Embedding Robots Into the Internet

Gaurav S. Sukhatme and Maja J. Matarić
gaurav,mataric@cs.usc.edu
Robotics Research Laboratory
Computer Science Department
University of Southern California
Los Angeles, CA 90089-0781

February 18, 2000

Abstract

With the explosive growth of embedded computing hardware, it is possible to conceive many new networked robotic applications for diverse domains ranging from urban search and rescue to house cleaning. Designing reliable software for such systems is a challenging problem. However, Internet communication can facilitate robotics by reducing uncertainty,, as well as providing direct user input and assistance, while robotics can facilitate communication by providing physical mobility at a distance. In this article we overview methods for control and coordination of embedded mobile systems (robots) which interact with other computers on a wireless network situated in human environments.

1 Introduction

Ubiquitous embedded computing [12] is here to stay. Information appliances, laptops, palmtops, and wearable computers are examples of the first wave of this new era. Two factors have contributed to the phenomenal increase in the number of computers in our environment: Moore's law [12]and improved network connectivity. It is now increasingly accepted that appliances of the future (whether they be for the office, home, or school) will be based around small embedded computers with limited (but growing) functionality, and a network connection. These devices all share one other often ignored common characteristic: they are not physically mobile on their own. Instead, they depend on human users for their placement and transport. In this article we focus on the class of embedded systems with autonomous mobility capabilities, better known as *robots*. The introduction of these devices into environments that are primarily built for people raises a number of very interesting and challenging questions:

- What is the best way to control and coordinate ubiquitous robots?
- How should such robots be used? What services can they provide?
- How does the software for such robots differ from software for other embedded systems?
- What are the safety and human factors issues involved?

We use the term *embedding* to reflect the fact that the robots are communicating over a wireless network. Although itself complex, such communication has the capability to provide richer interaction between robots, as well as between robots and other network resources. This has strong implications for task-sharing among robots, human-robot interaction, and on-the-fly re-programmability and adaptation.

At the University of Southern California's (USC) Robotics Research Laboratories we are involved in an NSF-funded research study[1] in collaboration with the USC Computer Networks and Distributed Systems

[1] The project, SCOWR (Scalable Coordination of Wireless Robots), is found at http://netweb.usc.edu/scowr

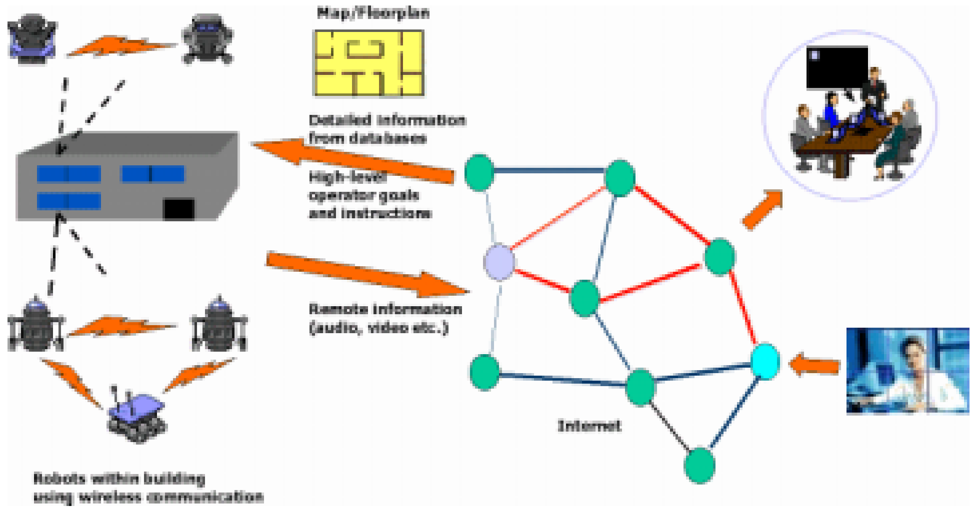

Figure 1: Example scenario: robots on a wireless network connected to the Internet, receiving information and instructions and providing in situ data.

Research Laboratory to address some of these issues. We are focusing on studying scalable algorithms for the distributed control and coordination of wireless nodes which may be robotic, i.e., autonomously mobile. Our goals are to address networking and robotics issues pertinent to the problem domain, while recognizing that the wireless network connects conventional computers, wearables, portables, immobile sensors, and robots.

We address here the issues and ideas related to the control and coordination of robots as entities embedded within a wireless network connected to the Internet. A stylized depiction of such a scenario is shown in Figure 1, in which several robots are exploring the interior of a building. The robots are on a local wireless network which is connected to the Internet backbone, which allows them to receive information and instructions, as well as send data back. For example, if the building being explored is partially collapsed due to an earthquake, a specialist may want to direct the robots to find trapped people and send back their vital signs. Other remote users might exploit audio and video information the robots can provide, and communicate with people in the building. Furthermore, connectivity to the Internet allows the robots to access information repositories on the web, such as building maps, to aid them in exploration.

The traditional style of doing robotics (largely before wireless networks) was an off-line programming process in which robot controllers were developed on a desk-top computer and downloaded to the robot's microcontroller, usually via a serial link. The new paradigm, enabled by wireless communication, is to develop controllers on the robot itself, since its computer is remotely accessible over the network at all times. In addition, by using wireless communication, robot controllers can now take advantage of a variety of networked resources, which may be physically attached to another robot, some immobile computer, or an on-line data-base.

We illustrate the major issues of embedding robots into the Internet in order to achieve a dual synergy: communication can facilitate robotics by *reducing uncertainty*, and robotics can facilitate communication by *providing physical mobility*. We discuss the key research problems associated with uncertainty and mobility, and propose some ideas towards their solution, and briefly touch upon related work in those areas.

2 Behavior-Based Robotics: A Methodology for Robot Controller Design

Behavior-based robotics is currently the most active and popular approach to mobile robot control in the multi-robot domain [7, 1]. The approach is based on the notion of *behavior*, a unifying representation for control, reasoning, and learning. Behaviors are real-time processes that take inputs from sensors and/or other behaviors and send outputs to the robot's actuators and/or other behaviors. The controller, then, is a network of such communicating, concurrently executed behaviors. This metaphor has excellent real-time and scaling properties. When applied to the multi-robot scenario, it can eliminate the distinction between a collection of processes on one robot and a collection of processes running on multiple robots across a wireless network. In both cases the system as a whole is a collection of communicating behaviors. Much of our work is applying just this metaphor.

The problem of behavior coordination within a robot controller (as well as between multiple robot controllers) is an active area of research. Various approaches have been effectively applied, ranging from fuzzy control to decision theory to neural network learning. Behaviors interact not only within a robot system but also through the environment, thus allowing the designer to exploit emergent properties. Several methods for principled behavior design and coordination have been proposed [1]. We introduced the concept of *basis behaviors*, a small set of necessary and sufficient behaviors that could be composed (by sequencing or fusion), as a means of handling controller complexity and simplifying design. This principle was demonstrated on large groups of robots performing exploration, described in Section 5 below, as well as other behaviors, including coordinated movement in the form of aggregation, dispersion, flocking, etc. We are currently expanding this principle to the more general problem of producing reusable robotics software[2] which can be effectively composed at run-time from a basis set in order to execute complex tasks. In that context, we are developing strategies for resource transport using a large robot group [11].

For convenience in experimentation as well as a teaching and research tool, we use a multi-robot simulator, Arena[3]. Arena simulates the movement and sensing of many small mobile robots in a 2D world. All sensing and actuation is modeled with low fidelity to achieve high update rates. Simple noise models for the robots' sensors and effectors are also provided. While our goal is to always validate our methods on real robots, a well-designed, simple simulator enables fast, incremental construction of controllers that run well in the presence of large perturbations injected into the simulated world. Furthermore, Arena uses a TCP/IP socket server that provides an identical controller/robot interface to that of our real Pioneer robots. This arrangement facilitates rapid transfer of controllers developed in simulation to the physical robots.

3 Internet Robots at USC

The major goals of our efforts to embed physical autonomous robots into the Internet are to develop, test, and characterize algorithms for scalable, application-driven, wireless network services using a heterogeneous collection of communicating mobile nodes. Some of these nodes will be autonomous (i.e., robots) in that their movements will not be human-controlled. The others will be portable, dependent on humans for transportation. While the focus of our work is on the mobile nodes, we include immobile computers on the network as well. We emphasize that most (though not all) of the mobile nodes will have modest sensing, computational, and communication resources.

To give a concrete example, consider an earthquake scenario where people are trapped within a partially collapsed building. The task for the rescue team is to quickly identify the areas where people are likely to be, so that heavy machinery may be introduced to assist them effectively. One solution to the problem is a group of small autonomous robots introduced into the building at various entry points. These robots communicate with each other and with the outside world through multihop wireless radio and explore the building, trying to detect the presence of people. Whenever one or more robots detects a person, the location and perhaps an image of the person are sent back. In more sophisticated versions, robots might also be used for bidirectional audio communication between people trapped in the building and the rescuers outside, or for delivering medicine and supplies to the victims; the possibilities are vast. It is also easy to imagine ubiquitous robots in everyday life, for applications ranging from mail delivery within buildings to cleaning and security.

Such small ubiquitous robots need a number of basic capabilities that will make them autonomous and generally useful. We focus here on three such basic capabilities: localization, exploration, and mapping. The first, localization, refers to the ability of the robots to use their sensors and wireless communication

[2] The project is supported under the DARPA Mobile Autonomous Robot Software (MARS) program and can be found at http://www-robotics.usc.edu/projects/mars

[3] Arena is available at ftp://deckard.usc.edu/pub/arena

to compute their position over time. The second, exploration, allows the robots to search and cover an area, perhaps with some guidelines from a user. The third, mapping, is the ability to collectively create a representation of the environment or to augment a representation provided by a user.

The interplay of these capabilities results in robots that are capable of functioning autonomously in relatively unstructured environments. As discussed in the next sections, we are pursuing concurrent development of these capabilities, and focusing on the following key guiding principles:

- In order to be robust, we investigate multi-robot solutions wherever possible, in particular to the key problems of collective mapping, exploration, and localization. The intuition is that, with careful design, multiple robots provide redundancy and hence fault tolerance. A well designed multi-robot solution also reduces global uncertainty, even if each individual robot is still a relatively noisy source of data.
- In order to be scalable, we investigate distributed, bottom-up strategies. We emphasize strategies that scale to large numbers of robots, thus favoring local, decentralized ones over global, centralized alternatives. The goal is to endow individuals with independent capabilities and minimal communication needs (each robots needs to communicate only with near-by neighbors) and seek globally coherent and efficient behavior.
- We treat the wireless network as a key resource in distributed robotics, and look for ways to exploit it without adopting many restrictive or simplifying assumptions.

Effective autonomous mobility and interaction with the physical world require a level of robustness that enables the robot to handle the inevitable uncertainties in sensing, action, communication, and control. Robot sensors provide incomplete, noisy information about the environment, and actuators are rarely precise. Most of this uncertainty is difficult to even characterize analytically. Finally, the behavior of other robots or humans in the environment is far from predictable. The central challenge in robotics is to perform robustly in the face of these uncertainties. Embedding robots into the Internet can potentially simplify some of these fundamental robotics problems.

In the sections that follow, we describe our work in embedding behavior-based robots into the Internet, in the context of the tree basic robotics capabilities we outlined above.

4 Robot Localization Algorithms

4.1 Using Inertial Sensing and Filtering

Techniques for accurately estimating the position (formally meaning position and orientation) of a robot lend themselves to a natural partitioning. One class of techniques relies on using on-board inertial sensing and odometry to keep track of changes in position. Integrating such small changes over time leads to an updated position estimate [2]. The second class of techniques uses some global sensing method (perhaps a map, or the Global Positioning System (GPS), if outdoors) to update the position estimate. The former is prone to drifting and depends on knowledge of the initial position, while the latter is dependent on global information often not easily obtained (e.g., no GPS signal if indoors). The two approaches have been combined with varying success. Both have been extensively studied in robotics, but given the uncertain nature of sensor measurements the problem of accurate position estimation remains difficult.

4.2 Using Radio Signal Strength and Range

We are using two approaches to investigating the use of radio as a basis for localization. The first is a coarse-grained approach in which multiple transmitters are placed in the environment. Each robot is equipped with a receiver that can distinguish the signature of the transmitters. In any location within the environment, the receiver can detect some subset of transmitters. The set of locations where the same transmitters can be detected form an equivalence class. An environment with N transmitters is thus divided into at most 2^N equivalence classes. Although straightforward, this approach is coarse and may not yield the desired granularity in realistic environments. An even more serious problem lies in the possibility that a given equivalence class may not be spatially connected, resulting in "holes" often encountered in wireless communication in cluttered environments.

The second approach we are pursuing is a fine-grained version of the first. Again each robot in the environment is equipped with a receiver and there are N transmitters in the environment. We endow each robot with the ability to not only distinguish transmission signatures, but to also detect signal strength. The goal is to acquire the mapping from signal strength to range for each transmitter and to then triangulate position based on the various range estimates available at each instant. The triangulation procedure uses a noise model of the range output to provide an estimate of position and an estimate of uncertainty. This approach, like the first, could be used in conjunction with inertial sensing and odometry to provide a better position estimate. One of the advantages of both approaches is that they are not specifically tailored for robotics applications; they can be equally well used to localize a person carrying a computer.

5 Robot Exploration Algorithms

Robot exploration approaches have been studied in different contexts. A common abstraction is the so-called "art gallery" analogy, where the robot's goal is to move from one position to another so as to maximize visual coverage of its surroundings, as one might try to do in a gallery. A complementary set of approaches addresses the pursuit-evasion problem [6] in which a robot tries to move so as to evade observation or capture by a group of moving trackers.

Several approaches to exploration have been developed, addressing the related goals of i) searching for a specific location/object, ii) space coverage, and iii) maximizing some measure of novelty. Task-specific heuristics can be applied to simplify the exploration as well make it more robust. For example, in recent work we chose an exploration strategy for indoor environment mapping that forces a robot to explore corridors to their end (depth first) [4]. Door openings on the way are recorded but not explored, with the goal of quickly generating a map of the overall structure of the building and only subsequently filling in the details. This strategy is heuristic and it is easy to imagine topologies in which it will be less than desirable.

Exploring space efficiently is a challenging problem, because of the multiple objectives involved. Detection of new and interesting features leads the robot into unexplored spaces, while staying localized constrains its movements to small feature-rich areas. As we discuss in the next section, the problem is even harder (but the exploration could be made faster) when multiple robots are involved. This is one of the areas where embedding robots into a communication network can significantly facilitate task performance.

5.1 Group Exploration for Maintaining Connectivity

While exploration strategies for single robots have been studied extensively, less is known about the problem in the multi-robot case. How should multiple robots coordinate themselves to explore a given environment so that they provide complete and efficient space coverage? When posed as global, top-down optimization problem, this question is extremely hard to answer in all but the most stylized, simple domains. We choose instead to explore this and other problems bottom up, in a decentralized fashion. Our past work has demonstrated effective distributed exploration for groups of up to 13 robots [8], and has applied different variations of behavior-based controllers (including homogeneous, heterogeneous, dominance hierarchies, and territorial solutions) [5]. We have developed methods for on-line interference estimation and minimization [5] as well as several approaches to adaptive multi-robot coordination, which use simple communication of sensory input and/or received feedback [9] to effectively improve and over time optimize group-level performance.

We are currently designing behavior-based robot controllers for a variety of scenarios. Two are particularly relevant to communication-based exploration. The objective of the first is for a group of robots to "fan out" from a common starting location and explore an area in search of some goal. When the goal is detected, an image is sent back to the starting location over the network. We are using a greedy exploration strategy in which each robot tries to maximize the amount of space it explores as long as it is within communication range with the other robots. When a robot goes out of range, it stops, stores its location, and backtracks until it re-establishes communication with at least one of the others. It then shares its stored location with this robot, and stores the second robot's location. These two locations, together, define a rendezvous pair,

Figure 2: A subset of the robot testbeds used in our research.

which is later used by both robots if they need to establish communication again. If the robots fall out of communication range again, they backtrack to the last stored rendezvous location and wait for contact.

The second scenario we are experimenting with assumes that a wave of robots has arranged itself in some pattern over the environment. These robots are most likely separated into disjoint groups that cannot directly communicate. Our “communication hole-filling” algorithm then explores the area by using a second wave of robots which detects communication voids and fills them either by dropping radio tags or stationing robots at “bridge” locations.

These exploration and space coverage techniques enable the basic capabilities that can be used for a variety of applications, including mapping.

6 Robot Mapping Algorithms

Robot mapping approaches principally fall into two categories: i) those that produce metric maps of the environment, and ii) those that produce topological ones. Significant research has been done in both categories, although more so in the former. Currently the best example of metric mapping, using laser rangefinders to produce precise floorplans, has been used to map the interior of a museum in order to equip it with a robotic tour guide. Experiences with the deployment and web-based control of the robotic tour-guide are found in [3]. The map was built by composing successive laser scans into a grid-based representation. The approach decides which cells of the grid are occupied and which are empty, and incrementally improves the confidence in each grid state with successive scans. However, in this approach mapping is made significantly easier if localization is perfect, since in truly dynamic environments accurate position estimates are needed to match successive scans. Similarly, localization is made significantly easier if a map is available. Unfortunately, simultaneous localization and mapping remain a difficult problem for autonomous robots.

The second class of approaches produces topological maps in which the significant or salient features in the environment (doors, windows, corners etc.), so-called landmarks, correspond to the nodes of a graph. Whenever such a feature is detected, the robot decides whether it has seen it before (in which case it may perhaps improve its position estimate) or whether the feature is new (in which case it can add it to its map with the appropriate links to the other already mapped features). Our early approach [7] to topological mapping using a graph representation introduced the notion of representation into behavior-based systems by developing an integrated system that did not distinguish between the control program and the map, embedding both into concurrent, communicating behaviors. An example of a recent approach to learning a topological map of an office building in the presence of odometric uncertainty is presented in [10].

Our current research focuses on multi-robot topological mapping. A group of robots (shown in Figure 2) concurrently builds individual topological maps of the environment with no *a priori* information about one another's respective locations. Each robot tracks its own position in a private reference frame; this information is communicated to other robots and a graph matching algorithm is used to combine individual maps. The matching algorithm seeks to find the transformation between the maps that maximizes “feature overlap” between the individual maps. To keep the number of candidate transformations manageable and thus keep the algorithm scalable, preprocessing heuristics are employed. The match produces a final transformation between the maps which is used to correct each robot's position estimates. The resulting map combines the features from each of the individual maps probabilistically, since the individual mapping algorithms keep track of a robot's belief in a feature once it is detected.

Mapping is a basic capability, which can be used by a robot to build, or augment, a representation of an environment, thereby helping it to stay localized. This in turn provides a basis for navigation and purposeful movement, all basic capabilities underlying various applications for ubiquitous Internet-embedded robots.

7 Summary

In this article, we presented some of the key challenges in embedding robots into the Internet and approaches to address them. We discussed the benefits of bottom-up, distributed control for this domain, and the use of behavior-based robotics for this purpose. We then focussed on three capabilities: localization, exploration, and mapping, which are important for mobile, robotic nodes on the Internet. We briefly described our ongoing projects in those areas, as well some other relevant work, which is utilizing communication to facilitate robot coordination, as well as providing physically mobile network nodes by introducing robots on the Internet.

While many interesting and difficult problems need to be solved to realize the goal of ubiquitous robots in human environments, we believe that the combination of the robotics technology with that of wireless communication, and the interaction of various types of such communicating nodes, is already a rich and promising area of research.

Acknowledgments

The authors gratefully acknowledge the USC SCOWR team for many helpful discussions. Interested readers may contact the individual team members directly, by visiting the SCOWR web page (http://netweb.usc.edu/scowr/). We thank Richard Vaughan for developing Arena. We also thank Deborah Estrin, Ramesh Govindan, and John Heidemann for putting together this special issue and inviting us to submit an article.

The described work is partially supported by the National Science Foundation under grant ANI-9979457, DARPA under contract DAAE07-98-C-L026 and grant DABT63-99-1-0015, and the Office of Naval Research under grants N0014-99-1-0162 and N00014-95-1-0759.

References

[1] Ronald C. Arkin. *Behavior-Based Robotics.* MIT Press, 1998.

[2] B. Barshan and H. F. Durrant-Whyte. Inertial navigation systems for mobile robots. *IEEE Transactions on Robotics and Automation*, 11(3):328–342, June 1995.

[3] W. Burgard, A.B. Cremers, D. Fox, D. Haehnel, G. Lakemeyer, D. Schulz, W. Steiner, and S. Thrun. Experiences with an interactive museum tour-guide robot. *Artificial Intelligence*, 114(1–2):3–55, 1999.

[4] Göksel Dedeoglu, Maja J. Matarić, and Gaurav S. Sukhatme. Incremental, online topological map building with a mobile robot. In *Proceedings of Mobile Robots XIV - SPIE 99*, pages 129–139, Boston, MA, 1999. SPIE.

[5] Dani Goldberg and Maja J Matarić. Coordinating mobile robot group behavior using a model of interaction dynamics. In *Proceedings, The Third International Conference on Autonomous Agents (Agents '99)*, Seattle, Washington, May 1999.

[6] L. J. Guibas, D. Lin J-C. Latombe, S. M. LaValle, and R. Motwani. Visibility-based pursuit-evasion in a polygonal environment. *International Journal of Computational Geometry and Applications*, To Appear.

[7] Maja J. Matarić. Integration of representation into goal-driven behavior-based robots. *IEEE Transactions on Robotics and Automation*, 8(3):304–312, 1992.

[8] Maja J Matarić. Designing and understanding adaptive group behavior. *Adaptive Behavior*, 4(1):50–81, December 1995.

[9] Maja J Matarić. Reducing locality through communication in distributed multi-agent learning. *Journal of Experimental and Theoretical Artificial Intelligence, Special issue on Learning in DAI Systems*, Ed. Gerhard Weiss, 10(3):357–369, 1998.

[10] Hagit Shatkay and Leslie Pack Kaelbling. Learning topological maps with weak local odometric information. In *Proceedings of the 15th International Joint Conference on Artificial Intelligence (IJCAI-97)*, pages 920–929, San Francisco, August 23–29 1997. Morgan Kaufmann Publishers.

[11] R. T. Vaughan, K. Stoy, G. Sukhatme, and M. J. Matarić. Whistling in the dark: Cooperative trail following in uncertain localization space. In *4th International Conference on Autononous Agents*, Barcelona Spain, June 3–7, 2000.

[12] M. Weiser. Some computer science problems in ubiquitous computing. *Communications of the ACM*, July 1993.