TopBot: Automated Network Topology Detection With a Mobile Robot

Paul Blaer and Peter K. Allen
Department of Computer Science, Columbia University, New York, NY 10027
{*pblaer,allen*}*@cs.columbia.edu*

***Abstract*— We have demonstrated that a properly-equipped mobile robot can easily construct a detailed map of the wireless coverage of an urban environment. The Autonomous Vehicle for Exploration and Navigation of Urban Environments (AVENUE) mobile robot was successfully used to generate such maps in both manual and autonomous modes of operation. The resulting database contained a wealth of information for many different positions in the region, with a list of all access points viewable from each location together with a quality measure (the signal-to-noise ratio) of every detected signal. At a later time, the AVENUE system effectively used the data in this map to determine the approximate position of the robot as it traveled through the urban area.**[*]

I. Introduction

In the process of exploring urban environments, a mobile robot passes through areas rich in wireless networks. Signals from the networks' access points can be collected and analyzed by the robot, with the acquired information incorporated into the robot's navigational system. During an initial exploratory phase, the robot can construct a detailed signal map of the geographical region. At a later time, the robot can use this map to assist in localization and navigation. The map can also identify deficiencies in the signal coverage of the area and help in the design of a better network infrastructure. The Autonomous Vehicle for Exploration and Navigation of Urban Environments, *AVENUE* [6], has been adapted to perform these wireless mapping tasks.

AVENUE is a mobile robot (see Fig. 1) whose overall goal is to automate the site modeling process which includes building geometrically accurate and photometrically correct models of complex outdoor urban environments. In particular, the AVENUE system can autonomously model a targeted building. The system plans a path to a desired viewpoint, navigates the mobile robot to that viewpoint, acquires images and three-dimensional range scans of the building, and then plans for the next viewpoint. The system then fuses all of the collected data into an accurate, texture-mapped, three-dimensional model of the targeted building.

In this paper, we describe how the AVENUE system has been extended so that it can autonomously map an outdoor region by utilizing signals from a wireless network. With the addition of new software modules to monitor and analyze wireless transmissions used in conjunction with the existing path planning and navigation software, AVENUE has been able to map the coverage of the numerous access points installed on the Columbia University campus. It has then been able to use this map for subsequent localization and navigation purposes, which is especially useful at times when GPS and odometry are not available or reliable.

Our paper is organized as follows. In the next section, we indicate previous work that is related to our project. We then describe in section III our equipment and the environment in which we worked. In section IV, we detail the process of building the wireless-signal maps of the northern part of Columbia's campus and we present a representative map of the locations at which wireless signals were measured. Some statistics associated with the viewable access points are also given. We then present and discuss in section V the implementation of one of the uses of the signal map, localization of the mobile robot. In the concluding remarks of section VI, we summarize our results and discuss additional possible uses of the signal map for the AVENUE and other projects.

II. Related Work

The work presented in this paper is part of the AVENUE project for automated modeling of an outdoor urban environment by a mobile robot ([1], [6]).

The use of existing 802.11b wireless network signals as a means of locating a user was originally presented in Microsoft Research's RADAR project [2]. The Microsoft group collected the signal data manually in an indoor environment and then used this information for estimating the position of a user at a later time. Other groups have also made use of manually-obtained 802.11b signals for indoor localization [8]. We have extended the work of these groups by having our mobile robot autonomously construct the database, while covering a much larger outdoor urban environment.

There have also been a number of systems [9] based on the characteristics of cellular signals and designed for geolocating cellular telephone users in outdoor environments. In addition, there have been attempts to use RF

[*]This work was supported in part by NSF grant ANI-00-99184.

based networks, as in the Daedalus project [7], to localize a user in an outdoor area.

III. The Platform and The Environment

Our mobile robot, AVENUE, has as its base unit the ATRV-2 model manufactured by Real World Interfaces, now part of iRobot, (see Fig. 1). The base unit has an onboard computer, odometry from wheel encoders, and a set of sonar units located around the perimeter of the robot. In addition to these base features, we have added additional sensors including a differential GPS unit, a laser range scanner, a camera mounted on a pan-tilt unit, an omnidirectional camera, a digital compass, and two 802.11b wireless network cards.

Communication with the networks' base stations is accomplished through an omnidirectional antenna which is mounted on the highest point of the robot and which is connected to the pcmcia wireless network card in the onboard computer. Software located on the robot's computer polls this wireless card and returns a list of access points that are in range together with a quality measure of the signals received from each base station. We use signal-to-noise ratios as our quality indicators. At the same time a wireless measurement is taken, we take readings from the robot's GPS receiver and odometry in order to tag the measurement with an accurate location. All of this information is then stored in a database and, at the same time, transmitted to the robot's operator.

The environment in which we conducted the experiments for our project was the Morningside Heights Campus of Columbia University (see Fig. 1). There are numerous wireless base stations installed throughout the campus for general use by the university community. This extensive wireless network provided an excellent setting in which to test our mapping system.

IV. Constructing the Map

A. *Building the Detailed Wireless Network Map*

As the AVENUE robot travels through the campus environment, a program running on the onboard computer accesses the robot's primary wireless Ethernet card and returns a vector of information. That information includes a list of all access points from which the robot is receiving any kind of signal. Each of these access points has an identifying hardware address which we use as a key into the data. We also tag each access point with the signal-to-noise ratio which represents the quality of the signal from that particular base station. (The base station with the best signal-to-noise ratio is generally chosen by the hardware's drivers as the one with which to communicate.) We also record which wireless access point is dominant (with the largest signal-to-noise ratio) for a particular location. Later, we reorganize all of our data into a set of arrays in which each array represents measurements for which a single access point is dominant. This helps to prune subsequent searches of the data.

Once the robot has recorded the information about the access points currently in sight, it proceeds to tag them with the robot's exact position at which these readings were taken. This is done by querying the onboard navigation software, which fuses together a number of inputs in order to determine as accurately as possible a location for the robot within the coordinate frame of the campus. The coordinates can be easily transformed into latitude and longitude; however, it is more convenient to use local coordinates. The inputs include those from an onboard GPS receiver, which is augmented by a differential receiver mounted on one of the tallest buildings on campus. Under perfect conditions, differential GPS will provide centimeter accuracy. However, under actual operating conditions, GPS can be much less accurate, depending on such conditions as the number of buildings in the way, satellite position, and time of day. The GPS input is then fused with information from the robot's onboard odometry, which is based on wheel encoders situated on each of the robot's motors. Finally, a precise location can also be calculated by matching a camera image of buildings close to the robot with the images in a previously collected database [5]. The combination of these three localization inputs (GPS, odometry, and fine-grained vision) provides a best estimate of the robot's position. This estimate, along with a time stamp, is used to tag a single reading from the robot's wireless card. The vector of these data is then stored in an internal data structure as an array.

The robot gathers and records all of the information described above at a rate of 1 location's set of readings per second. As the robot drives through the region to be measured, it automatically builds up the map. The information is stored and can be transmitted back to the operator's laptop where it can be visually displayed and then overlaid with a diagram of the region being explored.

B. *Autonomous Navigation of the Robot*

Our robot is currently equipped with a navigation system that includes an existing two-dimensional precision diagram of the operating environment. In our current project, this is the northern part of the Columbia University campus. The diagram indicates all of the free space in which the robot can safely operate. The robot also has a built-in path planner for computing safe paths between any two arbitrary points on the diagram. To collect data for the signal map, the robot follows the following procedure. First, it marks out a grid of points which are evenly spaced every 5 feet throughout the free space in the diagram. Then, from its starting point, it chooses the nearest point on the grid and plans an optimal safe path to it. Using the same localization software that was used

Fig. 1. The ATRV-2 Based AVENUE Mobile Robot (left), the outdoor campus environment as seen from above (center) and in outline form (right).

to tag the position of each wireless network reading, the robot navigates itself to that point. All along the path, measurements of the wireless network are continually taken at the rate of one location's readings per second. After reaching the intended grid point, the robot then plans a new path to the next closest grid point and executes that path. In this manner, data is obtained from a relatively dense set of points along the robot's entire path connecting all of the grid points in the region's free space.

To determine paths along which the robot can safely move through this environment, we use an approach based on the generalized Voronoi diagram of the two-dimensional diagram of buildings and obstacles in our operating environment. Once this diagram has been constructed, we can search it to find paths that pass, with maximal clearance, around the obstacles. The buildings and obstacles in the map are polygonal. To find the generalized Voronoi diagram for this collection of polygons, one can either compute the diagram exactly or use an approximation based on the simpler problem of computing the Voronoi diagram for a set of discrete points. We use the latter method. First, we approximate the boundaries of the polygonal obstacles with the large number of points that result from subdividing each side of the original polygon into small segments. Second, we compute the Voronoi diagram for this collection of approximating points using Fortune's sweepline algorithm [4]. Once this complicated Voronoi diagram is constructed, we then eliminate those Voronoi edges which have one or both endpoints lying inside any of the obstacles. The remaining Voronoi edges form a good approximation of the generalized Voronoi diagram for the original obstacles in the map. To navigate the robot from one point to another, we take the starting point and the destination point of the robot and compute the closest vertices of the Voronoi diagram to each of them. We then use these vertices on the diagram itself to search for a path using Dijkstra's algorithm. The robot then takes a straight line path from its starting point to the closest vertex and then follows the path. Finally, it takes a straight line path from the finishing vertex to the actual destination. When doing this, we have to check to be sure that the straight line path between the robot's actual location and the vertex on the Voronoi diagram does not pass through any obstacles, and if it does, we use the second closest vertex and so on until we find one with an acceptable path. In most cases, the first vertex works without any problem.

C. *Experimental Results*

We actually collected data for the network map in two different experiments, one in simple manual mode and the other in the autonomous mode described in the preceding subsection. Both experiments were conducted in similar environments and yielded detailed maps of the network's signals.

The first experiment was done in manual mode and did not use any path planning software. Instead, the operator used a joystick to dictate to the robot an appropriate zig zag path throughout the northern half of campus. As the robot traveled, its wireless monitoring system recorded measurements once every second. The manually constructed map covered most of the free space in which the robot could operate and was based on approximately one hour of data-taking from slightly more than 4,500 locations evenly distributed throughout the region. In Fig. 2, we depict a region of the entire map which has about half of the total number of data point locations. For reference purposes, we have divided the region into six subregions, each of which is approximately 220 feet by 80 feet. Some statistics about the viewable access points for

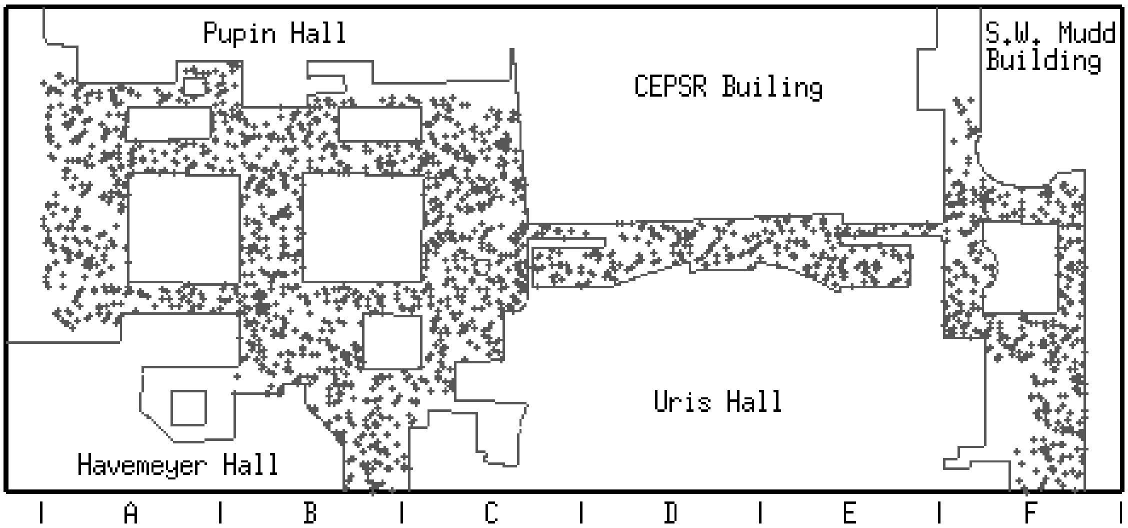

Fig. 2. A map of the locations at which wireless signals were measured in a region of the Columbia campus. The subregions A, B, C, D, E, and F extend in the north-south direction for approximately 220 feet and are each about 80 feet wide.

Subregion:	A	B	C	D	E	F
# of locations:	379	581	567	131	163	448
% of locations with:						
$N = 1$	96%	65%	15%	0%	3%	49%
$N = 2$	3%	25%	56%	3%	1%	48%
$N = 3$	1%	9%	25%	91%	64%	3%
$N \geq 4$	0%	0%	4%	6%	32%	0%
% of locations with:						
$S =$ very high	0%	2%	66%	72%	82%	42%
$S =$ high	3%	72%	25%	16%	13%	3%
$S =$ medium	36%	16%	5%	8%	3%	10%
$S =$ low	61%	10%	4%	4%	2%	46%

TABLE I

STATISTICS FOR EACH SUBREGION OF THE MAP IN FIG. 2. THE TABLE INDICATES THE NUMBER OF DATA POINT LOCATIONS IN EACH SUBREGION, THE PERCENTAGE OF THESE LOCATIONS WITH PRECISELY N VIEWABLE ACCESS POINTS, AND THE PERCENTAGE OF THESE LOCATIONS FOR WHICH THE DOMINANT ACCESS POINT HAS STRENGTH S. (THE FOUR DIFFERENT STRENGTH CATEGORIES REFER TO THE SIGNAL-TO-NOISE RATIO OF WHICHEVER ACCESS POINT IS DOMINANT.)

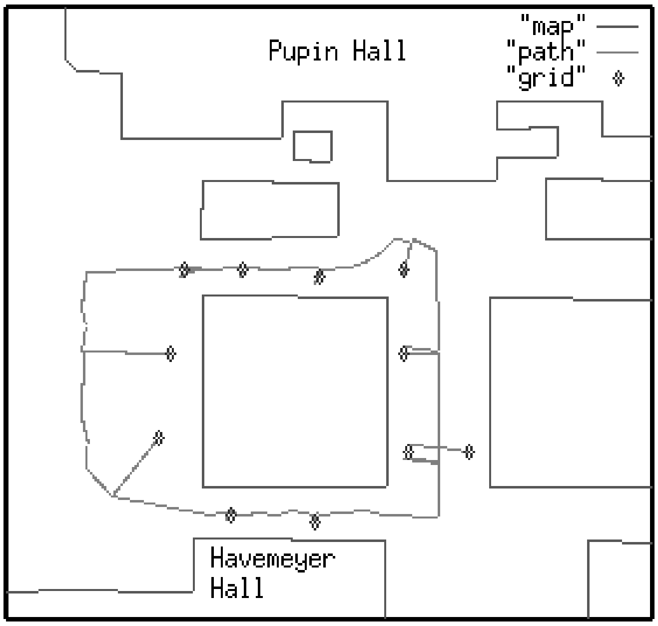

Fig. 3. The map shows the grid of 11 points that the robot used to survey autonomously a small area of campus. The paths followed between these points is also shown.

each of these subregions are given in table I. One should note that subregions C, D, and E have many locations for which the dominant access point has a very high signal-to-noise ratio. One should also note that these same three subregions have a large number of locations which receive discernible signals simultaneously from three or more access points. Determining the location of the robot at later times from signal measurements alone proved to be most successful in these three particular subregions.

The second experiment was done in autonomous mode and made use of our path planning and navigation software. We now chose to work in a small subregion of the area explored in the first experiment. In constructing the signal map of the second experiment, the robot chose a grid of 11 points (see Fig. 3) which were evenly spaced approximately every 5 feet throughout the free space. The robot then planned its optimal safe path connecting these points. These grid points were discerned in the resulting map as regions with a higher density of readings, because the robot slowed down as it approached each of its target grid points. The actual paths followed by the robot may not seem like the most obvious ones, because the robot is traveling along the edges of the Voronoi diagram which has been computed for the nearby obstacles. These paths result from the robot's attempt to stay as far away from all obstacles as is possible.

V. Localization

A. Localization Method

After a comprehensive signal map of a region has been constructed, it can be used later to determine the location of a mobile robot when GPS and odometry are not available. In the mapping stage of our project, the robot made a detailed set of measurements of the wireless signal strengths at many different points throughout the environment and stored all of these data. When the robot is in this region at a later time, it can take readings of the wireless signals available, compare these readings with its stored database of signals, and thereby determine its approximate location.

We have applied this localization method to the AVENUE robot. To determine our robot's unknown location somewhere in the northern part of the Columbia campus, we took a reading from the robot's wireless card to obtain a vector of visible access points and their corresponding signal strengths. We then compared this vector to the vectors in our large database and tried to find the best match. To minimize the amount of search time, we made the initial assumption that the current dominant access point (the one with which the wireless card was communicating at that moment) was the same as the dominant access point of the most closely matching vector in the database. This assumption substantially reduced the number of vectors needed for comparison.

To compare vectors in a meaningful way and decide which vector in the database was closest to the vector from an unknown location, we needed to define a suitable metric. In our work, we used a metric similar to the one used in the Microsoft RADAR project. The metric distance between two vectors was simply taken as the sum of the absolute values of the differences between signal strengths of corresponding access points. If there were access points that appeared in one vector but not in the other, we assumed that the other vector had that access point in its list but with a strength of zero. We stepped through each of the database vectors that had the same dominant access point as the vector from the unknown location and calculated the metric distance between each pair of database-unknown vectors. We then found the minimum distance and identified in this database subset the nearest neighbor of the unknown. If this minimum distance was within a certain specified range, we concluded that the location of the nearest neighbor gave an estimate of the robot's position. If this minimum distance was outside the specified range, we considered the similarity between this nearest neighbor and the unknown not to be sufficiently good. We then stepped through the entire database and computed all metric differences in order to find the global nearest neighbor and used that location as the estimate for the robot's position.

B. Results of Localization

To test our localization algorithm, we drove the robot through the northern part of the campus manually for a second time. Now, instead of building the signal map, we took single samples from the wireless card at various positions regarded as unknown. To check our final results, we also recorded the robot's actual location at each of these positions. We then applied our localization algorithm using the previously-constructed signal map to find the best database matches for the single samples. The signal map constructed in manual mode was actually used for this test because it spanned a much larger region of campus than did the autonomous-mode map.

Our resulting estimates for the robot's location varied in accuracy depending on the number of access points that were visible in a given location. In regions with three or more access points in view (such as subregions C, D, and E in Fig. 2), the localization was generally reliable enough to estimate the robot's position to within 25 feet of its actual position. This position estimate was necessarily a position at which we had previously recorded a database vector; however, this was not a significant drawback because of the high density of database locations. In regions with fewer than 3 access points visible, localization was much less reliable and could only identify the general region in which the robot was located by using the database region with the same dominant access point.

VI. Summary and Future Work

We have demonstrated that a properly-equipped mobile robot can easily construct a detailed map of the wireless coverage of an urban environment. The AVENUE mobile robot has successfully been used to generate such maps in both manual and autonomous modes of operation. The resulting database contains a wealth of information for many different positions in the region, with a list of all access points viewable from each location together with a quality measure of every detected signal.

This signal map was subsequently used by the AVENUE robot to determine its location at later times. This method of localization had reasonable accuracy only when many access points were viewable from a given location. However, even in areas in which only one or two wireless access points were visible, we were still able to give a coarse estimate of the robot's position by using knowledge of the dominant access point. We are planning to combine these signal map estimates with vision techniques in the AVENUE robot ([3], [5]) in order to obtain an integrated system for precise localization.

We are also planning to incorporate these wireless signal maps into the path planning software of the AVENUE system. Because the mobile robot often needs to communicate with distant computers situated in the laboratory or in the field, it is important to maximize the wireless coverage throughout the robot's entire path. We will therefore augment our Voronoi path optimization algorithm to take into account available network service as well as the previously-used clearance around obstacles and minimal path length.

The signal maps obtained by an autonomous robot could also be very useful in the design and construction of large area wireless networks. With an initial arrangement of trial base stations, the robot could quickly map the wireless coverage of the area and determine those regions which had inadequate signal quality. This immediate feedback would allow designers to reposition and adjust the transmitters appropriately for maximal coverage.

We have shown that wireless signal maps are easy to construct and have a variety of useful applications. These maps should therefore become an important component of all future mobile robot systems.

VII. REFERENCES

[1] P. K. Allen, I. Stamos, A. Gueorguiev, E. Gold, and P. Blaer. Avenue: Automated site modeling in urban environments. In *3DIM, Quebec City*, pages 357–364, May 2001.

[2] P. Bahl and V. N. Padmanabhan. RADAR: An in-building RF-based user location and tracking system. In *INFOCOM (2)*, pages 775–784, 2000.

[3] P. Blaer and P. Allen. Topological mobile robot localization using fast vision techniques. In *IEEE ICRA*, pages 1031–1036, May 2002.

[4] S. J. Fortune. A sweepline algorithm for Voronoi diagrams. In *Algorithmica 2 (1987)*, pages 153–174.

[5] A. Georgiev and P. K. Allen. Vision for mobile robot localization in urban environments. In *IEEE IROS*, October 2002.

[6] A. Gueorguiev, P. K. Allen, E. Gold, and P. Blaer. Design, architecture and control of a mobile site modeling robot. In *IEEE ICRA*, pages 3266–3271, April 24-28 2000.

[7] T. D. Hodes, R. H. Katz, E. S. Schreiber, and L. Rowe. Composable ad hoc mobile services for universal interaction. In *MOBICOM*, pages 1–12, September 1997.

[8] A. M. Ladd, K. Bekris, A. Rudys, G. Marceau, L. E. Kavraki, and D. S. Wallach. Robotics-based location sensing using wireless ethernet. In *MOBICOM*, 2002.

[9] S. Tekinay. Wireless geolocation systems and services. In *IEEE Communications Magazine*, April 1998.