



## Multipath Dissemination in Regular Mesh Topologies

D. Udaya Kumar Apandian,

Associate Professor,  
Department Of C.S.E,  
Bharath University – Selaiyur,  
Chennai – 600073, Tamil Nadu , India

V. Kaleeswari

Final Year – M.Tech (C.S.E)  
Department Of C.S.E  
Bharath University - Selaiyur  
Chennai – 600073, Tamil Nadu, India

---

**Abstract**—Mesh topologies are important for large-scale peer-to-peer systems that use low-power transceivers. The Quality of Service (QoS) in such systems is known to decrease as the scale increases. We present a scalable approach for dissemination that exploits all the shortest paths between a pair of nodes and improves the QoS. Despite the presence of multiple shortest paths in a system, we show that these paths cannot be exploited by spreading the messages over the paths in a simple round-robin manner; nodes along one of these paths will always handle more messages than the nodes along the other paths. We characterize the set of shortest paths between a pair of nodes in regular mesh topologies and derive rules, using this characterization, to effectively spread the messages over all the available paths. These rules ensure that all the nodes that are at the same distance from the source handle roughly the same number of messages. By modeling the multihop propagation in the mesh topology as a multistage queuing network, we present simulation results from a variety of scenarios that include link failures and propagation irregularities to reflect real-world characteristics. Our method achieves improved QoS in all these scenarios.

**Index Terms**—Wireless communication, network communications, packet-switching networks, routing protocols, mesh topology.

---

### Introduction

THE use of novel devices for computing and communication in highly engineered networked embedded systems such as streetlight management [1], reconfigurable conveyors [2], and critical infrastructures [3], presents new challenges and opportunities for monitoring and diagnostics. These systems contain a large number of nodes, with each node incorporating a tiny microcontroller, sensors, actuators, and an integrated low-power transceiver [4]. The nodes interact in a peer-to-peer manner over low-bandwidth wireless links to achieve the application objectives. To increase the number of simultaneous interactions between the nodes in the system, the transmission range of each node is limited so that it communicates directly only with its set of immediate neighbors; such an arrangement of nodes is referred to as a mesh topology [5].

Multihop communications are necessary in such systems to send messages from any source to any destination. For example, intermediate nodes must forward messages to a monitoring station from nodes that cannot communicate directly with the monitoring station. Routing protocols are used extensively in wired and wireless networks to support multihop communication [6]. Such protocols construct and maintain routing tables at each node by relying on system wide unique node identifiers. When the number of nodes is very large, such as in sensor networks, it is not feasible to use such identifiers. Several techniques, called dissemination methods, were developed at the network layer to regulate the flow of messages between nonadjacent nodes without relying on unique node identifiers or constructing routing tables using these identifiers [7], [8], [9]. In this paper, we consider highly engineered systems comprising of nodes arranged in a regular mesh topology. We focus on methods for effectively utilizing all the shortest paths available between a pair of nodes and present results to show that effective utilization of all the available paths significantly improves the Quality of Service (QoS). In many highly engineered systems, one can assume that the nodes have fixed relative locations. Often, the systems are designed to overlay on an underlying grid. For example, in automation systems, the regions demarcated by such a grid are called zones [10]; a zone is a commonly used abstraction to support the design, operation, and maintenance activities. Motivated by applications in such domains, we consider regular mesh topologies that arise by embedding the nodes in a 2 D Base grid. We show in Section 2 that several mesh topologies arise when the location in the 2D Base grid and the transmission range of the nodes change. Because the grid coordinates can specify the nodes, the shortest paths between any pair of nodes can be locally computed. Most routing protocols select only one of the shortest paths even when multiple such paths exist.

This results in reduced system level QoS [11]. We address the issue of how to effectively utilize all the shortest paths available. Since the resulting methods amount to a node making local decisions on how to distribute messages among its immediate neighbors, without having to dynamically construct any routing tables, we refer to this method of forwarding messages as dissemination in spite of the fact that nodes are identified by their global coordinates in the underlying 2D Basegrid. Because each node communicates directly with its immediate set of neighbors, there are multiple shortest paths between manypairs of nodes in amesh topology. Thenumber of such paths is limited by the

relative locations of the nodes. For example, the number of shortest paths between certain pairs of nodes is one, despite the mesh topology. We define a Contour as the union of all the shortest paths between a pair of nodes and present some results to precisely characterize the structure of contours. Using this structure, we show that when messages are spread in a round-robin manner, nodes along one path in the contour will always handle more messages than the nodes along other paths in the contour. Consequently, the benefits of the multiple paths cannot be fully realized. We then present a strategy for spreading messages to neighboring nodes that effectively exploits the available shortest paths and show that our rules for spreading the messages result in a balanced loading of all the available shortest paths.

We refer to this approach as Contour Guided Dissemination (CGD). CGD improves the QoS by disseminating the messages over all the available shortest paths. Typical application scenarios in which CGD will be useful are: 1) a node responding to a diagnostic query from an operator at a monitoring station, and 2) a node sending a recorded incident to a monitoring station in a surveillance application. CGD equitably disseminates the messages over all the available shortest paths in a manner that maximizes utilization of all available shortest paths. We present simulation results to demonstrate all these aspects. Routing protocols used in traditional wired and wireless networks are based on shortest path algorithms such as the Bellman-Ford algorithm [12] and Dijkstra's algorithm [6]. Similar protocols have been reported for ad hoc, wireless, and mobile networks [13], [14], [15], [16]. The QoS achieved in these systems has also been studied [17], [18], [19]. The dissemination method we describe in this paper is somewhat similar to a gradient dissemination scheme [13] with the cost metric being the deviation from evenness of load distribution on all available shortest paths. Recent efforts have focused on exploiting multipaths to improve the QoS in systems [7], [20], [21] using constrained node distribution models.

For example, in [21], the nodes are distributed randomly in a unit disk. In [7], the nodes are distributed in a narrow strip no wider than 0.86 times the Transmission range of a node. The multipath, multihop, approach we present in this paper also assumes a constrained distribution of the nodes, namely, nodes are on a 2D Basegrid. While there is no stochasticity in the node distribution, we precisely characterize the geometry of the shortest paths and show how to exploit it to achieve better QoS. In the current methods, the motivating factors for considering multipath routing include fault tolerance, higher aggregate bandwidth, and load balancing [22], [20], [23], [24], [25]. The QoS aspects of multipath routing have been addressed in [26], [27], [28], and [29]. The split multipath routing protocol maintains maximally edge-disjoint paths [30]. Braided multipaths discussed in [31] are useful when the routing is coupled with diagnostic or prognostic methods to select alternative paths. All such methods essentially focus on the discovery and maintenance of multiple paths that are useful under various constraints on the node distribution. In contrast, we assume regular mesh topologies and then precisely characterize the set of all shortest paths between any pair of nodes. We then use this geometric structure to propose rules for dissemination that result in all available shortest paths being utilized effectively so that QoS improves.

## 2. EMBEDDING FUNCTIONS

We consider mesh topologies that are obtained by embedding nodes on a 2D Basegrid and adjusting the transmission range of each node. Each location on a 2D Base grid is identified by a unique ordered pair  $\delta_i; j_P$ . The distance between two consecutive locations on the grid, i.e., between  $\delta_i; j_P$  and  $\delta_i; j_P + 1$  or between  $\delta_i; j_P$  and  $\delta_i + 1; j_P$ , is  $b$ . Let  $N$  denote a set of nodes. An embedding function  $_$  assigns a location on the 2D Basegrid and a transmission range to each node,  $n_i \in N$ . The location to which a node is assigned is assumed to be fixed for the application. Each node knows its own location and the number of neighbors it has. Further, we assume that all the neighbors of a node reliably receive messages sent by the node; however, in our simulations, we empirically explore robustness to temporary link failures. Each message contains the locations of the source and destination nodes as specified by the embedding function. Different mesh topologies are obtained by specifying different embedding functions. Suppose that the nodes in  $N$  must be embedded along  $R$  rows and  $C$  columns, starting at location  $\delta_X; Y_P$  of the 2D Basegrid. Let  $_q$  denote the embedding in which each node  $n_i \in N$  communicates directly with  $q$  neighbors. For  $q = 4$ , the embedding function yields a mesh topology. Since the transmission range for each node is  $b$ , it can only communicate with its four immediate neighbors. Similarly, for  $q = 8$ , the embedding function mesh topology shown in Fig. 1a. Similarly, other embeddings such as  $_6$  shown in Fig. 1b arise by changing the location and transmission radius of the nodes [33]. In the remainder of this paper, nodes that are assigned to location  $\delta_i; j_P$  on the 2D Basegrid are referred to as  $n_{i,j}$ . In the next section, we define contours and their structure precisely. The results for contours in the  $_8$  embedding are presented in detail. Contours in other embeddings are briefly discussed at the end of the section.

Fig. 2. Different

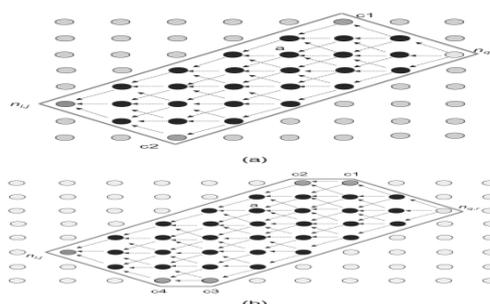


Fig. 2. Different contours in the  $_8$  embedding. Depending on the relative positions of the source and the destination, the

contour can be a rectangle as in (a) or a hexagon as in (b). (a) Contour in  $\_8$  enclosed by a Rectangle. (b) Contour in  $\_8$  enclosed by a Hexagon. Destination, the contour can be a rectangle as in (a) or a hexagon as in (b). (a)

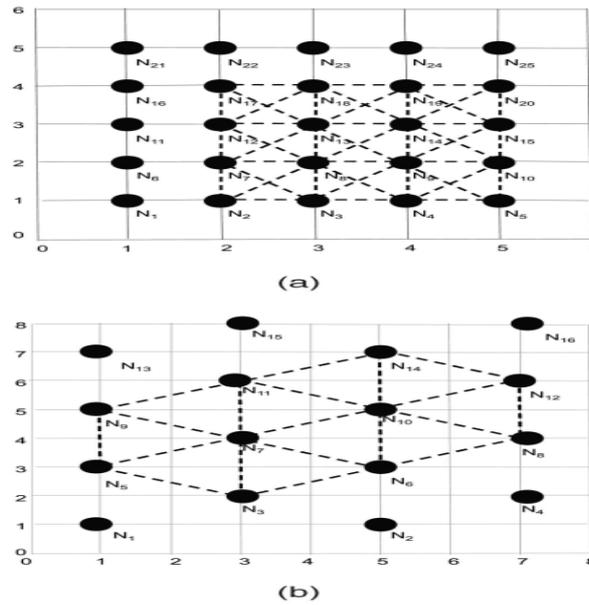


Fig. 1. Other embedding examples. (a)  $\_8$ . (b)  $\_6$ .

### 3. CONTOURS AND THEIR STRUCTURE

Definition 1. A contour is the union of all the shortest paths between a pair of nodes. Fig. 2 shows two example contours between a pair of nodes  $n_i:j$  and  $n_q:r$ . The shape of a contour depends on the relative locations of the source and the destination. For example, Fig. 2a shows a contour in  $\_8$  that is a rectangle enclosed by the four corner nodes:  $n_q:r$ ,  $c_1, n_i:j$ , and  $c_2$ . Fig. 2b shows another contour that is a hexagon enclosed by six corner nodes:  $n_q:r$ ,  $c_1$ ,  $c_2$ ,  $n_i:j$ ,  $c_4$ , and  $c_3$ .

Since the contour depends on the relative position of the source and destination nodes, its structure can be characterized by using only the differences in the x and y locations of the pair of nodes. To this end, we define the following quantities:

For example, in Fig. 2a,  $\_x \_4 7$ ,  $\_y \_4 3$ , and  $\_xy \_4 4$ , and in Fig. 2b,  $\_x \_4 8$ ,  $\_y \_4 3$ , and  $\_xy \_4 5$ . When the context is clear, we use  $\_x$ ,  $\_y$ ,  $\_xy$ , and  $\_y$  without specifying the source and the destination. The number of shortest paths in a contour increases as  $\_xy$  increases. In the  $\_8$  embedding, the length of the shortest path between two nodes  $n_i:j$  and  $n_q:r$  is  $d_{\delta} n_i:j; n_q:r \_4 \max\{ \_x; \_y \} + \delta$ . When viewed as the union of shortest paths, a contour is not symmetric with respect to the source and the sink because the paths have direction. In addition, a contour can also be viewed as a set of nodes. A node  $n_s:t$  is in a contour if and only if it is on a shortest path from the source to the destination. In this view, a contour is symmetric with respect to the source and the destination. From the well-known principle of optimality [12], a node,  $n_s:t$ , is in the contour of  $n_i:j$  and  $n_q:r$  if and only if  $d_{\delta} n_i:j; n_s:t \_4 d_{\delta} n_s:t; n_q:r \_4 d_{\delta} n_i:j; n_q:r$ .

#### 3.1 STRUCTURE OF CONTOURS IN $\_8$

We now present results to characterize the contour between two nodes  $n_q:r$  and  $n_i:j$ . To simplify the notation, we only consider the case when  $q > i$ ,  $r > j$ , and, whenever  $\_xy > 0$ ,  $q \_i > r \_j$ . All other cases can be handled in a similar manner.

Lemma 1. In  $\_8$ , the contour of  $n_i:j$  and  $n_q:r$  is a single shortest path if and only if  $\_xy \_4 0$ .

Proof. When  $\_xy \_4 0$ , we know that  $q \_i \_4 r \_j$ . The path  $\langle n_i:j; n_{i+1:j}; \dots; n_{q-1:r}; n_q:r \rangle$  is a shortest path in the embedding  $\_8$  between  $n_i:j$  and  $n_q:r$ . All other nodes  $n_s:t$  violate the criterion in (9) and are hence not in the contour. Next, we need to show that  $\_xy \_4 0$  when the contour is a single path. Suppose  $\_xy \_4 6 \_4 0$ , and without loss of generality, let  $\_x > \_y$ . By our notation, this means that the length of the shortest path is  $q \_i$ . Let  $K > 0$  be such that  $q \_i \_4 r \_j + K$ . Using (8), by direct calculation, we have  $d_{\delta} n_i:j; n_{i+K:j} \_4 d_{\delta} n_{i+K:j}; n_q:r \_4 d_{\delta} n_i:j; n_{q-K:r} \_4 d_{\delta} n_{q-K:r}; n_q:r \_4 d_{\delta} n_i:j; n_q:r$ . This implies

that nodes  $n_{q-K:r}$  and  $n_{i+K:j}$  have to be on optimal paths. However, using the distance criterion (9), it is easily seen that a path from  $n_i:j$  to  $n_q:r$  through both these nodes is not a shortest path. This implies we have to have at least two shortest paths. But since we are given the fact that the contour is a single path, we must have  $K \_4 0$ , which is same as  $\_xy \_4 0$ .  $\square$

Lemma 2. In  $\_8$ , the contour of  $n_i:j$  and  $n_q:r$  comprises two shortest paths that are enclosed by the nodes  $n_i:j$ ,  $n_{i+1:j}$ ,  $n_{q-1:r}$ , and  $n_q:r$  if and only if  $\_xy \_4 1$ .

Proof. Given the nodes as stated, we can use the distance criterion in (9) to verify that node  $n_{i+1:j}$  and node  $n_{q-1:r}$  are on

different shortest paths from  $n_{i;j}$  to  $n_{q;r}$ . It now suffices to note that nodes  $n_{i+2;j}$  and  $n_{q-2;r}$  do not satisfy the distance criterion in (9) and, hence, these are the only two shortest paths possible. Given two shortest paths between  $n_{i;j}$  and  $n_{q;r}$ , we now show that the paths are enclosed by the four nodes as stated. We know that a path from  $n_{q;r}$  to  $n_{i;j}$  is a shortest path only if  $\max\{\Delta_x, \Delta_y\}$  is reduced along each step of the path. Since by our notation,  $q_i > r_j$ ,  $\Delta_x$  must reduce along each step of the path. In the  $\Delta_8$  embedding, there are only three neighbors of  $n_{q;r}$  with a lower  $\Delta_x$ , namely,  $n_{q-1;r}$ ,  $n_{q-1;r+1}$ , and  $n_{q-1;r-1}$ .  $n_{q-1;r+1}$  does not satisfy the distance criterion in (9). Using similar arguments in the neighborhood of  $n_{i;j}$ , we conclude that only nodes  $n_{i+1;j+1}$  and  $n_{i+1;j}$  can be on any shortest path between  $n_{i;j}$  and  $n_{q;r}$ . Finally, we note that only the nodes on the two shortest Paths satisfy the distance criterion in (9), and hence, the four enclosing nodes are as stated.

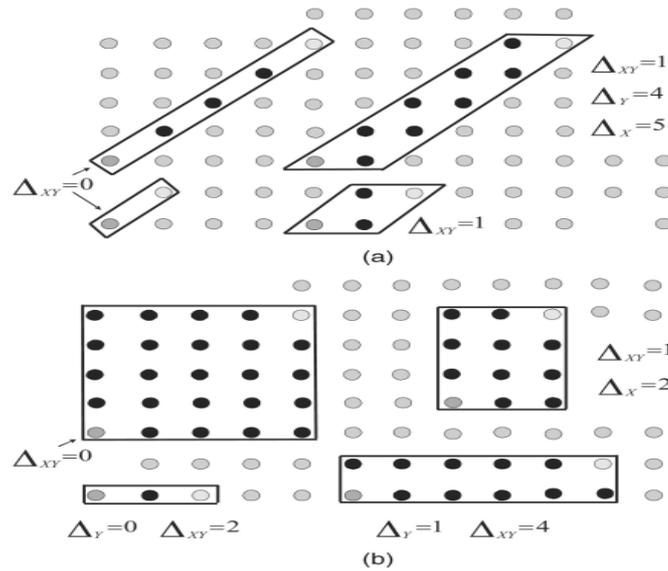


Fig. 3. Examples of different contour structures for various values of  $\Delta_{xy}$ . (a)  $\Delta_8$ . (b)  $\Delta_4$

### 3.2 Contours in Other Embeddings

The structure of contours in the embeddings  $\Delta_3$ ,  $\Delta_4$ , and  $\Delta_6$  are reported in [33]. In the  $\Delta_4$  embedding, the contour of  $n_{i;j}$  and  $n_{q;r}$  is a single path when either  $\Delta_x$  or  $\Delta_y \leq 0$ . Further, when  $\Delta_x > 0$  and  $\Delta_y > 0$ , the contour is always a rectangle bounded by the nodes  $n_{i;j}$ ,  $n_{i;r}$ ,  $n_{q;r}$ , and  $n_{q;j}$ . Fig. 3b shows a few example contours in the  $\Delta_4$  embedding. The structure of contours in the  $\Delta_6$  embedding is similar to the structure of contours in  $\Delta_4$ . In the  $\Delta_3$  embedding, contours have zero, one, or two pendant (i.e., adjacent only with a single edge) nodes [33].

### 3.3 Impact of Contour Structure on Dissemination

In a multihop routing scheme, each node receives messages from its upstream neighbor along some path and forwards this message to its downstream neighbor. In this paper, our interest is in mechanisms that exploit all the shortest paths. To utilize multiple paths, each node must receive messages from a set of upstream neighbors and forward these messages to a set of downstream neighbors. To utilize the results in this paper in practical applications, it is assumed that each message packet contains the location of both the source and the destination. As the messages propagate through the network, every intermediate node could easily compute (using, e.g., the distance criterion) which of its immediate neighbors are on a shortest path to the destination. (This is explained further in the next section). In addition, the results of this section allow the intermediate node to know its relative position in the contour. In Section 5, we show how this location information can be used to effectively exploit all the available shortest paths in a contour.

## 4. EFFECTS OF UNIFORM SPREADING

The first strategy we consider for spreading messages over all shortest paths will be called Uniform Spreading. This is a straightforward strategy where the source, as well as each intermediate node along every path in the contour, sends successive messages in a round-robin fashion to all its immediate neighbors in the contour. We present this algorithm and show that the nodes along one of the paths will always handle more messages than the nodes along other paths whenever this strategy is used. The following results characterize the nodes that are inside a contour.

### 4.1 Loading under Uniform Spreading in $\Delta_8$

Uniform spreading is achieved when all the nodes in the contour execute the Uniform Spreading algorithm. To characterize the effects of this algorithm, we first define what we call rows in a contour. Definition 2. A row in a contour is a set of nodes that are at the same distance (8) from the source. If the length of the shortest path is  $l$ , then there will be  $\lfloor l/8 \rfloor + 1$  rows in a contour. Note that the source and destination nodes are trivial rows with a single node in each. Fig. 4 shows a contour with eight rows that has a shortest path of 7 hops. The numbers in the circles show the number of packets handled by different nodes when all nodes use the uniform spreading algorithm and the source sends 99 packets. Even

though every node spreads the messages it handles over all the available neighbors, many nodes closer to the destination handle more messages than other nodes. This is the loading phenomenon that we will characterize precisely in this section. In the embedding  $\_8$ , notice in Fig. 4 that under uniform spreading, all the nodes  $n_{s;t}$  in rows  $1; 2; \dots; \_$  (where  $\_$  is as defined by (7)) of the contour divide the messages they handle into three parts and send each part along one of their three neighbors that are closer to the destination. Thus, we can conclude that the ratio of the messages handled by a node in a row to the smallest number of messages handled by another node in the same row follows the trinomial coefficients. The trinomial coefficient  $T\delta_{\_}; \_P$  is the coefficient of  $x_{\_}$  in the expansion of  $\delta 1 \_ x \_ \_ x 2P_{\_}$ . We use the following properties of trinomial coefficients in this section:

#### 4.2 Effects of Embedding on Loading

We observed similar loading effects under uniform spreading in  $\_4$  and  $\_6$ . Recall that the structure of the contours in  $\_6$  closely follows that in  $\_4$ . In these embeddings, only the nodes along paths on either the X or Y axis get loaded. If  $\_x > \_y$ , then the path along the X-axis in the contour will be loaded. Proofs of loading, similar to the ones presented for  $\_8$  in the preceding section, for contours in  $\_4$  and  $\_6$  follow from the properties of binomial coefficients.

### 5 OPTIMAL SPREADING

We now present an algorithm for spreading the messages so that all the available paths are effectively utilized. Recall that a row is a collection of nodes in the contour that are at the same distance from the source. Let  $w$  be the number of nodes in a row of a contour. We refer to  $w$  as the width of the row. If the source sends  $M$  messages and if every node in every row handles  $M/w$  messages, then we can say that the spreading is the best in the sense that all available paths are effectively used. This is the criterion of optimality that we choose. We will show that the algorithm presented in this section is optimal in this sense.

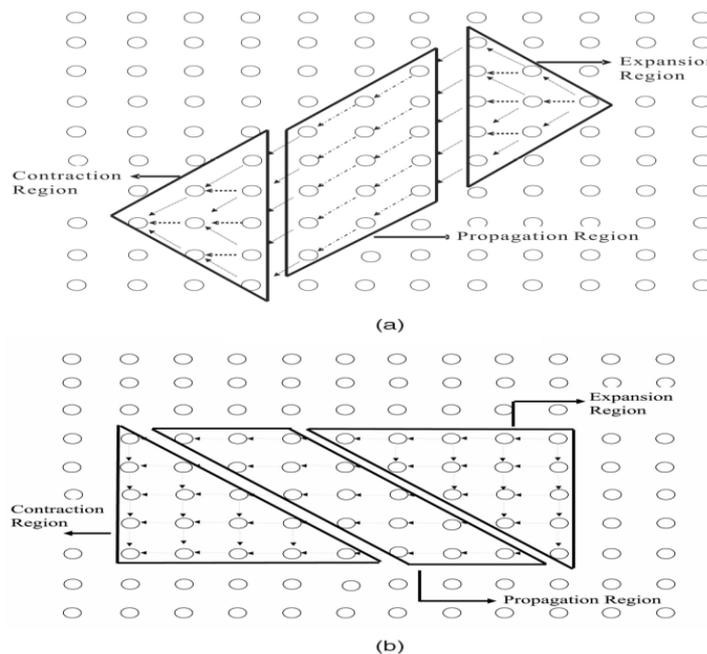


Fig. 6. Expansion, propagation, and contraction regions.

(a) Contour in  $\_8$ .

(b) Contour in  $\_4$ .

Fig. 6 shows examples of two contours—one in  $\_8$  and the other in  $\_4$ . Notice that the width of a row,  $w$ , varies with the row number in three ways in both the contours.  $w$  first increases monotonically, then remains constant, and then decreases monotonically. We refer to these three regions as the Expansion region, Propagation region, and Contraction region, respectively.

#### 5.1 Node Labeling

In any row, nodes on both sides of the middle node are labeled as  $1; 2; 3; \dots; bw=2c$ , as illustrated in Fig. 7. Let  $n_{mp}$  represent the node that is labeled  $p$  in the  $m$  throw of a contour. Note that  $n_{mp}$  is not the ID (or coordinates) of the node; in each row, two nodes with different IDs would have the same label. However, for any node except the middle one in row  $m$ , only one of the two nodes with label  $p$  would be a neighbor. Thus, for all nodes except the middle one in row  $m$ , the node label as per the convention shown in Fig. 7 uniquely identifies a neighbor in row  $m \pm 1$ . Under this labeling in the  $\_8$  embedding, node  $n_{mp}$  can communicate with nodes  $n_{m \pm 1, p}$  and  $n_{m \pm 1, p \pm 1}$ . For the middle node in row  $m$ , there would be two neighbors in row  $m \pm 1$  with the same label.

## 5.2 Optimal Spreading

Our algorithm for optimal spreading is given below. In the algorithm, the array  $ngbrs_{1/2}$  keeps track of the relevant neighbors for any node. As explained above, the middle node in a row has two relevant neighbors in the next row that have the same label.

## 6. RESULTS

To evaluate the QoS achieved by CGD, we designed and implemented a discrete-event simulation using the OMNet++ framework [34]. The results in this section demonstrate the performance of CGD. For all simulations, the nodes were embedded on the 2D Basegrid using the  $_8$  embedding with  $xy_{1/4} 4$ , one source, and one sink in the contour. In the scenarios where the number of messages was not varied, the source sent 5,000 messages, each of which is 36 bytes long. To study the effects of an increased number of messages, the number was changed between 5 and 50,000.

### 6.1 Simulation Approach

The multipath, multihop, dissemination of messages from the source to the sink was modeled as a multistage queuing network. The service time of each queue was exponentially distributed with a mean time of 30 ms to represent 1) the time required to propagate a message from the preceding node, 2) the time required to receive a message in the node, and 3) the time required to forward the message toward the sink. Using the channel modeling capability in OMNet++, the data rate (bandwidth) was set to 38.4 Kbps. Propagation delay and bit error rates were assumed to be zero, and all links between pairs of nodes were identical. The source generated messages with an inter arrival time that was exponentially distributed with a mean time of 20 ms.

To ensure that the simulations reflected real-world constraints on low-power wireless propagation and topology, we weakened the regular topologies, which were used in the analysis in the preceding sections, by modeling link failures and wireless propagation irregularities. To model link failures, the probability of success for each link was varied between zero (failure) and one (success). Before sending a message to a downstream node, a random number was drawn from a uniform distribution; the message was sent only if this random number was larger than the link failure threshold. To account for wireless propagation irregularities [35], a probability of success was assigned to each link in the neighborhood of a node that was drawn from a Weibull

```

let ngbrs[] be the neighbors of  $n_p \in \mathcal{N}$ .
let  $n_p$  be in row  $m$  that has width  $w$ .
let msgCount be a local variable.

initially msgCount := 0;

ngbrs[] = null;
if  $p \neq \lfloor \frac{w}{2} \rfloor + 1$  then {
    ngbrs[0] =  $n_p^{m+1}$ ;
    if  $n_p \in \text{expansion region}$  then {
        ngbrs[1] =  $n_{p+1}^{m+1}$ ;
    } else if  $n_p \in \text{contraction region}$  then {
        ngbrs[1] =  $n_{p-1}^{m+1}$ ;
    }
} else {
    ngbrs[0] =  $n_{\lfloor \frac{w+2}{2} \rfloor + 1}^{m+1}$ ;
    ngbrs[1] =  $n_{\lfloor \frac{w+2}{2} \rfloor}^{m+1}$ ;
    ngbrs[2] =  $n_{\lfloor \frac{w+2}{2} \rfloor}^{m+1}$ ;
}

foreach node  $n_p$  in the expansion region {
    receive  $m$ ;
    msgCount++;

    if  $p \neq \lfloor \frac{w}{2} \rfloor + 1$  then {
        // all nodes except the middle node
        send messages to ngbrs[0] and ngbrs[1]
        in the ratio  $w+2-2p : 2p$ ;
    } else {
        // the middle node
        send messages to ngbrs[0], ngbrs[1], and ngbrs[2]
        in the ratio  $\frac{1}{w+2} : \frac{1}{w(w+2)} : \frac{1}{w(w+2)}$ ;
    }
}

foreach node  $n_p$  in the propagation region {
    // forward messages to one neighbor on next row
    receive  $m$ ;
    send  $m$  to ngbrs[0];
}

foreach node  $n_p$  in the contraction region {
    receive  $m$ ;
    msgCount++;

    if  $p \neq \lfloor \frac{w}{2} \rfloor + 1$  then {
        // all nodes except the middle node
        send messages to ngbrs[0] and ngbrs[1]
        in the ratio  $w-2p : 2(p-1)$ ;
    } else {
        // the middle node sends all messages to the
        // middle node in the next row
        send  $m$  to ngbrs[1];
    }
}

```

distribution using the parameters reported in [35]. We varied the percentage of such irregular nodes from 0 (no nodes with irregular propagation) to 100 (all nodes have irregular propagation). To investigate the relationship between

wireless propagation irregularity and the structure of the contour, we also considered the case where irregular nodes were localized to the expansion and propagation regions of the contour.

**6.2 QoS Metrics**

The number of messages handled by each node, average end-to-end latency, jitter, and message loss rate were the primary QoS metrics. For each message, the time at which it was sent from the source,  $t_s$ , and the time at which this message arrived

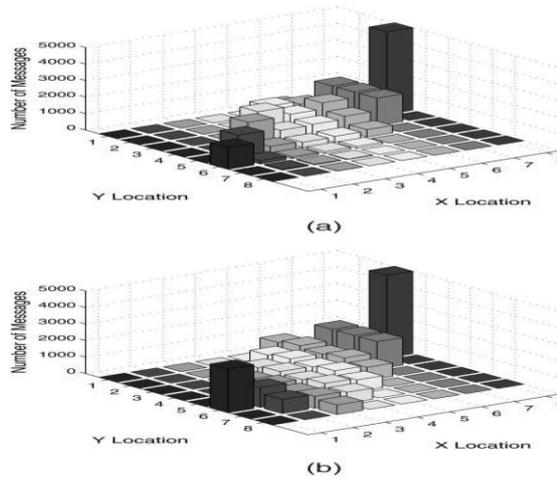


Fig. 8. In optimal CGD, all the nodes in a row of the contour handle roughly the same number of messages. This balance is not maintained in uniform CGD. (a) Uniform CGD. (b) Optimal CGD.

at the sink,  $t_0$ , were recorded. The end-to-end latency for this message is  $t_0 - t_s$ . The average end-to-end latency was the average value of  $t_0 - t_s$  over all the messages received at the sink. Jitter was the standard deviation of the end-to-end latencies. The ratio of the number of messages that were not received at the sink to the number of messages that were sent from the source was the message loss rate. To evaluate the effects of congestion on the performance, we considered the number of messages lost when 1) the service time in each node increases and 2) the number of messages sent by the source increases. In addition, throughput—defined as the number of messages arriving at the sink per unit time—was used as a metric to evaluate the scalability of CGD. Since none of the multipath routing methods described in Section 1 relies on effectively using all the available paths to improve QoS, the metrics discussed above were used to compare the performance of CGD with the performance of 1) flooding and 2) single-shortest-path routing (Unipath), which is typically used in distance vector routing. Our focus on regular topologies allowed us to hand-compute the shortest paths. We used flooding with duplicate elimination and only considered the communication costs incurred by nodes in the contour. We expected the average latency and jitter to be much better than flooding. Since CGD exploits multiple paths, Unipath served as a watermark for acceptable performance. The performance of CGD was investigated both under uniform spreading (uniform CGD) and under optimal spreading (optimal CGD).

**6.3 Performance of CGD**

Fig. 8 shows a histogram of the number of messages handled in each node in uniform CGD and optimal CGD. There were 56 nodes in the contour with  $\Delta x = \Delta y = 4$ . As shown

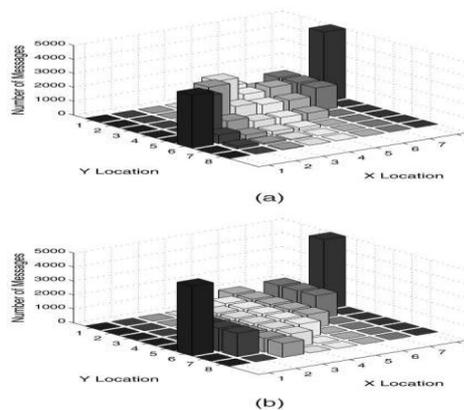


Fig. 9. Optimal CGD maintains a balance among the nodes in a row of a contour even in the presence of link failures. This result is obtained when the probability of link failure was 0.2 for every link in the contour. (a) Uniform CGD. (b) Optimal CGD.

## 7. Conclusions

Many future engineered systems that are based on peer-to-peer-connected mesh topologies are likely to have multiple paths between a pair of nodes. We defined a contour as the union of all shortest paths between a pair of nodes. Using a regular topology, we proved that when the messages are spread uniformly over the paths in a contour, nodes along one path handle more messages than other messages. We presented an optimal strategy for spreading messages in such systems, and our results demonstrate the effectiveness of the spreading strategy. Although the results are based on regular topologies, they represent upper bounds on what can be achieved in general topologies. These results reveal that to achieve optimal dissemination, some nodes must disseminate the messages over the available paths, and other nodes use only one of the available paths. Identifying these sets of nodes in general topologies is an interesting problem. In the future, the optimal dissemination techniques can be enhanced to improve QoS, mitigate interference, reduce hotspot effects, and design next-generation monitoring and surveillance systems based on wireless mesh topologies.

## References:

- [1] R.N. Murthy, A. Gosain, M. Tierney, A. Brody, A. Fahad, J. Bers, and M. Welsh, "CitySense: A Vision for an Urbanscale Wireless Networking Testbed," Proc. ACM Sixth Workshop Hot Topics in Networks (HotNets '07), Nov. 2007.
- [2] N. Hayslip, S. Sastry, and J. Gerhardt, "Networked Embedded Automation," Assembly Automation, vol. 26, no. 3, pp. 235-241, 2006.
- [3] J.Z. Hecker, "PORT SECURITY—Nation Faces Formidable Challenges in Making New Initiatives Successful," technical report, US General Accounting Office, 2002.
- [4] J. Hill, M. Horton, R. Kling, and L. Krishnamurthy, "The Platforms Enabling Wireless Sensor Networks," Comm. ACM, vol. 47, no. 6, pp. 41-46, June 2004.
- [5] I.F. Akyldiz, X. Wang, and W. Wang, "Wireless Mesh Networks: A Survey," Computer Networks, vol. 47, pp. 445-487, 2005.
- [6] G. Coulouris, J. Dollimore, and T. Kindberg, Distributed Systems Concepts and Design, fourth ed. Addison-Wesley, 2005.
- [7] J. Gao and L. Zhang, "Load-Balanced Short-Path Routing in Wireless Networks," IEEE Trans. Parallel and Distributed Systems, vol. 17, no. 4, pp. 377-388, Apr. 2006.
- [8] C. Intanagonwiwat, R. Govindan, D. Estrin, J. Heidemann, and F. Silva, "Directed Diffusion for Wireless Sensor Networking," IEEE/ACM Trans. Networking, vol. 11, no. 1, pp. 2-16, 2003.
- [9] J. Kulik, W. Heinzelman, and H. Balakrishnan, "Negotiation-Based Protocols for Disseminating Information in Wireless Sensor Networks," Wireless Networks, vol. 8, pp. 169-185, 2002.
- [10] J. Agre, L. Clare, and S. Sastry, "A Taxonomy for Distributed Real-Time Control Systems," Advances in Computers, vol. 49, pp. 303-352, 1999.