Robot Navigation in a 3D World Mediated by Sensor Networks

D. V. Karthikeya Viswanath* K. Madhava Krishna* Henry Hexmoor+ Sartaj Singh**

* Robotics Research Center, International Institute of Information Technology, Hyderabad, India
+Computer Science Dept, Southern Illinois University, Carbondale, IL, 62901, USA
** Center for Artificial Intelligence and Robotics, Bangalore, India.
viswanath@research.iiit.ac.in mkrishna@iiit.ac.in hexmoor@cs.siu.edu sartaj@cair.res.in

Abstract— *A methodology for real-time navigation of an all terrain vehicle capable of navigation between floors of a building mediated by a wireless sensor network is detailed in this paper. The map of the building is unknown to the robot. Sensor motes scattered in the building guide the robot to its desired destination. The guidance is in the form of specifying the next waypoint to the goal by the network. Local navigation between waypoints is achieved by a fuzzy logic based navigation system.*

1. INTRODUCTION

Sensor network mediated robot navigation has become popular [1-3] in recent years from different viewpoints. Firstly the network acts as a computing medium thereby reducing the computational payload on-board the robot. In a manner akin to swarm robotics where each of the individual entity has limited intelligence but the group in itself behaves as a sufficiently intelligent system, sensor network allows the robotic agent to be possessed with minimal decision making capabilities but the network plus the robot behaves as a system of enhanced intelligence. Secondly the network provides for fault tolerance capabilities for if the onboard sensors fail or misbehave the robotic agent can look up to the sensor network for information about the environment. Thirdly the network supplements the computational capacity of the robot. Efficiently designed sensor fusion algorithms can agglomerate intelligence gathered through onboard as well as off board resources to come up with robust decisions.

In this paper we describe the problem of navigation in a 3D environment where the decision making happens through the network while minimal obstacle avoidance maneuvers based on fuzzy inference system [4] exists onboard the agent. The system is being developed for 3D navigation of an all terrain vehicle (ATV) developed at the Center for AI and Robotics, Bangalore [5].

The essential novelty of this work is that among the survey of papers on a similar theme the authors have not come across one that provides for navigation in a 3D world mediated by a sensor network. The methodology uses the hop count propagation method to find the next waypoint to the goal [1]. However unlike in [1] since the sensor motes are scattered in a 3D world motes need to be programmed to discard packets that are received from sender motes that are not in the same z plane as the sender. This constraint is relaxed for motes on stairways since stairways are the gateways for navigating between planes. For this purpose sensor motes are classified as corridor or stairway or room motes depending on where they are placed. We also introduce the idea of building topology of a region from a sensor network which aids the navigation of the all terrain vehicle.

The work most similar to the current is found in [1,2]. In [1] sensor network mediated robot navigation is accomplished through a potential aggregation protocol. Motes develop potential depending on the closeness to the goal node and their proximity to dangerous regions in the habitat. This method is more suitable for navigation in an open 2D area that can also be ridden with some danger rather than an office like environment comprising of rooms, floors and stairways. In [2] an indoor navigation protocol mediated by sensor network is accomplished through a value iteration method. The next waypoint to reach is probabilistically chosen through the value iteration method. However this paper does not suggest ways of extending it in a 3D setting.

2. PROBLEM FORMULATION

Given: A building whose map is unknown and with multiple floors and sensor motes placed in corridors, rooms and staircases.

Objective: To navigate the robot from its current position to a desired destination.

Assumptions:

a. The positions of the sensor motes are known to themselves with respect to a global reference frame
b. Each sensor mote knows the ids of two of its connected neighbors.
c. The initial position of the robot is known with respect to the same reference frame and the robot can know its neighborhood based on signal strength received from motes.

d. The kinematics of the robot permits it to climb and descend stairs as well as travel on flat surfaces.
e. The robot has no *a-priori* knowledge of its environment.

Assumption 'a' is commonly used in approaches that use network topology to guide the robot such as in [1,3]. Motes can be programmed to know their positions manually and these positions can be manually ascertained or could be localized by a navigating robot aware of its position [3]. Assumption 'b' is useful in the construction of robot's surrounding region topology which is discussed in the later sections. Assumption 'c' is again used in network mediated robot navigation algorithms. For example [2] described a DP algorithm to locate the robot's neighborhood. Eventually the algorithm described in this paper will be implemented on the ATV developed at CAIR, Bangalore [5]. Figure 1 shows an image of this vehicle consisting of articulated limbs on wheels capable of climbing and descending stairs and moving up and down slopes. The deployment strategy of motes is as follows:

- Place a mote labeled as 'R' at all four corners of a room.
- Place two motes labeled as 'D' at every door in the building, one on the left side and the other on the right side of the door
- Place motes labeled as 'C' along the walls of corridor at regular distances, every corner of the corridor must have a *corridor* mote.
- Place motes labeled as 'S' on staircases.

Figure 2 shows how the motes will be deployed.

Figure 1: The all terrain vehicle developed at CAIR, Bangalore. The proposed testbed for the current paper

3 COMPUTING ATV PATHS & NAVIGATION

3.1 ATV Path Computation

The path computation for ATV follows the hop count model discussed in [1] however with the following differences:

- Motes are labeled as corridor (C) or stairway (S) or doorway (D) or room (R) depending on where they are placed.
- Motes discard packets received from other motes if they do not belong to the same z plane (in the same floor) unless they are stairway motes.

Packets are sent from the mote closest to the goal area and are received by motes that are within one hop of the sent mote. The sender packet consists of the following information: *moteId, moteLabel, hop count and z coordinate* of the mote. The receiver mote updates its hop count if the hop count of the received packet is less than its current hop value and only then does it broadcast it to others after incrementing the updated hop-count by 1. Thus each mote stores its hop count distance from the Goal. The algorithm for computing the safest path to the goal is given in Algorithm 1.

ALGORITHM1: Safest path to goal

1. Let G be the sensor mote nearest to the goal coordinates
2. Let m_r be the mote fitted on the robot.
3. G broadcasts a message $msg = (moteid = G, moteLabel = L(G), hop(G) = 0, z = z(G))$
4 **for** all motes m_i initialize hop count to $hop_{iG} = \propto$
5. **for** all motes m_i where $m_i \neq m_r$ **do**
6. **for** all received messages m = (m_{id}, L_{id}, hops, z_{id}) **do**
7. **if** $hops+1 < hop_{iG}$ **then**
8. **if** $z_{id} = z_i$ or ($z_{id} \neq z_i$ and L_{id} = S) **then**
9. Broadcast msg = (m_i, L_i, $hop_{iG} = hop_{iG}+1$, z_i)

3.2 Deciding the Next Mote

Algorithm 1 gives the method by which each mote stores its hop count distance from the Goal. The robot whenever it is supposed to decide the mote to which it should move to next sends an inquiry message to the sensor network. The motes which receive this inquiry message reply with their ids and positions. The robot then moves to the mote which has sent the message with least hop count. The algorithm for deciding the next mote to move to is given by Algorithm 2.

3.3 Guiding Navigation to the Next Mote

Since the environment consists of rooms and corridors, moving to a mote is not straightforward. Imagine a scenario where the robot is in a room and the mote to which the robot is supposed to reach is in the corridor (figure 3). The path to the next mote cannot be traversed in a single straight line. We have come up with a strategy wherein we use the topological information of the robot's surroundings to plan the path to the next mote. Whenever the robot has to move to a mote m_j, it broadcasts an inquiry message, when the robot's surrounding motes receive the inquiry message they reply with the labels and positions of themselves and two of their connected neighbors with respect to the global frame. The robot based on the replies it receives constructs a topology of its surrounding region. This topological construction can be done by following a set of rules:

- If the robot receives four replies from motes labeled as 'R' and two replies from motes labeled as 'D' then the robot is in a room.
- If the robot receives replies from motes labeled as 'C' then the robot is in the corridor.
- If the robot receives replies from motes labeled as 'S' then the robot is on the staircase.

After the robot identifies its surrounding motes and the connectivity between them it can construct the actual topology from the positions of the motes. An artificial repulsive potential field is made to act along the walls of the topological map constructed by the robot (figure 4). This potential field keeps the robot away from brushing against the walls while in transit. The path of the robot to the mote m_j is divided into a sequence of steps with each step being the robot reaching the farthest reachable point in the path leading to the *next mote*. The point to which the robot should move at each step can be determined from the following method:

- whenever the robot is in room (determined from the topology), the reachable point will be the door of the room.
- if the goal is in a room and the robot is in the corridor, the reachable point is again the door of the room.
- if the robot and the *next mote* are both in the same room or corridor then the reachable point is the mote itself.
- if the robot and the *next mote* are in different corridors then the next reachable point is the end of the corridor nearest to the mote.

At the end of reaching an intermediate point in the path to a mote, the robot should recompute the path to the mote by broadcasting another inquiry message so that it can identify its new surroundings. Figure 5 shows the actual path in which the robot moves to reach the desired mote. The robot always moves at a distance from the walls because of the potential field acting along the walls. Any obstacle that comes in the path of the robot is avoided by using the fuzzy inference system mentioned in [4]. The algorithm for guiding navigation to the *Next Mote* is given in Algorithm 3.

ALGORITHM2: Choosing the Next Mote

1. **if** m_i is the mote fitted on robot **then**
2. **if** not at the goal G **then**
3. Broadcast inquiry message
4. **for** all received messages m = (m_{id}, L_{id}, pos_{id}, $hops_{idG}$, z_{id}) **do**
5. **if** $z_{id} = z_i$ or ($z_{id} \neq z_i$ and L_{id} = S) **then**
6. Choose the message m with the least hop count
7. Let m_j be the id for the sender of this message and pos_j be its position
8. Move the Robot towards pos_j

ALGORITHM3: Guiding Navigation to Next Mote

1. **if** m_i is the mote fitted on robot **then**
2. **if** not at the goal G **then**
3. Broadcast inquiry message
4. **for** all received messages m = (m_{id}, pos_{id}, pos_{id1}, pos_{id2}, L_{id}, z_{id}) **do**
5. **if** $z_{id} = z_i$ or ($z_{id} \neq z_i$ and L_{id} = S) **then**
6. construct topology and apply repulsive potential field along the walls.
7. move the robot to the next reachable point using fuzzy logic for obstacle avoidance and target reaching.
8. repeat the process till the robot reaches the desired mote
9. **if** m_i is not the robot mote **then**
   if receive inquiry message **then**
   reply with (m_i, pos_i, pos_{i+1}, pos_{i+2}, L_i, z_i) where pos_{i+1} and pos_{i+2} are the positions of the motes connected neighbors

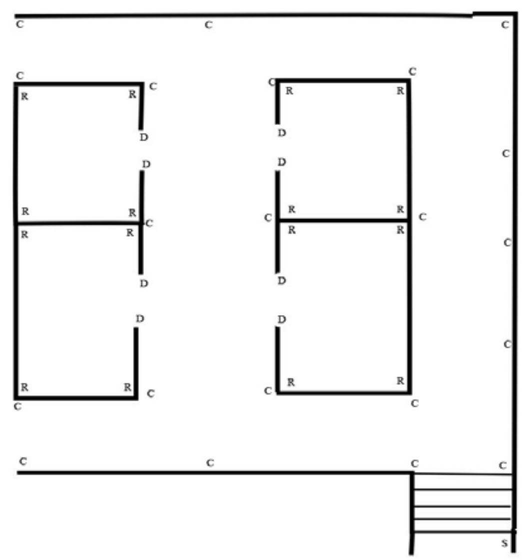

Figure 2: Mote Deployment

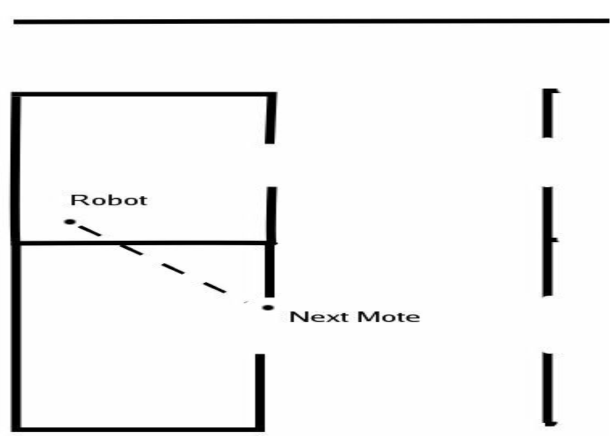

Figure 3: The path to the next mote is not a straight line

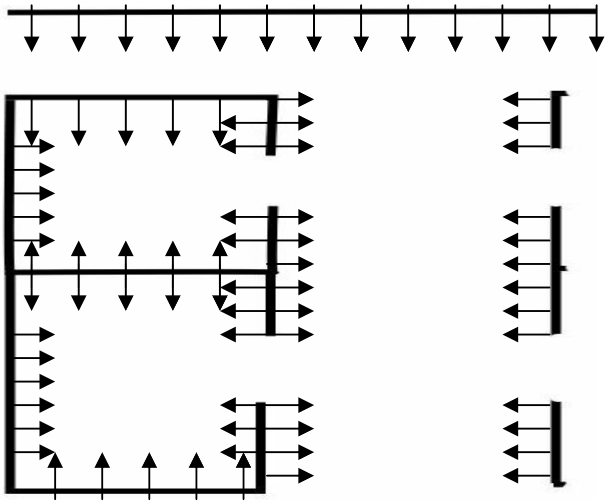

Figure 4: Potential Field acting along the walls.

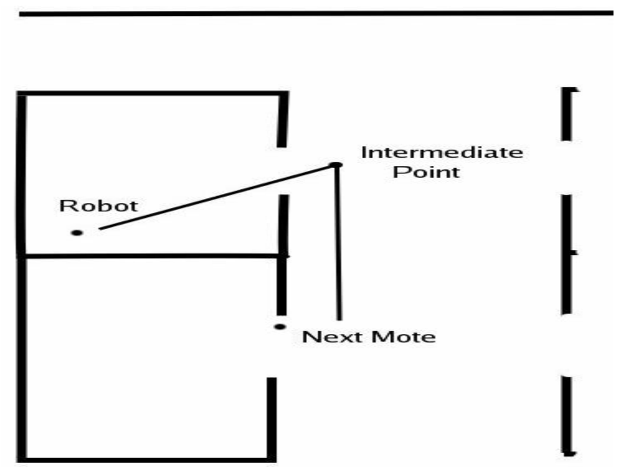

Figure 5: Actual path of the robot.

4 SIMULATION RESULTS

4.1 Simulation Environment

Low-power wireless radios used by real sensor network platforms (*e.g.*, Berkeley motes) are known to have highly irregular communication range and probabilistic link characterization [6]. The simplifying assumptions on wireless radio propagation made by a network simulator may cause simulation results to differ significantly from real-world experimental results. Accurate simulation to the characterization of real wireless radios with different transmission powers is the key for evaluating the realistic performance of our proposed methodology. For this purpose, we have implemented our work in the prowler network simulator [7]. Prowler is a Matlab-based network simulator that employs a layered event-driven structure similar to TinyOS, which allows us to easily implement new network modules and to port our work to Berkeley motes in future. Prowler has been modified to include 3D capabilities and robot navigation. The path obtained from the prowler simulator has been plotted in a realistic 3D office environment as shown in the following figures.

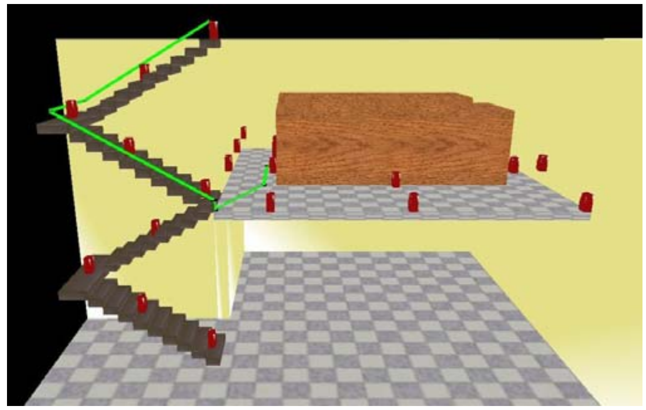

Figure 6: The test environment

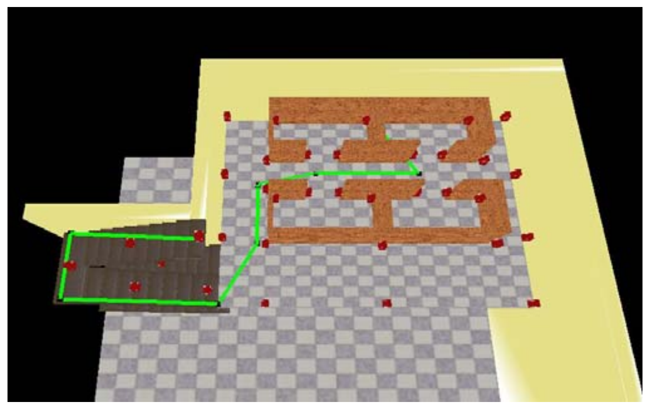

Figure 7: Top view of the test environment

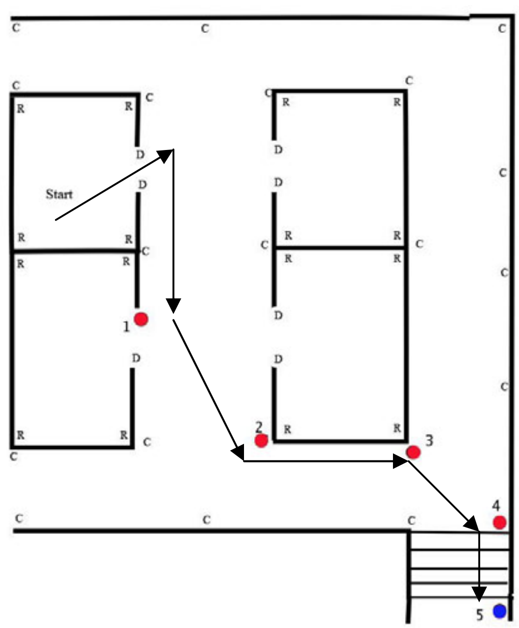

Figure 8: 2D view of the first floor

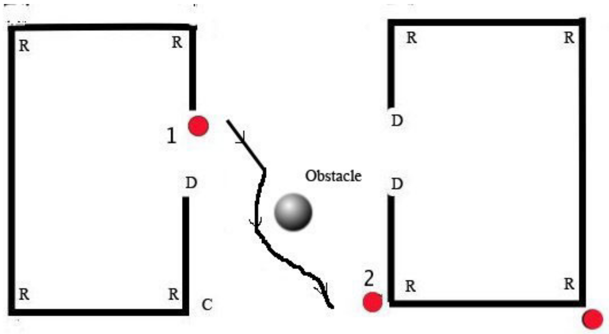

Figure 9: obstacle avoidance using fuzzy inference system

4.2 Simulation Results

Figures 6 & 7 show the front and top view of the environment in which our work was tested. The red cylinders represent the motes. The green line indicates the path that the robot has taken to reach its Goal. Here the Goal is the second floor and robot is in the first floor room 1. Figure 8 shows the 2D view of the first floor. The arrows indicate the path of the robot with each arrow representing one step of the robot navigation. The red dots represent the motes which the robot has tried to reach. The robot starts from the position indicated by 'Start' in the figure. It receives the least hop count message from the mote labeled as 1 in the figure, since the mote is in the corridor and the robot is in a room, the robot breaks its path into two steps, it first reaches the door of the room and then reaches the actual mote. From then onwards the robot reaches those motes from which it receives the least hop count message. Ultimately it reaches the staircase mote indicated by a blue dot. From the staircase it can reach higher floors. Notice that the mote is always at a safe distance from the walls that is due to the repulsive potential field acting along the walls. Figure 9 shows the path of a robot when there's an obstacle in front of it. The fuzzy inference system makes sure that the obstacle is avoided and the robot reaches its goal.

5 CONCLUSIONS AND SCOPE

A new methodology for real-time navigation of an all terrain vehicle capable of navigation between floors of a building mediated by a wireless sensor network is presented. Topological information of the area surrounding the robot is inferred from the sensor network motes, which helps in the robust navigation of the robot. Future scope of this work would be to remove the necessity of a global reference frame, the assumption that the motes know their position with respect to the global reference frame could be overcome by developing a methodology to localize the motes depending on the strength of the signal received by the robot. The number of motes required could be minimized by developing heuristics for the navigation guiding algorithm.

ACKNOWLEDGEMENT

We sincerely thank Mr. Hemanth Kumar, RRL, IIIT Hyderabad, for his work in developing the graphics shown in the simulation section.

REFERENCES

[1] Q. Li, M. DeRosa, and D. Rus, "*Distributed algorithms for guiding navigation across a sensor network*," in The 2nd International Workshop on Information Processing in Sensor Networks (IPSN '03), 2003.

[2] Maxim Batalin, Gaurav S. Sukhatme, and Myron Hattig, "Mobile Robot Navigation using a Sensor Network," In *IEEE International Conference on Robotics and Automation*, Apr 2004.

[3] Peter Corke, Ron Peterson and Daniela Rus, "Localization and Navigation Assisted by Cooperating Networked Sensors and Robots" In *International Journal of Robotics Research*, 2005.

[4] K Madhava Krishna and Prem K Kalra, "Solving the Local Minima Problem for a Mobile Robot by Classification of Spatio-temporal Sensory Sequences", *Journal of Robotic Systems*, John Wiley and Sons.

[5] All Terrain Vehicle based on articulated suspension, Center for Artificial Intelligence and Robotics, Internal Report.

[6] Jerry Zhao and Ramesh Govindan, "Understanding packet delivery performance in dense wireless sensor networks," in *Sensys*, November 2003

[7] http://www.isis.vanderbilt.edu/projects/nest/prowler/index.html