ARTICLE IN PRESS

Available at www.ComputerScienceWeb.com
POWERED BY SCIENCE @ DIRECT®

Robotics and Autonomous Systems 1048 (2002) 1–11

Robotics and Autonomous Systems

www.elsevier.com/locate/robot

Towards robotic assistants in nursing homes: challenges and results

Joelle Pineau [a,*], Michael Montemerlo [a], Martha Pollack [b], Nicholas Roy [a], Sebastian Thrun [a]

[a] Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15232, USA
[b] Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48019, USA

Abstract

This paper describes a mobile robotic assistant, developed to assist elderly individuals with mild cognitive and physical impairments, as well as support nurses in their daily activities. We present three software modules relevant to ensure successful human–robot interaction: an automated reminder system; a people tracking and detection system; and finally a high-level robot controller that performs planning under uncertainty by incorporating knowledge from low-level modules, and selecting appropriate courses of actions. During the course of experiments conducted in an assisted living facility, the robot successfully demonstrated that it could autonomously provide reminders and guidance for elderly residents.
© 2002 Published by Elsevier Science B.V.

Keywords: Robot control; Human–robot interaction; Planning; Scheduling; Probabilistic reasoning

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 [31]. 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 robotics 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 a prototype autonomous mobile robot, shown in Fig. 1 [23]. This robot primarily interacts with the world through speech, visual displays, facial expressions and physical motion. It differs from earlier workplace robots in that it goes 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.

From the many services a nursing-assistant robot could provide [12,19], 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 for the elderly community.

* Corresponding author. Tel.: +1-412-268-4857; fax: +1-412-268-5571.
E-mail address: jpineau@cs.cmu.edu (J. Pineau).

0921-8890/02/$ – see front matter © 2002 Published by Elsevier Science B.V.
doi:10.1016/S0921-8890(02)00381-0

UNCORRECTED PROOF

Fig. 1. Nursebot Pearl.

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/s) 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. Rather, nurses often provide important cognitive help, guidance and motivation, in addition to valuable social interaction.

Several factors make this task a challenging one for a robot to accomplish successfully. First, many elderly have difficulty understanding the robot's synthesized speech, as well as articulating an appropriate response in a computer-understandable way. In addition, walking abilities vary drastically between individuals. People with walking aids are usually an order of magnitude slower than people without, and people often stop to chat or catch their breath along the way. It is therefore imperative that the robot adapt to individuals—an aspect of interaction that has been poorly explored in AI and robotics.

The work presented in this paper seeks to address these challenges, focusing on three software components most pertinent to human–robot interaction: 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 that uses efficient particle filter techniques to detect and track people; and finally a high-level robot controller that uses probabilistic reasoning techniques to arbitrate between information-gathering and performance-related actions, while also incorporating information obtained through both navigation sensors (e.g. laser range finder) and interaction sensors (e.g. speech recognition and touch-screen).

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-on-one interactions 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.

2. Hardware and software description

Fig. 1 shows an image of the nursing robot Pearl. It 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 input from nurses and medical experts, Pearl also features two sturdy handle-bars, a compact design that allows for cargo space, a removable tray, and a sophisticated head unit.

On the software side, the robot features off-the-shelf autonomous mobile robot navigation system [5,29], speech recognition software [25], speech synthesis software [4], fast image capture and compression software for online video streaming, face detection tracking software [26], 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 [5,29], which had only rudimentary abilities to interact with people.

UNCORRECTED PROOF

3. Plan management with Autominder

The Autominder software component contains the intelligent *cognitive orthotic* system. It is designed to provide elderly people with reminders about their daily activities [24]. The idea of using computer technology to enhance the performance of cognitively disabled people dates back nearly 40 years [13]. More recently, cognitive orthotics have enabled reminders to be provided using the telephone [14], personal digital assistants [11], and pagers [16]. Related work has also been done on improved modeling of users' activities [18,21], 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 Fig. 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 care-giver of an elderly user inputs a description of the person'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 the four ways: (i) the user or care-giver may add new activities; (ii) the user or care-giver 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 propagation as needed. To adequately represent user plans, it is essential to support a rich set of temporal constraints; we achieve this goal by modeling user plans as Disjunctive Temporal Problems (DTPs) and reasoning about them using efficient algorithms [30].

The CM incorporates sensor information gathered by the robot to infer activities performed by the user. The relevant sensor information comes from laser readings, as well as touch-screen and speech input.

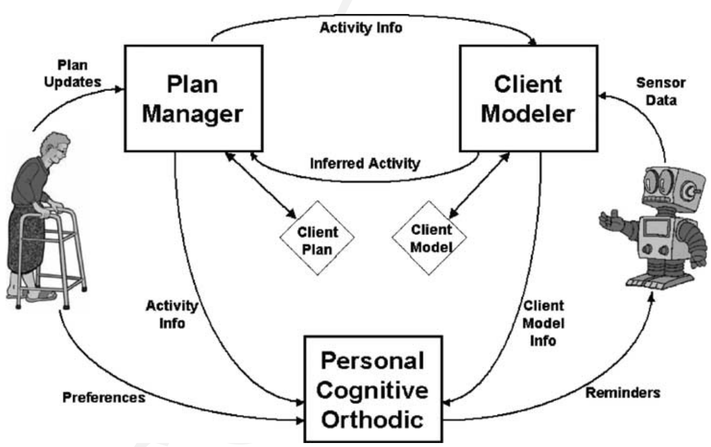


Fig. 2. Autominder architecture.

ARTICLE IN PRESS

The laser readings are used to track the user and reason about site-specific tasks (e.g. going into the kitchen for a period of time can indicate meal-taking). The touch-screen and speech are used to confirm compliance to reminders (e.g. whether medication has been taken). 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 [6].

The final component of the Autominder is the PCO [22], 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 care-giver. Finally, the PCO provides justifications as to why particular activities warrant a reminder. The PCO treats the generation of a reminder plan as a satisfying problem and uses a local-search approach called Planning-by-Rewriting (PbR) [2] to produce a high-quality reminder plan that takes into account the user's expected behavior, preferences, and interactions amongst planned activities.

The Autominder system was initially designed to interact with a specific individual, rather than a community of users. In the nursing home environment, Autominder would need to maintain parallel plans for each individual, and would need to identify the appropriate person for each action. This is particularly important when issuing key health reminders (e.g. medication and appointments). The current robot system does not fully address this problem: it simply assumes that the target person can be found in his/her room, and thus identifies individuals by their initial location. In the future, person identification could best be handled by camera-based face identification, or requiring the user to verbally confirm his/her identity. Though we have not focused on the problem of *person identification*, we do address the question of *person finding*, as described in the next section.

4. Locating people

In order to track users and guide them 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.[1] Previous approaches to people tracking in robotics are feature-based: they analyze sensor measurements (images, range scans) for the presence of features [15,27] 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. Fig. 3 shows an example map acquired using preexisting software [29].

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

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

where $y_{n,t}$ with $1 \le n \le N$ 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 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:

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

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 [3], or exponential in N with particle filters [9]. Neither of these approaches is applicable: Kalman filters cannot globally localize the robot, and particle filters would be computationally prohibitive.

[1] Depending on the task at hand, additional dimensions such as orientation or velocity and bearing may be of interest, but we ignore these features for our particular problem.

UNCORRECTED PROOF

J. Pineau et al. / Robotics and Autonomous Systems 1048 (2002) 1–11

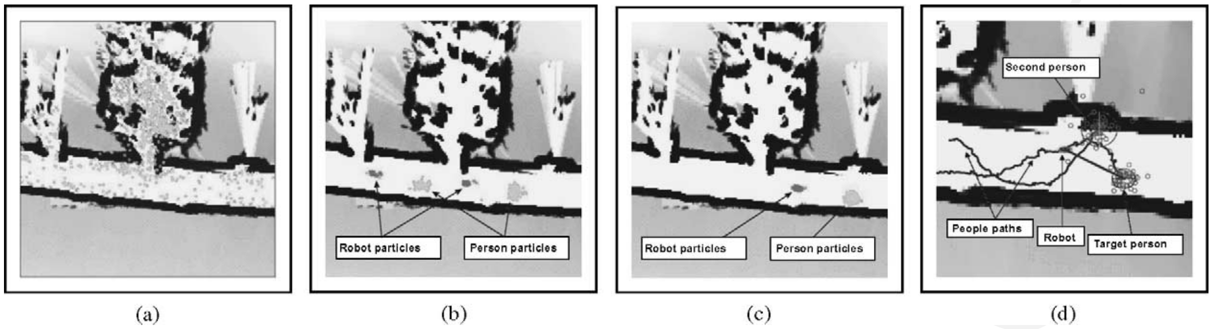


Fig. 3. (a)–(c) Evolution of the conditional particle filter from global uncertainty to successful localization and tracking. (d) The tracker continues to track a person even as that person is occluded repeatedly by a second individual.

Luckily, under mildly restrictive conditions (discussed below) the posterior (Eq. (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 [10]. First, the robot path x^t is estimated using a particle filter, as in the Monte Carlo localization (MCL) algorithm for mobile robot localization [7]. 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 [3]. 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 AIC criterion for model selection [1], with a prior that imposes a complexity penalty exponential in N.

Fig. 3 shows results of the filter in action. In Fig. 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 Fig. 3b. Finally, as the robot is localized, so is the person (Fig. 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 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, Fig. 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 Fig. 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.

UNCORRECTED PROOF

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 modeled as a partially observable Markov decision process (POMDP) [17]. 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 Eq. (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 the wrong reminder, 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 best exact POMDP algorithms can tackle [17]. 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 [8],

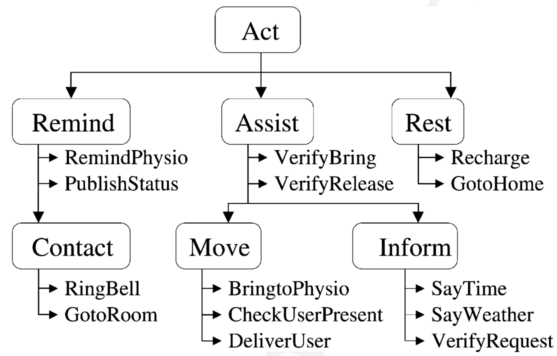


Fig. 4. Dialog problem action hierarchy.

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 Fig. 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 conveying or 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

ARTICLE IN PRESS

Table 1
Sample dialog demonstrating the role of clarification actions[a]

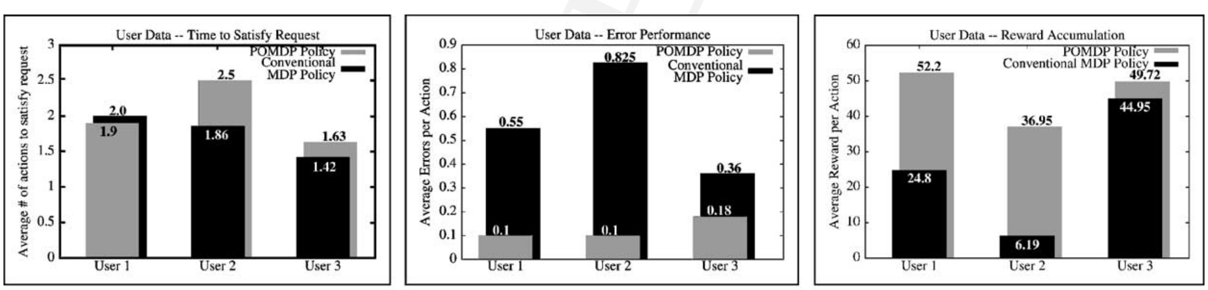
Observation	True state	Action	Reward
Pearl hello	request_begun	say_hello	100
Pearl what is like	start_meds	**ask_repeat**	−100
Pearl what time is it for will the	want_time	say_time	100
Pearl was on abc	want_tv	ask_which_station	−1
Pearl was on abc	want_abc	say_abc	100
Pearl what is on nbc	want_nbc	**confirm_channel_nbc**	−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

[a] The *observation* describes the output from the speech recognizer. The *true state* is manually extracted by a human observer. The *action* is chosen by the POMDP policy. Actions in bold font are clarification actions, chosen by the POMDP because of high uncertainty in the speech signal. *Reward* indicator describes the appropriateness of the action for the given state.

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, using real data collected in our lab. In the first experiment, we investigated the importance of considering the uncertainty arising from the speech interface. In particular, we compared Pearl's performance (using a POMDP to select actions) to a similar system that ignores that uncertainty. The second system uses an MDP policy, similar to the one described in [28]. Fig. 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 (Fig. 5a). However, its error rate is much higher (Fig. 5b), which negatively affects the overall reward received by the robot (Fig. 5c). These results clearly demonstrate the importance of considering uncertainty at the highest robot control level, specifically with poor speech recognition.

In the second experiment, 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 guiding a person to a



(a) (b) (c)

Fig. 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.

UNCORRECTED PROOF

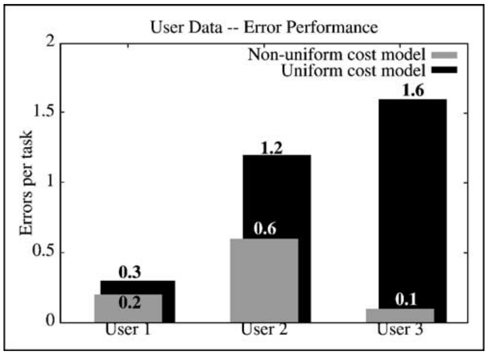


Fig. 6. Empirical comparison between uniform and non-uniform cost models. Results are an average over 10 tasks. Depicted are three example users, with varying levels of speech recognition accuracy. Users 2 and 3 had the lowest recognition accuracy, and consequently more errors when using the uniform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

wrong location, or guiding a person who does not want assistance. 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 Fig. 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.

These experiments confirm the need to reason about observation uncertainty during planning, and thus validate our choice of POMDPs as the appropriate model for robot interaction. Although the experiments described in this section focused principally on the uncertainty stemming from the speech interface, other robot sensors are also prone to measurement uncertainty which can be equally handled by the POMDP framework.

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 3 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 delivering sweets. This allowed us to gauge people's initial reactions to the robot.

Following this, we performed 2 days of formal experiments during which the robot autonomously conducted 12 test scenarios, involving six different elderly people. In each scenario, the robot was required to provide a timed reminder (e.g. scheduled appointment) to the test subject, lead the subject between locations in the facility, and verbally interact with the

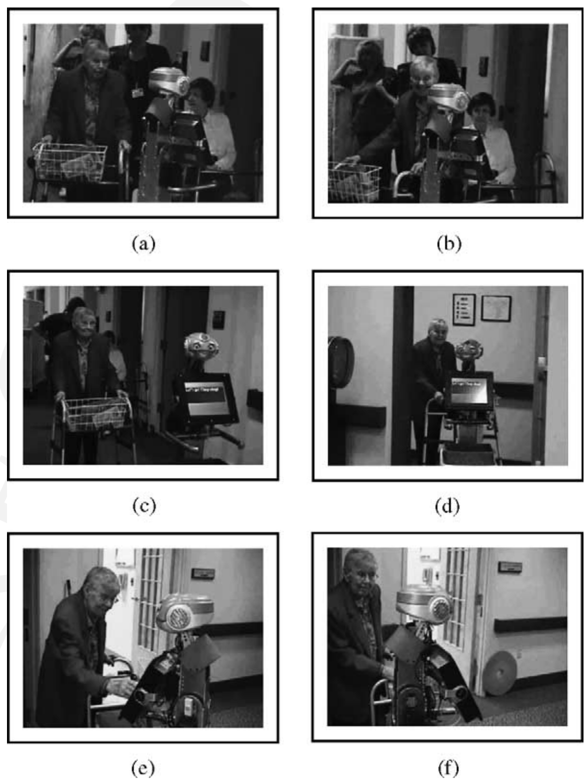


Fig. 7. Example of a successful guidance experiment: (a) Pearl picks up the patient outside her room; (b) reminds her of a physiotherapy appointment; (c) guides the person to the physiotherapy department; (d) enters the department; (e) satisfies a request for the weather report; (f) terminates the interaction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

UNCORRECTED PROOF

user. Fig. 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.

Each test subject received a short (approximately 5 min) training session with the robot, before completing the scenario. In all trials, the task was performed to completion, without any outside intervention. All reminders were successfully delivered (as confirmed through a touch-screen press by the user), and in all but one trial, the robot guided the subject to their appointment. The exception occurred when a test subject communicated to the robot that she did not require assistance, and the robot therefore appropriately returned to its home base rather than proceed with the guidance.

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, including spatial motion, touch-screen I/O, and speech output. Earlier trials with a poorly adjusted speech recognition system, and fixed velocity robot motion, both caused difficulties. These were addressed by increasing the role of the touch-screen, 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 a nursing home, where the robot autonomously provided reminders and guidance to elderly residents.

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. This was demonstrated in part by the robot's ability to complete the assigned task, but also by the fact that the response from the elderly participants was uniformly positive.

Second, this project also demonstrated the effectiveness of probabilistic tracking and decision making for interactive robots. Pearl is one of a few robots to use POMDPs, and the first to apply POMDP planning to the highest level of decision making. The ability to represent the uncertainty inherent in a person's behavior, and formulate plans accordingly, allowed the robot to robustly handle difficult situations, including noisy communication and crowded environments.

One of the key lessons learned while developing this robot is the imperative need for techniques that can cope with individual differences. This is especially true when designing robots for elderly users, which exhibit a great range of skills as a result of age-related decline. We had to make specific adjustments to accommodate varying walking speeds, voice levels, and auditory acuity.

Given the pressures of an aging population, we view the area of assistive technology as a prime source for great AI problems in the future.

Uncited reference

[20].

Acknowledgements

The authors wish to thank the many members of the Nursebot team for their invaluable contribution. In particular, the Autominder component was developed with the help of Laura Brown, Dirk Colbry, Colleen McCarthy, Cheryl Orosz, Bart Peintner and Ioannis Tsamardinos. The robot design and user studies benefited greatly from the suggestions of our Nursing and HCI colleagues: Jacqueline Dunbar-Jacobs, Sandra Engberg, Sara Kiesler, Francine Gemperle, Jennifer Goetz and Judith Matthews. This work was supported by the National Science Foundation Grant IIS-0085796.

References

[1] H. Akaike, A new look at statistical model identification, IEEE Transactions on Automatic Control 19 (1974) 716–723.

[2] J.L. Ambite, C.A. Knoblock, Planning by rewriting, Journal of Artificial Intelligence Research 15 (2001) 207–261.

UNCORRECTED PROOF

ARTICLE IN PRESS

[3] Y. Bar-Shalom, T.E. Fortmann, Tracking and Data Association, Academic Press, New York, 1998.

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

[5] W. Burgard, A.B. Cremers, D. Fox, D. Hähnel, G. Lakemeyer, D. Schulz, W. Steiner, S. Thrun, The interactive museum tour-guide robot, in: Proceedings of AAAI'98.

[6] D. Colbry, B. Peintner, M.E. Pollack, Execution monitoring with quantitative temporal Bayesian networks, in: Proceedings of AIPS'02.

[7] F. Dellaert, D. Fox, W. Burgard, S. Thrun, Monte Carlo localization for mobile robots, in: Proceedings of ICRA'99.

[8] T. Dietterich, The MAXQ method for hierarchical reinforcement learning, in: Proceedings of ICML'98.

[9] A. Doucet, N. de Freitas, N.J. Gordon (Eds.), Sequential Monte Carlo Methods in Practice, Springer, Berlin, 2001.

[10] A. Doucet, N. de Freitas, K. Murphy, S. Russell, Rao-Blackwellised particle filtering for dynamic Bayesian networks, in: Proceedings of UAI'00.

[11] M.M. Dowds, K. Robinson, Assistive technology for memory impairment: Palmtop computers and TBI, in: Proceedings of the Braintree Hospital Traumatic Brain Injury Neurorehabilitation Conference, 1996.

[12] G. Engelberger, Services, in: Handbook of Industrial Robotics, Wiley, New York, 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 (1998) 95–101.

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

[16] N.A. Hersh, L. Treadgold, Neuropage: The rehabilitation of memory dysfunction by prosthetic memory and cueing, NeuroRehabilitation 4 (1994) 187–197.

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

[18] 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.

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

[20] G. Lakemeyer (Ed.), Notes, in: Proceedings of the Second International Workshop on Cognitive Robotics, Berlin, 2000.

[21] R. Levinson, PEAT—the planning and execution assistant and trainer, Robotics and Autonomous Systems 23 245–252.

[22] C.E. McCarthy, M. Pollack, A plan-based personalized cognitive orthotic, in: Proceedings of AIPS'02.

[23] M. Montemerlo, J. Pineau, N. Roy, S. Thrun, V. Verma, Experiments with a mobile robotic guide for the elderly, in: Proceedings of AAAI'02.

[24] M. Pollack, Planning technology for intelligent cognitive orthotics, in: Proceedings of AIPS'02.

[25] M. Ravishankar, Efficient algorithms for speech recognition, Ph.D. Thesis, School of Computer Science, Carnegie Mellon University, 1996.

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

[27] D. Schulz, W. Burgard, D. Fox, A. Cremers, Tracking multiple moving targets with a mobile robot using particles filters and statistical data association, in: Proceedings of ICRA'01.

[28] S. Singh, M. Kearns, D. Litman, M. Walker, Reinforcement learning for spoken dialogue systems, in: Proceedings of NIPS'00.

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

[30] I. Tsamardinos, Constraint-based temporal reasoning algorithm with applications to planning, Ph.D. Thesis, Intelligent Systems Program, University of Pittsburgh, 2001.

[31] Health and Aging Chartbook, Health, US Department of Health and Human Services, United States, 1999.

Joelle Pineau is a Ph.D. candidate in Robotics at Carnegie Mellon University. She received her B.A.Sc. (1998) in systems design engineering from the University of Waterloo. Her research interests are in artificial intelligence and robotics, and more specifically in developing probabilistic planning techniques to control robots under uncertainty.

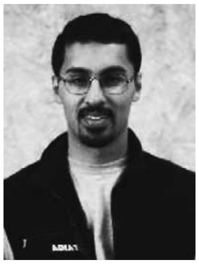
Michael Montemerlo is a doctoral student at the Robotics Institute, Carnegie Mellon University. He received his B.S. and M.S. in electrical/computer engineering in 1997 from Carnegie Mellon University. His research interests include simultaneous localization and mapping and people tracking.

Martha Pollack is Professor of Computer Science and Engineering at the University of Michigan. She was previously on the Faculty of the University of Pittsburgh and the research staff of the Artificial Intelligence Center, SRI International. Pollack, who received her Ph.D. from the University of Pennsylvania in 1986, is a fellow of the American Association for Artificial Intelligence, and a recipient of

UNCORRECTED PROOF

ARTICLE IN PRESS

an NSF Young Investigator's Award, the Computers and Thought Award, and the University of Pittsburgh Chancellor's Distinguished Research Award. She is currently the executive editor of the "Journal of Artificial Intelligence Research". Her research interests are in automated plan management, constraint-based temporal reasoning, and the design of cognitive orthotic systems. 24



Nicholas Roy is a Ph.D. candidate in robotics at Carnegie Mellon University. He received his B.Sc. in physics and M.Sc. in computer science from McGill University. His research interests include mobile robot navigation and exploration, human–robot interaction, speech dialogue management, probabilistic reasoning and machine learning.

25

Sebastian Thrun is the Finmeccanica Associate Professor of Computer Science and Robotics at Carnegie Mellon University, and one of the founders of the Nursebot project. Thrun pursues research in artificial intelligence, machine learning, and robotics.

UNCORRECTED PROOF