Look-ahead Geographic Routing for Sensor Networks

Jiaxi You[1], Dominik Lieckfeldt[1], Qi Han[2], Jakob Salzmann[1], and Dirk Timmermann[1]
[1]University of Rostock, Germany
[1] {jiaxi.you, dominik.lieckfeldt, jakob.salzmann, dirk.timmermann}@uni-rostock.de
[2]Colorado School of Mines, USA
[2]qhan@mines.edu

***Abstract*— As sensor networks are deployed over various terrains, the complexity of their topology continues to grow. Voids in networks often cause existing geographic routing algorithms to fail. In this paper, we propose a novel geographic routing algorithm called "Greedy Forwarding with VIrtual Position (ViP)". We introduce virtual position as the middle position of all direct neighbors of a node. Instead of comparing nodes' real geographic positions, ViP employs virtual positions when selecting the next hop. Such virtual position reflects the neighborhood of a sensor node, as well as the tendency of further forwarding. For sparsely-deployed networks, ViP increases success rate of packet routing without introducing significant overhead. Furthermore, multiple levels of virtual position can be obtained with localized iteration. We propose the "Greedy Forwarding with Multi-level Virtual Position (MVP)" algorithm and the "Greedy Forwarding with Hierarchical Virtual Position (HVP)" algorithm. Different levels of virtual positions and their combinations are used alternatively to increase success rate of packet routing in sensor networks.**

***Key words*— geographic routing, greedy forwarding, routing hole, wireless sensor networks.**

I. Introduction

Wireless Sensor Networks (WSNs) are composed of many sensor nodes, which are capable of sensing physical parameters, processing data and communicating wirelessly [1]. Sensor nodes are usually battery powered and left unattended after deployment. Since radio communication among sensor nodes is the main drain of energy in WSNs, energy efficiency remains a critical design issue for WSNs.

Among various routing algorithms, on-demand routing algorithms such as AODV [2] are popular, while their flooding-based route discovery often lead to high control overhead [3]. In contrast, single-path geographic routing with Greedy Forwarding (GF) is attractive for WSNs [1]. In a basic GF algorithm, a node communicates only with its direct neighbors (1-hop). The neighboring node that further minimizes the remaining distance of a packet to its destination will be selected as the next hop. Such localized approach is effective and can be dynamically adapted to changes, which only require position information of sensor nodes [1]. The Greedy algorithm [4] and the Most Forwarding progress within Radius (MFR) algorithm [5] are two fundamental approaches of GF based on distance. Fig. 1 illustrates the principle of the Greedy algorithm and the MFR algorithm.

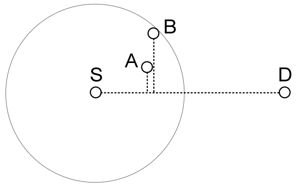


Fig. 1. An example of Greedy and MFR geographical routing. A packet is sent from node S to node D. The Greedy algorithm selects node A as the next-hop, since A is the neighbor that is the closest to the destination. The MFR algorithm selects node B as the next hop, since B makes the greatest progress along the direction of packet forwarding (S-D). Both Greedy and MFR requires the next hop to be closer to the destination than the current node.

As modern WSNs are becoming popular in various applications, their topology is becoming complicated. Due to limited precision of deployment, voids can cause routing holes in the network, where often lead traditional GF algorithms to fail [6]. The reason is the local minimum phenomenon illustrated in Fig. 2. In single-path GF routing algorithm (e.g. Greedy or MFR), forwarding of packets towards the sink can fail at node A, since there is no direct neighbor closer to the destination than node A itself. Besides, uneven power consumption on sensor nodes can also lead to new routing holes in the later lifetime of WSNs.

The objective of our work is to improve the success rate of GF for sparse WSNs or WSNs with small routing holes. To this end, we present a novel geographic routing algorithm

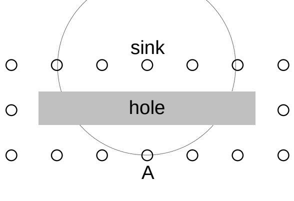


Fig. 2. An example of local minimum.

This work is partially financed by the German Research Foundation (DFG) (post graduate program MuSAMA, GRK 1424).

named “Greedy Forwarding with VIrtual Position (ViP)", as well as its improved versions called the “Greedy Forwarding with Multi-level Virtual Position (MVP)" algorithm and the "Greedy Forwarding with Hierarchical Virtual Position (HVP)" algorithm. Each proposed algorithm has two variants using the Greedy and MFR principle respectively. The main advantage of our approaches is that the algorithms simply employ GF throughout the routing process, and inherently results in high routing efficiency as the basic GF algorithms. In the mean time, the amount of control overhead of the proposed algorithms is strictly limited. The basic assumption of our approaches is that each node in the network is aware of its own geographic position as well as the positions of its direct neighbors.

II. Related Work

Various geographic routing algorithms [6] have been proposed in recent years to address the local minimum problem of GF. Most of the routing algorithms are composed of two phases. Such algorithms start with a basic GF algorithm, and recover from local minimum with their complementary hole-bypassing phases.

The Greedy Perimeter Stateless Routing (GPSR) is one of the fundamental algorithms based on planar graphs [7]. A planar graph is a sub-graph with non-crossing edges, which represents the same connectivity as the original network. GPSR uses GF and switches to its perimeter mode when a local minimum is met. The right-hand rule is employed in the perimeter mode, where packets are routed counterclockwise along the edge on the face of a planar graph. Bose et al. proposed the FACE-1 and FACE-2 [8] algorithms that use the perimeter of the planar graph formed at each node. Such approaches have high success rate of packet routing, but also high control overhead due to the planarization processes and the maintenance of the planar graph information on every sensor node.

In the abovementioned 2-phase algorithms, the performance during the GF phase is much better than during their complementary phases [9]. To increase the ratio of GF during routing, Liu et al. presented the idea of Aligned Virtual Coordinate System (AVCS) [9]. Virtual coordinates are based on integer number of hops to the reference nodes (anchor nodes) and represent a coarse approximation of node location. The success rate of greedy routing is improved with AVCS, while flooding of control message from anchor nodes to the whole network causes large control overhead during the setting up of AVCS.

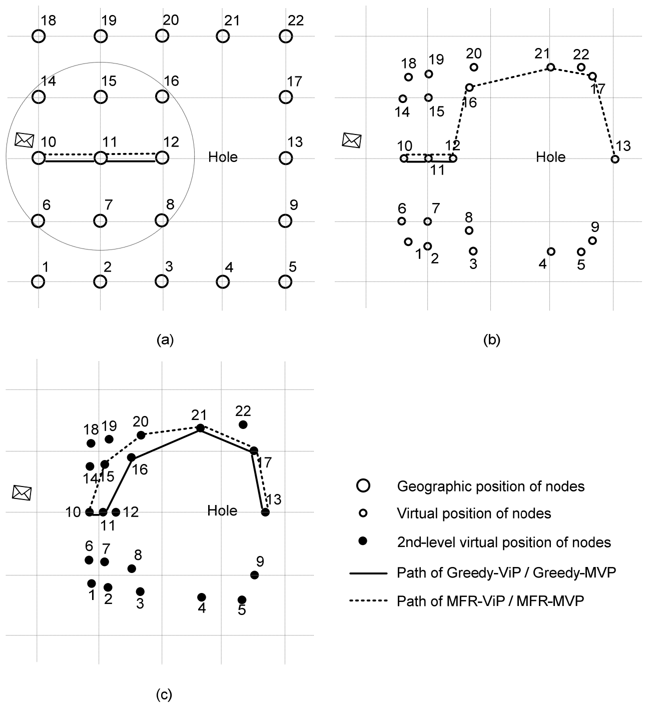
In [10], Stojmenovic et al. proposed to use GF with information of 2-hop neighbors (2-hop GF). The success rate of the GF algorithms is improved with each node aware of its direct neighbors as well as its 2-hop neighbors (neighbors of its neighbors). Among all direct and 2-hop neighbors, packets are sent to the node which is the closest to the destination via 1-hop or 2-hop forwarding. Since farther neighbors are available for the selection of next-hop, small routing holes can be avoided efficiently. The authors also proved that such GF-based methods are inherently loop-free. The method in [10] solely employs the basic idea of GF during routing. Such a method limits control overhead since it only requires local knowledge of sensor nodes (information of 1-hop and 2-hop neighbors) and avoids sophisticated complementary phases to recover from local minimums.

III. Algorithm Design

As mentioned earlier, 2-hop GF uses extra knowledge of 2-hop neighbors in GF. Such information on farther neighbors improves the success rate of GF in a straight-forward manner. However, 2-hop GF is not strictly localized to the direct neighbors, therefore, it is not scalable for further extension. For example, when further improvement of success rate is demanded, nodes need to have information of neighbors within N hops (N>=3). For a 2D wireless network, the number of neighbors stored on each sensor nodes increases proportionally to N^2. Similarly is the control overhead for the maintenance of neighbor information within N hops.

A. Greedy Forwarding with Virtual Position (ViP)

We introduce the *virtual position* of a node as the middle point of all its direct neighbors. If a node $A(x_A,y_A)$ has n direct neighbors $V_{A,1}(x_{A,1},y_{A,1})$, $V_{A,2}(x_{A,2},y_{A,2})$, …, $V_{A,n}(x_{A,n},y_{A,n})$, the virtual position of node A is:



18
19
20
21
22
14
15
16
17
10
11
12
Hole
13
6
7
8
9
1
2
3
4
5
(a)
(b)
(c)
Geographic position of nodes
Virtual position of nodes
2nd-level virtual position of nodes
Path of Greedy-ViP / Greedy-MVP
Path of MFR-ViP / MFR-MVP

Fig. 3. An example of using virtual positions in GF. The packet is routed from node 10 to node 13. When using the geographic positions of nodes, node 12 becomes a local minimum for both Greedy and MFR, as shown in (a). With virtual positions, a path can be found by MFR-ViP, as shown in (b). With 2nd-level virtual position, both Greedy-MVP (2-level) and MFR-MVP (2-level) find paths around the hole, as shown in (c).

$$(x_A', y_A') = ((x_{A,1} + x_{A,2} + \ldots + x_{A,n})/n, (y_{A,1} + y_{A,2} + \ldots + y_{A,n})/n) \quad (1)$$

Each node calculates its virtual position according to (1), and broadcasts its virtual position to its direct neighbors. The information of virtual position is stored on nodes themselves and their direct neighbors. In other words, each node has the knowledge of its own virtual position, and the virtual positions of its direct neighbors. Fig. 3(a) shows the geographic positions of nodes. Fig. 3(b) shows the calculated virtual positions of nodes. The virtual position of node 11 is the same as its real geographic position, since its direct neighbors (node 6, 7, 8, 10, 12, 14, 15, 16) are located uniformly in the transmission circle centered at the geographic position of node 11. In contrast, the virtual position of node 12 (Fig. 3(b)) locates to the left of the geographic position of node 12 (Fig. 3(a)), because the positions of its direct neighbors (node 7, 8, 11, 15, 16) are left-biased in the transmission circle of node 12.

The virtual position of a node provides an indication of how the direct neighbors are located around the node on average, hence it is a suitable metric to demonstrate the tendency of further forwarding during geographic routing. Consider the example in Fig. 3, for a packet that needs to be sent from node 10 to node 13, both Greedy and MFR get stuck at node 12, since there is no neighbor that can make further progress towards the destination. In contrast, the virtual position of node 12 is strongly left-biased due to the void on the right side of node 12. As a result, MFR-ViP finds a path (10-11-12-16-21-17-13) around the hole using virtual positions of nodes.

We propose "Greedy Forwarding with Virtual Position (ViP)", a look-ahead geographic routing algorithm based on the coordinate system of virtual positions. Instead of using farther neighbors as in 2-hop GF, ViP uses the virtual positions of nodes to involve farther neighbors in the look-ahead routing process. Our algorithm is strictly localized, where a node only interacts with its direct neighbors. ViP has two variants called Greedy-ViP and MFR-ViP, which are based on the principle of the Greedy and MFR algorithm, respectively. As an example for a 2D WSN, when a packet with destination $D(x_D,y_D)$ arrives at node $A(x_A,y_A)$, Greedy-ViP evaluates its own virtual position (x_A',y_A') and the virtual positions of its direct neighbors. The neighbor has the virtual position that is the closest to node D is selected as the next hop. According to the Greedy algorithm, the virtual position of the selected neighbor must be also closer to the destination than the virtual position of current node A. Namely, the next-hop node $V_{A,i}$ of Greedy-ViP meets the following two conditions:

$$\arg\min_{V_{A,i}}(\sqrt{(x_{A,i}'-x_D)^2+(y_{A,i}'-y_D)^2}) \quad (2)$$

and:

$$\sqrt{(x_{A,i}'-x_D)^2+(y_{A,i}'-y_D)^2} < (\sqrt{(x_A'-x_D)^2+(y_A'-y_D)^2} \quad (3)$$

Similarly, MFR-ViP selects the neighbor that makes the greatest progress towards the destination of a packet. Given the vectors *DA* and $DV_{A,i}$, the next-hop node $V_{A,i}$ of MFR-ViP is the one that has the minimal dot product of the two vectors:

$$\arg\min_{V_{A,i}}(DA \cdot DV_{A,i}) \quad (4)$$

and also satisfies equation (3). Greedy-ViP and MFR-ViP terminate when there is no neighbor that has the virtual position to make further progress towards the destination of a packet.

With this idea, nodes only need to interact with their direct neighbors to obtain information of virtual position. Therefore, setting up virtual positions is strictly localized. The message overhead in the worst case is O(M), where M is the total number of nodes in a network.

B. Greedy Forwarding with Multi-level Virtual Position (MVP)

To further improve the success rate of GF, we introduce the concept of "multi-level virtual position" that considers farther nodes (neighbors of K-Hop, K>=1). If we refer to the virtual position derived using Equation (1) as the *1st-level virtual position*, then the *Kth-level virtual position* (K>=1) of node A can be done with K iterations of local broadcast between nodes and their direct neighbors $V_{A,1}(x_{A,1},y_{A,1})$, $V_{A,2}(x_{A,2},y_{A,2})$, …, $V_{A,n}(x_{A,n},y_{A,n})$, as follows.

$$(x_A{}^K, y_A{}^K) = ((x_{A,1}{}^{K-1} + x_{A,2}{}^{K-1} + \ldots + x_{A,n}{}^{K-1})/n, (y_{A,1}{}^{K-1} + y_{A,2}{}^{K-1} + \ldots + y_{A,n}{}^{K-1})/n) \quad (5)$$

Such Kth-level virtual positions of nodes indicate how the K-hop neighbors are located on average, also the tendency of further forwarding during geographic routing. Using this concept, we introduce the "Greedy Forwarding with Multi-level Virtual Position (MVP)" algorithm based on Kth-level virtual position (K>=1), where each node knows its own Kth-level virtual position, and the Kth-level virtual positions of its direct neighbors. MVP equals ViP when K is 1. The 2 variants Greedy-MVP and MFR-MVP evaluate the Kth-level virtual positions according to the principle of the Greedy and MFR algorithm, respectively. Comparing to ViP, MVP employs a higher level of virtual positions to further improve the success rate of GF. Fig. 3(c) is an example of MVP using 2nd-level virtual position, which demonstrates that using virtual position of higher level indicates better forwarding tendency in geographic routing, since it takes farther neighbors into consideration. Greedy-MVP and MFR-MVP stop when there is no neighbor that has the Kth-level virtual position that makes further progress towards the destination of a packet.

C. Greedy Forwarding with Hierarchical Virtual Position (HVP)

Using Kth-level virtual positio*n* (K>=1) can introduce new local minimums during routing. For instance, in Fig. 4, a packet needs to be sent from node 5 to node 4. When using the geographic positions of nodes, the packet can be successfully delivered (path 5-3-4 in Fig. 4(a)). While using 1st-level virtual position, node 5 has no neighbor to make further progress towards node 4, thus becomes a local minimum of Greedy-

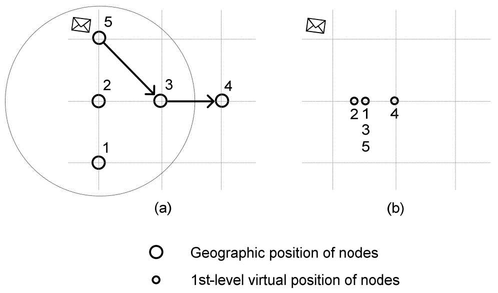

5
2
3
4
1
(a)
2 1
3
5
4
(b)
Geographic position of nodes
1st-level virtual position of nodes

Fig. 4. An example of routing holes introduced by using virtual positions.

ViP and MFR-ViP.

To address this problem, we introduce the "Greedy Forwarding with Hierarchical Virtual Position (HVP)" algorithm (Table. 1 shows the pseudo code of HVP). HVP uses the combination of all K-level virtual positions (K>=1) and the geographic positions of nodes in a down-hill fashion. Namely, when a local minimum is met using Kth-level virtual position, HVP will lower the level of virtual position from Kth-level to (K-1)th-level (K>=2), or from 1st-level virtual position to real geographic position. HVP requires nodes to store the geographic positions, as well as all the K-level virtual positions of itself and its direct neighbors. A flag *flag_level* is added to the packets to indicate the current level of virtual position (using geographic position when K=0). In our example (Fig. 4(b)), the packet stuck (using 1st-level virtual position) at node 5 can set its *flag_level* to be "0", and reach its destination via the path 5-3-4 by using the geographic positions of nodes. The two variants Greedy-HVP and MFR-HVP start with Kth-level virtual position (K>=1) according to the principle of the Greedy and MFR algorithm, respectively.

To ensure that HVP is loop-free, such down-hill process is unidirectional. Namely, for a packet initialized with Kth-level virtual position and adjusted to (K-1)th-level virtual position after meeting a local minimum, the packet stays on (K-1)th-level and can only be further adjusted to the lower levels (or from 1st-level virtual position to geographic position).

TABLE 1
PSEUDO CODE OF GF-HVP ON NODE N

```
PROCEDURE handle_packet

WHILE there is local minimum AND packet.flag_level>=0
{
        packet.flag_level--;
}

IF packet.flag_level>=0
{
        forward packet using virtual position of level packet.flag_level
        indicated by the packet (using geographic position when
        packet.flag_level is 0);
}
ELSE
{
        routing failed;
}
```

D. Algorithm Properties

In a GF algorithm, a node needs to evaluate all the entries of neighbors stored on the nodes, and select the next hop based on a certain criterion. Therefore, the average storage and computational overhead is proportional to the number of neighbor entries on nodes. This metric also reflects the control message overhead to maintain the information of virtual positions. Table 2 shows control overhead of related algorithms. Here we assume that nodes are deployed with the same density in a 2D network.

TABLE 2
ALGORITHM OVERHEAD

Algorithm	Overhead
GF, GF-ViP, GF-MVP	$O(\pi\cdot R^2\cdot D)$
GF-HVP starts with *Kth-level virtual position*	$O(\pi\cdot R^2\cdot D\cdot(K+1))$
N-hop GF	$O(\pi\cdot (N\cdot R)^2\cdot D)$

R: transmission range of nodes.
D: average number of nodes in a unit area

The proposed ViP, MVP and HVP algorithms are strictly localized, and can terminate based on local knowledge, i.e., they are loop-free.

Lemma 3.1. For any static network, ViP is loop-free.

Proof: As proved in [10], single-path routing algorithms based on the Greedy algorithm or the MFR algorithm is inherently loop-free. ViP employs the virtual position coordination system which is the 1-to-1 mapping of the original geographic coordinate system of a network. Instead of the geographic positions, ViP evaluates the corresponding virtual positions of nodes based on the coordinate system of virtual position. Thus, Greedy-ViP and MFR-ViP are also inherently loop-free. Namly, ViP is loop-free.

Lemma 3.2. For any static network, MVP is loop-free.

Proof: Assume that MVP of Kth-level virtual position (K>=1) is loop free. Since MVP of (K+1)th-level virtual position employs the 1-to-1 mapping of the Kth-level virtual position coordinate system. Therefore, MVP of (K+1)th-level virtual position is loop-free. As proved in Lemma3.1, MVP of 1st-level virtual position is loop free. Thus, MVP of Kth-level virtual position is loop free for any K>=1.

Lemma 3.3. For any static network, HVP is loop-free.

Proof: HVP starts with Kth-level virtual position (K>=1) and adjust the *flag_level* of packets unidirectionally until "0", which refers to the coordinate system of geographic position. As proved in Lemma3.2, MVP of Kth-level virtual position is loop free. Therefore, each step in the down-hill scaling of HVP is loop-free. Thus, HVP starts with Kth-level virtual position is loop free for any K>=1.

IV. Performance Evaluation

We implemented the proposed algorithms using Matlab. The simulated network was a 1000 m × 1000 m square plane, where 500 sensor nodes were randomly deployed. Packets were generated with random pairs of source-destination addresses. Different communication ranges of sensors are used. We used a simple disc-communication model: sensor nodes in the transmission range can receive signals from a transmitter without loss. For the same deployment density, smaller communication range implies sparser layout of a network and bigger chance to encounter routing holes during packet forwarding.

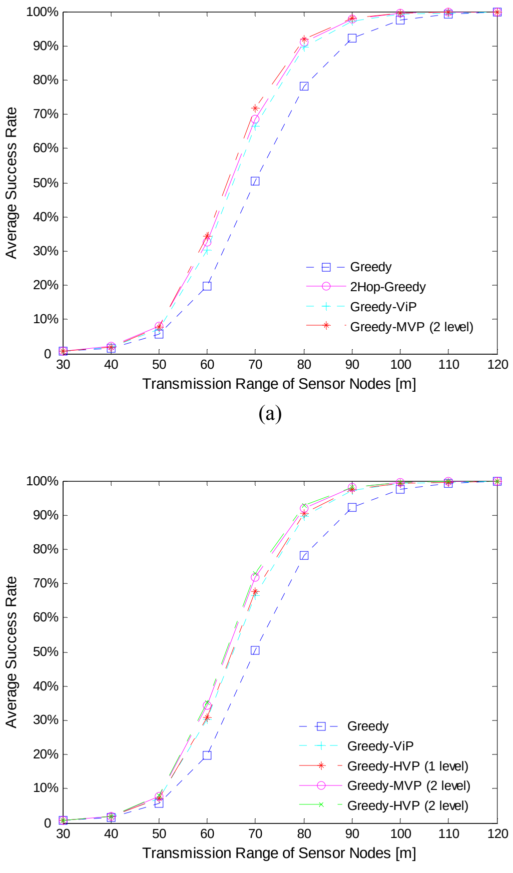
The objective of our experiments is to demonstrate the performance of our algorithm in sparsely-deployed WSNs, or WSNs with small voids due to non-uniform deployments. We show only the results of the variant Greedy-ViP, Greedy-MVP and Greedy-HVP. The performance of MFR-ViP, MFR-MVP and MFR-HVP are similar from our simulation. The results were compared with the 2-Hop Greedy algorithm [10] and its extension, the 3-Hop Greedy algorithm.

We use the following metrics to compare the performance of the simulated routing algorithms:

- *Success rate of GF:* This is defined as the percentage of packet delivery with random packet forwarding. 500 packets with random source-destination pairs were generated and forwarded in the simulated network. When the transmission range of nodes is small, voids appear frequently in the network. This metric illustrates the ability of avoiding routing hole of the simulated algorithms.
- *Delay of GF:* This is defined as the average hop counts of packets from their source to the destination. Given the same success rate, small packet delay (in hops) represents better performance in terms of routing efficiency and energy consumption.
- *Algorithm Overhead*: The number of neighbor entries stored on sensor nodes implies the storage overhead, as well as the computational overhead by choosing of next-hop. Furthermore, the control message overhead of maintaining neighbor information is also proportional to the number of entries.

Experimental Results:

Fig. 5 shows the average percentage of successful packet forwarding from our simulation. Fig. 5(a) compares the performance of the Greedy algorithm, 2Hop-Greedy, Greedy-ViP and Greedy-MVP (using 2nd-level virtual positions). When the communication range of sensor nodes is short, nodes have relatively few neighbors, which leads to low success rates of the simulated algorithms. The 2Hop-Greedy and Greedy-ViP consider the 2-hop neighbors of nodes. As the communication range increases, 2Hop-Greedy and Greedy-ViP demonstrate similar success rates, which are higher comparing to Greedy. Greedy-MVP using 2nd-level virtual position results in even better success rate since it considers 3-hop neighbors. All the simulated algorithms reach full delivery of packets in densely-deployed networks (when transmission range reaches 120m in Fig. 5(a)). Fig. 5(b) illustrates the performance of the down-hill scaling in the Greedy-HVP algorithms start with 1st-level and 2nd-level virtual positions. Comparing to the corresponding Greedy-MVP algorithms using the same level of virtual positions, Greedy-HVP has higher success rate due to its additional search on virtual positions of lower levels and the geographic positions of nodes.



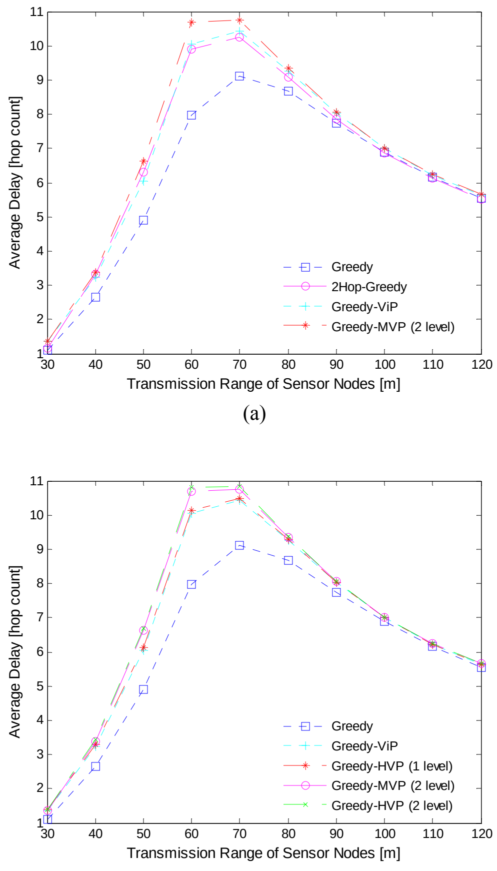
(a)

(b)

Fig. 5. Average success rate of the Greedy algorithms with different communication ranges

Fig. 6 shows the average delay of the successfully delivered packets from our simulation. When the network is sparsely-deployed (transmission range < 70m in Fig. 6(a)), the length of routing paths increases with the increasing of the communication range. The reason is that more packets with distant source and destination can be delivered when the connectivity of a network increases. 2Hop-Greedy, Greedy-ViP and Greedy-MVP (using 2nd-level virtual position) have bigger average delay, since additional packet with source-destination pair of longer distance can also be delivered with the look-ahead algorithms. As the transmission range reaches 70m (Fig. 6(a)), the hop count starts to fall due to the high success rates and the increasing step-size of forwarding. We observe that the results of the simulated algorithms are similar when the success rates reach 100%. Therefore, the look-ahead routing algorithms (2Hop-Greedy, Greedy-ViP and Greedy-MVP) have similar routing efficiency as the Greedy algorithm.

Fig. 7 illustrates the control overhead of the simulated



(a)

(b)

Fig. 6. Average delay of the Greedy algorithms with different communication ranges

algorithms indicated by the number of neighbor entries on nodes. With similar performance in terms of success rate, Greedy-ViP results in lower storage and computational overhead compared to the 2-Hop Greedy algorithm. As an extension of the 2-Hop Greedy algorithm, the rapidly increasing overhead of the 3-Hop Greedy algorithm shows that the N-Hop Greedy idea is not scalable. In contrast, the number of neighbor entries of our Greedy-ViP and Greedy-MVP algorithms stays low as the Greedy algorithm. The storage and computational overhead of Greedy-HVP increases linearly along with the level of virtual position it starts with.

V. Conclusion and Future Work

In this paper, we present a novel geographic routing algorithm named “Greedy Forwarding with Virtual Position (ViP)", as well as its two extensions called the “Greedy Forwarding with Multi-level Virtual Position (MVP)" algorithm and the “Greedy Forwarding with Hierarchical Virtual Position (HVP)” algorithm. Each of the three algorithms has two variants based on the principle of the Greedy algorithm and the “Most Forwarding progress within Radius (MFR)” algorithm, respectively. Simulation results demonstrate that our proposed algorithms improve the success rate of geographic routing for sparsely-deployed WSNs and WSNs with small routing holes. The proposed algorithms solely employ Greedy Forwarding (GF) throughout the routing processes, and inherently results in high routing efficiency as the basic GF algorithms. Furthermore, the control overhead of our proposed algorithms is strictly limited.

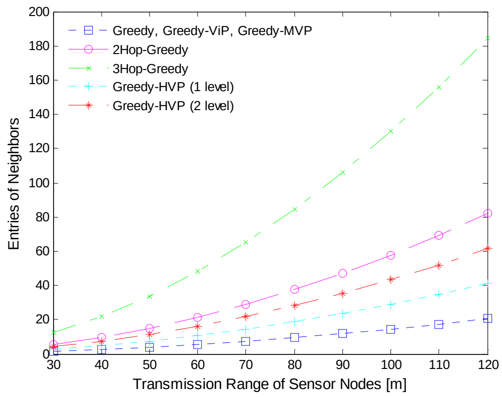


Fig. 7. Number of neighbor entries with different communication ranges

For future work, the trade-off between the overhead caused by using higher levels of virtual positions and the resulted performance will be formally analyzed. Context information such as remaining energy will be considered as the routing parameter for our algorithms.

References

[1] K. Akkaya and M. Younis, "A survey of routing protocols for wireless sensor networks", Elsevier Ad Hoc Network Journal, 2005, Vol. 3/3, pp. 325-349.

[2] C.E. Perkins and E.M. Belding-Royer, "Ad hoc on-demand distance vector (AODV) routing", in IEEE Workshop on Mobile Computing Systems and Applications, Feb. 1999.

[3] Pham, N. N., Youn, J., and Won, C. "A Comparison of Wireless Sensor Network Routing Protocols on an Experimental Testbed", In Proceedings of the IEEE international Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing - Vol 2 - Workshops - Volume 02 (June 05 - 07, 2006). SUTC. IEEE Computer Society, Washington, DC, 276-281.

[4] G. G. Finn. "Routing and addressing problems in large metropolitan-scale internetworks", Technical Report ISI/RR-87-180, Information Sciences Institute, Mars 1987.

[5] H. Takagi and L. Kleinrock, "Optimal Transmission Ranges for Randomly Distributed Packet Radio Terminals", IEEE Trans. Comm., vol. 32, no. 3, pp. 246-257, 1984.

[6] Nadeem Ahmed , Salil S. Kanhere , Sanjay Jha, "The holes problem in wireless sensor networks: a survey", ACM SIGMOBILE Mobile Computing and Communi-cations Review, v.9 n.2, April 2005.

[7] Brad Karp and H. T. Kung, "GPSR: Greedy perimeter stateless routing for wireless networks", in Proceedings of the ACM/IEEE MobiCom '2000, pages 243-254, 2000.

[8] P. Bose, P. Morin, I. Stojmenovic, and J. Urrutia, "Routing with guaranteed delivery in ad hoc wireless networks", Wireless Networks, 7:609-616, 2001.

[9] Ke Liu, Nael Abu-Ghazaleh, "Aligned Virtual Coordinates for Greedy Routing in WSNs", Mobile Adhoc and Sensor Systems (MASS), 2006 IEEE International Conference on , vol., no., pp.377-386, Oct. 2006.

[10] Stojmenovic, I. and Lin, X, "Loop-Free Hybrid Single-Path/Flooding Routing Algorithms with Guaranteed Delivery for Wireless Networks", IEEE Trans. Parallel Distrib. Syst. 12, 10 (Oct. 2001), 1023-1032.