

# RFID Based Localization for a Miniaturized Robotic Platform for Wireless Protocols Evaluation

Vikram P. Munishwar, Shailendra Singh, Christopher Mitchell<sup>†</sup>, Xiaoshuang Wang,  
Kartik Gopalan, Nael B. Abu-Ghazaleh

Computer Science, Binghamton University (State University of New York)

<sup>†</sup>S\*ProCom<sup>2</sup>, The Cooper Union for the Advancement of Science and Art

Contact: {kartik,nael}@cs.binghamton.edu

**Abstract**—The proliferation of wireless-enabled portable computing devices has spurred a growing need for efficient and powerful networking protocols. The key challenge in the development of robust wireless networking protocols is an ability to conduct effective and efficient evaluation of the protocol in order to ensure its successful working in real-world settings. We proposed *MiNT-2*, a fresh re-design of the original *MiNT* framework developed at Stony Brook University. One of the fundamental requirements of *MiNT-2* is to provide location awareness of all the nodes within the network. In this paper, we demonstrate the use of radio-frequency identification (RFID) technology in order to carry out localization of the mobile nodes within the system. We also demonstrate the application of the localization system of constructing different scenarios for wireless protocols evaluation.

## I. INTRODUCTION AND MOTIVATION

The widespread use of wireless communication devices has led to an extensive research into new powerful and robust wireless networking protocols and applications. However, evaluation of these protocols is still based on two contrasting methodologies: (1) *software-only simulation*, and (2) *large-scale testbeds*. Although network simulators [1]–[3] present a convenient way of evaluating wireless protocols for different network configurations, they can not account for radio propagation effects such as non-uniform path loss, multi-radio interference, and multi-path fading. On the other hand, large-scale, custom-built wireless network testbeds [4]–[6] preserve the RF propagation characteristics, but are limited in terms of their ability to provide a range of experimental scenarios for evaluation, and are extremely expensive from the standpoints of experiment setup and routine maintenance.

In order to address these limitations, the Stony Brook University created the *MiNT* testbed [7], [8], which used mobile robots to transport wireless network nodes, and radio signal attenuators to “miniaturize” the physical space requirements of a multi-hop wireless testbed. We developed the second generation of *MiNT* robotic platform, *MiNT-2* [9], for wireless protocol development and testing with focus on improving the original testbed design and operation along the way through use of more effective algorithms and technologies.

Nodes in *MiNT-2* are designed for low cost and maximal functionality using the latest generation of iRobot Create robots. The Create offers a mobility platform with well-documented x86-compatible libraries and a bidirectional serial

protocol for movement control and feedback, battery charge level, proximity/cliff sensor data, etc. A low-power x86-based embedded controller board (the Soekris net5501) interfaces with the robot and an RFID reader, provides multiple wireless interfaces necessary for multi-channel protocols, and runs a distributed “hybrid” version of the NS2 software package.

Compared to the original *MiNT* testbed that used camera-based localization, our *MiNT-2* prototype uses a simpler and more effective three-stage system of motor commands, distance sensor feedback, and RFID tags on the floor of the testbed that provide authoritative position information. The Roomba robots used in the original *MiNT* prototype could not sense their own positions. As a result, they relied exclusively on the central controller which used a vision-based position/orientation tracking system consisting of six ceiling-mounted webcams with overlapping image planes. This system was used to track node position and orientation and to command the robots on which direction to move at any instant. To account for growing discrepancy over time between the central controller’s per-node position information and the actual node positions, mobile robots were periodically manually brought to fixed locations to re-synchronize their logical and physical node positions. Each node was identified using unique color patches mounted on the mobile nodes. Since this system relied on visual identification, it tended to develop inaccuracies over time if any of the six cameras moved slightly, or color patterns on the nodes faded, or lighting varied.

We redesigned the localization mechanism from scratch in the *MiNT-2* testbed to overcome the limitations of its predecessor. Our new localization mechanism uses inexpensive RFID technology coupled with enhanced mobility sensors within the Create robot to reduce maintenance overheads, which helps achieve high levels of accuracy. Since the RFID tags are distributed within the testbed area, a node can localize itself whenever it crosses an RFID tag, by assigning the position of the tag to itself.

## II. OVERVIEW OF *MiNT-2*

The main components contributing to the new design of the *MiNT-2* node are the iRobot Create platform to achieve node mobility and the Soekris net5501 x86 embedded board (with multiple wireless cards attached) to run networking applications, simulation packages (e.g. NS2), and applications

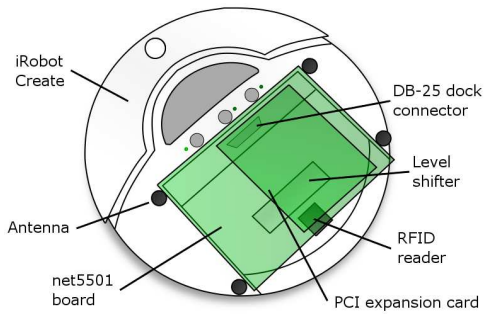


Fig. 1. Physical layout of a MiNT-2 node.

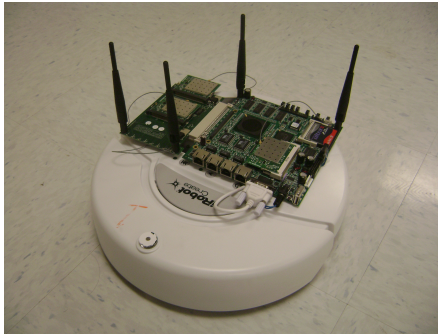


Fig. 2. Picture of a MiNT-2 node.

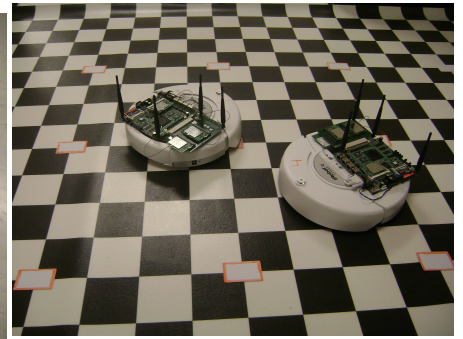


Fig. 3. Picture of the MiNT-2 testbed. RFID cards are marked red.

to control movements of the robot. For automatic recharging of the node's battery, Create robots are designed to use docking stations. Each node has at least two wireless communication interfaces: one to exchange control packets and the other to exchange data packets. The wireless interfaces used in the node design are the R52 802.11a/b/g cards based on the Atheros AR5414 chipset. Each wireless interface is attached to an external antenna through a fixed attenuator, in order to miniaturize the radio communication range. A low frequency RFID reader (125KHz) is also attached to each node in order to be able to localize itself upon detection of an RFID tag. Figures 1, 2 and 3 present the details of the *MiNT-2* node and testbed design.

The key features offered by the *MiNT-2* testbed include:

- *Miniaturization:* *MiNT-2* significantly reduces the physical space requirement of a multi-hop wireless network testbed, by attenuating the radio signals in a controlled fashion. By combining the hardware attenuator (fixed attenuator of 20 dB) and software attenuation control (by varying the transmission power of wireless cards), we are able to adjust the range of each wireless interface at a fine granularity.
- *Autonomic reconfigurability and management:* A key requirement for the *MiNT-2* infrastructure is to be an autonomic testbed that is remotely accessible for 24x7 operation without human intervention. Thus, automating the support for configuring an arbitrary initial network topology as well as setting up an arbitrary node mobility pattern during a simulation run, is an essential component of the system.
- *Support for protocol development, testing, and debugging:* *MiNT-2* is aimed primarily as a platform for wireless protocol implementation, testing, and debugging. We are currently enhancing *MiNT-2* to support network fault injection to test robustness of the protocols, a distributed debugger that will allow a protocol developer to pause/resume, single-step, breakpoint, and roll-back a simulation run, and a visualization interface to enable analysis of the network state at any given instance of time.
- *Running existing simulation code on MiNT-2:* Many

existing wireless protocols are written as NS2 simulation models. Just as with original *MiNT*, *MiNT-2* provides the ability to directly execute existing NS2 scripts and models on the testbed.

### III. NODE LOCALIZATION

The iRobot Create has two built-in sensors to track the robot's movement. These sensors can be queried via the serial interface, and return both distance traveled by the robot since the last query (in mm) and angle the robot has rotated through since the last query (in degrees). However, the accumulated sensor measurements may grow inaccurate due to rounding errors, wheel slippage, and encoder inaccuracy, over time. In addition, a node may be manually picked up and moved to a new location. Thus, we used an RFID based system to periodically re-calibrate the node's position and orientation in the testbed space.

An array of fixed RFID tags deployed on the floor of the testbed allows each robot to determine its absolute location with an uncertainty equal to the maximum tag-sensing radius of the RFID reader (2.25cm). When a robot crosses an RFID tag, the tag value is used to determine the node's absolute  $(x, y)$  position within the testbed.

The heart of the localization algorithm is a section of code running periodically every 50ms that is in charge of acquisition and processing of data from movement sensors and the RFID reader. Figure 4 presents a more structured view of the localization algorithm. Every time the localization tick runs, it reads the delta change in distance and orientation since the last sensor access, adds the changes to the last known position and orientation respectively, and clears the sensors. In addition, more precise position and orientation calibration is performed when passing over RFID tags to remove accumulated error from the Create's sensors feedback. Once at least two tags have been read, the node can determine its orientation from the coordinates of each tag (say  $(x_1, y_1)$  and  $(x_2, y_2)$ ) as:

$$\theta = (\tan^{-1}[(y_2 - y_1)/(x_2 - x_1)] + \Delta\theta_{RFID}/2) \bmod 360 \quad (1)$$

If the node travels in a straight line, then  $\Delta\theta_{RFID}$  is 0. However, if it traveled in a constant-radius arc, its deviation

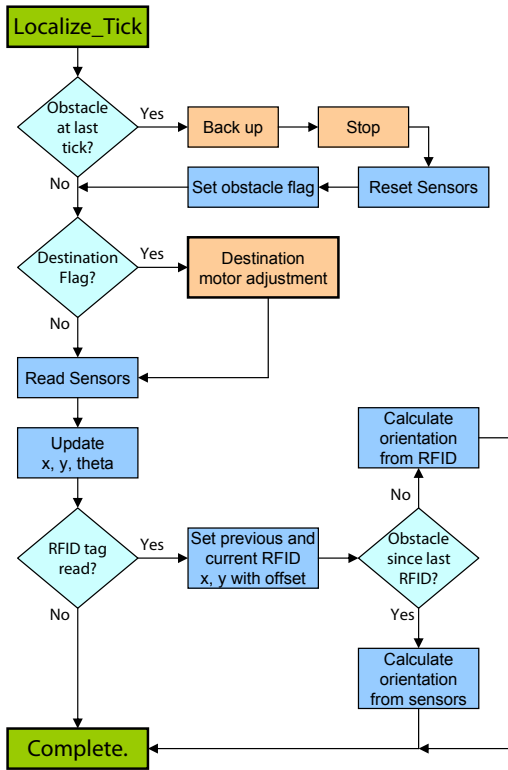


Fig. 4. Scheduled localization tick algorithm.

from the straight line path between the two tags is equal in magnitude at both the tags. Thus the effective change in angle,  $\Delta\theta_{RFID}/2$  is added to the net angle. Figure 5 elaborates more the robot's orientation calculation for the constant-radius arc movement.

In order to account for the situations, where researchers or testbed operators manually drop the robot anywhere in the testbed area, the robots are programmed to initialize themselves with correct coordinates and orientation information. The *position initialization* process is only considered successful if the node reads two RFID tags over a straight line movement.

#### IV. DEMONSTRATION OVERVIEW

In the demonstration, we will show the use-case for the RFID-based localization mechanism, where the nodes can automatically position themselves based on the user specified requirements. Essentially, the user can specify the topology as well as locations of the nodes in the target network, before a scenario execution. After the *position initialization* process, each node travels to the user-specified initial position and orients itself in a specific direction before an experimental scenario begins.

User can also modify the initial location of a node and have the centralized controller (or a base station) instruct the corresponding robot to move accordingly. During this process, every testbed node is constantly measuring the radio signal strength between itself and each of its neighbors within hearing

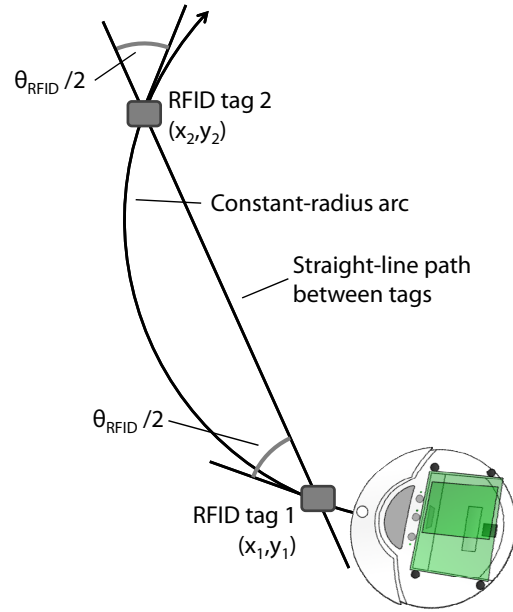


Fig. 5. Orientation calculation for constant-radius arc movement.

range, and relaying the information to the central controller, which then feeds it back to the user. Such interactive initial placement and feedback greatly simplifies the network setup effort because the administrator no longer needs to manually move the robots.

#### REFERENCES

- [1] Information Sciences Institute, "The Network Simulator – NS-2" <http://www.isi.edu/nsnam/ns/>.
- [2] X. Zeng, R. Bagrodia, and M. Gerla, "GloMoSim: A Library for Parallel Simulation of Large-Scale Wireless Networks," in *Workshop on Parallel and Distributed Simulation*, May 1998.
- [3] Scalable Network Technologies Inc., "QualNet WiFi simulator," in [http://www.scalable-networks.com/products/qualnet\\_wifi.php](http://www.scalable-networks.com/products/qualnet_wifi.php), 2004.
- [4] D. Maltz, J. Broch, and D. Johnson, "Experiences Designing and Building a Multi-Hop Wireless Ad-Hoc Network Testbed," in *Technical Report 99-116, School of Computer Science, CMU*, Mar 1999.
- [5] H. Lundgren, D. Lundberg, J. Nielsen, E. Nordstrom, and C. Tscudin, "A Large-scale Testbed for Reproducible Ad Hoc Protocol Evaluations," in *Proceedings of Wireless Communications and Networking (WCNC)*, 2002.
- [6] Daniel Aguayo et al., "MIT Roofnet Implementation" <http://www.pdos.lcs.mit.edu/roofnet/design/>, August 2003.
- [7] P. De, R. Krishnan, A. Raniwala, K. Tatavarthi, N. A. Syed, J. Modi, and T. cker Chiueh, "Mint-m: An autonomous mobile wireless experimentation platform," in *Proceedings of Mobisys*, 2006.
- [8] P. De, A. Raniwala, S. Sharma, and T. cker Chiueh, "Mint: A miniaturized network testbed for mobile wireless research," in *Proceedings of IEEE Infocom*, 2005.
- [9] C. Mitchell, V. Munishwar, S. Singh, X. Wang, K. Gopalan, and N. Abu-Ghazaleh, "Testbed design and localization in mint-2: A miniaturized robotic platform for wireless protocol development and emulation," in *Proc. First International Conference on Communication systems and Networks (COMSNETS)*, Jan. 2009.