Available online at www.sciencedirect.com

ScienceDirect

Computer Communications 29 (2006) 2550–2555

computer communications

www.elsevier.com/locate/comcom

Localized routing with guaranteed delivery and a realistic physical layer in wireless sensor networks

Milos Stojmenovic *, Amiya Nayak

SITE, University of Ottawa, Ottawa, Canada

Available online 6 March 2006

Abstract

Routing is the problem of sending a packet from a source node to a destination node in the network. Existing solutions for sensor networks assume a unit disk graph model, where messages between nodes are received correctly if and only if the distance between them is up to the transmission radius, which is equal for all nodes. Sensor networks require localized solutions, where nodes make decisions based on their own positions, the positions of their neighbouring nodes and the position of the destination. Existing localized routing algorithms include greedy routing and GFG (with guaranteed delivery) for the unit disk graph model and expected progress routing for a physical layer model. In a physical layer model, the packet reception probability depends on the distance between nodes. We propose to combine the expected progress routing with face routing to define a localized routing algorithm that guarantees delivery under realistic physical layer models. The proposed localized routing protocol is called EFE (Expected progress–Face–Expected progress). We have implemented the proposed algorithm and compared it with the shortest weighted path scheme. They both assume an ideal medium access layer, where packets between two neighbouring nodes are delivered in the number of attempts equal to the expected hop count between them (which is the expected number of transmissions and acknowledgments). The experiments show that localized EFE is efficient compared to the globalized shortest path algorithm, especially for dense networks.

Keywords: Wireless sensor networks; Routing; Guaranteed delivery; Localized algorithm

1. Introduction

Sensor networks are an active research area due to their potential applications in a wide variety of scenarios. They are forecasted to contribute greatly in situations such as terrain monitoring in potential earthquake zones and volcanoes, forest fire control, rescue operations and military operations such as battlefield monitoring. The actual sensors used are typically very small, i.e., 1 cm^3 in volume, and can therefore be positioned virtually everywhere. The main problem with these sensors is their limited battery life. It can be very expensive to replace sensors in distant or inaccessible and dangerous places. It is therefore very important that sensors make full use of their battery life. Much of the battery is used in communication between sensors and the base station. Efficient and possibly optimal routing algorithms are necessary to make the most of each sensor's battery.

Under current network layer routing assumptions, each node has a fixed transmission radius r. This assumption is referred to as the unit disk model. A sensor receives a message if and only if it is within the transmission radius of the sending sensor. This standard assumption has recently been challenged with more realistic transmission assumptions. In the network model, we will be analysing and testing, the reception of a message by a sensor is a probabilistic event that is directly related to the distance of the sensor to the message's sender.

Figs. 1 and 2 illustrate realistic physical layer assumptions. Fig. 1 is a packet reception probability graph as a function of distance. Fig. 2 conveys the same idea presented in Fig. 1, except it is illustrated on a sensor graph model.

* Corresponding author.
E-mail addresses: milos22@gmail.com (M. Stojmenovic), anayak@site.uottawa.ca (A. Nayak).

0140-3664/$ - see front matter © 2006 Elsevier B.V. All rights reserved.
doi:10.1016/j.comcom.2006.02.013

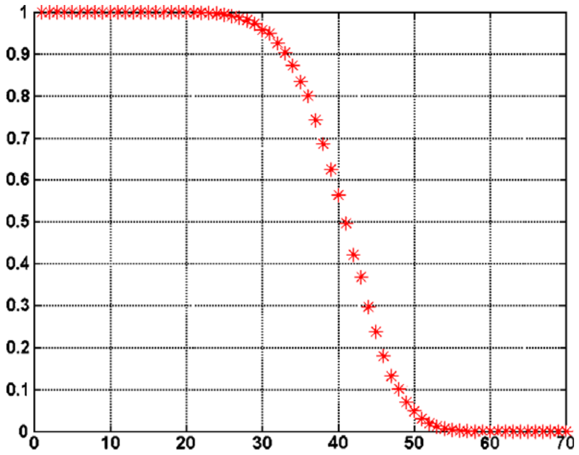


Fig. 1. Packet reception probabilities.

The darker areas in Fig. 2 represent higher reception probabilities. The lighter areas in Fig. 2 represent lower reception probabilities. The described physical layer characteristics cover a number of specific transmission and channel models (such as lognormal shadowing or Raleigh fading) in wireless (not only sensor) networks.

We assume that all sensors use the same transmission power when sending messages. We also assume that there is no other traffic in the network except the considered routing task. Therefore, the packet reception probability purely depends on the distance between transmitting and receiving nodes.

In Section 2, we describe localized routing protocols for the unit disk model of sensor networks. In Section 3, we elaborate on physical layer assumptions used in this article. A localized routing scheme with guaranteed delivery (assuming an ideal medium access layer) is described in Section 4. Section 5 presents the results of our experimental evaluation and comparison with the corresponding globalized shortest weighted path protocol. Conclusions and references are given at the end of this article.

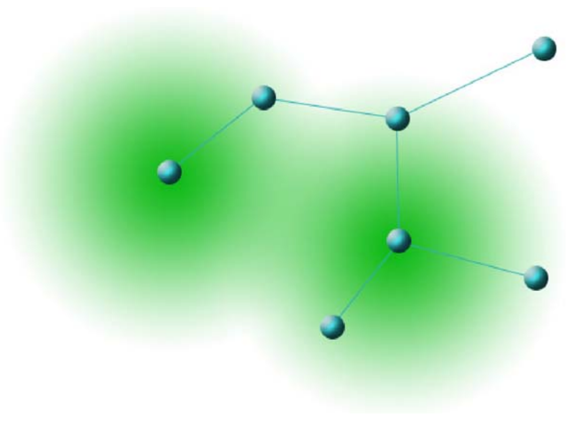


Fig. 2. Reception distribution illustration.

2. Current routing approaches in the unit disk model

One of the simplest and most common routing algorithms used today is the greedy method. *Greedy* routing [2] between source and destination recursively passes the message to the neighbouring node that is physically closest to the destination. This method only needs local knowledge of the network. This means that each node only knows its position, the position of its neighbours and the position of the destination. In the case of sensor networks, the destination is usually the sink or base station that collects the information from the network. This approach is beneficial in the sense that it does not require extra overhead in the form of global information of all the positions of all the nodes in the network. Such information might be useful in determining the optimal path between source and destination, but it is in no way optimal considering the battery power necessary to maintain accurate global positions of all the nodes in the network. The *shortest path* routing takes into account all of the nodes in the network and finds the optimal path using Dijkstra's algorithm. It is usually used as a basis of comparison of all other routing algorithms since it finds the optimal route. The goal of most new routing approaches is to use local information in determining the route between source and destination. Fig. 3 illustrates both the greedy method and the shortest path method on a network.

The shortest path method uses four hops to get the message from source to destination. The greedy method uses five hops in this case. There is however a substantial risk in using the greedy algorithm. Like all other greedy approaches, it has a tendency of reaching a local maximum in many cases, which means that messages are not guaranteed to be delivered, although the source could otherwise be connected to the destination.

A new routing method was proposed by Bose et al. [1] that combines the greedy routing method with face routing. It is called greedy-face-greedy (GFG). This new method guarantees delivery but uses only local information. In order to be able to use GFG, the network must be planar, which means that there can be no intersecting edges in the graph. Gabriel graphs are used to achieve this effect. Fig. 4 depicts a network in the form of a Gabriel graph (GG). Nodes *A* and *B* have an edge between them in a Gabriel

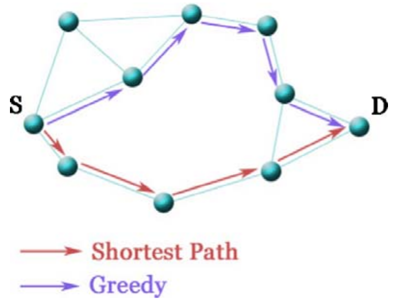


Fig. 3. Greedy and shortest path routing.

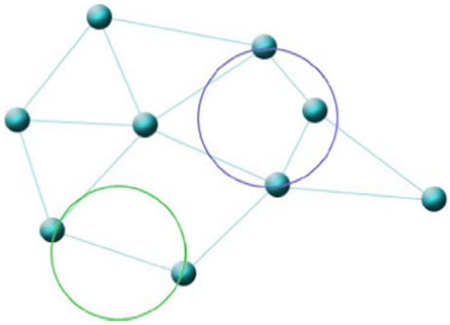


Fig. 4. Gabriel graph.

graph if and only if the disk with diameter AB between nodes A and B contains no other nodes. Fig. 4 shows two such cases, of an edge belonging and not belonging to the GG.

Once a planar graph is obtained, the *face routing* algorithm [1] can be used. We will use the Gabriel graph in Fig. 5 to illustrate the *face routing* procedure. Face routing is a procedure which uses local information yet achieves guaranteed delivery if a path exists from the source to the destination. It usually requires significantly more hops to accomplish face routing compared to the shortest path. The name 'face routing' implies what the algorithm does. Since the graph that we are working with is planar, it can be observed as a set of faces. Messages travel along these faces, as can be seen in Fig. 5. Face routing first draws an imaginary line from the source S to the destination D and is depicted as the dotted line in Fig. 5. This line is used to change faces as a message is passed from node to node. Each face is traversed in the clockwise or counter-clockwise direction until the line segment crosses the imaginary line. The message then travels along the next face until the destination is reached, or the line is crossed again. The example network chosen in Fig. 5 is possibly a poor choice for applying the face routing scheme since it is easily traversable by the greedy routing algorithm.

The GFG routing algorithm [1] is a combination of greedy routing and face routing. As the name suggests, the algorithm starts off in greedy mode and switches to face mode only if greedy mode gets stuck, and cannot proceed further. Once a node that is closer to the destination is reached in face mode, greedy mode once again takes over. Consider the example in Fig. 6. The routing starts off in greedy mode from node S. We see that the node currently holding the message in Fig. 6, node X, cannot forward it to any node that is closer to the destination than itself. Greedy mode has failed, and we switch to face mode. An imaginary line is first drawn from the node that currently holds the message to the destination. In Fig. 6, it is the dotted line XD. This part of the algorithm is called face mode and it is used to find a node that is closer to the destination than the node at which face routing started. Once such a node, Y, is found, greedy routing resumes. In our example, greedy routing from Y will successfully deliver the message to destination D.

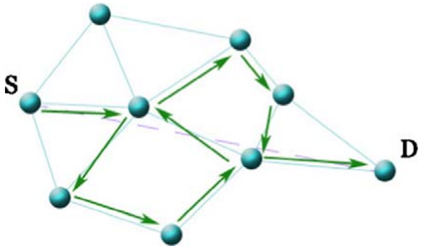


Fig. 5. Face routing.

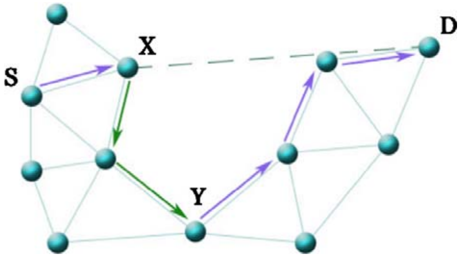


Fig. 6. Three phases in GFG: greedy SX, face XY, greedy YD.

3. Greedy routing with realistic physical layer assumptions

The Greedy [2] and GFG [1] routing algorithms described to this point assume the unit disk model of communication. That is, if a link exists between two nodes, a message can be passed from one to the other with a 100% delivery rate. It does not take into account the distance between the nodes, only the fact that there exists a link between them. In reality, the delivery rate very much depends on the distance between nodes. In this report, we will refer to $p(x)$ as the probability that a message between two nodes is received, where x is the distance between the two nodes. This probability, $p(x)$, decreases with distance x, as shown in Fig. 1. The exact function for $p(x)$ is quite complex, and depends on the exact model used (combined Friis and two-ray ground model in [5]; lognormal shadowing model [6,7]). In this study, we follow the lognormal shadowing model, and $p(x)$ is approximated by a formula that achieves a 4% error, following [3]. First we determine the distance R between two neighbouring nodes for which $p(R) = 0.5$. Then,

$$p(x) = \left(1 - \frac{(x/R)^{2\beta}}{2}\right) \quad \text{for } x < R \quad \text{and}$$

$$p(x) = \frac{\left(2 - \frac{x}{R}\right)^{2\beta}}{2} \quad \text{for } x \geqslant R.$$

In this formula, β is the power attenuation factor, $2 \leqslant \beta \leqslant 6$. This formula is based on packets of length 120 bits. We make several more assumptions as follows [3]. The message is divided into fixed size packets, and each packet is transmitted independently. The acknowledgement message is of the same size as the packet. Each bit is transmitted independently, with the same bit error rate, and the packet is received correctly only if all bits are correctly received. We assume a fixed SNR (signal to noise ratio).

Consider the impact of a decision about what the transmission radius is and strictly follow the greedy position based routing protocol by Finn [2]. Instead of using the traditional hop count measure, which counts each link on a path once, we will now consider a more realistic path optimality measure. The expected hop count (EHC) needs to be used instead of the hop count, which measures all the transmissions on each hop. It may take more than one transmission of a message between sender and receiver for the message to arrive at the receiver. Since this is the case, we also need the receiver to send an acknowledgment to the sender once the message is successfully received. We consider the expected hop count to be the sum of all retransmissions of all sent messages and all acknowledgements.

We consider an 'ideal' protocol at the medium access layer. The sender will keep retransmitting the packet, with a sufficient pause between any two retransmissions (to allow the receiver time to respond) until the packet is acknowledged successfully. Upon successful reception of any message, the receiver will respond with u acknowledgements. The optimal u value depends on x, such that $u * p(x) \approx 1$ [3]. The specific value used in [3] is $u = round(1/p(x) - 0.1)$.

The expected hop count between two nodes is a function $f(x,u)$, where x is the already defined distance between the nodes, and u is the number of times the receiver acknowledges each received message. It is in fact a function of only one variable, x, because u is determined by x. For the above described protocol, it is determined that $f(x,u) = 1/(p(x)(1-(1-p(x))u)) + u/(1-(1-p(x))u)$ [3].

EHC can be used as a weight on an edge between two nodes. The shortest weighted path algorithm then finds the path between the source and destination which has minimal EHC. It is then recognized as the optimal algorithm that can be used as a benchmark for other protocols. Several localized routing protocols based on a realistic physical layer are described in [3]. We have only used the one with the best reported performance. The selected protocol is called the *expected progress routing* protocol [3]. It is based on an expected progress measure. This expected progress measure of an edge takes into account the expected hop count in traversing that edge and the progress obtained by doing so. In Fig. 7, we see a configuration that adequately illustrates this point.

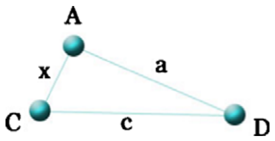


Fig. 7. Expected progress.

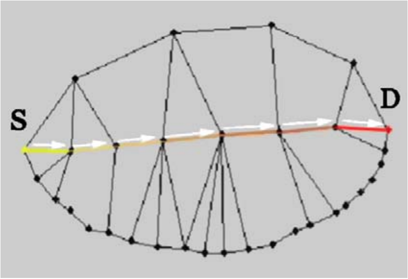


Fig. 8. Expected progress routing algorithm.

In Fig. 7, node C is the sender and node D is the destination. Labels x, c and a are the distances between the pairs of nodes. Node C contemplates where to send the message. The expected progress of sending it to node A is $(c-a)/f(x,u)$. If destination D is a neighbour of C, the expected progress of sending the message directly to node D is then $c/f(c,u)$. The message will be sent to the neighbour whose expected progress is greatest [3].

Fig. 8 illustrates the expected progress routing scheme [3] on an example. The S is the source, and the D is the destination. Only the edges of the corresponding Gabriel graph are drawn. Notice that the marked path was the output of the routing algorithm. The nodes above the marked line are spaced relatively sparsely between each other. They are placed almost at the limit of their transmission radius. The nodes below the marked line are spaced relatively densely. The path that was given as output had nodes that were relatively moderately spaced; each neighbour is roughly at half of the transmission radius of each of its neighbours in the path. The sparsely spaced nodes that form a path to the destination were not chosen by the expected progress routing algorithm since the cost of passing messages between them far outweighs the progress they achieve. According to our experiments, it would cost anywhere between 20 and 80 transmissions to hop from one node to another in such a sparsely spaced graph. The densely spaced nodes that form a path to the destination were not chosen by the greedy algorithm since the progress they achieve per hop is not large enough. The path that was chosen illustrates a sort of middle ground for the expected progress routing approach. The shortest path algorithm produced the same result when tested on the graph in Fig. 8. This confirms that it is best to make moderate, but not small, gains when faced with realistic assumptions.

4. Routing with guaranteed delivery and a realistic physical layer

The contribution in this article is to describe an efficient localized routing scheme that will guarantee delivery when a realistic physical layer is considered. We assume that the medium access layer is ideal, in the sense that the message between two neighbouring nodes is successfully received and acknowledged with the number of packets being equal to the EHC between them (more precisely, equal to $f(x,u)$).

Our proposed algorithm is an adaptation of the GFG protocol [1] to the realistic physical layer. It combines expected progress routing (instead of greedy routing) with face routing. This routing protocol, with guaranteed delivery, is refereed to here as EFE (Expected progress–Face–Expected progress). The expected progress routing (EPR), is used from the source (or another node currently holding the packet) as long as there exist neighbours closer to the destination. If the node currently holding the packet is at a local maxima (that is, has no neighbour closer to destination than itself) then face routing is applied to recover. Face routing is not changed from the unit graph model. It is still based on the Gabriel graph (suitably defined), and follows the same edges (with EHC measure being applied to them). When a neighbour closer to the destination than the node that entered the face routing is found, expected progress routing may resume.

5. Experimental results

In this article, we adapted java software developed by Aaron Zollinger [4] to suit our needs. Zollinger developed a java environment for testing various routing protocols. Weighting schemes had to be updated in his models in order to test our models of various routing protocols. We applied the new expected progress weighting scheme to all of the edges of each network, effectively replacing the original weighting of 1 for each edge. The shortest weighted path algorithm did not need other changes. Face routing procedure also did not need any other modification. The greedy routing scheme [2], however, needed to change the criterion for selecting the next neighbour. Instead of merely looking for a neighbour closest to the destination, it checked the ratios of progress and EHC for each neighbour and selected the best neighbour. As a result, a different neighbour is selected in most cases.

Network density d is an input parameter of the program. It is used to determine the expected number of neighbours for each node. The following approximate equation was used to find the number of nodes n that need to be dispersed around the area where the network is located.

$$d \approx (n-1)\frac{r^2\pi}{a^2}.$$

The length/width measure of the square area in which the network is located is a. In our experiments, $a = 10$. This formula is an approximation, and small error occurs at nodes close to the borders of the area.

We need to clearly define when two nodes are considered neighbours. Following [3], we consider two nodes as neighbours if the packet reception probability between them is $\geqslant w$ for a relatively small constant w (we used $w = 0.05$, as in [3]). The considered graph then is a kind of unit graph, with a properly interpreted transmission radius. Thus, the 'transmission radius' of each node is r denotes the distance for which $p(r) = w$. That is, two nodes are considered neighbours if there exists a meaningful chance of communication between them. The transmission radius r is used to determine R value in the formula for $p(x)$ as follows: $R = r/1.4377$. This corresponds to $p(R) = 0.5$.

Experiments were conducted as follows. For each generated graph, a random pair (source and destination) is selected. Dijkstra's shortest path algorithm is then applied to test whether they are connected. If not, this triple (graph, source and destination) is ignored. Otherwise, the described routing schemes are applied and averages are measured. This process is repeated with various densities. For each density, twenty graphs were generated, and the results shown represent the average values of hops needed to traverse the graphs using these algorithms. Our main goal was to compare the shortest weighted path algorithm (denoted by SP) with EFE, to show how efficient localized routing is with respect to optimal globalized routing. The expected progress routing (EPR) algorithm was also tested, and two measures for it were taken: fail rate, and average expected hop count when it was found successful. The percentage of failure of the greedy routing algorithm is seen in the last column. Table 1 for results, where value $\beta = 2$ is used.

It is a general trend in Table 1 that as the node density decreases, the expected hop count increases for all of the routing algorithms. Exceptions are seen for the EPR algorithm since it fails often in sparsely packed networks, and these failed numbers were not taken into account when the average was computed; however, they were considered when computing the averages for shortest path and GFG. This is why they seem lower. For dense networks (starting with $d = 20$ in Table 1), EFE performs very close to SP (14% overhead at $d = 10$, down to 3.5% at $d = 80$), confirming the competitiveness of localized approaches. Taking the indicated densities one at a time, starting at density 10, the overhead

Table 1
EHC values for routing protocols for $\beta = 2$

Density	SP	EPR	EFE	EPR fail rate
80	25.5	26.4	26.4	0
40	29.4	32	32	0
20	36.1	41.3	41.3	0
10	51.2	70	81.6	0.11
8	61.2	79.4	153	0.47
6	66.7	52.6	251.4	0.50
5	94	63.6	128.8	0.44

Table 2
EHC values for routing protocols for $\beta = 4$

Density	SP	Greedy	GFG	Greedy fail rate
80	21.1	21.7	21.7	0
40	25	26.6	26.6	0
20	22.8	26.7	26.7	0
10	38.8	68	101.4	0.15
8	67.3	221.6	337.2	0.33
6	265.3	151.8	945.9	0.53
5	379	74.8	1573.5	0.84

of EFE with respect to SP is 60%, 150%, 277% and 37%, respectively. Thus, EFE performs almost the same as SP on very densely packed networks, and its performance tremendously drops off in sparse networks. On the other hand, the EPR protocol works well on densely packed networks, but its delivery rate becomes unacceptable for sparsely packed networks. Note that EFE became 'suddenly' again competitive to EHC at very low densities. This is because in sparse networks, below a critical density level, all connected paths tend to consist of only one hop; therefore, EFE then performs as SP. This means also that the overhead of EFE, considering over all possible densities, with respect to SP is limited. Table 2 presents the corresponding data for $\beta = 4$.

Acknowledgements

We thank the Ontario graduate scholarship fund and NSERC for partly funding this work.

References

[1] P. Bose, P. Morin, I. Stojmenović, J. Urrutia, Routing with guaranteed delivery in ad hoc wireless networks, in: ACM DIAL M, Aug. 1999, pp. 48–55, Wireless Networks 7 (6) (2001) 609–616.

[2] G.G. Finn, Routing and addressing problems in large metropolitan-scale internetworks, ISI Research Report ISU/RR-87-180, 1987.

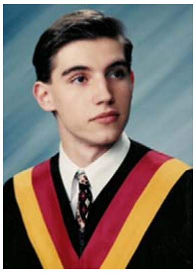
[3] J. Kuruvila, A. Nayak, I. Stojmenovic, Hop count optimal position based packet routing algorithms in ad hoc wireless networks with a realistic physical layer, in: The 1st IEEE International Conference on Mobile Ad-hoc and Sensor Systems MASS, Fort Lauderdale, October 2004, pp. 398–405, IEEE Journal of Selected Areas in Communications 23 (6) (2005) 1267–1275.

[4] F. Kuhn, R. Wattenhofer, Y. Zhang, A. Zollinger, Geometric ad-hoc routing: of theory and practice, in: Proceedings of ACM PODC, 2003.

[5] T. Nadeem, A. Agrawala, IEEE 802.11 fragmentation-aware energy-efficient ad-hoc routing protocols, in: Proceedings of the 1st IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS), Fort Lauderdale, October 2004.

[6] L. Qin, T. Kunz, On-demand Routing in MANETs: the impact of a realistic physical layer model, in: Proceedings of the 2nd International Conference on Ad-hoc Networks and Wireless (ADHOC-NOW), 2003, pp. 37–48.

[7] I. Stojmenovic, A. Nayak, J. Kuruvila, F. Ovalle-Martinez, E. Villanueva-Pena, Physical layer impact on the design and performance of routing and broadcasting protocols in ad hoc and sensor networks, Computer Communications 28 (10) (2005) 1138–1151.



Milos Stojmenovic is currently working on his Ph.D. in Computer Science at the University of Ottawa. He completed his Master's thesis titled "Real time car detection with an AdaBoost machine learning approach and a small training set", at Carleton University, Ottawa in 2005. His research interests include Computer Vision, and wireless and ad hoc sensor networks.

Amiya Nayak is an associate professor at School of Information Technology & Engineering, University of Ottawa since 2002. He received his B.Math. degree from the University of Waterloo in 1981, and Ph.D. in computer engineering from Carleton University in 1991. He has over 15 years of industrial experience in software engineering in avionics and telecommunication applications, working at CMC Electronics and Nortel Networks. His current research interests are in the area of fault tolerance and testing of VLSI systems, distributed systems and mobile ad hoc networks with over 70 publications in refereed journals and conference proceedings. He is in the Editorial Boards of International Journal of Parallel, Emergent and Distributed Systems and International Journal of Computing and Information Science.