# Efficient Communications in Mobile Hybrid Wireless Networks

Xueli Huang and Xiaojiang Du
Dept. of Computer and Information Sciences
Temple University
Philadelphia, PA, 19122 USA
Email: {xueli.huang, dux}@temple.edu

Xiaoming Li and Kaigui Bian
School of Electrical Engineering and Computer Science
Peking University
Beijing, China
Email: {lxm, bkg}@pku.edu.cn

***Abstract*—In this paper, we present an efficient routing protocol for *Mobile Hybrid Wireless Networks* (MHWNs), which consists of an infrastructure wireless network and a few mobile ad hoc networks. MHWNs have several advantages over traditional wireless networks (such as cellular networks and WLANs). For example, MHWNs can achieve higher throughput, larger coverage, and better load balancing cross cells. In this paper, we present an efficient routing protocol for MHWNs. The protocol utilizes node location information, moving speed, and remaining battery life for routing decision. The protocol is energy efficient (for mobile nodes) and can find robust routing path. We evaluate its performance by NS-2 simulations. Our results show that the protocol significantly reduces the routing overhead, and the routing path is relatively stable.**

***Index Terms*—Communications, hybrid wireless networks, routing**

## I. Introduction

There has been fast grow of wireless access in the past few years due to the wide deployment of Wi-Fi, 3G, and 4G, e.g., Long Term Evolution (LTE) and WiMAX. According to report [1], the smart phone shipments in 2011 grew by 62.7 percent compared to 2010. The shipments of 159 million units in the fourth quarter of 2011 represent a 56.6 percent increase from the same period in 2010. The wide use of smart phones and their ability to operate in ad hoc mode make it possible for an emerging type of wireless networks - *mobile hybrid wireless networks* (MHWNs).

A MHWN consists of an infrastructure wireless network and a few mobile ad hoc components. MHWNs can achieve better performance than traditional wireless networks (such as cellular networks and WLANs). For example, MHWNs can achieve higher throughput, larger coverage, and better load balancing cross multiple cells.

A number of literatures have discussed the concept of MHWNs. However, few works have studied efficient routing in MHWNs. In this work, we design an efficient routing protocol for MHWNs, which is referred to as Hybrid routing protocol. Our MHWN model has an open architecture, and it is not limited to any particular wireless standards. For the ease of discussions, in this paper we adopt 4G cellular network (e.g., LTE) for the infrastructure part and IEEE 802.11 ad hoc mode for the ad hoc part of MHWNs.

## II. Related work

A lot of works have been done on mobile ad hoc networks (MANETs) routing. Ad hoc on demand Distance Vector routing (AODV) uses on-demand route discovery [2]. In AODV, when a source node wants a route to a destination, it broadcasts a route request (RREQ) packet in the network. A node receiving the RREQ may send back a route reply (RREP) if it is either the destination or it has a route to the destination; otherwise, it re-broadcasts the RREQ to its neighbors. An old RREQ is discarded.

With the proliferation of GPS-based technology in mobile devices such as smart phones, more location-based routing algorithms have been proposed. In MFR (Most Forward within Radius) algorithm [3], a packet is sent to a neighbor with greatest progress towards destination node. In DREAM [4] and LAR [5], location information is used to direct routing discovery. LAR takes into account the physical location of destination node, and limits the routing discovery to a smaller "request zone" of the network, which reduces the number of routing messages. Energy Efficient LAR (EELAR) [6] reduces the area of discovering a new route to a smaller one by having a BS in the center of an ad hoc network. The BS is used to collect node's information and send the destination's location to a source node. In EELAR, the BS divides the network into 6 zones. Each node only forwards messages from nodes in the same zone. This reduces the routing discovery overhead. However, this method significantly increases the cost because a BS is much more expensive than a mobile node. In PBHRA [7], a portion of nodes function as master nodes and they assist routing discovery of other nodes. A source node requests path information of destination from master nodes. In PBHRA, master nodes consume more energy than other nodes.

AODV and most other position-based routing algorithms require that a source knows the location of the destination. This requirement either causes large routing overhead or makes it hard to satisfy. In addition, the routing overhead of AODV becomes very large when node dense is high (e.g., in metropolitan areas), because much more RREQ packets are broadcasted.

In our protocol, each node only sends its RREQ packet to its favorite neighbor that is selected according to location,

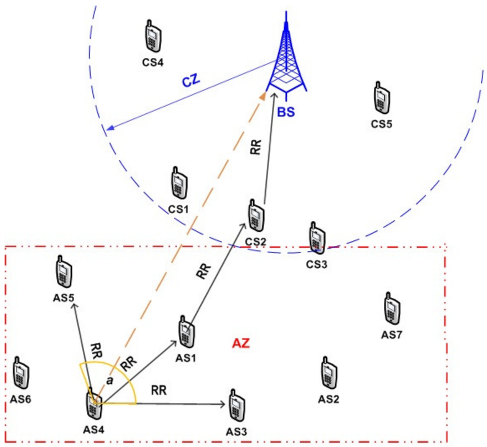


Fig. 1. The architecture of MHWNs

moving speed and remaining energy of neighbor nodes. A gateway node is one that can directly communicate with the BS. If the next hop node is not a gateway node, it forwards the packet to its own favorite neighbor. Otherwise it sends the packet directly to the BS. According [8], our algorithm is loop-free. When the BS receives RREQ packets, it finds the best path of which nodes have less mobility and more remaining energy, and send reply packets to nodes along the path. The General Purpose Computing on the GPU (GPGPU) may be used to increase the speed 9 to 12 times over the traditional CPU [9]. A BS may use GPGPU to calculate routing paths in a MHWN.

## III. The Architecture of MHWNs

A MHWN consists of both infrastructure nodes and ad-hoc nodes. A MHWN may be divided into two zones, as illustrated in Fig. 1.

1) Cell Zone (CZ) - Subscriber Stations (SSs) within the transmission range of the BS. A Cell zone SS (CS) with 4G cellular radios can directly communicate with the BS. The MAC used in CZ is OFDMA.
2) Ad hoc Zone (AZ) - SS outside the transmission range of a BS. Ad hoc zone SS (AS) use multi-hop relay to connect to a CS and then to a BS or RS. AS use IEEE 802.11 (Wi-Fi) as the MAC protocol. Note: A CZ node also uses Wi-Fi to communicate with an AZ node.

Based on the above architecture, communications in MH-NWs are divided into two parts: (1) communications in the AZ and (2) communications in the CZ. The CS nodes that connect with AS nodes are referred to as gateway nodes, for example, CS2 and CS3 in Fig. 1. The gateway node plays an important role for routing in MHWNs, since it inter-connects the two zones. To make the routing path stable, one should select nodes with slow moving speed and/or high remaining energy as gateway nodes.

## IV. Efficient Routing in MHWNs

In this work, we design an efficient routing protocol for MHWNs. The routing protocol consists of several parts, which are presented in the following subsections.

### A. *Node's classification algorithm*

According to the node classification, the node is a CS if it can directly receive packet from BS; otherwise it is an AS. Beacon packets are used to exchange information between mobile nodes and their neighbor nodes. Two types of beacon packets are defined in our algorithm: type 0 is for node labeling and type 1 is for finding its favorite neighbor. Both beacon packets contain the source node ID and a label.

In Hybrid routing, the network operation is divided into multiple rounds. Similar to the Hello packet in AODV, beacon packets are broadcasted periodically by both BS and mobile nodes. For a node just moved into the AZ, it does not have the BS's location. An existing AZ node sends the BS's location to it. A new node assigns itself with an unknown label, and then it updates the label periodically. At the end of each round, each node updates its label as follows: (1) if it can only receive packets from the BS, then it is a CZ node (CS label); (2) if it can receive packets from both the BS and AZ nodes, then it is a gateway node (GS label); (3) if it cannot receive any packets from the BS, then it is an AZ node (AS label).

**Algorithm 1** Node Label

```
Input:the packet coming in.
for each mobile wireless node do
  if timer interval is expired then
    if N_BS > 0 then
      label = CS; /*in Cell Zone */
      if N_AS > 0 and Energy > E and Speed < S
      then
        label = GS; /* gateway node */
    else
      if The node joins the network then
        store the BS's location;
      lable = AS; /*in ad hoc zone */
      N_AS ← 0 and N_BS ← 0;
  else
    if Packet is from BS then
      N_BS ← N_BS + 1;
    if Packet is from AS then
      N_AS ← N_AS + 1;
    if Packet is from new joint node then
      unicast BS's location to the new node;
```

A node needs to store the following information: current and previous locations, the travel time, remaining energy, its label, an angle of the communication sector (see Fig. 1), the favorite neighbor node that is one hop away and has a high score.

### B. *The algorithm for selecting favorite neighbors*

When CS nodes have traffic, they directly send their request packets to the BS, and the BS determines and assigns channels to them. For AS nodes, they send the routing request packet to their favorite neighbors. If an AS node cannot connect with

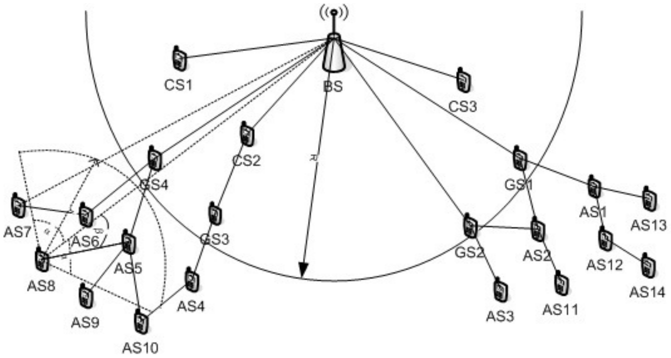


Fig. 2. A MWHN with two separate MANETs

any of its favorite neighbors or has no favorite neighbor, it broadcasts type-1 beacon packets to its one-hop neighbors. A mobile wireless node $N_j$ that receives this kind of packet from node $N_i$ checks if it has enough energy and slow moving speed. If yes, it checks if the same packet was received during the last time slot. If not, it calculates the angle $\beta$ of the three nodes - $\angle(BS, N_i, N_j)$. If $N_i$ finds that it received the same packet at the last time slot, it will just use the previous $\beta$. After $N_i$ calculates the angle $\beta$, it will mark the packet if $\beta$ is larger than the angle $\alpha$ of $N_i$, which means that node $N_j$ is outside of $N_i$'s predefined communication sector. If $\beta$ is no larger than $\alpha$, it will unicast to $N_i$ the following: its node ID, remaining energy, moving speed, angle $\beta$, its gateway node and distance from the BS. For nodes (say $N_k$) outside the sector, the beacon packet is discarded if it does receive a new beacon packet in the next round, which means node $N_i$ already found its favorite neighbor. If $N_k$ receives a new beacon packet in the next round, it just uses the result calculated in the previous round. Node $N_i$ determines its favorite neighbor at the end of each time slot. $N_i$ increases it angle if it cannot receive any packet from its neighbor. If there are too many neighbor nodes in its sector, it decreases the angle. For each packet from neighbor $N_j$, $N_i$ calculates a gain value as follows:

$$\begin{aligned} A(i,j,BS) =& \gamma \cdot \sin\left(Dis(i,j)/R\right) \cdot (\pi/2)) + \\ & \epsilon \cdot \cos\left((Dis(j,BS)/R_{BS}) \cdot \pi/2)\right) + \\ & \mu \cdot \cos\left((S_j/S)(\pi/2)\right) + \delta(E_j/E) \end{aligned} \quad (1)$$

where $Dis(i,j)$ stands for the distance between node $N_i$ and $N_j$; $E_j$ is the remaining energy in node $N_j$; $E$ is the maximum energy; $S_j$ is the moving speed of node $N_j$. If two favorite neighbors use the same gateway node, node $N_i$ chooses the one with the most gain value. Each node stores no more than three favorite neighbors.In our simulations, we use the following values: $\gamma = 1$, $\epsilon = 1$, $\mu = 1.2$, and $\delta = 1$.

Fig. 2 illustrates a MWHN that has two separate MANETs, one on the left and one on the right. Let's take node AS8 as an example. When AS8 has traffic to send, it broadcasts type-1 beacon packet to its one-hop neighbors, and nodes AS5, AS6, AS7, AS9 and AS10 receive the packet. Each of the nodes that have enough energy and slow moving speed calculates the angle $\beta$ , and unicasts its information to AS8 when angle $\beta$ is less than $\alpha$. If AS8 does not receive any reply packet from its one-hop neighbors, it increases its angle $\alpha$, and broadcasts the beacon packet again in the next time interval. Neighbors that calculated $\beta$ in the previous time slot marked the packet just obtain the previous $\beta$ value and then do the following things.

**Algorithm 2** Selecting favorite neighbors

Input: neighbor beacon packet with type 1, which contains source node location and its angle
**for** each 1-hop neighbor node $N_j$ of source node $N_i$ **do**
  **if** $E_j > E_{threshold}$ and $S_j < S_{threshold}$ **then**
    **if** packet cannot be found in the marked packet **then**
      $\beta \leftarrow CalculateAngle(N_i, N_j, BS)$;
    **else**
      Get $\beta$ from the buffer;
      Discard the marked packet;
    **if** $\beta < \alpha$ **then**
      $Unicast(nodeid, location, E_j, S_j, \beta, gatewaynode, distancefromBS)$ to $N_i$;
    **else**
      **if** packet cannot be found in the marked packet **then**
        Mark the packet;
      **else**
        Discard the packet;
  Discard the packet;
**for** each source node $N_i$ **do**
  **if** timer is expired **then**
    **if** $n < 1$ and $\alpha \leq \pi$ **then**
      $\alpha \leftarrow \alpha + \alpha$;
      Broadcast neighbor the beacon packet again;
      Reset timer;
    **else**
      **if** $n > N_{threshold}$ and $\alpha > 0$ **then**
        $\alpha \leftarrow \alpha/2$;
    $n \leftarrow 0$ and $m \leftarrow 0$;
  **else**
    $n \leftarrow n + 1$;
    **if** $n < N_{threshold}$ **then**
      $g_{i,j} \leftarrow A(i, j, BS)$;
      **if** $N_j$ has same gateway node with $N_k$ which has been chosen as $N_i$'s favorite neighbor **then**
        **if** $g_{i,j} > g_{i,k}$ **then**
          Replace $N_k$ with $N_j$;
          $Min \leftarrow \max\{Min, g_{i,j}\}$;
    **if** $m < 3$ and $n \geq N_{threshold}$ **then**
      $m \leftarrow m + 1$;
      $Min = \min\{N_i^{\prime}\text{s all favorite neighbor}\}$;
    **if** $m \geq 3$ and $n \geq N_{threshold}$ and $g_{i,j} > Min$ **then**
      Replace the $Min$ neighbor node with $N_j$;
      $Min = \min\{N_i^{\prime}\text{s all favorite neighbor}\}$;

As we can see from Fig. 2, AS8 gets reply from node AS5 and AS6, then it calculates their gain values respectively according equation (1). Since AS5 and AS6 have the same

**Algorithm 3** Routing discovery

```
for each node sending a routing request packet do
  if node's label is CS or GS then
    Unicast routing request packet directly to BS;
  else
    for each of its favorite neighbor node do
      Unicast routing request packet;
for each node in the mobile hybrid wireless network do
  if packet is routing request then
    if node'label is GS then
      Unicast the routing request packet to BS directly;
    else if node is on one routing path then
      Unicast the packet along its path to BS;
    else
      for each of its favorite neighbor node do
        Unicast routing request packet;
```

gateway node GS4, it chooses the node that has larger gain value as its favorite neighbor. For node AS10, it receives reply packet from node AS4 and AS5, and records them both as its favorite neighbors because they have different gateway nodes.

### C. Routing discovery

As mentioned above, CS or GS nodes send routing request packets to the BS directly because they are only one hop away from the BS. An AS node broadcasts its routing request packet to all of its favorite neighbor nodes. If a favorite neighbor knows a good path from itself to the BS, then it uses that path as part of the route (from the original source node to the BS). Otherwise, it sends the packet to its favorite nodes. This process continues until the route discovery packet reaches the BS.

Take AS10 node for example, it multicasts its routing request packet to AS4 and AS5. Then AS4 (AS5) forwards the packet along the path AS4→GS3→BS (AS5→GS4→ BS) to the BS. It is the BS that calculates the routing path for each of mobile nodes that have traffic to send.

### D. Routing decision

When a node of class 1 receives a beacon from the BS, it sends its information to the BS. Gateway nodes send their information first, and then relay other AS nodes' information to the BS. The BS selects gateway nodes for each AS. The BS balances the relay load among all gateway nodes.

Routing paths among ad hoc nodes are calculated as follows. The BS computes the routing path from each AS to its gateway node by using a shortest path algorithm and also considering the following: try not to use nodes with high moving speed and/or low remaining energy. This makes the routing path more stable.

## V. Performance evaluation

In this section, we evaluate our Hybrid routing protocol using the NS-2 simulator and compare its performance with current AODV routing protocol.

TABLE I

DEFAULT ENVIRONMENT SETTINGS OF THE NS-2 SIMULATIONS

| Parameter | Value |
|---|---|
| Mobile Node Transmission range | 30 meters |
| Mobile Node's Maximum Moving Speed | $0.8, 1.0, 1.2, 1.4, 1.6 (m/s)$ |
| Node Propagation Model | TwoRayGround |
| Number of Node over CBR Traffic | 20% of Mobile Nodes |
| Mobile Node's Initial Energy | Uniformly Distribution |
| Mobile Node's Max Initial Energy | 100 Joules |
| Simulation Duration | 50 seconds |
| Number of Mobile Nodes in $30m \times 30m$ | 10, 20, 30, 40, 50 |

### A. Simulation setup

In this subsection, we will introduce the simulation environment. The default environment parameters used in our simulations are listed in Table I.

Nodes (except the BS) in the simulation move according to the "random waypoint" model. The movement scenario files we used for each simulation are characterized by a pause time. When the simulation begins, each node remains stationary for the pause time. It then selects a random destination in the $30m \times 30m$ square and moves to that destination at a speed uniformly distributed between 0 m/s (meter per second) and a maximum speed, given in Table I. Upon reaching the destination, the node stops again for the pause time, selects another destination, and does the same thing. Each simulation ran for 50 seconds of the simulated time. We used a fixed pause time of 1 second.

Constant Bit Rate (CBR) is used as data traffic at a rate of 8 packets per second and the size of each packet is 64 bytes. 20% of mobile nodes in the left ad hoc network are randomly chosen as CBR sources; and 20% mobile nodes in the right ad hoc network are randomly selected as destinations. Each simulation was run for 50 seconds. We run simulations for different numbers of nodes: 10, 20, 30, 40 and 50. In all these cases, we recorded the average overhead of routing discovery. Under our routing protocol, all the traffic goes through the BS. In the simulations, on average 20% mobile nodes are within the BS transmission range. The network topology is shown in Fig. 3. The terrain dimension is $60m \times 120m$, and there is a $30m \times 30m$ square at the left side and right side. A MANET is form in each square. The distance between the two squares is larger than the mobile node's transmission range. Hence, communications from one square to another need to go through the BS.

### B. Results and discussion

We measure the overhead of routing discovery and routing maintenance, and compare the performance of our protocol with AODV. In the first set of experiments, there are 40 mobile nodes in each of the two ad hoc networks, and we vary the node maximum moving speed. The results are plotted in Fig. 4, which shows that our MHWN routing protocol has much smaller routing overhead than AODV, for all the tested moving speeds. Our protocol chooses more stable nodes for relay and

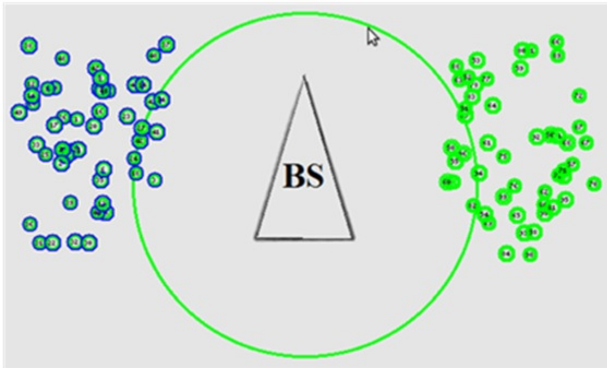


Fig. 3. The layout of simulation

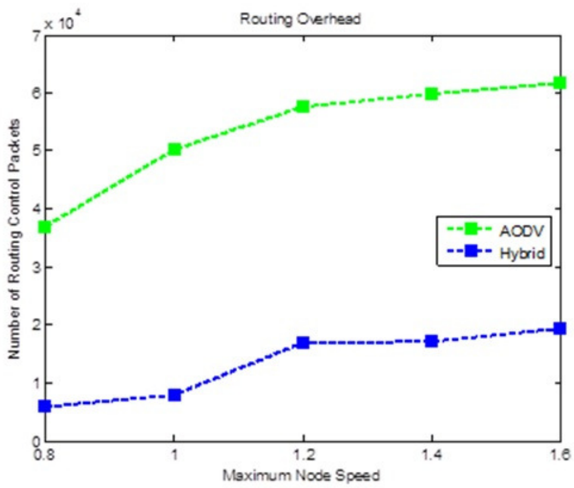


Fig. 4. Comparison of routing overhead under different moving speeds

hence has less broken links. If there is a broken link, the node discovering this only forwards the route error packet to the BS, instead of flooding it (as in AODV). This also reduces the routing overhead.

In the second set of experiments, we fix the maximum moving speed to $1.2m/s$, while varying the number of nodes in each ad hoc network. Fig. 5 plots the results, and it shows that: (1) our protocol always has less overhead than AODV; (2) when the node density is low, the overhead of AODV is similar to that of our protocol; however, when the node density increases, AODV's overhead increases much faster than ours. Under our protocol, nodes in higher density networks have more choices to select their favorite neighbors, and hence there are more stable neighbors and route, which reduce the number of packets to find their favorite neighbors. Furthermore, in our protocol, the routing request packet is only sent to favorite neighbors (not all neighbors). However, in AODV, when a node sends out routing request packet, all of its neighbors that have not received the same packet must re-broadcast the request packet, which dramatically increases the overhead when node density increases.

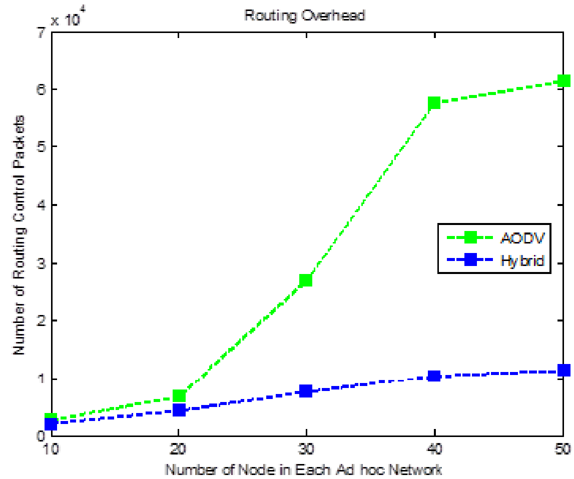


Fig. 5. Comparison of routing overhead for different node densities

## VI. Conclusion

In this paper, we designed an efficient routing protocol for Mobile Hybrid Wireless Networks (MHWNs). The protocol utilizes node location information, moving speed, and remaining energy for routing decision. The protocol is energy efficient and can find robust routing path. We evaluated the routing performance by using NS-2 simulations, and compared its performance with a popular ad hoc routing protocol - AODV. Our results showed that our protocol significantly reduces the routing overhead.

## Acknowledgment

This research was supported in part by the US National Science Foundation (NSF) under grants CNS-0963578, CNS-1002974, CNS-1022552, and CNS-1065444, as well as the US Army Research Office under grant W911NF-08-1-0334.

## References

[1] D. Poeter, "Report: Smartphone shipments overtake pcs in 2011," tech. rep., 2011.

[2] S. Das, C. Perkins, and E. Royer, "Ad hoc on demand distance vector (aodv) routing," *Mobile Ad-hoc Network (MANET) Working Group, IETF*, 2002.

[3] H. Takagi and L. Kleinrock, "Optimal transmission ranges for randomly distributed packet radio terminals," *IEEE Transactions on communications*, 1984.

[4] S. Basagni, I. Chlamtac, V. Syrotiuk, and B. Woodward, "A distance routing effect algorithm for mobility (DREAM)," in *Proc. of the 4th annual ACM/IEEE international conference on Mobile computing and networking*, 1998.

[5] Y. Ko and N. Vaidya, "Location-aided routing (lar) in mobile ad hoc networks," *Wireless Networks*, 2000.

[6] M. Mikki, "Energy efficient location aided routing protocol for wireless MANETs," *International Journal of Computer Science and Information Security*, 2009.

[7] R. Kara, I. Ozcelik, and H. Ekiz, "A new routing algorithm in MANETs: Position based hybrid routing," *Sci. Res. Essays*, 2010.

[8] X. Lin and I. Stojmenovic, "Gps based distributed routing algorithms for wireless networks," tech. rep., University of Ottawa, 2000.

[9] J. Soman, K. Kishore, and P. Narayanan, "A fast GPU algorithm for graph connectivity," in *Parallel & Distributed Processing, Workshops and Phd Forum (IPDPSW), 2010 IEEE International Symposium on*, 2010.