used by each node. It is possible to increase the probability of a message meeting its deadline by using policies such as Minimum–Deadline–First among messages of the same criticality class. However, we have chosen to use the first-come-first-served policy because of its analytic tractability.

In addition to M1–M5, we will make the following two assumptions only for the purpose of efficiently determining the number of copies of each message to be sent. Even without these two assumptions, the benefits of the multiple copy approach are substantial as shown later in Section 4.3.

M7. The length of a message is regenerated at each intermediate node of its route and is independent of its length at other intermediate nodes.

---

[2] A message is said to have *higher criticality* if it belongs to class with a larger average cost.

M8. The delivery time of messages through disjoint routes are mutually independent.

M7 is commonly referred to as Kleinrock's independence assumption [4]. Although M7 is unrealistic in practice, several empirical studies have shown that the mean message delivery times computed using this assumption closely matches the actual mean message delivery times [4]. However, the knowledge of mean message delivery times is not sufficient to determine the expected cost incurred as a result of messages failing to meet their deadlines. To compute the expected cost, we need to determine the probability distribution function (PDF) of message delivery times.

The PDF of message delivery times cannot be accurately determined using M7 and M8. However, even a crude approximation of the PDF seems to be sufficient to determine the number of copies of each message that results in a substantial reduction in the expected cost incurred by the system. This result is demonstrated through several numerical experiments with the help of a simulator that does not make use of these two assumptions (see Section 4.3).

3 Proposed Scheme

The basic idea of the proposed scheme is as follows. When a message is selected for transmission at the source node, the message scheduler at that node determines the number of copies of this message to be sent on the network. These copies will be routed to the destination node through a set of disjoint routes. The rationale behind this approach is that more the number of copies, the higher the probability that at least one of these copies will reach the destination before the deadline. However, as the number of copies increases, the message traffic on the network increases, thereby increasing the delivery time for each copy. There is therefore a tradeoff between the number of copies of each message and the expected cost incurred as a result of messages missing their deadlines.

Intuitively, the number of copies selected by the source node for a given message should depend on three factors: its criticality and deadline, and the number of hops it has to traverse. For example, when the other two factors are identical, it is advantageous to send more copies of a message with higher criticality, or the message with a shorter deadline or the message that has to traverse fewer hops. However, determining the optimal number of copies to be sent based on all the three factors can be computationally very expensive. Therefore, in this paper, the number of copies for a particular message will be based only on the criticality and the number of hops the message has to traverse. Since our numerical experiments have indicated that the percentage reductions in the expected cost are in the range of 70–90% for realistic network loads, we believe that ignoring the deadline factor while determining the number of copies is not a major problem.

3.1 Formal Description

The message handler at each node can be described in terms of two concurrent processes: *S_process* and *D_process*. The *S_process* is responsible for transmitting the messages to the neighboring nodes while *D_process* is responsible for receiving the messages destined for the local node. A formal description of *S_process* and *D_process* are shown in Fig. 1. In this description each message is represented by a triple (c, d, h) where c is the criticality of the message, d is the deadline, and h is the number of hops the message has to traverse. The description is also made in terms of the following global parameters and functions.

ρ_g: The normalized message generation rate at each node. That is, it is the ratio of the message generation rate at each node (λ_g) to the message service rate at that node (μ).

Copies$[\rho_g][c][h]$: A pre-computed array containing the number of copies of a message to be sent given the normalized message generation rate (ρ_g), the criticality of the message (c), and the number of hops the message has to traverse (h). An algorithm to compute this array is discussed in Section 4 2.

next( ): Returns the next message to be transmitted based on some message scheduling policy. This call blocks if there are no messages for transmission.

send(M, k): Determines k disjoint routes to the destination of message M and then transmits copies of the message to the neighboring nodes identified by the disjoint routes.

relay(M): Relays a single copy of the message to the neighbor specified in the route of the message.

receive( ): Returns a message that has not been processed from the set of messages that are destined for the local node. This call blocks if there are no messages for processing.

S_process : This process is responsible for transmitting the messages from a node. If the message being transmitted was generated locally, then *S_process* determines the number of copies of the message to be sent and the route for each of the copies. It then transmits the copies to the neighbors identified by the disjoint routes. On the other hand, if the message

```
S_process
loop forever
    $(c, d, h)$ := next( );
    if $(c,d,h)$ was generated locally
        no_of_copies := Copies[$\rho_g$][$c$][$h$];
        send(($c$, $d$, $h$), no_of_copies);
    else
        relay(($c$, $d$, $h$));
    endif
endloop.

D_process
loop forever
    $(c, d, h)$ := receive( );
    if $(c, d, h)$ is the first copy to be received
        deliver $(c, d, h)$ to the waiting process;
    else
        discard $(c, d, h)$;
    endif
endloop.
```

Figure 1: *S_process* and *D_process*

originated from some other node, then *S_process* simply relays the message to the neighbor specified in the route of the message.

When *S_process* has to determine the number of copies of a message to be sent, it looks up a table of pre-computed values. The main advantage of this table-driven scheme is that the overhead is minimal when the system is in operation. At first glance, this scheme might seem to be memory-intensive. However, as we will show later, the optimal number of copies of each message does not change rapidly with ρ_g. Therefore, one can easily partition the range of ρ_g, namely $[0, \beta)$, for some $\beta < 1$, into a few disjoint intervals and assume that the optimal solution does not change within each interval. It is necessary to restrict ρ_g to the range less than $[0, 1)$ because each node has to service not only all the messages generated at that node, but also all the transit messages that are being relayed through the node. The value of β depends on the topology, the communication pattern between the real-time tasks and the message routing algorithm used. For a given topology, one can easily compute the value of β if the associated queueing network has a product-form solution [3]. The assumptions M1–M8 are sufficient to ensure a product-form solution for the associated queueing network. For a regular homogeneous topology satisfying the assumptions M1–M8, one can show that the range of ρ_g (or equivalently the range of λ_g) should be such that $\rho_t = \lambda_t/\mu < 1$ where λ_t is related to λ_g by the equation $\lambda_t = \lambda_g \sum_{h=1}^{H} h \cdot q_h$, where H is the maximum number of hops a message has to traverse in this topology and q_h is as defined in M4.

If more than one copy of the message are to be sent, then *S_process* has to determine a set of disjoint routes (one for each copy) to the destination. This can be easily done for most regular topologies such as the hypercube [8] and C-wrapped hexagonal mesh [1].

D_process :This process is responsible for receiving the messages destined for a local node. Since the local node will receive one or more copies of each message, *D_process* has to identify the duplicate copies of each message. This can be done by using message identifiers. Such techniques are currently being used to recover from lost messages. Therefore, there is no real additional penalty at the destination end in using the proposed scheme.

It is clear from the above description that the time overhead of using the proposed scheme is minimal while the system is in operation. However, to demonstrate the viability of our scheme, we need to show that:

- The pre-computed table of values used by *S_process* can be easily determined in an off-line mode.
- There is significant reduction in the expected cost incurred as a result of using the proposed scheme.

The rest of this paper addresses these two issues.

4 Determination of the Optimal Number of Copies

In this section, we first describe the derivation of the probability distribution function (PDF) of message delivery times in a system satisfying the assumptions M1–M8 in Section 2. This PDF is required to analytically estimate the expected cost incurred as a result of messages missing their deadlines. An expression for this analytic estimate using the PDF is also derived in this section. Finally, in this section, we present some numerical examples to illustrate the benefits of using the multiple copy approach.

4.1 PDF of Message Delivery Times

It is very difficult to compute the exact PDF of message delivery times in a queueing network with multiple classes of messages and prioritized service. Therefore, we approximate the PDFs as follows. Without loss of generality, one can assume $K_1 \geq K_2 \geq \cdots \geq K_n$. Consider the messages in class 1. Since the messages in this class have the highest priority, their delivery times are unaffected by the existence of the other classes of messages. Therefore, for this class, the system behaves as a "product-form network" [3] in which each node acts as an independent M/M/1 queue with arrival rate $\lambda_{t,1}$ and service rate μ, where $\lambda_{t,1}$ is the total throughput at each node due to messages of class 1 generated at that node and in-transit messages of class 1 relayed through that node. For a regular homogeneous topology, $\lambda_{t,1}$ can be derived from λ_g as:

$$\lambda_{t,1} = c_1 \lambda_g \sum_{h=1}^{H} h \cdot q_h$$

where c_1, λ_g, H, and q_h are as introduced in Section 2.

From the above facts, one can derive the PDF of delivery times for a message of class 1 traversing h hops [4] as:

$$\begin{aligned} F_h(\lambda_g, \mu, t \mid 1) &= 1 - e^{-\Lambda_1 t} \sum_{k=1}^{h-1} \frac{(\Lambda_1 t)^k}{k!} && (4.1) \\ \text{where} \quad \Lambda_1 &= \mu\left(1 - \frac{\lambda_{t,1}}{\mu}\right). && (4.2) \end{aligned}$$

For other classes, then computation of the PDF of message delivery times is not easy because these messages have to wait behind messages from higher priority classes. Therefore, we approximate the PDF by using expressions similar to that of class 1 except with higher message generation rates. That is, let the PDF of delivery times of any class $i = 1, 2, \ldots, n$ be

$$\begin{aligned} F_h(\lambda_g, \mu, t \mid i) &= 1 - e^{-\Lambda_i t} \sum_{k=1}^{h-1} \frac{(\Lambda_i t)^k}{k!} && (4.3) \\ \text{where} \quad \Lambda_i &= \mu\left(1 - \frac{\lambda_{t,i}}{\mu}\right) \quad \text{and} \\ \lambda_{t,i} &= \lambda_g \sum_{k=1}^{i} c_k \sum_{h=1}^{H} h \cdot q_h. \end{aligned}$$

4.2 Formulation of the Objective Function

In this subsection, we formalize the objective used to compute the number of copies of a message to be sent given its criticality class and deadline, and the number of hops it has to travel. The objective is to minimize the expected cost incurred as a result of messages missing their deadlines.

The expected cost can be expressed as

$$E[\text{cost}] = \sum_{i=1}^{n} \sum_{h=1}^{H} c_i \cdot q_h \cdot K_i \cdot P[\text{mess. missing deadl.} \mid i, h], \quad (4.4)$$

where H is the maximum number of hops a message has to traverse in the interconnection network, c_i, K_i and q_h are as defined in modeling assumptions M1–M4, and $P[\text{mess. missing deadl.} \mid i, h]$ is the conditional probability of message missing its deadline given that it belongs to class i and that it has to traverse h hops.

The $P[\text{mess. missing deadl.} \mid i, h]$ is a function of the network traffic and the deadline of the message. It can be computed from the PDF of message delivery times as follows. Let $m_i(\rho_g, h, t)$ denote the design parameter that is equal to the number of copies used by the source node for a message of class i, traversing h hops and having a deadline t when the normalized message generation rate at each node is ρ_g. Then, the increased message traffic on the network caused by the additional copies of each message can be thought of as an increased message generation rate. That is, the effective message generation rate at each node, λ_e, is given by

$$\lambda_e = \lambda_g \cdot \sum_{i=1}^{n} \sum_{h=1}^{H} \int_{t=0}^{\infty} c_i \cdot q_h \cdot m_i(\rho_g, h, t) \cdot f_{Dh}(t) dt. \quad (4.5)$$

In the above equation, $c_i \cdot q_h$ is the probability of a message being in class i and traversing h hops, $m_i(\rho_g, h, t)$ is the number of copies of that message and $f_{Dh}(t)$ is the probability density function of the deadline of the message. If the multiple copy approach is not being used, then $m_i(\rho_g, h, t) = 1$ for all i, ρ_g, h and t and therefore, $\lambda_e = \lambda_g$. However, if the multiple copy approach is being used, then $m_i(\rho_g, h, t) > 1$ for some i, ρ_g, h and t. In that case, $\lambda_e > \lambda_g$.

For this effective message generation rate, the probability of timely delivery of a single copy of a given message of class i with deadline t and traversing h hops is $F_h(\lambda_e, \mu, t \mid i)$ where $F_h(\lambda_e, \mu, t \mid i)$ is given by Eq. (4.3). Therefore, from M8, if $m_i(\rho_g, h, t)$ copies of the message are sent, then

$$P[\text{mess. missing deadl.} \mid i, h] = \int_{t=0}^{\infty} (1 - F_h(\lambda_e, \mu, t \mid i))^{m_i(\rho_g, h, t)} \cdot f_{Dh}(t) dt. \quad (4.6)$$

Combining Eqs. (4.4) and (4.6), we get

$$E[\text{cost}] = \sum_{i=1}^{n} \sum_{h=1}^{H} \int_{t=0}^{\infty} c_i \cdot q_h \cdot K_i \cdot f_{Dh}(t) \cdot [1 - F_h(\lambda_e, \mu, t \mid i)]^{m_i(\rho_g, h, t)} dt. \quad (4.7)$$

The problem then is to determine a set functions $m_i(\rho_g, h, t)$ that minimizes the expected cost in Eq. (4.7). Clearly, the above objective is difficult to minimize. We therefore impose the following restrictions on the possible solutions for $m_i(\rho_g, h, t)$.

1. $m_i(\rho_g, h, t_1) = m_i(\rho_g, h, t_2)$ for all $t_1, t_2 \in [0, \infty)$.
2. $m_i(\rho_g, h, t) \leq M_h$ for some constants M_h, $h = 1, 2, \ldots, H$ and for all $t \in [0, \infty)$.
3. $K_i > K_j$ implies $m_i(\rho_g, h, t) \geq m_j(\rho_g, h, t)$ for all $t \in [0, \infty)$.
4. $m_i(\rho_g, h-1, t) \geq m_i(\rho_g, h, t)$ for all $h = 2, 3, \ldots, H$ and for all $t \in [0, \infty)$.

The first condition means that the deadline of the message is ignored when making decisions on the number of copies to be sent. The second condition arises from the connectivity of the interconnection topology. Basically, M_h denotes the total number of disjoint routes between any two nodes that are h hops away from each other. The third condition implies that more copies of a message are sent if the message has higher criticality. This is intuitively appealing because a higher cost will be incurred if a more critical message misses its deadline. The fourth condition implies that more copies of a message are sent if the message has to traverse fewer hops. This is because messages traversing fewer number of hops are more likely to miss their deadlines since they have fewer chances to make up the unduly large delays they may encounter at some nodes[3]. This, of course, will be partially offset by the fact that messages traversing more hops are more likely to encounter unduly large delays at some nodes. However, we observed from a simulator (cf. Section 4.3) that the first situation dominated the second one and messages traversing fewer number of hops were more likely to miss their deadlines.

With the above simplifications, it is possible to search the space for an optimal solution. The results obtained by minimizing the simplified objective for a C-wrapped hexagonal mesh and hypercube topologies are given in the following section. Due to the above simplifications and due to the approximations made in estimating the PDF of message delivery times, the results obtained are sub-optimal. The numerical examples in the following section also show the further improvements that can be achieved by using the multiple copy approach if we could efficiently determine the optimal number of copies without any of the above simplifications.

4.3 Numerical Examples

In this section, the benefits of the proposed scheme are illustrated through a numerical example for a C-wrapped hexagonal mesh topology [1]. Similar examples were conducted for a hypercube topology and the results for those examples are presented in [7].

The numerical experiments were conducted as follows. Given a set of input parameters, the heuristic proposed in Section 4.2 was used to determine the number of copies to be sent for a given message. The number of copies as determined by the heuristic was used in a simulator that implemented both the traditional single copy approach and the proposed multiple copy approach. This allowed us to compare the expected cost incurred in the traditional single copy approach and the proposed multiple copy approach without some of the simplifying assumptions made in the heuristic method. For instance, the simulator relied only on assumptions M1–M6 and not on assumptions M7 and M8. The simulator also took into account other realistic constraints such as assuming that all messages are at least 64 bytes long to account for header information. Further, unlike the heuristic method, the simulator did not make any assumptions about PDF of message delivery times. Consequently, the reduction in the expected cost observed in the simulator reflects the actual reduction one would observe in practice as a result of using the proposed scheme. In the rest of this section, we present the results so obtained from the simulator for the C-wrapped hexagonal mesh topology and the hypercube topology for various input parameters.

A C-wrapped hexagonal mesh topology [1] is a regular, homogeneous graph in which each node has six neighbors. The key aspect of this topology is that there are six disjoint routes between any two nodes. However, the six disjoint routes between a given pair of nodes are not necessarily of the same length. One of the reasons for studying this topology is to see whether the multiple copy scheme is advantageous even when the multiple routes are not of the same length.

Table 1 shows the number of copies as determined by the proposed heuristic method in a C-wrapped hexagonal mesh of dimension five. This table is provided to the system (in our case, the simulator) at startup time. When a node wants to send a message, it selects the number of copies based on the estimated network load, the shortest distance to the destination, the type of the destination depending on whether or not the destination lies on the principal axes (PA) and the criticality class of the message.[4] The table only

---

[3] We have implicitly assumed that in a good design messages traversing more hops will usually have larger deadlines.

[4] The source has to determine whether or not the destination lies on the principal axes because the lengths of six disjoint

Hops	Type	Load 0.1			Load 0.2			Load 0.6		
		Class			Class			Class		
		1	2	3	1	2	3	1	2	3
1	PA	3	2	2	3	2	2	2	1	1
2	PA	2	2	2	2	1	1	1	1	1
2	NPA	2	2	2	2	1	1	1	1	1
3	PA	2	1	1	2	1	1	1	1	1
3	NPA	2	1	1	2	1	1	1	1	1
4	PA	2	1	1	2	1	1	1	1	1
4	NPA	2	1	1	2	1	1	1	1	1

PA: in principal axes; NPA: not in principal axes

Table 1: Number of copies as determined by the proposed heuristic method

shows a few selected loads because, for loads between 0.2 and 0.6 the solution was the same as that for 0.2. Similarly, the solution was the same as that for 0.6 for loads in the range of 0.6 to 0.8. This is significant because the pre-computed set of copies can be stored for on-line use without a considerable memory overhead.

The other parameters that were used in determining these copies are described below. These parameters were chosen solely for the sake of illustration. We have conducted many experiments by varying these parameters. Although the actual results change with the parameters, the conclusions that we can draw from the following set of parameters are typical.

1. Messages were assumed to belong to one of three classes. The probabilities of a message being in class 1, 2 or 3 were 0.1, 0.25 and 0.65, respectively.
2. The costs incurred when a message of class 1, 2 or 3 misses its deadline were assumed to be 625.0, 25.0, 1.0.
3. The probability of a node communicating with any other node was assumed to be inversely proportional to the length of shortest route between the two nodes.
4. The deadline of a message was a function of its criticality class as well as the number of hops it had to traverse. Given the criticality class of a message and the number of hops it has to traverse, the deadline of that message was taken from a uniform distribution centered around five times the mean delivery time of that message when the network load is 0.3. The deadlines varied from 3/5ths to 7/5ths of the mean deadline. In particular, note that the deadline distribution did not depend on the actual network load.
5. The message service rate at each node was a function of its length. Each node was assumed to be capable of transmitting at a rate of 667,000 bytes/sec to a neighboring node in the hexagonal mesh.

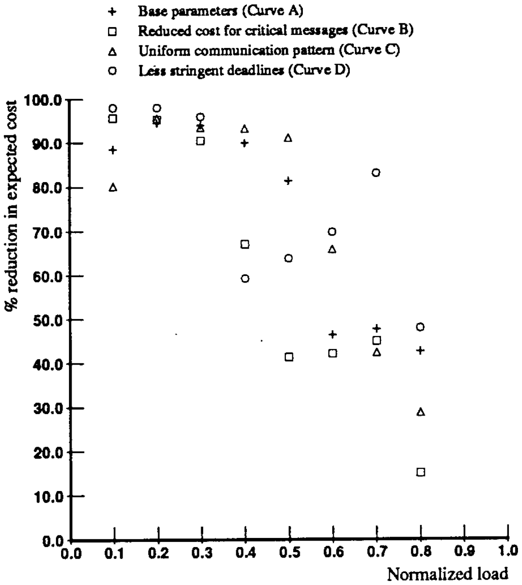

Figure 2: % reduction in expected cost for different input parameters in C-wrapped hexagonal mesh of dimension 5

Curve A in Fig. 2 shows the percentage reduction in the expected cost as observed in the simulator when multiple copies of a message were sent using Table 1. Curve B has the same parameters as in Curve A except that the costs incurred when a message of class 1, 2 or 3 misses its deadline were assumed to be 100.0, 10.0, 1.0 instead of 625.0, 25.0 and 1.0. Although the number of copies to be sent changes with the relative cost of different classes (not shown in the figure), the percentage reduction in the expected cost does not change significantly. In fact, as the ratio of cost of class 1 to cost of class 3 decreases, the number of copies to be sent decreases. When the ratio is one, all messages belong to the same class and it is not beneficial to send more than one copy of any message. At

---

routes in a C–wrapped hexagonal mesh depend on that factor [7].

Approach	Criticality class		
	1	2	3
Single copy	163	197	585
Multiple copy	4	139	535

Table 2: Count of messages out of 200,000 that failed to meet their deadlines

the other end — that is, as the ratio of class 1 to cost of class 3 increases — the number of copies to be sent does not increase beyond a certain point. Further, we observed through a series of experiments that the number of copies is not very sensitive to the actual ratio of the costs. This is important because we may not be able to accurately determine the cost incurred for different classes of messages.

Curve C has the same parameters as Curve A except that the probability of a node communicating with other nodes was assumed to be independent of the distance between the two nodes. We assumed that each node is equally likely to communicate with every other node. Curve D also has the same parameters as Curve A except that the deadline constraints are less stringent. Given a message the deadline of that message was taken from a uniform distribution centered around seven times (as opposed to five times for curve A) the mean delivery time of that message when the network load is 0.3. The deadlines varied from 5/7ths to 9/7ths of the mean deadline. Here, again the percentage reductions in the expected cost are substantial.

Table 2 shows the the actual count of messages out of a total of 200,000 messages that failed to meet their deadline in the traditional single copy approach and the multiple copy approach when the network load was 0.3 and the other parameters as in Table 1. It is clear from this table that there is a considerable reduction in the number of highly critical messages that missed their deadline without much change in the number of non-critical messages missing their deadlines.

5 Conclusion

A scheme was proposed to minimize the expected cost incurred as a result of messages missing their deadlines. The scheme made use of the multiple disjoint routes in a point-to-point interconnection network to send one or more copies of a message depending the criticality and the number of hops the message has to traverse. A numerical example was presented to illustrate the reduction in the expected cost as a result of using the proposed approach. The numerical example showed that reductions of more than 70% can be achieved at low to moderate loads. At high loads the reductions were in the range of 10–40%. Since the proposed scheme imposes minimal overhead while the system is in operation, it can be easily implemented along with the existing schemes for message passing under deadline constraints to achieve substantial improvements in message delivery times.

References

[1] M.-S. Chen, K. G. Shin, and D. D. Kandlur, "Addressing, routing and broadcasting in hexagonal mesh multicomputers," *IEEE Transactions on Computers*, vol. 39, no. 1, pp. 10–18, January 1990.

[2] D. Dolev, "The Byzantine generals strike again," *Journal of Algorithms*, vol. 3, no. 4, pp. 14–30, 1982.

[3] J. R. Jackson, "Networks of waiting lines," *Operations Research*, vol. 5, pp. 518–521, 1957.

[4] L. Kleinrock, *Communication Nets: Stochastic Message Flow and Delay*, McGraw-Hill, 1964 (also Dover, 1972).

[5] J. F. Kurose, M. Schwartz, and Y. Yemini, "Controlling window protocols for time-constrained communication in multiple access networks," *IEEE Trans. Communications*, vol. 36, no. 1, pp. 41–49, January 1988.

[6] A. Orda and R. Rom, "Routing with packet duplication and elimination in computer networks," *IEEE Trans. on Communications*, vol. 36, no. 7, pp. 860–866, July 1988.

[7] P. Ramanathan and K. G. Shin, "A multiple copy approach for delivering messages under deadline constraints," Technical report ECE-91-6, Department of Electrical and Computer Engineering, University of Wisconsin-Madison, 1991.

[8] Y. Saad and M. H. Schultz, "Topological properties of hypercubes," *IEEE Trans. on Computers*, vol. 37, no. 7, pp. 867–872, July 1988.

[9] J. K. Strosnider, T. Marchok, and J. P. Lehoczky, "Advanced real-time scheduling using the IEEE 802.5 token ring," in *Proc. Real-Time Systems Symposium*, pp. 42–52, December 1988.

[10] W. Zhao and K. Ramamritham, "Virtual time CSMA protocols for hard real-time communications," *IEEE Trans. Software Engineering*, vol. SE-13, no. 8, pp. 938–952, August 1987.