The History of the Mobot Museum Robot Series: An Evolutionary Study

Thomas Willeke
Mobot, Inc.
Pittsburgh PA
twilleke@cs.stanford.edu

Clay Kunz
Mobot, Inc.
Pittsburgh PA
clay@cs.stanford.edu

Illah Nourbakhsh
Carnegie Mellon University
Pittsburgh PA
illah@cs.cmu.edu

Abstract:

This paper describes a long-term project to install socially interactive, autonomous mobile robots in public spaces. We have deployed four robots over the last three years, accumulating a total operational time of about six years. We introduce the robots, then focus on the lessons learned from one deployment to the next. The evolution of the robot-human interface is of particular interest, although other aspects of the robots' operations are briefly described.

Introduction

The history of autonomous mobile robotics research has largely been a story of closely supervised, isolated experiments on platforms which do not last long beyond the end of the experiment. In January 1998, we and others started work on Chips, an autonomous robot intended to be more than an experiment. Chips would become a permanent installation and member of the museum staff at the Carnegie Museum of Natural History in Pittsburgh, PA (Nourbakhsh et al. 1999).

Shortly thereafter, Mobot, Inc. was incorporated with a charter to improve and extend the Chips technology in a series of robot installations. Following Chips, three more robots have been developed in succession; three of the four still operate every day. Together, these robots have logged more than 2,000 total days of operation in their real-world public spaces.

In striving to deploy autonomous mobile robots in a social niche, we have two high-level goals. First, the robots must be autonomous to the largest extent possible. Human supervision of a full-time social robot is unacceptable. At most, the robots should only require occasional human help, and should request that help explicitly. Even the routine trip to the battery charger should be performed autonomously.

Second, since the robots would be deployed in public, they must have sufficiently rich personalities to achieve compelling and fruitful interaction with humans in their environments. A moving object without expressive interactivity would soon be moved into the closet.

We begin by presenting a brief overview of each of the four robots. Following this we discuss the evolution of our robot design in view of the goals of autonomy and personality.

From: FLAIRS-01 Proceedings. Copyright © 2001, AAAI (www.aaai.org). All rights reserved.

Robot Overview

The four robots compared in this paper share the same operating system (RedHat Linux); the same robot platform (Nomadic Technologies XR4000); and the same programming environment (Gnu C++). The first robot, Chips, began work at the Carnegie Museum of Natural History on May 22, 1998. Chips operates exclusively in Dinosaur Hall, which contains large bone collections of *T. Rex* and other massive dinosaurs as well as ancillary exhibits focusing on topics such as paleogeology and ancient aquatic life. Chips's charter is to provide tours in Dinosaur Hall, presenting audiovisual information regarding both the large bone collections as well as the less frequented, smaller exhibits. Thus far Chips has been operating for almost 3 years, covering a total travel distance greater than 323 kilometers.

The second robot, Sweetlips, conducts tours in the Hall of North American Wildlife, also at the Carnegie Museum of Natural History (see Fig. 1). This space is composed of a number of dioramas in which preserved wildlife are shown in naturalistic settings. This portion of the museum has very low visitor traffic, so Sweetlips's charter is to both attract additional visitors to the Hall and to bring the static dioramas to life with high-quality footage of the same wildlife in their natural habitats. Sweetlips has been operating since April 1999, covering a total distance greater than 145 kilometers autonomously.

The third robot, Joe Historybot, operates in the atrium of the Heinz History Center. Its mission is to welcome visitors to this historical museum and provide both information and a tour of the atrium, which itself houses a number of significant exhibits. Joe provides historical information in an entertaining multimedia format. The robot also provides tutorials on speaking with a Pittsburgh accent and remotely triggers sound and light events associated with atrium exhibits. Joe has been operating since July 1999, covering a total distance greater than 130 kilometers autonomously.

Copyright © 2001, AAAI. All rights reserved.

end of a long cane held by the user. It does not provide balance or support to its user and so its control system is different from ours. Finally, the most similar work in assistive technology is the PAMAID project [11][11]. PAMAID is a walker that has been augmented with sensors and effectors in order to assist elderly, visually impaired users in taking independent exercise. While our device is mechanically similar to PAMAID, we are focusing on automatic adaptation to user input and inference of user goals based solely on forces applied to the walker, i.e. using the traditional rollator interface.

This research also involves problems in general mobile robotics, namely path planning [14][21] and self-localization [5][18]. Probabilistic localization methods [18] maintain multiple location hypotheses, adjusting the probability that the robot is located at any particular one as new information becomes available. Landmark recognition combined with triangulation [5], is another popular localization method. Mobile robots incorporating such technology, e.g. the 'nursebot' Flo [2], provide robotic assistance for the elderly in the form of a social and data collection agent, but do not provide physical support. Passive robots for material handling [7] or surgery [16] are similar in concept to our walker because they lack the capacity to move on their own. Instead they steer their joints to particular positions based on how the user moves the robot.

Future Work

The control systems discussed in this paper have been evaluated in only the most primitive forms using an early prototype walker that is not described here. Clinical trials are now set to begin. We will be providing various walkers, both with and without smart control systems to seniors at a senior center in Charlottesville, Virginia. Walker users will be asked to cover a basic course, where they must maneuver around obstacles, move through tight spaces and follow paths. For each type of walker (including the walker with no control system), we will measure variables such as the number of collisions, the time to complete the course and the user's feelings of satisfaction, safety and control.

References

[1] Alliance for Aging Research. 1998. Independence for Older Americans: An Investment in our Nation's Future, http://www.agingresearch.org/Resources/brochures/indep.htm.

[2] Baltus, G., Fox, D., Gemperle, F., Goetz, J., Hirsch, T. Magaritis, D., Montemerlo, M., Pineua, J., Roy, N., Schulte, J. and Thrun, S. 2000. Towards personal service robots for the elderly. http://www.cs.cmu.edu/~thrun/papers/thrun.nursebot-early.pdf

[3] Borenstein, J. and Ulrich, I. 1997. The GuideCane – A Computerized Travel Aid for the Active Guidance of Blind Pedestrians. *IEEE Int. Conf. On Robotics and Automation*: 1283-1288.

[4] Brooks, R. A. 1986. A Robust Layered Control System for a Mobile Robot. *IEEE Journal of Robotics and Automation* RA-2: 14 - 23

[5] Brill, F., Wasson, G., Ferrer, G. and Martin, W. The Effective Field of View Paradigm: Adding Representation to a Reactive System. 1998. *EAAI* 11: 189-201.

[6] Ciole R. and Trusko, B. 1999. HealthCare 2020: Challenges for the Millennium, *Health Management Technology*: 34-38.

[7] Colgate, J., Wannasuphoprasit, W. and Peshkin, M. 1996. Cobots: Robots for Collaboration with Human Operators, *Proc. Intl. Mech. Eng. Conf. And Exhib.*, DSC-Vol. 58: 433-439.

[8] Gomi, T. and Griffith, A. 1998. Developing Intelligent Wheelchairs for the Handicapped. Technology and Artificial Intelligence, eds. Mittal et. al., Springer-Verlag: 150-178.

[9] Gunderson, J. P. 2000. Adaptive Goal Prioritization by Agents in Dynamic Environments, *Proceedings of the IEEE Systems, Man, and Cybernetics*: 1944-1948

[10] Kramarow, E., Lentzner, H., Rooks R., Weeks, J. and Saydah S. 1999. Health and Aging Chartbook, Health United States, National Center for Health Statistics. http://www.cdc.gov/nchs/data/hus99cht.pdf.

[11] Lacey, G. and MacNamara, S. 2000. Context-Aware Shared Control of a Robot Mobility Aid for the Elderly Blind. *Intl. Journal of Robotics Research* 19(11): 1054-1065.

[12] Lacey, G., MacNamara, S. and Dawson-Howe, K. 1998. Personal Adaptive Mobility Aid for the Frail and Elderly Blind. Assistive Technology and Artificial Intelligence, eds. Mittal et. al. Springer-Verlag: 211-220.

[13] Levine, S., Bell, D., Jaros, L., Simpson, R., Koren, Y. and Borenstein, J. 1999. The NavChair Assistive Wheelchair Navigation System. *IEEE Trans, on Rehabilitation Engineering* 7(4): 443-451.

[14] Lozano-Perez, T. 1987. A Simple Motion-Planning Algorithm for General Robot Manipulators, *IEEE Robotics and Automation* vol. RA-3 (3): 224-238.

[15] National Center for Health Statistics. 1994. Number of Persons Using Assistive Technology by Age of Person and type of Device. http://www.cdc.gov/nchs/about/major/nhis_dis/ad292tb1.htm

[16] Schneider, O., Troccaz, J., Chavanon, O. and Blin, D. 1999. Synergistic Robotic Assistance to Cardiac Procedures, Computer Assisted Radiology and Surgery: 23-26.

[17] Statistic Bureau and Statistics Center, Ministry of Public Management, Home Affairs, Posts and Telecommunications, Government of Japan. 2001. Japan Statistical Yearbook, http://www.stat.go.jp/english/1431-02.htm

[18] Thrun, S., Fox, D. and Burgard, W. 1998. A Probabilistic Approach to Concurrent Mapping and Localization for Mobile Robots. *Autonomous Robots* 5: 253-271.

[19] Wasson, G. 1999. Design of Representation Systems for Autonomous Agents. Ph.D. Dissertation. Computer Science Department. University of Virginia.

[20] Yanko, H. 1998. Wheelesley: A Robotic Wheelchair System: Indoor Navigation and User Interface. Assistive Technology and Artificial Intelligence, eds. Mittal et. al., Springer-Verlag: 256-268.

[21] Zilberstein, S. and Russell, S. 1999. Anytime Sensing, Planning and Action: A Practical Model for Robot Control. *IJCAI-99*: 1402-1407.

where $d_{(walker,path)}$ is the length of the normal from the walker's location to that planned path and d_{max} is the tolerance for how far off the path the user can be without the control system taking corrective action. $\angle_{path}$ is the angle to the path expressed as –1, 0 or +1 meaning that the walker has to turn left, stay straight, or turn right to move closer to the path. BF is the force applied to the brakes. Finally, $\triangle CT_2$, $\triangle CT_3$ and $\triangle CT_4$ are correction times.

First, let's examine rules 7 – 10. These rules when the walker is far enough from the path that the control system will try to pull the walker back toward it. Rule 7 is used to steer the walker back toward the path for $\triangle CT_2$. Once $\triangle CT_2$ expires, if the walker is still d_{max} beyond the path, rule 8 causes the control system to apply minimum braking force for $\triangle CT_3$. This will not stop the walker, but will provide increased resistance as a signal to the user. It is hoped that this will cause the user to realize that they are straying. If the user persists (for more than $\triangle CT_3$) in moving away from the path, the control system relents and the walker moves as the user wishes. The best path to the goal from the walker's new location will now be computed (rule 9). This continues until the user indicates (through a push-button) that the goal has changed or that the walker may begin leading again. Should the user move back toward the path during any of these intervals, the walker resets its timers and allows the motion (rule 10).

Rules 11 and 12 apply when the user is within d_{max} of the path. As long as the user steers approximately parallel to the path, rule 11 causes the walker to take no action. If the user is not traveling parallel to the path, the walker gently angles them back toward it (rule 12). After the wheel again parallels the path or $\triangle CT_4$ expires, the walker releases control to the user.

To make this control system comfortable to walker users, the correction times must be long enough to make the control system's desire apparent, yet not so long that the user feels they are struggling with the device. Experiments will help us to estimate these parameters, which we initially feel will be between 1 and 5 seconds.

Discussion

A goal of this project is to give the user the same interface to our walker as they have to a conventional wheeled walker, and to provide mobility assistance. The shared control approach requires that the two control inputs, the operator and the walker control system act in a collaborative manner. While we cannot directly measure the goals of the operator, we can use sensors to infer likely goals, and to continually update the confidence in those inferences. We believe that this will not be a severe limitation because the control system can never turn in the opposite direction of the user input (it can either go in that direction or straight) and can only steer differently than user input for a limited time. Therefore, the walker's motion should conform to user expectations.

The control systems discussed in the previous sections are designed to provide safety, support and guidance to their users. In order to meet all these criteria, several design principles were developed.

1. Diverge from user input slowly

Many walker users move slowly and require frequent stops. Since they depend on the walker frame for balance and support, the walker should not make any large movements away from the user's desired input. The control system must ensure that it does not cause the front wheel to be realigned at a rapid pace.

2. Keep walker turns in direction indicated by user

Both the user and the control system work to direct the walker's motion. When the control system is active, the walker should not turn in the opposite direction of user input. In other words, if the user is indicating that they want to go left, the actual command executed by the walker should not cause the platform to turn right (only less left or more left).

3. The walker should be compliant

The walker should mostly follow the user's commands, as this is crucial for user acceptance. For the safety braking and the safety braking and steering control systems, the control system only influences the motion when obstacles or cliffs are near the user. In other words, the walker is, typically, fully user controlled. For all other situations, including path following, rules 6, 9, 10, 11 represent the control system submitting to the user's desire. This does not mean that the control system shuts down, or does not provide the usual safety features. In fact, all of the control systems fall back on their emergency braking to keep the user safe. When the control system has had to brake to avoid an obstacle or has given up trying to lead the user on a particular path, the user must disengage the brakes (via a pushbutton) or re-engage the path following (again via a pushbutton) to regain control or allow collaboration again. This lets the user select the walker's mode manually when they disagree with the control system's choices.

Related Work

Assistive technology is an active area of research. In the last several years, strides have been made in locomotive assistance for the disabled, mostly through the development of "smart" wheelchairs[8][13][20]. These wheelchairs provide a shared-control system that assists the user with maneuvers such as obstacle avoidance, door passing, and wall following. This work differs from ours in two ways. First, our users are not willing to use a wheelchair because they are capable of walking (with a little support) and they want to. Second, although most smart wheelchairs perform maneuvers in a manner that is smooth and comfortable to the user, the user is essentially riding a mobile robot. In our case, the user is not fully under the control of their mobility aid (because they are walking behind it) and the control system must take this into account.

Pedestrian mobility aids for the blind have also been developed. The GuideCane [3] is a small robot base on the

controlling the direction that the front wheel is pointing. The sensor system monitors the immediate environment for danger (obstacles or cliffs). If there is no danger, the walker is completely under the user's control. If danger is detected, the control system either brakes or steers to try and avoid it. In this control strategy, the walker attempts to make decisions based upon an estimate of the users goals. Using the direction to the obstacle, and a predicted avoidance direction, it adjusts the steering of the walker. It then monitors the sensors to adapt its estimate of the goals. Using a hierarchical strategy based on the instantiation of beliefs from sensor data [19], it selects the control strategy that most likely conforms to the users actions. This selection evolves over time, based on the input from the user, and the nature of the sensory data.

If the danger is closer than a certain minimum distance, the above braking rules are applied. Otherwise, the following steering rules are in effect.

$$\text{If } \mathrm{sign}(V_{user}) = \mathrm{sign}(P_{walker}) \rightarrow \quad (4)$$
$$V_{wheel} \sim 1/(\textit{distance to object})$$
$$|V_{wheel}| \leq V_{max}$$
$$\text{until } P_{wheel} = P_{walker}$$

$$\text{If } \mathrm{sign}(V_{user}) \neq \mathrm{sign}(P_{walker}) \rightarrow \quad (5)$$
$$V_{wheel} = V_{user} + \mathrm{sign}(P_{walker}) * c$$
$$\text{until } P_{wheel} = 0 \text{ or } \Delta CT_1 \text{ expires}$$

$$\text{If } \mathrm{sign}(V_{user}) \neq \mathrm{sign}(P_{walker}) \text{ and } \quad (6)$$
$$\Delta CT_1 \text{ expired} \rightarrow$$
$$V_{wheel} = V_{user}$$

where V_{user} is the steering force applied by the user (direction and magnitude) and $\mathrm{sign}(V_{user})$ is simply the direction of the force, i.e. the way the user is pushing the walker[1]. P_{walker} is the heading computed by the control system to avoid the obstacle detected by the sensors. P_{wheel} is the actual angle of the wheel and V_{wheel} is the rate at which the wheel's orientation is changing (capped at V_{max}). Finally, c is a constant and ΔCT_1 is the correction time allowed by the control system. All wheel velocities and angles are in a coordinate system in which the wheel orientation at the origin (0°) cause the walker to move straight ahead, a positive orientation turns the walker left and a negative orientation turns the walker right.

Rule 4 causes the walker to turn toward its desired wheel angle (P_{walker}) when the user and the control system both want to turn left or both want to turn right. The wheel's turning velocity is inversely proportional to the distance to the obstacle being avoided. Rule 5 represents a time-limited compromise when the control system and the user wish to travel in different directions. The walker will steer in the user's intended direction, but along a shallower angle (i.e. closer to 0°). This will only continue until either the wheel is pointing straight ahead or the correction time (ΔCT_1) expires. If this timer expires, then rule 6 is in effect and the walker merely follows the user's commands.

[1] The difference between V_{user} and $\mathrm{sign}(V_{user})$ is important. We can only estimate V_{user} by observing the motion of the motion of the walker's frame. However, our handle sensors can determine $\mathrm{sign}(V_{user})$ without the frame moving.

The interaction of these rules with the user is of primary importance. Rule 4 assumes that if the user is turning in the direction that the control system wants to turn then the user is also trying to avoid the obstacle. Of course, this need not be the case because rule 4 does not consider the magnitude of the turn, merely the sign. However, because the rules are continually evaluated, the walker will reach the desired angle of either the user or the control system first and then rule 5 will soon apply.

When the user and the control system wish to turn in different directions, the correction timer starts and rule 5 is in effect. The walker turns in the user's intended direction, but the control system keeps edging the wheel back straight. This compromise rule helps to keep the walker's response within the user's expectations of how the device should move based on their input. However, if the walker deviates from the user's course for too long, the user will feel like they are fighting with the walker. Rule 6 applies in this situation. The correction time is meant to try and steer the user away from the obstacle, or into a position where they begin taking appropriate action of their own, causing rule 4 to be in effect. Note that the user is not in danger when rule 5 or 6 are in effect because the braking rules will still slow (eventually stopping) the walker if an obstacle or cliff gets too close.

Path Following

This type of control system is similar to the safety steering and braking control system, but it can effect the walker's wheel orientation even when there is no obstacle or cliff to avoid. This type of control is the most complex because, if it is done incorrectly, it is most likely to make the user feel that they are not in control of the device. The main benefit to allowing the control system to guide the walker's user along a certain path is that the control system can adjust to the capabilities of the user as they vary and provide more or less guidance as needed.

The following rules are in effect when no obstacles are detected. The rules are divided into two sets depending on whether or not the user is within some distance d_{max} of the path selected by the control system.

$$\text{If } d_{(walker,path)} \geq d_{max}$$

$$\text{If } \mathrm{sign}(V_{user}) \neq \angle_{path} \rightarrow \quad (7)$$
$$V_{wheel} = \angle_{path} * V_{min}$$
$$\text{until } \Delta CT_2 \text{ expires}$$

$$\text{If } \mathrm{sign}(V_{user}) \neq \angle_{path} \text{ and } \quad (8)$$
$$\Delta CT_2 \text{ expires} \rightarrow$$
$$BF = BF_{min} \text{ until } \Delta CT_3 \text{ expires}$$

$$\text{If } \mathrm{sign}(V_{user}) \neq \angle_{path} \text{ and } \Delta CT_2 \text{ expires} \quad (9)$$
$$\text{and } \Delta CT_3 \text{ expires} \rightarrow$$
$$V_{wheel} = V_{user} \text{ (recompute path)}$$

$$\text{If } \mathrm{sign}(V_{user}) = \angle_{path} \rightarrow V_{wheel} = V_{user} \quad (10)$$

$$\text{If } d_{(walker,path)} < d_{max}$$

$$\text{If } (P_{wheel} \mid\mid \text{path}) \rightarrow V_{wheel} = V_{user} \quad (11)$$
$$\text{If } \neg(P_{wheel} \mid\mid \text{path}) \rightarrow V_{wheel} = \angle_{path} * V_{min} \quad (12)$$
$$\text{until } \Delta CT_4 \text{ expires}$$

trying to navigate. Rather the walker must attempt to use small changes in the operator's choice of direction and speed to maintain an estimate of the operator's goals. Using these estimates, the walker control system must adapt its control in a smooth, collaborative manner.

A Variety of Control System Designs

The Medical automation Research Center (MARC) is designing several walker control systems with different capabilities. As mentioned previously, all of our walkers lack propulsion and rely on their user to provide "forward" motion. All control systems can actuate the walker's brakes and most have the ability to steer, by controlling the orientation of walker's front wheel. The walkers are equipped with sonar, infrared sensors, and wheel encoders, to determine their location in the local environment. The control systems help walker users avoid obstacles and "cliffs". Cliffs are sudden drops in the user's walking surface, such as stairs or a sidewalk curb, which can cause the user to fall. In addition, some control systems can estimate their global position within a known environment (like a private home, rest home, or health care facility) to help guide the user to particular locations. The design of a control system that meets the above criteria depends on the service(s) that the walker is to provide. We are currently developing four types of control systems for the walkers (see figure 1).

Figure 1. MARC Smart Walker

These control systems begin with informational support only, and then add layers of control. These control layers progress from safety control, through simple navigational aid, to more complex goal achievement assistance. Each successive layer builds additional competencies into the walker in a subsumptive model based on that proposed by Brooks [4]. In effect, these layers of control act as a set of possible goals for the walker, and based on the evolving user inputs and sensor data, the walker reprioritizes these goals [9] to collaborate with the user.

Warning System Only

The warning system only control system cannot steer or brake the walker; it can only alert the user that there is impending danger. Small motors mounted in the walker's handles cause them to vibrate when the sensors detect a nearby obstacle. We feel that this has an advantage over an audible warning because it is felt only by the user and does not call attention to them. However, a disadvantage is that the user must scan the nearby area to locate the obstacle detected by the sensors.

This walker has no means of effecting its own motion, and therefore collaborates with the user in only the simplest manner. It will be used as one of the baselines in our user tests.

Safety Braking Only

This type of control system can brake the walker's wheels, but cannot steer. The walker frame's original, bicycle-style, hand brakes have been augmented with a lead-screw system that can depress the brake pads. This system can be triggered whenever the walker comes to a stop, or when an imminent collision is detected. The brake system allows the walker to be slowed gradually, as an obstacle is approached, stopping at some minimum distance.

The control system has the goals of preventing collisions and keeping the user upright. Preliminary experiments with a solenoid braking system showed that fast-firing brakes can be disconcerting to the user. Instead, brakes should come on and off slowly so that the user feels the resistance increase in stages. This leads to these control rules:

```
If the walker is in motion (as                    (1)
determined by the wheel encoder), the
force applied to the brake pads is
inversely proportional to the
distance to obstacles.
If the walker is stopped, the brakes              (2)
should be fully applied to provide a
stable base on which the user can
rest
When the walker is stopped and the                (3)
user wishes to move again, the brakes
should come off slowly to prevent the
walker from lurching forward
```

Rule 1 is self-explanatory. Rule 2 is used to provide a stable frame configuration when the user stops to rest. Rule 3 raises two interesting questions. First, how to distinguish between the forces applied to the walker while leaning against it to rest and forces that indicate that the user wishes to move and second how to deal with users with a slow or halting gait. The first question is difficult and the subject of our work in designing force sensors for the walker handles. Our initial prototype will have a handle-mounted push button with which the user can toggle the walker back into the "user wishes to move" state. The second question relates to the time between steps in the user's gait. If the user stops for too long between steps, rule 2 will activate and the user will have to keep toggling the walker mode. The walker's control system will have to be tuned for the user by clinicians experienced in mobility.

Safety Braking and Steering

This control system can respond to danger by both braking and steering. Recall that only the user provides motive force and so the following discussion is entirely about