Simple Mapping and Path-planning with the Roomba

Jon Olson
College of Computing Georgia Tech

Ivan Kirillov
College of Computing Georgia Tech

John Sweeney
College of Computing Georgia Tech

Abstract

Reactive architectures have gain wide acceptance as a robust and effective architecture design that yields results through the flexibility of its simplicity. Such architectures have made their way into many commercial products, including the popular iRobot Roomba vacuum cleaner. The existing reactive architecture does not always produce results efficiently and by adding mapping functions and planning to the existing hardware cleaning can be improved without adding expensive hardware that would drive up the cost of the cheap, efficient Roombas. Data points are gathered during normal cleaning runs and later used to probabilistically estimate a map of the cleaning space. The mapping space is then tested by directing the Roomba to move to a remote position in the cleaning space.

1 Introduction

Simple robots with a reactive architecture are currently the best available technology to meet the needs of a nascent market for commercial robots. Commercial robots must meet market needs for robustness, simplicity of operation and cost as well as the technical challenges of autonomous operation in the real world. iRobot has offered a series of simple, autonomous cleaning robots for the home and the Roomba is the most popular of all its products. The Roomba relies on a very simple, reactive architecture to vacuum the home of its user through the use of impact sensors and limited use of infrared sensors. iRobot's Roomba does not use any planning or mapping as part of its cleaning protocols. If mapping and planning abilities could be added to these simple robots their cleaning abilities could be expanded and cleaning improved. Vacuuming could be accomplished in less time or more thoroughly. By gathering movement data from the Roomba, a map of the path that had been travelled during the vacuuming process was created. With a map available, it was possible for the Roomba to travel a direct path to a destination that was explicitly called out by the user within its probabilistically mapped space. This directed travel could be further expanded to enable more thorough cleaning of a home by intentionally moving into spaces that are remotely positioned from the Roomba's charging base.

2 Related Work

Different researchers have attempted to make improvements to the cleaning abilities of the Roomba since its introduction in 2002 [3]. The iRobot cooperation has encouraged this experimentation and creativity with its robots, even creating development platform versions of its simple cleaning design. Attempts have been to introduce human-aided planning into the coverage algorithm. One attempt, by Kim, Kim and Kim [2] used existing hardware, which is a good comparison to our own experimental efforts. Introduction of navigational waypoints has been shown to produce better, more thorough coverage in less time. A human user selected starting points for the Roomba to move towards in partitioned areas of the cleaning space. This waypoint selection significantly improved the Roomba's performance over its own random walk. This work was done in simulation though and has obvious limitations. This testing does not reveal the extent of improvement that might be found in a real environment filled with complex obstacles and terrain variations. Further, these improvements require a human user inputs and gains created in cleaning performance would probably be offset by frustration and user anxiety as owners have to program and interface with their vacuums. The simplicity of the Roomba's operation is a key selling point with the public and alterations to operation that increase complexity from the owner's perspective may be met with resistance. Mapping efforts by Rodriguez-Losada, Matia, Jimenez and Galan using SLAM were also reviewed [1]. SLAM faces issues with indoor mapping due to difficulties with pose and position estimation and so extended Kalman filtering was used as a measure to correct the irregularities associated with these estimation errors. The intentional limitation to proprioceptive sensors sets up challenging scenarios those are certainly worth studying because of the number of real-world activities that require proprioceptive sensing of an environment without any absolute reckoning. Our efforts were focused on improvements to a specific product and robotic system though and the simple sensors available on the Roomba might not be capable of the detail necessary for the mapping algorithms used in these experiments. The discussion about global map divergence could provide insight into difficulties fusing sensor maps from different runs of the Roomba through the cleaning space. As newer maps are added to the mapping space how to relate this data to older maps and use it is an important question. Newer data should not just supplant older data, but the maps should be incorporated together to make the map more reliable and detailed.

3 Methods

A wireless Bluetooth connection allows for polling of data from the Roomba. Encoder data can be turned into information about the location information. This information can be used to create a map of the path that the Roomba has travelled while vacuuming. Roomba vacuum cleaners come reequipped with firmware that records movement data for wheel encoders as well as the status of Roomba. Our wireless connection retrieves data for distance moved, stall status, change in rotational angle, and the status of the impact sensor. This impact sensor toggles more quickly than the Bluetooth is polled so this sensor does not ever ”appear” to be on. This data is recorded by the Roomba during the regular cleaning cycle at a frequency of 66.6 Hz. The Bluetooth model polls this data as quickly as possible given the latency involved with communicating between the Roomba and the computer. It is estimated that the latency effects of the Bluetooth module allow for the movement data to be gathered at a rate of 6 to 7 Hz. When the cleaning cycle is concluded and the Roomba returns to its home base to charge its batteries and all the data from the cleaning run is stored. The data that the Roomba returns for the change of rotational angle is provided in degrees, but is truncated. This can lead to significant error compounding, especially during slow turns. As a result, it was necessary to gather data directly from the encoders and process it to determine the angle that the Roomba was turning. All of the discrete data points gathered while vacuuming are plotted and mapped. Map points gathered during previous cleaning runs around the environment are used to create probabilistic maps of the cleaning space using Gaussian approximations of the space. The data points are reduced to 10 percent of their original value. This is done to create continuous lines for the path. Raw position data returned from the Roomba results in gaps spread across the cleaning space. Since, the project does not need to track the path so precisely the condensed data is sufficient for mapping the environment. The mapping concludes that areas that have a high density of points is likely to be free space and that low density of position points are likely to be walls or the space beyond them. A probabilistic approximation is then made around each point using a sigma value. The corners of the map, the extrema, are updated as well if necessary to expand the boundaries of the map as data points are added.

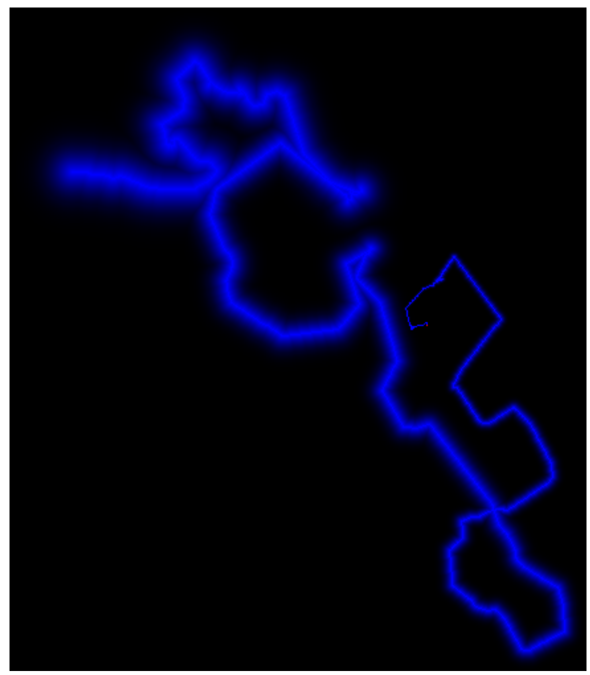


Figure 1: Probabilistic Map created by Gaussian expansion of real data gathered during one cleaning run.

These maps are then used as the basis for planning a path from selected starting point, the home base to the selected end point, the most uncovered areas in apartment. An example path is show below in Figure 2. The path planning works by selecting points that are near neighbors and linking them together to form a path to the selected end point. This methodolgy does not create the most efficient route from the start point to the end point, but does create an effective path.

4 Experiments

Experimental results of five different test runs resulted in probabilistic maps built of the test apartment. The wider,

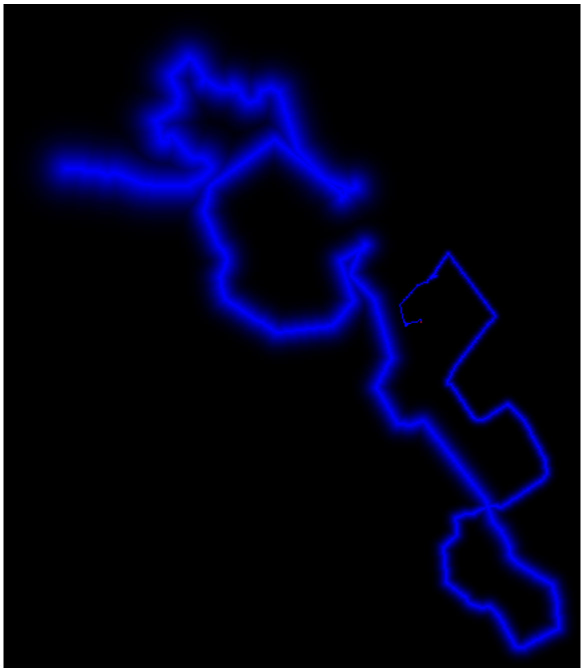
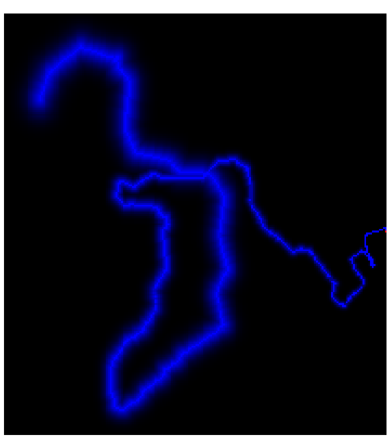
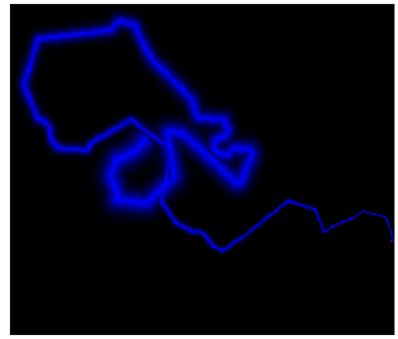


Figure 2: A planned route through the apartment.

brighter areas of the maps below in Figure **??** represent areas receiving more coverage during the random-walk cleaning cycles of the data gathering process. The path planning algorithms overlays the mapped space and compares it to the floor layout of the apartment. This comparison is then searched for the area of highest untraveled space during the Roomba's previous runs. This area is considered to be poorly explored, or cleaned, and is in need of a more direct approach. This algorithm enables the Roomba to effectively search for areas that it has not found during its random-walk cleaning cycles and improve its cleaning coverage. This method of coverage improvement is superior to the human-aided coverage improvement studied previously because it does not impose any additional burdens upon the user.





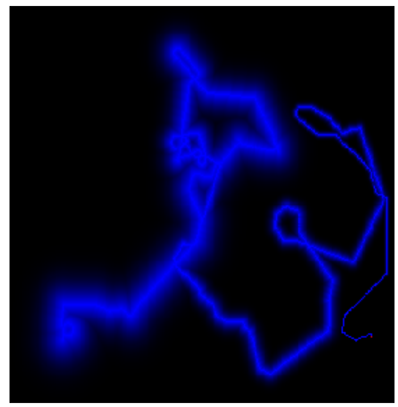


Figure 3: Three probabilistic floor maps of the Test Apartment

5 Analysis

The Roomba's ability to learn about its environment was added to a stock Roomba. Through the use of data gathered from unmodified trial runs using the preprogrammed subsumption architecture the Roomba was able to build a map of its environment without any additional sensors. Use of the data provided by the Roomba during a cleaning cycle for map generation was proven to be reasonably effective. The lack of small changes in trajectory limited the effects of truncation error. This error was further mitigated by the addition of half a degree added to each reported measurement (which assumed a symmetric distribution of fractional degrees prior to truncation). The Roomba was able to determine which parts of free space it was not covering adequately and create new abilities to purposefully travel to those areas of free space. Due to limits in the sampling rate we could achieve with the Bluetooth wireless link it was necessary to drive the robot slowly. At low speeds truncation error quickly becomes a dominating factor. Thus while a map could be generated, a target cleaning point selected, and a path plan created to reach that point, actually executing this plan was impossible. The robot would either col-

Figure 4: Floor plan of the Test Apartment

lide with walls because the low sampling rate prevented it from being able to make a necessary course correction, or the Roomba would become effectively lost due to overwhelming error. Without reliable localization any path generated is effectively meaningless. Attempting to use this data to reliably navigate the robot to a given point, however, presented sufficient difficulty that it made true evaluation of the cleaning performance impossible.

6 Discussion

The ability to sample and control the robot at a higher frequency would likely allow reliable execution of cleaning plans. While the Roomba does offer a high frequency sampling mode, this mode was not available given this experimental setup. The Bluetooth serial module used to communicate with the robot had very limited buffer space. Additionally, the Bluetooth module must occasionally wait to transmit. The Roomba's fixed sampling rate of 66Hz overwhelms this buffer almost immediately when one of these waits occurs. These facts, combined with the bugginess of the module itself (when it is receiving a constant transmission from the robot side it refuses to accept new incoming Bluetooth connections) forced us into utilizing a polling method which, in addition to unreliable latency, only permitted us to sample at between 6 and 7Hz. Executing a planned path is the basis for determining whether or not the movement data gathered and the map built upon this data is useful for improving the cleaning movements of the Roomba. If the Roomba can be directed from room to room using a map of its own construction then further improvements to the Roomba could direct a learning algorithm to find rooms and then travel to them more directly during the cleaning activities. Thus far, the simple mapping efforts selected in this experiment have proven too error prone to sufficiently contribute planning to improve upon the subsumption architecture of the Roomba.

References

[1] DIEGO RODRIGUEZ-LOSADA, F. M. A. J. R. G. Local map fusion for real-time indoor simultaneous localization and mapping. *Journal of Field Robotics 23*, 5 (2006), 291–309. 10.1002/rob.20120.

[2] KIM, S., KIM, K., AND KIM, T.-H. Human-aided cleaning algorithm for low-cost robot architecture. In *Human-Computer Interaction. HCI Intelligent Multimodal Interaction Environments*. 2007, pp. 366–375. 10.1007/978-3-540-73110-8$_3$9.

[3] WIKIPEDIA. Roomba — wikipedia, the free encyclopedia, 2008. [Online; accessed 11-December-2008].