Logical Coordinate Assignment for Geographic Routing in Wireless Sensor Networks

Jang-Ping Sheu
Department of Computer Science and Information Engineering National Central University, Chung-Li, 32054 Taiwan, R.O.C.; E-mail: sheujp@csie.ncu.edu.tw
Yu-Chia Chang
Department of Computer Science and Information Engineering National Central University, Chung-Li, 32054 Taiwan, R.O.C.; E-mail: jimchang@axp1.csie.ncu.edu.tw
Gang-Hua Song
Department of Computer Science and Information Engineering National Central University, Chung-Li, 32054 Taiwan, R.O.C.; E-mail: silva@axp1.csie.ncu.edu.tw

Received: October 29, 2005; revised: December 07, 2005

***Abstract*— In this paper, we propose a distributed protocol to build a logical coordinate system based on the hop counts of each node to four selected landmarks, and the real location information is not needed. Our designed protocol uses the sink node as one of the landmarks and then selects three other sensor nodes near the corners of the sensor network as landmarks. The simulation results show that our proposed protocol has the superior performance in packet delivery ratio, average hop counts among nodes, and communication overhead to previous works.**

***Index Terms*— Geographic routing, logical coordinates system, wireless sensor networks.**

I. Introduction

In recent years, advances in microprocessors, low power wireless radio devices, sensing units, and devices smaller in size are making low cost and multifunctional sensor nodes more available. These tiny sensor nodes also make the deployment of densely distributed sensor networks for a wide area possible. Many applications are designed for very large scale sensor networks and extend the ability to monitor the real world, such as environment monitoring applications, military operations, biological observations and other useful applications. Since most applications will produce lots of sensing data, the main functions of sensor nodes are aggregating the huge amount of the sensing data to the sink nodes (base stations) in a multi-hop manner. Geographic routing [1], [2], [3] is a location-based routing protocol and it uses greedy forwarding as its basic rule to route the packets to the destination. Nodes look up its neighbor table and deliver the packets to the closest neighbor to the destination. Many geographic routing protocols [1], [2], [3] have been proven to provide extreme performance improvement over existing routing protocols. With location information, routing protocols can accomplish good performance and conserve battery energy of sensor nodes. Therefore, geographic routing is very suitable for large scale sensor networks. One important reason geographic routing can work efficiently and correctly is that sensor nodes have accurate real location information. Individual node locations play a key role in forwarding decision. Geographic routing protocols have been shown to be correct and efficient with exact location information. With location errors, geographic routing may find the wrong node to pass the packets, and it has been shown that errors in node location information lead to routing failures [4], [5] . In [4], the authors show that even small location errors (10% of the radio range or less) can in fact lead to incorrect geographic routing results and noticeable degradation of performance. Harsh environment and imperfect hardware can cause location inaccuracy even without node mobility. Real location information usually can be acquired by using Global Positioning System (GPS) or location estimation algorithms [6], [7], [8], [9], [10], [11], [12]. Attaching a GPS receiver to each node may not be a good solution because of the following reasons. First, the GPS technology cannot get 100% accurate real location information, the position error might still be 10 20 meters [9], and this position error may be larger than the distance of two nodes. Second, the GPS signal would be affected by weather, topography, the satellite coverage, and the obstructions in the path of satellite signals. These factors cause location inaccuracy and GPS cannot work indoors. Third, it is expensive to equip GPS chips or devices for each small sensor node in a large scale sensor network. Finally, sensor nodes are powered by batteries, and the GPS devices may consume power heavily causing the sensor nodes to die quickly. Another way to acquire real locations for wireless sensor nodes is to use location estimation protocols. These algorithms mostly use measurements of signal strength [8], Angle of Arrival (AOA) [10], Time of Arrival (TOA) [7], Time Difference of Arrival (TDOA) [7], [11], and Distance Vector (DV) based algorithm [9]. Most of them assume that GPS receivers are available at some nodes, or the positions of some nodes are known a prior, these nodes are so called reference (beacon) nodes, and let all other nodes derive their positions from these reference nodes. Some location estimation algorithms [8] use received signal strength indicator (RSSI) technique assuming that radio model is ideal; this assumption is strong and is not true in the real environment. Therefore, to assign each node an accurate location is a challenge issue in wireless sensor networks. Due to the reasons mentioned above, the logical coordinate system based on hop counts is proposed. Nodes only maintain hop counts to a small number of landmarks (or anchors) and do not need the real location or other information. Previous works [13], [14], [15] have shown that the logical coordinate system can support standard geographic routing efficiently in large scale sensor networks. In this paper, we propose a distributed protocol to construct a logical coordinate system based on four landmarks (anchors). We assume the sensor nodes are deployed in a rectangle-like area and the sink node is placed in one of the corners of the rectangle. Our designed protocol uses the sink node as one of the landmarks, and then selects three other sensor nodes near the corners of the sensor network as landmarks. The logical coordinate vector consists of hop counts to four landmarks, and a node can make greedy routing decision by using its logical coordinate vector. Simulation results show that the logical coordinate system constructed by our protocol can support geographic routing efficiently and has comparable performance to previous works [13], [14]. Our protocol also has less flooding overhead compared with VCap, which does not use the trading time technique [13]. The rest of this paper is organized as follows. Section 2 gives the preliminaries and related works. In section 3, we will describe our logical coordinate assignment protocol. Section 4 shows the performance results. Finally, conclusions are drawn in section 5.

II. Preliminary and related works

In this section, we first introduce the logical coordinate system and geographic routing. Then, we review some logical coordinate systems which are based on hop counts to landmarks.

A. Logical coordinate system

The main concept of logical coordinate system is that each node maintains hop counts to a small number of landmarks (or anchors). These hop counts form a vector, called logical coordinate vector. For example, in Fig. 1, four landmarks, *W*, *X*, *Y*, *Z* and 16 sensor nodes are deployed in the network. Every landmark generates a control packet containing its own ID and a hop counter. By flooding this packet to the entire network, each node acquires a hop count to all landmarks with the logical coordinate vector (w, x, y, z). The logical coordinate vector of node *A* is (3, 4, 4, 3). It means that node *A* is three hops away from landmark *W*, four hops away from landmark *X*, four hops away from landmark *Y*, and three hops away from landmark *Z*.

The logical coordinate system has features described as follows. First, logical coordinate has no absolute relations with real locations of sensor nodes. It can reflect the true connectivity between nodes in the sensor network graph, rather than real distance. For example, physical obstacles can easily prevent two geographically close nodes from communicating directly, causing them to be far apart in the logical coordinate system. Second, the dimensionality of the logical coordinate vector is determined by the number of landmarks. In [14], the simulation result shows that selecting an appropriate number and positions of landmarks can reach a good packet delivery ratio and make it possible to eliminate the existence of voids or obstacles in the logical coordinate system despite their existence in the physical place. Increasing the number of landmarks can improve robustness with respect to the geographic routing protocol, but it would also increase the flooding overhead. Third, at a high enough network density, coordinates propagate as circular coronas centered on the initiator landmark. In Fig. 2, nodes with first hop centered on landmark *X* resemble a circle with radius equal to the communication range. Nodes with second hop also resemble a circular corona centered on landmark *X*, and the radius is equal to the first hop plus communication range. Fourth, the number of landmarks must be larger than 2.

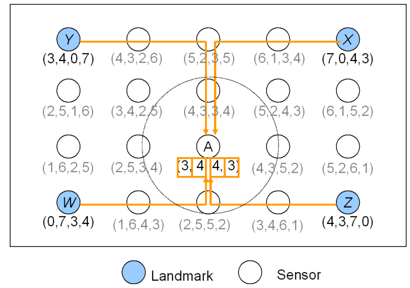

Fig. 1. An example of logical coordinate system.

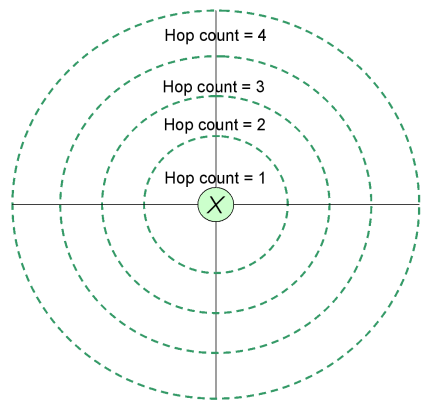

Fig. 2. Hop counts propagate as circular coronas centered in the landmark X.

The authors in [13] show that if two landmarks are deployed in the network, there would exist the situation that zones symmetric to the directrix connecting to two landmarks have the same logical coordinate vector. As shown in Fig. 3, nodes in zones *A* and *B* share the same logical coordinate. If the destination is in zone *A*, the forwarding algorithm may route

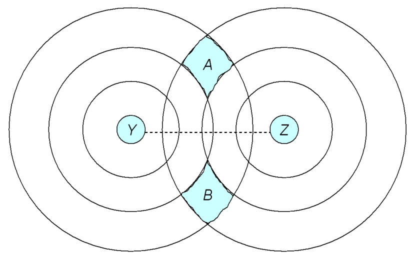

Fig. 3. Nodes in zones A and B share the same logical coordinate.

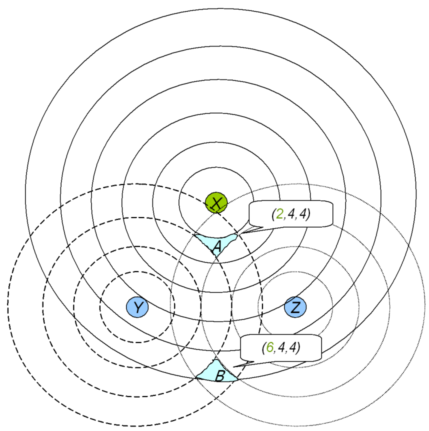

Fig. 4. Zones A and B can be identified by using a third landmark X.

the packet to zone *B*. In Fig. 4, by adding another landmark *X*, zones *A* and *B* apart can be identified. Finally, the logical coordinate system does not assure that each node has a unique logical coordinate vector. A node may share the same logical coordinate with its neighbors in the same zone. In Fig. 4, nodes in zone *A* share the same logical coordinate (2, 4, 4).

While applying geographic routing over the logical coordinate system, one node chooses the neighbor whose logical distance from destination is the smallest, as the relay node. In the logical coordinate system, the logical distance *D* between two logical coordinate vectors *A* and *B* is defined as follows:

$$D = \sqrt{\sum_{i=1}^{n} (A_i - B_i)^2}$$

where A_i and B_i are elements in vectors *A* and *B*, respectively. Each node needs to maintain its neighbors' logical coordinate for choosing the best relay node greedily. In addition to greedy forwarding rule, a simple backtracking technique also can be used to improve routing performance [14].

B. Related works

Many algorithms are proposed to construct a coordinate system in wireless ad hoc and sensor networks. Those protocols can be classified into two categories: One is to find the logical (or relative) coordinate [12], [13], [14], [15], [16], [17], [18]. The goal of finding a logical coordinate system is to find an embedding of the nodes into multi-dimensional space that result in the same neighbor relationships as the underlying network. The second is to find absolute coordinate [6], [7], [8], [9], [10], [11], [12]. The goal of an absolute coordinate system is to determine the real location of all the nodes. In this paper, we are interested in the logical coordinate system. In the following, we would like to introduce some previous works on the logical coordinate system. The authors in [12] and [17] proposed connectivity-based [18] approaches to construct a logical coordinate system. Particularly, in [12], logical coordinates can be transformed into absolute ones if sufficient landmarks are available. The main shortcoming of these protocols is that they are based on centralized approaches which are not quite feasible for a large wireless sensor network scenario. In addition to connectivity based approaches, hop counts based approaches are also proposed by several researchers. In [**?**], the goal of the work is to propose a distributed protocol of coordinate assignment which aims at assigning the sensor nodes with logical coordinates. The key idea of [15] is to use a relaxation algorithm that associates a logical coordinate to each node. The virtual coordinate are then used to perform geographic routing. The approach in [15] can construct a coordinate system to support standard geographic routing efficiently, but the communication overhead and memory cost are quite heavy. The authors in [14] propose a scalable logical coordinate framework in wireless sensor network. The main concept of the logical coordinate framework is to maintain hop counts to a small number of landmarks. These hop counts form a vector which is the logical coordinate vector of the node. Nodes work on the relative logical coordinate system and run a geographic routing protocol while transmitting packets. A backtracking mechanism is also proposed to improve the packet delivery ratio, in the paper. The authors also investigate the effect of the number of landmarks and their positions on routing performance. The simulation results show that, in a square network, four landmarks put in the corners (4-corner case) of the network can reach almost 100% packet delivery ratio, which is the same results achieved with the 6-corner case. If the landmarks are randomly placed, even when the number of landmarks is more than 4, the routing performance is worse than the 4-corner case. It means that the proper positions and number of landmarks can reach the balance of performance and flooding overhead. The disadvantage of the work in [14] is that the positions of landmarks are controllable. This is not applicable in a real sensor network. Unlike [14], the authors in [13] proposed a distributed protocol to identify three landmarks. At the end of the protocol, each node is assigned with a triplet of logical coordinate vectors. In this protocol, a sink node (or any node designed for this protocol) will initiate the protocol, and the protocol uses four phases to identify the three landmarks. In the first phase, the sink node floods a message containing a hop counter to the whole network. In the second phase, each node compares the hop count to sink node with neighbors within two hops to determine whether it is the candidates of the first landmark. Those candidates will generate a control message containing their ID and a hop counter and flood the packets to the entire network. In the following two phases, nodes use the existing hop counts acquired from previous phases to elect other landmarks in sequence. In this protocol, many global flooding will occur in the last three phases because many candidates are selected locally. To reduce the number of broadcasts overhead, the authors use a trading time technique [13] for communications and only four global flooding are needed. This technique can reduce considerable flooding overhead, but it needs the entire network to be synchronized. As is known, synchronization protocol is not an easy problem to solve in the real sensor networks [19]. Furthermore, using such a technique in each phase would require a time close to *n x 2t* (*t* is the time needed to propagate a message throughout the entire network, *n* is the number of nodes in the network) to complete in the worst case. Our work proposes a distributed algorithm to find landmarks which are near the corners of a sensor network and to assign each node a logical coordinate vector in the network. We use a simple protocol to reduce the number of global flooding rather than the trading time technique used in [13]. In addition, two-hop neighbors' information is needed in [13]; in our protocol, nodes only need to collect one-hop neighbor's information. In a large scale wireless sensor network, the memory and communication costs of constructing two-hop neighbor tables are much higher than constructing one-hop ones. Thus, our protocol has less memory and communication overhead than the protocol proposed in [13].

III. Logical Coordinate Assignment (LCA) protocol

In this section, we present a logical coordinate assignment protocol (LCA) to select four landmarks which are located near the corners of the network. According to the simulation results in [14], four landmarks put in the corners of the network can reach almost 100% packet delivery ratio. Using more than four landmarks only brings limited improvement and will incur more communication overhead. Therefore, our protocol is designed to find four landmarks which are located as near the corners of the network as possible. Each sensor node in the network will then be assigned a four-dimensional logical coordinate vector without any real location information. A logical coordinate system based on hop counts to the landmarks can be established and can support geographic routing efficiently without the GPS device. Our protocol consists of four phases: *W*-Phase, *X*-Phase, *Y*-Phase, and *Z*-Phase. In the first phase, *W*-Phase, we treat the sink node as landmark *W*, and deploy it at one of the four corners in the sensor network. This is reasonable because once the network is deployed; the placement of the sink node can be decided by our will. Initially, landmark *W* will generate a *W_msg* packet and broadcast this message to its neighbors; the *W_msg* packet includes its ID, a hop counter (initial set to 0), and *W_threshold*. The *W_threshold* is used to select landmark *X* in the next phase. When a node receives the *W_msg* packet, it increases the hop count by one and rebroadcasts the packet to its neighbors. Each node will keep the information of the smallest hop count packet while receiving multiple *W_msg* packets. At the end of *W*-Phase, each node will be assigned a hop count to landmark *W*, called *w* coordinate. In the second phase, *X*-Phase, a node will be selected as landmark *X* which is the farthest node to landmark *W* in the network. Assume

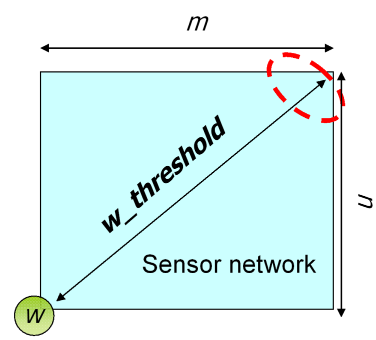

Fig. 5. The definition of W_threshold.

a sensor network is bounded in a rectangle and the edge lengths of the rectangle are *m* and *n* for long side and short side, respectively. The parameter, *W_threshold*, represents the minimum hop counts from landmark *W* to its diagonal of the network as shown in Fig. 5. That is,

$$\mathit{W\_threshold} = \sqrt{m^2 + n^2}/T_x\_Range$$

where T_x_Ranger is the transmission range of the radio.

In *X*-Phase, each node will first broadcast its *w* coordinate to one-hop neighbors if its *w* coordinate is larger than or equal to *W_threshold*. Then every node can determine whether it is a candidate of landmark *X*. A node will become a candidate if its *w* coordinate is maximum within one-hop neighbors. Note that, if two nodes have the same *w* coordinate value, we select the node with smaller ID as a candidate. Since the candidate is decided locally by each node, there may exist more than one candidate of landmark *X*. Therefore, each candidate node will flood a control packet (*W_local_msg*) to the network and find one of the candidates as the landmark *X*. The control packet includes candidate node's ID, *w* coordinate, and a TTL (time to live). Since the *w* coordinate of any candidate is larger than or equal to *W_threshold*, the control packets only need to forward to the nodes whose *w* coordinate *W_theshold*. For example, in Fig. 6, it is assumed that the *W_threshold* is 14 and nodes 55 and 65 are the candidates of landmark *X*. Both nodes 55 and 56 will flood *W_local_msg* packet to the network. Any node with *w* coordinate smaller than *W_threshold* will drop the received control packet. Each node will also drop the received control packet if the packet's *w* value is smaller than one of the previously received one. Such local flooding can reduce a large number of control packets overhead in *X*-Phase. After a predetermined time period T_x, the node with maximum *w* value will find that it is the landmark *X*, where T_x is equal to $TTL \times t$ (*t* is the time needed to broadcast a packet from a node to its neighbors). In our simulations, TTL = (*W_threshold* / 2) is enough to obtain good results. The selected landmark *X* will flood an *X_msg* control packet including its ID, *w* value, and a hop counter (initial set to zero) to the whole network. Each node will obtain its *x* coordinate from the control packet. For example, in Fig. 6, node 55 will consider that it is the landmark *X* after finishing the local flooding.

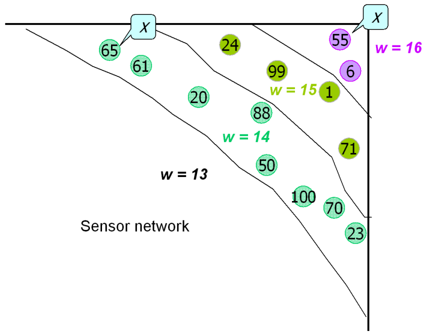

Fig. 6. A local flooding example for W_threshold = 14.

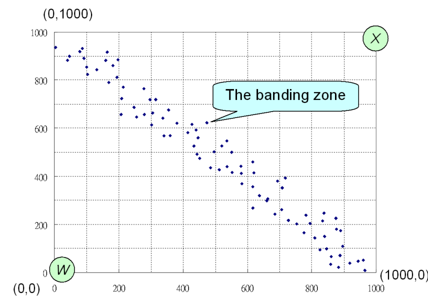

Fig. 8. An example for the distribution of nodes which satisfy w = x or w = x ± 1.

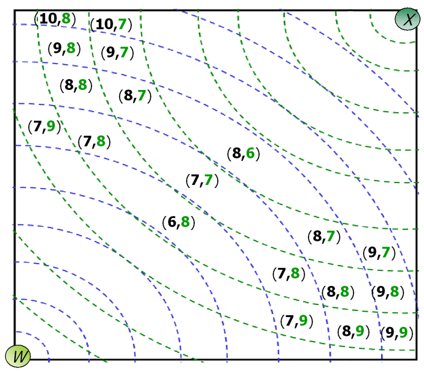

Fig. 7. The result after W-Phase and X-Phase is finished.

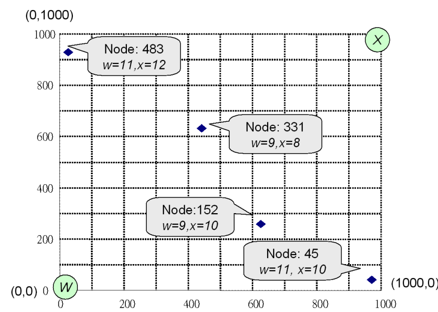

Fig. 9. Nodes 483, 331, 152, and 45 are Y candidates.

After executing the X-phase, each node can obtain its w and x coordinates as shown in Fig. 7. We can find that the value of w + x of nodes near the center of the network are smaller than those near the corners of the network. In the third phase, Y-Phase, we would like to choose the landmark Y located in the upper-left or lower-right corner. Therefore, the possible candidates of the next landmark are located in a banding zone of the network from upper-left to lower-right corners. This banding zone can be defined as a set of nodes which their coordinates w and x satisfy the following equation: $w = x$ or $w = x \pm 1$. For example, we randomly deploy 500 nodes in a 1000 m x 1000 m network; a banding zone is shown in Fig. 8. Each node belonging to the banding zone will broadcast its coordinate to one-hop neighbors. A node will become a candidate of landmark Y if its x + w value is maximum among its one-hop neighbors.

Like in X-Phase, there are more than one Y candidates in the banding zone. For example, in Fig. 9, nodes 483, 331, 152, and 45 are candidates of the landmark Y. To select one of them as the landmark, each candidate floods a *Y_local_msg* control packet containing its ID, w, and x coordinates to the network. Nodes located in the banding zone will rebroadcast the control packets. When each node receives a *Y_local_msg* packet, the node will discard the control packet if the packet's w + x value is smaller than the previously received one. If two nodes have the same w + x value, the node ID is used to break the tie. After a predetermined time T_y, the node with the maximum w + x will claim that it is the landmark Y. Note that, $T_y \geq y_{max} \times t$, where y_{max} is the w value of landmark X in the network. Then the landmark node will flood a control packet *Y_msg* containing its ID and a hop counter (initial set to 0) to notify all the nodes. And each node can get a y coordinate from the control packet. In Fig. 9, node 483 will become the landmark Y since its sum of w and x is the maximum in the banding zone.

In the last phase, Z-Phase, the landmark Z is the farthest node to the landmark Y. Thus, the candidates of the landmark Z are located in the same banding zone with landmark Y. When a node receives the *Y_msg* packet, it has a hop count to the Y landmark. Each node in the banding zone broadcasts its y coordinate to one-hop neighbors. The node that has the maximum y value among its one-hop neighbors becomes the candidate of the landmark. Note that, if two nodes have the same y coordinate value, the node ID is used to break the tie. Similarly, each candidate node will flood a control packet *Z_local_msg* containing its ID, y coordinate, and a TTL. In our simulations, TTL = 2 is enough. Each candidate node waits $2t$ time periods to determine whether it is the landmark Z. Then the landmark Z floods a control packet *Z_msg* including its ID and a hop counter (initial set to 0) to notify all the sensor nodes and each sensor node can get a z coordinate. Finally, the landmarks W, X, Y, and Z and each node can acquire a logical coordinate vector which consists of w, x, y, and z. Although some nodes might not receive the control packets sent from some landmarks, they can fill their logical coordinate vector by exchanging the coordinate data with its one-hop neighbors. Then, the logical coordinate system is constructed completely and geographic routing can be applied in the system. We will show the performance of our proposed protocol by simulations in the next section.

IV. Simulations

In this section we evaluate the performance of our proposed protocol (LCA) through simulations. We implemented our protocol in Glomosim [20], a discrete event simulator developed by UCLA. We consider that sensor nodes have no mobility and are distributed uniformly in a 1000 m x 500 m rectangle. The number of nodes is from 150 to 900. Each node has the same transmission range of 100 m. The propagation delay is 1 μs. The simulation results are computed as the average of 100 runs. We have three experiments in our simulations. First, we measure the packet delivery ratio of greedy forwarding [7] in the logical coordinate system built by our protocol, LCA and VCap (without trading time technique), proposed in [13] and the 4-corner case in [14]. We also evaluate the routing performance while sensor nodes use accurate real coordinate system. We do not use any mechanism like backtracking in [14] to improve the routing performance because we want to show the superiority of the logical coordinate system. Second, we evaluate the average path length with the logical coordinate system constructed by LCA and previous works and evaluate the real coordinate system. Finally, we compare the flooding overhead (in number of packets) and completion time of LCA and VCap with and without the trading time technique for the purpose of studying the communication overhead as two protocols constructing the logical coordinate system.

A. packet delivery ratio

In our simulations, we apply simple geographic routing based on logical and real coordinate systems. We randomly choose 100 pairs of sources and destinations. We carefully control the time which each source node generates a routing packet in order to avoid too many routing pairs at the same time. This action decreases the probability of collisions and helps us to evaluate the packet delivery ratio more precisely. The simulation results in Fig. 10 show that, while the network density (average number of neighbors of each node) increases, the reachability of the four logical and real coordinate systems also increases and almost reaches 100%. In general, the routing performance with real coordinate performs better than the four logical coordinate systems. In the situation of low density, our logical coordinate system has more than 90% reachability, very close to the performance of the 4-corner case in [14]; VCap has poorer performance than others while network density is low. We found that the logical coordinate system have better performance than the real coordinate system in very low density. This is because logical coordinate can reflect the real connectivity of nodes.

B. Average packet length

In this section, we evaluate the average path length with the four logical and real coordinate systems. In Fig. 11, the results show that the real coordinate system has the shortest average

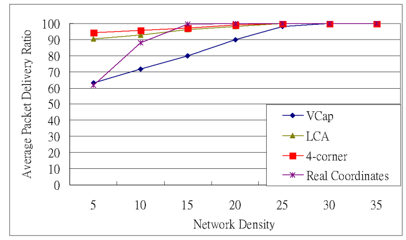

Fig. 10. Average packet delivery ratio with network size 1000 m x 500 m.

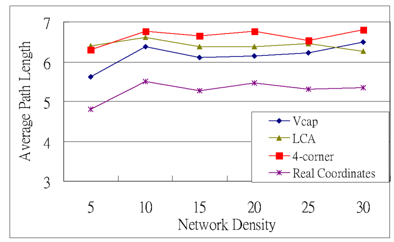

Fig. 11. Average path length with network size 1000 m x 500 m.

path length. This is because sensor nodes can get more precise information while routing to the destination for real coordinate. From the simulation results, we can know that the average path lengths of four logical coordinate systems are 10% longer than the real coordinate system. Our coordinate system has longer path length than that of VCap in low network density. With the network density increases, the average path lengths of LCA and 4-corner case are shorter than that of VCap.

C. Flooding overhead

In this section, we evaluate the flooding overhead of LCA and VCap with and without trading time technique which is measured by the number of control packets sent during the simulation period. In Fig. 12, our protocol needs more flooding packets than VCap with trading time technique. The result shows that LCA needs more control packets than the VCap with trading time. This is because our protocol has additional local flooding in the last three phases. In our observation, the cost of local (and banding zone) flooding is acceptable. As expected, we can see that VCap without trading time technique needs much more flooding overhead than LCA while the number of nodes increases. This is because many global flooding occurred in the last three phase of VCap protocol without trading time technique.

D. Execution time

Finally, we evaluate the execution time of LCA and VCap spending to construct the logical coordinate system. In particular, we evaluate the time of VCap with and without trading time technique. In Fig. 13, the execution time of our protocol is slightly higher than VCap without trading time technique. In Fig. 14, we show the execution time of LCA and VCap

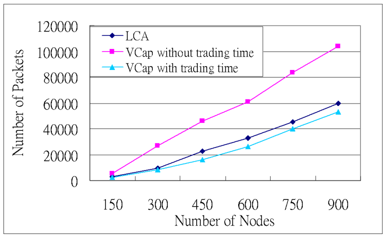

Fig. 12. The number of control packets with network size 1000 m x 500 m.

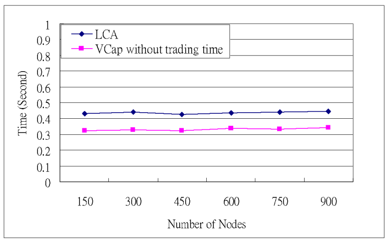

Fig. 13. The execution time with network size of 1000 m x 500 m.

with trading time technique. We can observe that the average execution time of VCap with trading time technique increases as the number of nodes increases. The VCap with trading time technique takes much more time than our protocol LCA. In each phase of VCap with trading time technique, it may take a time close to $n \times 2t$ in the worst case as we mentioned in section 3. On the other hand, the execution time of LCA is only affected by the network size.

V. Conclusion

In this paper, we presented a distributed protocol to build a logical coordinate system based on hop counts to landmarks. Nodes make routing decision with their logical coordinates, not real ones. The main idea of our protocol is to use a distributed protocol to identify four landmarks where each one is as near the corner of the network as possible and assign each node a logical coordinate vector. Simulation results show that the packet delivery ratio with our logical coordinate system is almost 100% and very close to the 4-corner case in [14] and the real coordinate system. The real coordinate system has shorter average path length than the logical coordinate system. Our LCA protocol has better routing performance than that of VCap in low network density. Moreover, the communication overhead of our protocol LCA is lower than VCap without trading time technique. Although the communication overhead of our protocol is a little higher than the VCap with trading time technique, the execution time of VCap with trading time technique is much higher than our protocol.

Acknowledgment

This work was supported by National Science Council, Republic of China under Grants NSC 94-2213-E-008-024 and NSC 94-2752-E-007-003-PAE.

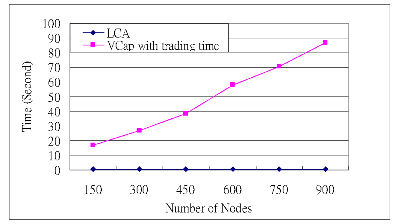

Fig. 14. The execution time with network size of 1000 m x 500 m.

References

[1] B. Karp and H. -T. Kung, ”GPSR: Greedy Perimeter Stateless Routing for Wireless Networks,” *in Proceedings of the 6th Annual ACM/IEEE International Conference on Mobile Computing and Networking*, pp. 243-254, Boston, Mass, USA, August 2000.

[2] Y. -B. Ko and N. H. Vaidya, ”Location-Aided Routing in Mobile Ad Hoc Networks,” *in Proceedings of the 4th ACM/IEEE International Conference on Mobile Computing and Networking*, pp. 66-75, Dallas, Texas USA, October, 1998.

[3] W. -H. Liao, J. -P. Sheu, and Y. -C. Tseng, ”GRID: A Fully Location-Aware Routing Protocol for Mobile Ad Hoc Networks”, Telecommunication Systems, Volume 18, pp. 37-60, 2001.

[4] K. Seada, A. Helmy, and R. Govindan, ”On the Effect of Location Errors on Geographic Face Routing in Sensor Networks,” in Proceedings of the 3rd International Symposium on Information Processing in Sensor Networks, pp. 71-80. Berkely, CA, USA, April, 2004.

[5] D. Son, A. Helmy, and B. Krishnamachari, ”The Effect of Mobility-Induced Location Errors on Geographic Routing in Mobile Ad Hoc and Sensor Networks: Analysis and Improvement Using Mobility Prediction, ” in Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC), Atlanta, Georgia, USA, March, 2004

[6] T. He, C. Huang, B. Blum, J. A. Stankovic, and T. Abdelzaher, ”Range-Free Location Schemes in Large Sensor Networks,” *in Proceedings of the 9th Annual ACM/IEEE International Conference on Mobile Computing and Networking*, pp. 81-95, San Diego, CA, USA, September, 2003.

[7] A. Savvides, C. -C. Han and M. B. Strivastave, ”Dynamic Fine-Grained Localization in Ad-Hoc Networks of Sensors,” *in Proceedings of the 7th Annual ACM/IEEE International Conference on Mobile Computing and Networking* (Mobicom), pp. 166-179, Rome, Italy, July, 2001.

[8] P. Bahl and V. N. Padmanabhan ”RADAR: An in-building RF-based User Location and Tracking System,” *in Proceedings of 19th Annual Joint Conference of the IEEE Computer and Communications Societies*, Volume 2, pp. 775-784, Tel-Aviv, Israel, March, 2000.

[9] D. Niculescu and B. Nath, ”Ad Hoc Positioning System (APS),” *in Proceedings of IEEE Global Communications* (GLOBECOM), Volume 5, pp. 2926-2931, San Antonio, Texas, November, 2001.

[10] A. Nasipuri and K. Li, ”A Directionality Based Location Discovery Scheme for Wireless Sensor Networks,” *in First ACM International Workshop on Wireless Sensor Networks and Applications* (WSNA), pp. 105-111, Atlanta, GA, September, 2002.

[11] N. B. Priyantha, A. Chakraborty, and H. Padmanabhan, ”The Cricket Location Support System,” *in Proceeding of sixth ACM International Conference on Mobile Computing and Networking* (MOBICOM), pp. 32-43, Boston, MA, Augest, 2000

[12] Y. Shang, W. Rum, Y. Zhang, and M. Fromherz. ”Localization from Mere Connectivity,” *in Proceedings of International Symposium On Mobile Ad Hoc Networking and Computing* (MobiHoc), pp. 201-212, Annapolis, MD, USA, June, 2003.

[13] A. Caruso, S. Chessa, S. De, and A. Urpi, ”GPS Free Coordinate Assignment and Routing in Wireless Sensor Networks,” *in Proceedings of IEEE International Conference on Computer Communication* (INFOCOM), Miami, March, 2005.

[14] Q. Cao, and T. Abdelzaher, ”Scalable Logical Coordinates Framework for Routing in Wireless Sensor Networks,” *in Proceedings of 25th IEEE International on Real-Time Systems Symposium* (RTSS), pp. 349 - 358, Lisbon, Portugal, December, 2004.

[15] A. Rao, C. Papadimitriou, S. Shenker, and I. Stoica, ”Geographic Routing Without Location Information,” *in Proceedings of the 9th Annual*

International Conference onMobile Computing and Networking, pp. 96-108, San Diego, CA, USA, 2003.
[16] S. Capkun, M. Hamdi, and J. P. Hubaux, ”GPS-free Positioning in Mobile Ad Hoc Networks, ” *in Proceedings of Hawaii International Conference on System Sciences*, pp. 3481-3490, Maui, HW, January, 2001.
[17] T. Moscibroda, R. O’Dell, M. Wattenhofer, and R. Attenhofer, ”Virtual Coordiantes for Ad Hoc and Sensor Networks,” *in Proceedings of the ACM Joint Workshop on Foundations of Mobile Computing* (DLALM-POMC), pp. 8-16, Philadelphia, Pennsylvania, USA, October, 2004.
[18] R. R. Bischoff, and R. Wattenhofer, ”Analyzing Connectivity-based, Muti-hop Ad Hoc Positioning,” *in Proceedings of the 2nd IEEE International Conference on Pervasive Computing and Communications*, pp. 165-176, Orlando, Florida, USA, March, 2004.
[19] F. Sivrikaya and B. Yener, ”Time Synchronization in Sensor Networks-a Survey,” *IEEE Networks*, Vol. 18, No. 4, pp. 45-50, July/August 2004.
[20] Glomosim: Website, http://pcl.cs.ucla.edu/projects/glomosim/

Jang-Ping Sheu received the B.S. degree in computer science from Tamkang University, Taiwan, Republic of China, in 1981, and the M.S. and Ph.D. degrees in computer science from National Tsing Hua University, Taiwan, Republic of China, in 1983 and 1987, respectively. He joined the faculty of the Department of Electrical Engineering, National Central University, Taiwan, Republic of China, as an Associate Professor in 1987. He is currently a Professor of the Department of Computer Science and Information Engineering and Director of Computer Center, National Central University. He was a Chair of Department of Computer Science and Information Engineering, National Central University from 1997 to 1999. He was a visiting professor at the Department of Electrical and Computer Engineering, University of California, Irvine from July 1999 to April 2000. His current research interests include wireless communications, mobile computing and parallel processing. He was an associate editor of Journal of the Chinese Institute of Electrical Engineering, from 1996 to 2000. He was an associate editor of Journal of Information Science and Engineering from 1996 to 2002. He was an associate editor of Journal of the Chinese Institute of Engineers from 1998 to 2004. He is an associate editor of the IEEE Transactions on Parallel and Distributed Systems and International Journal of Ad Hoc and Ubiquitous Computing. He has served as a Program Chair and Vice Program Chair for a number of international conferences including IEEE ICPADS’02, ICPP’03, and IEEE MSN’05. He received the Distinguished Research Awards of the National Science Council of the Republic of China in 1993-1994, 1995-1996, and 1997-1998. He was the Specially Granted Researchers, National Science Council, from 1999 to 2005. He received the Distinguished Engineering Professor Award of the Chinese Institute of Engineers in 2003. He received the Distinguished Professor award of the National Central University in 2005. Dr. Sheu is a senior member of the IEEE, a member of the ACM, and Phi Tau Phi Society.

Yu-Chia Chang received the B.S. degree from National Central University, Taiwan, in 2000. He is presently acquiring both his M.S. and Ph.D. degrees in the department of Computer Science and Information Engineering, National Central University. He once participated in the Program for Promoting Academic Excellence of Universities, carried out by the MOE, in 2000-2001. He has been an associate in the Design and Implementation of Location-Aware Mobile Ad Hoc Networks, promoted by the National Science Council, the Executive Yuan of the ROC, since 2000. His research interests include Wireless Ad Hoc Networks, Wireless Sensor Networks, and Bluetooth technology.

Gang-Hua Song received the B.S. and M.S. degree from National Central University, Taiwan, in 2002 and 2004.