Probablistic Control of Human Robot Interaction: Experiments with A Robotic Assistant for Nursing Homes

Joelle Pineau[1], Michael Montemerlo[1], Martha Pollack[2], Nicholas Roy[1] and Sebastian Thrun[1]
[1]Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, 15232, USA
[2]Dept. of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI, 48019, USA
http://www.cs.cmu.edu/∼nursebot

This paper describes an implemented robot system designed to assist nurses and elderly persons in institutionalized settings. The robot Pearl has been developed as a multi-functional robotic assistant. Its primary task involves guiding people through a nursing facility, reminding them of upcoming events and the need to take medication, and providing them with information very much like a mobile information kiosk. In the past 24 months, the robot has been deployed more than six times in an elderly care facility in Oakmont, PA. In this paper, we present three software modules relevant to ensure successful robot performance in a dynamic and interactive task domain: an automated reminder system; a people-tracking and detection system; and finally a high-level robot controller which performs planning under uncertainty by incorporating knowledge from low-level modules, and selecting appropriate courses of actions.

1. Introduction

The US population is aging at an alarming rate. At present, 12.5% of the US population is of age 65 or older (33). It is widely recognized that this ratio will increase as the baby-boomer generation moves into retirement age. Meanwhile, the nation faces a significant shortage of nursing professionals. The Federation of Nurses and Health Care Professionals has projected a need for 450,000 additional nurses by the year 2008.

This acute need provides significant opportunities for roboticists and AI researchers to develop assistive technology that can improve the quality of life of our aging population, and help nurses become more effective in their activities. The *Nursebot Project* was conceived in response to this challenge. It is formed by a multi-disciplinary team of investigators from the fields of health-care, HCI/psychology, and AI/robotics. The overall goal of the project is to develop mobile robotic assistants that can assist nurses and elderly people in their daily activities.

To this aim, the team has developed two prototype autonomous mobile robots, shown in Figure 1 (25). These robots primarily interact with the world through speech, visual displays, facial expressions and physical motion. They differ from earlier workplace robots in that they go beyond simply interacting with an (often static) environment, to interacting with human users and bystanders. Thus we leverage earlier technology for navigation, localization and mapping, and specifically focus on developing new algorithmic approaches to track people, predict their behavior, and react appropriately.

The idea of building robotic companions for the elderly is not new (11; 12; 16; 20; 21; 34). From the many services a nursing-assistant robot could provide, the work reported here considers the task of reminding people of events and guiding them through their environments. Both of these tasks are particularly relevant with the elderly community. Decreased memory is a common effect of age-related cognitive decline, which often leads to forgetfulness about routine daily activities (e.g. taking medications, attending appointments, eating, drinking, bathing, toileting) thus the need for a robot that can offer cognitive reminders. In addition, nursing staff in assisted living facilities frequently need to escort elderly people walking, either to get exercise, or to attend meals, appointments or social events. The fact that many elderly people move at extremely slow speeds (e.g. 5 cm/sec) makes this one of the most labor-intensive tasks in assisted living facilities. It is also important to note that the help provided is often not strictly of a physical nature, as many elderly people select walking aids over physical assistance by nurses. Rather, nurses often provide important cognitive help, in the form of reminders, guidance and motivation, in addition to valuable social interaction.

From an AI point of view, several factors make this task a challenging one for a robot to accomplish successfully, particularly because of the prevalence of uncertainty in the task domain. The type of uncertainty relevant to robot decision-making is two-fold. First, we are concerned with estimating the effects of a robot's actions. For example, a robot travelling at great speed may quickly tire an elderly person following it. Second, we are concerned with handling partial or erroneous sensor measurements. For example, when escorting a person through busy hallways the robot faces the risk of losing that individual or confusing him/her with another. Our approach explicitly considers these forms of uncertainty when optimizing a control strategy. Our approach also considers the costs of suboptimal control actions, which can vary widely, from the cost of unnecessarily asking a clarification question to incorrectly moving to a remote location.

The work presented focuses on three key software components of our robotic architecture: an automated reminder system that incorporates knowledge of a person's typical schedule with observations of recent activities, and issues pertinent reminders about upcoming events; a module which uses efficient particle filter techniques to detect and track people; and finally a high-level robot controller which uses probabilistic reasoning techniques to arbitrate between information-gathering and performance-related actions, as well as incorporate information obtained through both navigation sensors (e.g. laser range-finder) and interaction sensors (e.g. speech recognition and touchscreen).

In systematic experiments conducted at a nursing home, we found the combination of techniques to be highly effective in dealing with elderly test subjects. In particular, during a sequence of one-one-one scenarios between Pearl and residents of the nursing home, the robot demonstrated the ability to contact a resident, remind them of an appointment, accompany them to that appointment, as well as provide information of interest to that person, for example weather reports or television schedules.

Figure 1. Nursebots Flo (left) and Pearl (right)

2. Hardware and Software Description

Figure 1 shows images of the robots Flo (first prototype, now retired) and Pearl (the present robot). Each robot is equipped with a differential drive system, two on-board PCs, wireless ethernet, laser range finders, sonar sensors, microphones for speech recognition, speakers for speech synthesis, touch-sensitive graphical displays, actuated head units, and stereo camera systems. As a result of feedback from nurses and medical experts following deployment of the first robot, Flo, the second robot Pearl also features an improved visual appearance, two sturdy handle-bars, a more compact design that allows for cargo space and a removable tray, doubled battery capacity, a second laser range finder, and a significantly more sophisticated head unit.

On the software side, both robots feature off-the-shelf autonomous mobile robot navigation system (4; 31), speech recognition software (27), speech synthesis software (3), fast image capture and compression software for online video streaming, face detection tracking software (28), as well as the three major new software modules described in this paper. These modules are principally concerned with people interaction and control. They overcome important deficiencies of the work described by (4; 31), which had only rudimentary abilities to interact with people.

3. Plan management with Autominder

The Autominder software component is designed as an intelligent *cognitive orthotic* system, providing elderly people with reminders about their daily activities (26). The idea of using computer technology to enhance the performance of cognitively disabled people dates back nearly forty years (13). More recently, cognitive orthotics have enabled reminders to be provided using the telephone (14), personal digital assistants (10), and pagers (17). Work has also been done on improved modelling of users' activities (19; 23), where in one case a hand-device uses AI planning technology to model the user's plans, and provide visual and audible cues about its execution.

In the Nursebot project, the goal of this software system is to make principled decisions about what reminders to issue and when, balancing the following potentially competing objectives: (i) ensure that the user is aware of activities s/he is expected to perform, (ii) increase the likelihood that s/he will perform at least the required activities (e.g. taking medicine), (iii) avoid annoying the user, and (iv) avoid making the user overly reliant on the system. To attain these goals, the system must be flexible and adaptive, responding to the actions taken by the user.

The Autominder architecture is shown in figure 2. As depicted, the system maintains an accurate model of a user's daily schedule, monitors performance of activities, and plans reminders accordingly. The three main components are: a Plan Manager (PM), which stores the user's plan of daily activities in the *Client Plan*, and is responsible for updating it and identifying any potential conflicts in it; a Client Modeler (CM), which uses information about the user's observable activities to track the execution of the plan, storing its beliefs about the execution status in the *Client Model*; and a Personal Cognitive Orthotic (PCO), which reasons about any disparities between what the user is supposed to do and what s/he is doing, and makes decisions about when to issue reminders.

To initialize the system, the caregiver for an elderly user inputs a description of the user's daily activities, as well as any constraints on, or preferences regarding, the time or manner of their performance. This plan may then be changed in one of four ways: (i) the user or caregiver may add new activities; (ii) the user or caregiver may modify or delete activities already in the plan; (iii) the user may execute one of the planned activities; or (iv) the simple passage of time may cause automatic changes to be made in the plan. Whenever a change occurs, the PM updates the user plan, performing plan merging and constraint

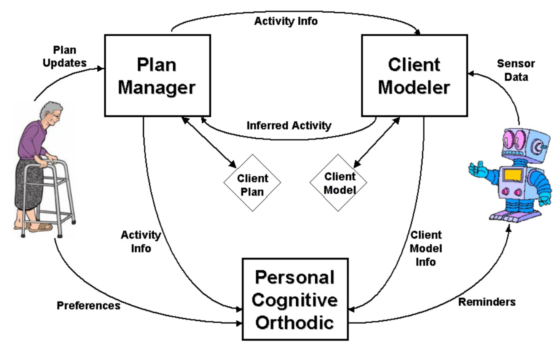


Figure 2. Autominder Architecture

propagation as needed. To adequately represent user plans, it essential to support a rich set of temporal constraints; we achieve this goal by modelling user plans as Disjunctive Temporal Problems (DTPs) and reasoning about them using efficient algorithms (32).

Throughout the day, sensor information is gathered by the robot and sent to the CM, which uses this information to try to infer what activities the user is performing. If the likelihood is high that a planned activity has been executed, the CM reports this to the PM, which can then update the user's plan by recording the time of execution, and propagate any affected constraints accordingly. The user model is represented using a Quantitative Temporal Bayes Net (QTBN), which was developed to handle the need both to reason about fluents and about probabilistic temporal constraints (5).

The final component of the Autominder is the PCO (24), which uses both the user plan and the user model to determine what reminders should be issued and when. The PCO identifies activities that may require reminders based on their importance and their likelihood of being executed on time as modeled in the CM. It also determines the most effective times to issue each required reminder, taking account of the expected user behavior, and any preferences explicitly provided by the user and the caregiver. Finally, the PCO provides justifications as to why particular activities warrant a reminder. The

PCO treats the generation of a reminder plan as a satisficing problem and uses a local-search approach called Planning-by-Rewriting (PbR) (1) to produce a high-quality reminder plan that takes into account the user's expected behavior, preferences, and interactions amongst planned activities.

4. Locating People

In order to issue reminders and, when appropriate, guide users to their activities, it is necessary to interact with people spatially, and most specifically to be able to locate people in their living environment. The problem of locating people is the problem of determining their x-y-location relative to the robot. Previous approaches to people tracking in robotics are feature-based: they analyze sensor measurements (images, range scans) for the presence of features (15; 29) as the basis of tracking. In our case, the diversity of the environment mandates a different approach. Pearl detects people using map differencing: the robot learns a map, and people are detected by significant deviations from the map. Figure 3 shows an example map acquired using preexisting software (31).

Mathematically, the problem of people tracking is a combined posterior estimation problem and model selection problem. Let N be the number of people near the robot. The posterior over the people's positions is given by

$$p(y_{1,t}, \ldots, y_{N,t} | z^t, u^t, m) \tag{1}$$

where $y_{n,t}$ with $1 \leq n \leq N$ is the location of a person at time t, z^t the sequence of all sensor measurements, u^t the sequence of all robot controls, and m is the environment map. However, to use map differencing, the robot has to know its own location. The location and total number of nearby people detected by the robot is clearly dependent on the robot's estimate of its own location and heading direction. Hence, Pearl estimates a posterior of the type:

$$p(y_{1,t}, \ldots, y_{N,t}, x^t | z^t, u^t, m) \tag{2}$$

where x^t denotes the sequence of robot poses (the path) up to time t. If N was known, estimating this posterior would be a high-dimensional estimation problem, with complexity cubic in N for Kalman filters (2), or exponential in N with particle filters (8). Neither of these approaches is, thus, applicable: Kalman filters cannot globally localize the robot, and particle filters would be computationally prohibitive.

Luckily, under mildly restrictive conditions (discussed below) the posterior (2) can be factored into $N + 1$ conditionally independent estimates:

$$p(x^t | z^t, u^t, m) \prod_n p(y_{n,t} | z^t, u^t, m) \tag{3}$$

This factorization opens the door for a particle filter that scales linearly in N. Our approach is similar (but not identical) to the Rao-Blackwellized particle filter described in (9). First, the robot path x^t is estimated using a particle filter, as in the Monte Carlo localization (MCL) algorithm for mobile robot localization (6). Each particle in this filter is associated with a set of N particle filters, each representing one of the people position estimates $p(y_{n,t} | z^t, u^t, m)$. These *conditional* particle filters represent people position estimates *conditioned* on robot path estimates—hence capturing the inherent dependence of people and robot location estimates. The data association between measurements and people is done using maximum likelihood, as in (2). Under the (false) assumption that this maximum likelihood estimator is always correct, our approach can be shown to converge to the correct posterior, and it does so with update time linear in N. In practice, we found that the data association is correct in the vast majority of situations. The nested particle filter formulation has a secondary advantage that the number of people N can be made dependent on individual robot path particles. Our approach for estimating N uses the classical AIC criterion for model selection, with a prior that imposes a complexity penalty exponential in N.

Figure 3 shows results of the filter in action. In Figure 3a, the robot is globally uncertain, and the number and location of the corresponding people estimates varies drastically. As the robot reduces its uncertainty, the number of modes in the robot pose posterior quickly becomes finite, and each such mode has a distinct set of people estimates, as shown in Figure 3b. Finally, as the robot is localized, so is the person (Figure 3c). When guiding people, the localization estimate of the person is used to determine the velocity of the robot, so that the robot maintains roughly a constant distance to the person. In our experiments in

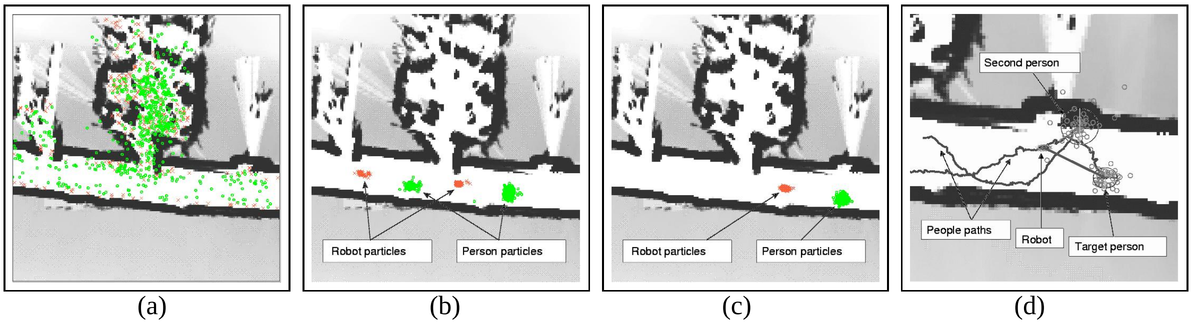


Figure 3. (a)-(d) Evolution of the conditional particle filter from global uncertainty to successful localization and tracking. (d) The tracker continues to track a person even as that person is occluded repeatedly by a second individual.

the target facility, we found the adaptive velocity control to be absolutely essential for the robot's ability to cope with the huge range of walking paces found in the elderly population. Initial experiments with fixed velocity led almost always to frustration on the people's side, in that the robot was either too slow or too fast.

Finally, Figure 3d illustrates the robustness of the filter to interfering people. Here another person steps between the robot and its target subject. The filter obtains its robustness to occlusion from a carefully crafted probabilistic model of people's motion $p(y_{n,t+1}|y_{n,t})$. This enables the conditional particle filters to maintain tight estimates while the occlusion takes place, as shown in Figure 3d. During in-lab experiments involving 31 tracking instances with up to five people at a time, the error in determining the number of people was 9.6%. The error in the robot position was 2.5 ± 5.7 cm, and the people position error was as low as 1.5 ± 4.2 cm, when compared to measurements obtained with a carefully calibrated static sensor with ± 1 cm error.

5. High Level Robot Control and Dialog Management

The most central module in Pearl's software is a probabilistic algorithm for high-level control and dialog management. This module integrates observations from lower-level modules (e.g. the Autominder, the people tracker, the speech recognizer, etc.) and uses this information to select appropriate behaviors and responses.

Pearl's high-level control architecture is a hierarchical variant of a partially observable Markov decision process (POMDP) (18). The POMDP is a model for calculating optimal control actions under uncertainty. The control decision is based on a probabilistic belief over possible states.

In Pearl's case, this distribution is defined over a collection of multi-valued state variables:

- robot location (discrete approximation)
- person's location (discrete approximation)
- person's status (inferred from speech recognizer)
- motion goal (where to move)
- reminder goal (what to inform the user of)
- user initiated goal (e.g., an information request)

The value of the *person's location* variable is observed through the people tracker, and similarly the *reminder goal* variable is set by the Autominder module. Overall, there are 516 possible states. The input to the POMDP is a factored probability distribution over these states, generated by a state estimator, such as in Equation (2). Uncertainty over the current state arises predominantly from the localization modules and the speech recognition system. The consideration of uncertainty is especially important in this domain, as the costs of giving a reminder to the wrong person, or unnecessarily sending the robot to a location can be large.

Unfortunately, POMDPs of the size encountered here are an order of magnitude larger than today's

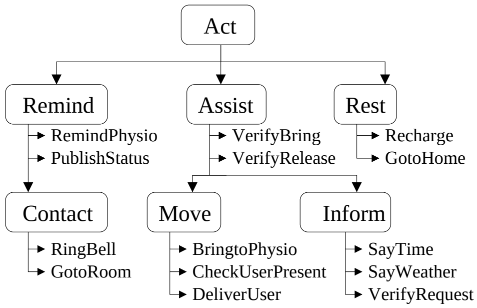


Figure 4. Dialog Problem Action Hierarchy

Observation	True State	Action	Reward
pearl hello	request_begun	say_hello	100
pearl what is like	start_meds	**ask_repeat**	-100
pearl what time is it for will the	want_time	say_time	100
pearl was on abc	want_tv	ask_which_station	-1
pearl was on abc	want_abc	say_abc	100
pearl what is on nbc	want_nbc	**confirm_channel_nbc**	-1
pearl yes	want_nbc	say_nbc	100
pearl go to the that pretty good what	send_robot	ask_robot_where	-1
pearl that that hello be	send_robot_bedroom	**confirm_robot_place**	-1
pearl the bedroom any i	send_robot_bedroom	go_to_bedroom	100
pearl go it eight a hello	send_robot	ask_robot_where	-1
pearl the kitchen hello	send_robot_kitchen	go_to_kitchen	100

Table 1
Sample dialog demonstrating the role of clarification actions. Actions in bold font are clarification actions, chosen by the POMDP because of high uncertainty in the speech signal.

best exact POMDP algorithms can tackle (18). However, Pearl's domain is highly structured, since certain actions are only applicable in certain situations. To exploit this structure, we developed a *hierarchical* version of POMDPs, which breaks down the decision making problem into a collection of smaller problems that can be solved more efficiently. Our approach is similar to the MAX-Q decomposition for MDPs (7), but defined over POMDPs (where states are unobserved).

The basic idea of the hierarchical POMDP is to partition the action space—not the state space, since the state is not fully observable—into smaller chunks. For Pearl's guidance task the action hierarchy is shown in Figure 4, where *abstract actions* (shown in circles) are introduced to subsume logical subgroups of lower-level actions. This action hierarchy induces a decomposition of the control problem, where at each node all lower-level actions, if any, are considered in the context of a local sub-controller. At the lowest level, the control problem is a regular POMDP, with a reduced action space. At higher levels, the control problem is also a POMDP, yet involves a mixture of physical and abstract actions (where abstract actions correspond to lower level POMDPs.)

It is important to notice that such a decomposition is especially appropriate when the optimal control transgresses down along a single path in the hierarchy to receive its reward. This is approximately the case in the Pearl domain, where goals are satisfied upon successfully delivering a person, or successfully gathering information through communication.

Using the hierarchical POMDP, the high-level decision making problem in Pearl is tractable, and a near-optimal control policy can be computed off-line. Thus, during execution time the controller simply monitors the state (calculates the posterior) and looks up the appropriate control. Table 1 shows an example dialog between the robot and a test subject. Because of the uncertainty management in POMDPs, the robot chooses to ask a clarification question at three occasions. The number of such questions depends on the clarity of a person's speech, as detected by the Sphinx speech recognition system.

An important remaining question concerns the importance of handling uncertainty in high-level control. To investigate this, we ran a series of comparative experiments, all involving real data collected in our lab. In one series of experiments, we investigated the importance of considering the uncertainty arising from the speech interface. In particular, we compared Pearl's performance to a system that ignores that uncertainty, but is otherwise identical. The resulting approach is an MDP, similar to the one described in (30). Figure 5 shows results for three different performance measures, and three different users (in decreasing order of speech recognition performance). For poor speakers, the MDP requires less time to "satisfy" a request due to the lack of clarification questions (Figure 5a). However, its error rate is much higher (Figure 5b), which negatively affects the over-

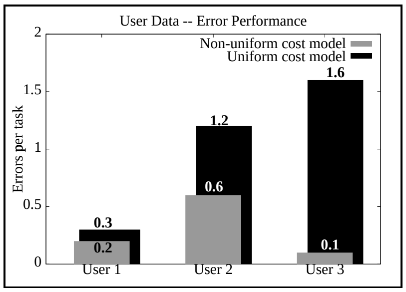


Figure 6. Empirical comparison between uniform and non-uniform cost models. Results are an average over 10 tasks. Depicted are 3 example users, with varying levels of speech recognition accuracy. Users 2 & 3 had the lowest recognition accuracy, and consequently more errors when using the uniform cost model.

all reward received by the robot (Figure 5c). These results clearly demonstrate the importance of considering uncertainty at the highest robot control level, specifically with poor speech recognition.

In the second series of experiments, we investigated the importance of uncertainty management in the context of highly imbalanced costs and rewards. For example, in Pearl's case, asking a clarification question is in fact much cheaper than accidentally delivering a person to a wrong location, or guiding a person who does not want to be walked. We therefore compared performance using two POMDP models which differed only in their cost models. One model assumed uniform costs for all actions, whereas the second model assumed a more discriminative cost model in which the cost of verbal questions was lower than the cost of performing the wrong motion actions. A POMDP policy was learned for each of these models, and then tested experimentally in our laboratory. The results presented in figure 6 show that the non-uniform model makes more judicious use of confirmation actions, thus leading to a significantly lower error rate, especially for users with low recognition accuracy.

6. Results

Following integration of the three software modules onto Pearl, the robot was deployed in a retirement community located near Pittsburgh, PA. This section describes experiments involving elderly residents of this facility, with mild cognitive, perceptual, or physical limitations.

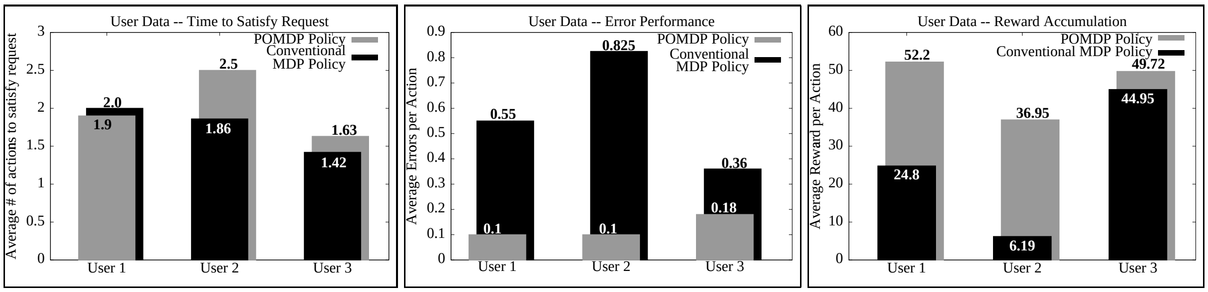
We tested the robot in five separate experiments, each lasting one full day. The first three days focused on open-ended interactions with a large number of elderly users, during which the robot interacted verbally and spatially with elderly people with the specific task of delivered sweets. This allowed us to gauge people's initial reactions to the robot.

Following this, we performed two days of formal experiments during which the robot autonomously led 12 full guidances, involving 6 different elderly people. Figure 7 shows an example guidance experiment, involving an elderly person who uses a walking aid. The sequence of images illustrates the major stages of a successful delivery: from contacting the person, delivering the reminder, walking her through the facility, and providing information after the successful delivery—in this case on the weather.

In all trials, the task was performed to completion. Post-experimental debriefings illustrated a uniform high level of excitement on the side of the elderly. Overall, only a few problems were detected during the operation. None of the test subjects showed difficulties understanding the major functions of the robot. They all were able to operate the robot after less than five minutes of introduction. Earlier trials with a poorly adjusted speech recognition system, and fixed-velocity robot motion, both caused difficulties. These were addressed early on by increasing the role of the touchscreen, and including adaptable velocities.

7. Discussion

This paper described a mobile robotic assistant for nurses and elderly residents in assisted living facilities. The system has been tested successfully in experiments in an assisted living facility. The experiments were successful in two main dimensions. First, they provided some evidence towards the feasibility of using autonomous mobile robots as assistants to nurses and institutionalized elderly. Second, they demon-



(a) (b) (c)

Figure 5. Empirical comparison between POMDPs (with uncertainty, shown in gray) and MDPs (no uncertainty, shown in black) for high-level robot control, evaluated on data collected in the assisted living facility. Shown are the average time to task completion (a), the average number of errors (b), and the average user-assigned (*not* model assigned) reward (c), for the MDP and POMDP. The data is shown for three users, with good, average and poor speech recognition.

strated that various probabilistic tracking and planning techniques are well-suited to solve problems pertaining to human-robot interactions.

One of the key lessons learned while developing this robot is that the elderly population requires techniques that can cope with individual differences (e.g. walking speed), age-related decline (e.g. memory loss) and noisy perception (e.g. poor speech recognition). We view the area of assistive technology as a prime source for great AI problems in the future.

References

[1] J.L. Ambite and C.A. Knoblock. Planning by rewriting. *JAIR*, 15:207-261. 2001.

[2] Y. Bar-Shalom and T. E. Fortmann. *Tracking and Data Association.* Academic Press. 1998.

[3] A.W. Black, P. Taylor, and R. Caley. *The Festival Speech Synthesis System.* University of Edinburgh. 1999.

[4] W. Burgard, A.B., Cremers, D. Fox, D. Hähnel, G. Lakemeyer, D. Schulz, W. Steiner, and S. Thrun. The interactive museum tour-guide robot. *AAAI-98*.

[5] D. Colbry, B. Peintner and M.E. Pollack. Execution monitoring with quantitative temporal bayesian networks. *AIPS-02*.

[6] F. Dellaert, D. Fox, W. Burgard, and S. Thrun. Monte Carlo localization for mobile robots. *ICRA-99*.

[7] T. Dietterich. The MAXQ method for hierarchical reinforcement learning. *ICML-98.*

[8] A. Doucet, N. de Freitas, and N.J. Gordon, editors. *Sequential Monte Carlo Methods In Practice.* Springer. 2001.

[9] A. Doucet, N. de Freitas, K. Murphy, and S. Russell. Rao-Blackwellised particle filtering for dynamic bayesian networks. *UAI-2000.*

[10] M.M. Dowds, and K. Robinson Assistive technology for memory impairment: Palmtop computers and TBI. In *Braintree Hospital Traumatic Brain Injury Neurorehabilitation Conference.* 1996.

[11] S. Dubowsky, F. Genot, S. Godding, H. Kozono, A. Skwersky, H. Yu, and L. Yy. PAMM: Aid to the elderly for mobility assistance and monitoring: A helping hand for the elderly. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 570–576, 2000.

[12] G. Engelberger. Services. In *Handbook of Industrial Robotics*,. John Wiley and Sons. 1999.

[13] D.C. Englebart. A conceptual framework for the augmentation of man's intellect. In *Vistas in Information Handling.* Spartan Books. 1963.

[14] R.H. Friendman. Automated telephone conversation to assess health behavior and deliver behavioral interventions. *Journal of Medical Systems.* 22:95-101. 1998.

[15] D. M. Gavrila. The visual analysis of human movement: A survey. *Computer Vision and Image Understanding.* 73(1). 1999.

[16] B. Graf. Reactive navigation of an intelligent robotic walking aid. In *Proceedings IRS-2000*, pages 252–259, Montreal, CA, 2000.

[17] N.A. Hersh and L. Treadgold. Neuropage: The rehabilitation of memory dysfunction by prosthetic memory and cueing. *NeroRehabilitation.* 4:187-197. 1994.

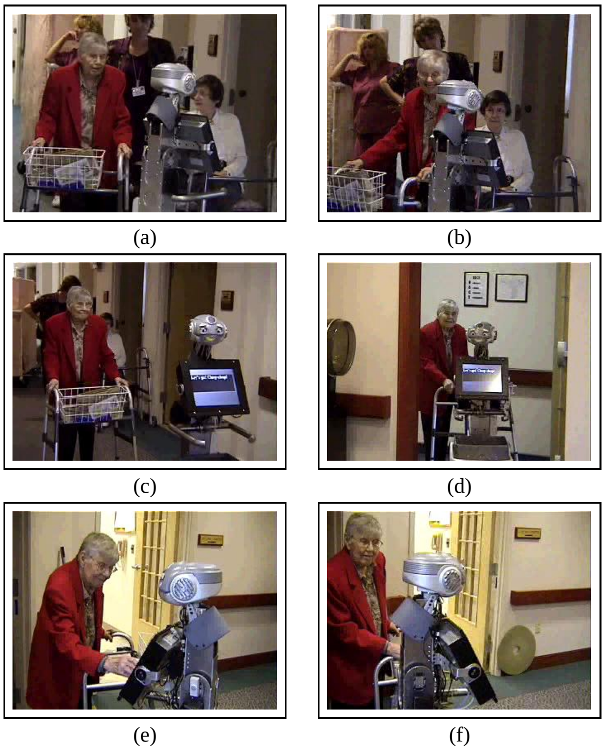


Figure 7. Example of a successful guidance experiment: a) Pearl picks up the patient outside her room, b) reminds her of a physiotherapy appointment, c) guides the person to the physiotherapy department, d) enters the department, e) satisfies a request for the weather report, and f) terminates the interaction and leaves.

[18] L.P. Kaelbling, M.L. Littman, and A.R. Cassandra. Planning and acting in partially observable stochastic domains. *Artificial Intelligence.* 101. 1998.

[19] N. Kirsch, S.P. Levine, M. Fallon-Kureger, L.A. Jaros. The microcomputer as an 'orthotic' device for pations with cognitive deficits. *Journal of Head Trauma Rehabilitation.* 2:77-86.

[20] G. Lacey. Adaptive shared control of a robot mobility aid. In *Field and Service Robotics*, pages 25–30. Springer Verlag, 1999.

[21] G. Lacey and K.M. Dawson-Howe. The application of robotics to a mobility aid for the elderly blind. *Robotics and Autonomous Systems*. 23. 1998.

[22] G. Lakemeyer, editor. Notes *Second International Workshop on Cognitive Robotics.* Berlin. 2000

[23] R. Levinson. Peat-the lanning and execution assistant and trainer. *Robotics and Autonomous systems.* 23:245-252.

[24] C.E. McCarthy, and M. Pollack. A Plan-Based Personalized Cognitive Orthotic. *AIPS-2002*.

[25] M. Montemerlo, J. Pineau, N. Roy, S. Thrun and V. Verma Experients with a Mobile Robotic Guide for the Elderly *AAAI-2002*

[26] M. Pollack Planning technology for intelligent cognitive orthotics *AIPS-2002*.

[27] M. Ravishankar. Efficient algorithms for speech recognition. 1996. Internal Report.

[28] H.A. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence.* 20(1). 1998.

[29] D. Schulz, W. Burgard, D. Fox, and A. Cremers. Tracking multiple moving targets with a mobile robot using particles filters and statistical data association. *ICRA-2001*.

[30] S. Singh, M. Kearns, D. Litman, and M. Walker. Reinforcement learning for spoken dialogue systems. *NIPS-2000*.

[31] S. Thrun, M. Beetz, M. Bennewitz, W. Burgard, A.B. Cremers, F. Dellaert, D. Fox, D. Hähnel, C. Rosenberg, N. Roy, J. Schulte, and D. Schulz. Probabilistic algorithms and the interactive museum tour-guide robot Minerva. *International Journal of Robotics Research*. 19(11). 2000.

[32] I. Tsamardinos. *Constraint-Based Temporal Reasoning Algorithm with Applications to Planning.* Ph.D. Dissertation. University of Pittsburgh Intelligent Systems Program. 2001.

[33] US Department of Health and Human Services. Health, United states, 1999. Health and aging chartbook, 1999.

[34] G. Wasson, J. Gunderson, S. Graves, and R. Felder. An assistive robotic agent for pedestrian mobility. In *Proceedings of the International Conference on Autonomous Agents*, pages 169–173, 2001.