POMDP-based Long-term User Intention Prediction for Wheelchair Navigation

Tarek Taha, Jaime Valls Miró and Gamini Dissanayake

Abstract— **This paper presents an intelligent decision-making agent to assist wheelchair users in their daily navigation activities. Several navigational techniques have been successfully developed in the past to assist with specific behaviours such as "door passing" or "corridor following". These shared control strategies normally require the user to manually select the level of assistance required during use. Recent research has seen a move towards more intelligent systems that focus on forecasting users' intentions based on current and past actions. However, these predictions have been typically limited to locations immediately surrounding the wheelchair. The key contribution of the work presented here is the ability to predict the users' intended destination at a larger scale, that of a typical office arena. The systems relies on minimal user input - obtained from a standard wheelchair joystick - in conjunction with a learned Partially Observable Markov Decision Process (POMDP), to estimate and subsequently drive the user to his destination. The projection is constantly being updated, allowing for true user-platform integration. This shifts users' focus from fine motor-skilled control to coarse control broadly intended to convey intention. Successful simulation and experimental results on a real wheelchair robot demonstrate the validity of the approach.**

I. MOTIVATION

The world's aging population and the large number of people effected by motor disabilities has motivated past researchers to develop assistive technologies to give otherwise immobile people freedom of movement, dramatically increasing independence and improving the quality of life of those affected. Systems such as robotic walkers [1], smart blind sticks [2] and robotic wheelchairs [3]–[8] have been developed with this goal in mind. Out of these, robotic electric wheelchairs are particularly desirable given their social acceptance and ubiquity. Yet depending on the users' type of disability, safely and effectively driving the wheelchair may be difficult in general, for example people with severe tremors. Furthermore, wheelchairs are also large in comparison to the passageways typical of indoors environments such as offices, nursing homes, hospitals and the home environment. This means that for some users, apparently trivial tasks such as passing through a doorway or navigating a hallway may be quite challenging. This work proposes an intelligent driving system designed to assist these users by understanding and complying with their intentions. The interaction takes place transparently to the user, demanding a minimal input from them to automatically perform the required fine motion control for the given situation.

This work is supported by the Australian Research Council (ARC) through its Centre of Excellence programme, and by the New South Wales State Government. The ARC Centre of Excellence for Autonomous Systems (CAS) is a partnership between the University of Technology Sydney, the University of Sydney and the University of New South Wales.

All authors are with the ARC Centre of Excellence for Autonomous Systems (CAS), Faculty of Engineering, University of Technology Sydney (UTS), NSW2007, Australia {t.taha,j.vallsmiro,g.dissanayake}@cas.edu.au

II. INTRODUCTION AND RELATED WORK

If a truly user-machine integrated system is to be developed, the type of cooperation between user and machine must be comparable to the cooperation between a horse and its rider [9]: the rider navigates, while the horse avoids (small) dangerous obstacles on the ground. To achieve this level of user-machine integration the machine must grow and learn with the user so that a relationship may form (such as that between a horse and its rider) and so that the machine can predict the users intention and autonomously enact the intention with only minimal corrective input from the user.

The fundamental component of this relationship is intention recognition. The primary considerations for an intention recognition system are whether an accurate representation/model of the environment is required and whether the system is going to be reactive or deliberative. Reactive refers to systems that do not use a representation of the environment and therefore are usually weak in decision making and long-term prediction. They rely on local or temporal information collected on-line which might not be sufficient to develop correct long-term plans. They establish a direct link between the perceptions obtained by their sensors and their effectors; the control doesn't comply to a model but simply happens as a low level response to the perception. Systems with limited resources like processing power, memory and communication mediums often use a reactive system but the scope of their possible applications and intelligence is limited. Several wheelchair platforms such as Rolland III [4], Bremen Autonomous Wheelchair, Sharioto [3], RobChair [5], Senario [6], VAHM [7], Wheelesley [8], and Navchair [10] employ reactive control algorithms, limited by either the set of operating modes that the user must select (e.g. manual, fully autonomous or semi-autonomous), and/or by the limited scope of their navigation algorithms, reduced to a local scale.

In the last few years some wheelchair assistive reactive techniques have emerged which overcome some of these restrictions by capturing the users local intentions in order to facilitate a limited set of tasks such as avoiding obstacles, following a wall, entering a room, going towards an object or other local area of interest. These algorithms are based on systems that can *act intelligently* but not *think intelligently*. In other word, they try to make the link between perception and action as direct as possible by combining decisions made by both the human and the machine [9], [10]. Even though such techniques appear to work well in predicting the user's intentions at a local scale (same room or same open space), they lack the cognitive capabilities to autonomously recognize them; rather, those must be manually specified prior to system operation. Moreover, they lack the markedly higher scope of assistance that would support specific user's activities with the appropriate sequence of actions beyond the local boundaries, such as going to the bathroom or out the main door.

This paper presents an intention recognition and goal prediction assistance strategy that uses environmental knowledge to plan and interact with the user in a deliberate manner, but at a larger scale. Typically, disabled users requiring wheelchair assistance have a known set of target locations that they go to during their daily activities, such as the bathroom, kitchen or T.V. room. Through monitoring a wheelchair user through his daily routine/activities, the technique hereby proposed first determines the locations of interest that the user regularly frequents and builds knowledge about these locations using machine learning techniques. This knowledge is then used to predict the users intended destination from sensed inputs, as derived from a POMDPs model. In this scheme of things, the wheelchair is considered an intelligent agent with an internal representation of the environment which effectively translates these target destinations into a plan of action to reach them in the presence of uncertainties. An intelligent controller subsequently performs the lower level navigational tasks such as local path planning, collision avoidance and actuating motion control. It is important to emphasize that whilst in motion the user remains in complete control of the system, providing continuous (or discrete) action/course correcting feedback to the system through the intention recognition algorithm.

III. The POMDP Model

Partially Observable Markov Decision Processes (POMDP) provide a general framework for sequential decision making in environments where states are hidden and actions are stochastic. POMDPs were recently used in assistive applications and they proved to be reliable and efficient when modelled properly [11], [12]. A POMDP model represents the dynamics of the environment, such as the probabilistic outcomes of the actions (the transition function T), the reward function R, and the probabilistic relationships between the agents observations and the states of the environment (the observation function O). In POMDP terminology, system states are typically referred to as "hidden states", since they are not directly observable by the agent. The POMDP framework is a systematic approach that uses belief states to represent memory of past actions and observations. Thus, it enables an agent to compute optimal policies using the MDP framework, as depicted in Fig. 1.

A POMDP model is defined by $< S, A, T, R, Z, \gamma, O >$, a seven tuple where:

- S: A set of states that represents the state of the system at each point in time.
- A: A set of actions that an agent can take (can depend on the current state).
- T: $A \times S \times S \rightarrow [0, 1]$: The state transition function, which maps each state action pair into a probability distribution over the state space. The next distribution over the state space depends only on the current state action pair and not on the previous state action pairs. This requirement ensures the $Markovian\ property$ of the process. We define $T(s, a, s')$ as the probability that an agent took action a from state s and reached state s'.
- R: $S \times A \rightarrow \Re$: The immediate reward function which indicates the reward for doing an action in some state.
- Z: A set of observations.
- γ : A discount factor used to reduce the award given to future (and more uncertain) steps.
- O: $A \times S \times Z \rightarrow [0, 1]$: A function that maps the action at time t-1 and the state at time t to a distribution over the observation set. We define $O(s', a, z)$ as the probability of making observation z given that the agent took action a and landed in state s'.

The belief is a sufficient statistic for a given history and it is updated at each time step according to 1, where $Pr(o|a, b)$ is a normalizing constant [13], [14].

$$b'(s') = \frac{O(s', a, o) \sum_{s \in S} T(s, a, s') b(s)}{Pr(o|a, b)} \quad (1)$$

Given a POMDP model, the goal is to find a sequence of actions, or policy, $a0, ..., at$ that maximizes the expected sum of rewards $E[\sum_t \gamma^t R(s_t, a_t)]$. Since the states are not fully observable, the goal is to maximize the expected reward for each belief [15]. The function $V^*(s)$ that solves the Bellman equation (2) is called the value function, and its associated optimal policy can be formulated using (3).

$$V^*(s) = max_a[R(s, a) + \gamma \sum_{s' \in S} T(s, a, s') V^*(s')] \quad (2)$$

$$\pi^*_t = argmax_a[R(s, a) + \gamma \sum_{s' \in S} T(s, a, s') V^*_{t-1}(s')] \quad (3)$$

IV. POMDP Problem Specification

Within this POMDP framework, our intention recognition problem is transferred into a planning problem where the wheelchair is transformed into a decision maker agent required to find the best plan (optimal policy) that represents the user's intention by reducing the uncertainty in the belief state, categorized by the $Destination$ the user is trying to reach. The state space is described by the cross product of two features, the $WchairLocation = \{s1, ..., sx\}$ and the $Destination = \{d1, ..., dy\}$ resulting in a $StateSpace = \{s1d1, s2d1, ..., sxdy\}$. The wheelchair starts from a known position and the plan finishes when the $WchairLocation$ is the same as the $Destination$. The $Wchair$ can have one of the following actions:

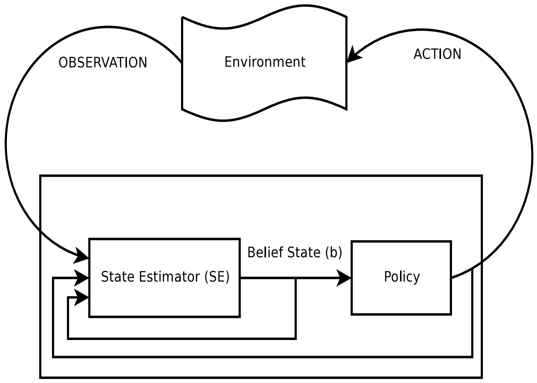

Fig. 1. A POMDP agent is made up of two main components. The state estimator module receives observations from the environment, the last action taken and the previous belief state and produces an updated belief state. The policy module maps belief state to an action.

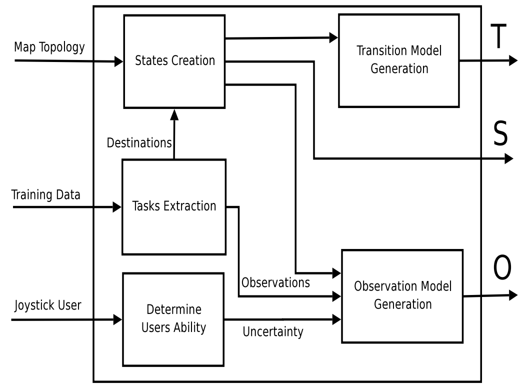

Fig. 2. The POMDP model generation architecture. The map topology together with the training data are used to determine the transition model. The training data is also used to determine the observation model of the POMDP. User's Joystick calibration determines the uncertainty in the observations.

$\{North, South, East, West, DoNothing\}$ indicating the global direction of travel. A reward of -1 is given for each motion step and +100 reward is given when the $Wchair$ performs an action that leads to $Destination$. It is assumed the $WchairLocation$ is fully observable via a localizer, but not the destination, and the effect of an action has a predictable deterministic effect as the example described by (4):

$$Pr(Wchair = Sx|Wchair = Sy, South) = 1 \qquad (4)$$

The position of the $Destination$ is unobservable until the wheelchair reaches its destination. At each state the joystick input is observed and is represented by a set of discrete states = $\{Up, Down, Right, Left, NoInput\}$, and the uncertainty in the user's input is taken into consideration while generating the observation model (further explained in section V-C).

V. POMDP GENERATION

To obtain a efficient POMDP system, we need to have proper $Transition$, $Observation$ and $StateSpace$ models. Our model generation consists of three major parts as depicted in Fig. 2. These three steps will be explained in the subsections below:

A. State Space

In our assistive system, we want the user to be able to navigate in a high level topological manner. This means that the user should be focusing on driving the wheelchair from one room to another, or from one spatial location to another without having to worry about the intermediate steps that comes in between (planning wise). In order for us to do so, only significant spatial feature are considered, such as a hallway intersection, a door opening or a room.

The ability to learn tasks and represent environments [16], [17] is essential in our application as it creates the bases for the long term intention recognition and prediction. This is done by simplifying the encapsulation of spatial and activity information. For this to happen, the wheelchair should have the ability to represent the spatial information

TABLE I
LIST OF TASKS RECORDED FROM THE USER'S ACTIVITIES

	Start	End	Path
Task1	Lab	Office	26/D - 25/L - 24/L - 22/D - 23/N
Task2	Office	Meeting	42/U - 40/L - 43/U - 44/N
Task3	Office	Bathroom	3/D - 4/L - 5/D - 6/N

of the environment in a simplistic topological manner that can make it easier to store, extract and update information.

For our POMDP platform, the state space consists of two features: the $WchairLocation$ and the intended $Destination$. The cross product of the above two feature will form the $StateSpace = \{s1d1, s2d1, ..., sxdy\}$, these features are separately extracted in two different steps described below:

1) Spatial States: The spatial representation we are using is based on the topological graph representation of the environment, where vertices are locations in the environment and edges represent a viable path connecting two locations as a result of performing an action. In our research we are mainly targeting indoor office or home environments. For such environments there has been a lot of research done on how to build maps and extract topological representation accurately. For simplicity, we assume that the maps are already available and that the topological map representation is hand coded and programmed. It might be more convenient in the future to consider a complete system that can build maps and extract topological representations simultaneously but this is out of the scope of the current research. The map topology will be represented by a graphical tree of nodes and connections (segments), where the set of nodes $WchairLocation = \{s1, ..., sx\}$ represents a location in the map and the connection represents a physical path that connects two locations. The hand coded spatial configuration of the domain used for planning illustrated in Fig. 3.

2) Destinations States: Identifying places of interest is not an easy task and there is no direct method to achieve this as it is an application and environment dependent problem. For the prediction problem we are trying to solve, it's sufficient

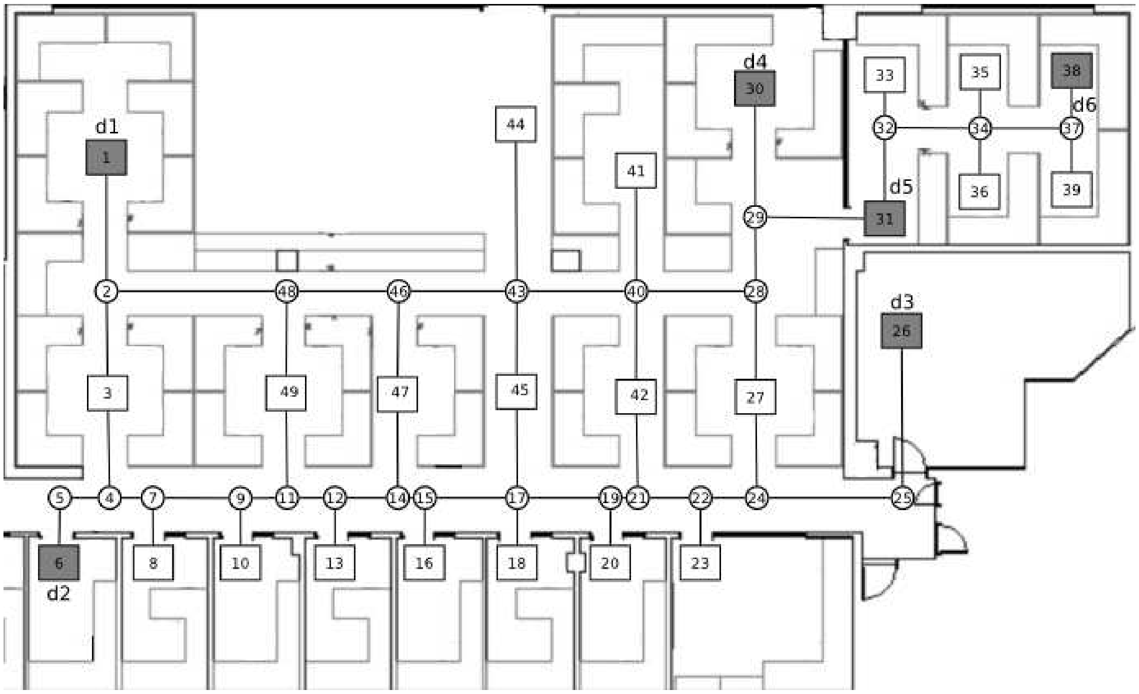

Fig. 3. The map topology used for our intention recognition. Circles represent intersections and cannot be a destination while squares represent rooms or open spaces and can be considered as a possible destination. The numbers inside the circles represent the state number and is used to build the transition model. Gray shaded rectangles represent learned destinations.

to think about the place of interest as a spatial location in the environment where the user spends significantly most of his/her time. After observing the user's activities we can determine the time that the user need to stay in the same place for it to be considered as a place of interest. In general staying few minutes in a certain location can nominate that location to the place of interest set. For POMDP model generation purposes we log the activities of the user over a period of time, then in that log we determine the locations of interest $Destination = \{d1, ..., dy\}$ based on the above criteria.

B. Transition Model

Transition model specifies the translation from one state to another given a certain action $T(s, a, s')$. In our model specifications, the actions are global navigation commands $\{North, South, East, West, Stop\}$ and determines the spatial node that we will end up at if we are in location s and executed the action a. The transition mode is built directly from the map topology. This transition is deterministic and independent of the intention, i.e., it is derived regardless of where the user wants to go. The result of executing an action in the same location will be the same. For example $T(s3d1, North, s2d1)$ = $T(s3d2, North, s2d2)$ = 1 in Fig. 3.

C. Observation Model

The observation model defines the probability of observing z given that the wheelchair took an action a and landed in state s' $O(s', a, z)$. To generate a proper observation model that correctly models the user's intention and activities, we use a training data from that particular user. In an indoor environment, the wheelchair users usually perform a repetitive set of tasks that represents navigating from one place to another. A task can be for example going from the living room to the bathroom or to the kitchen. This set of tasks can be defined by the user himself or extracted from a set of data recorded by monitoring the user's activities. The tasks are defined by a starting location, intermediate locations, end location and the joystick inputs/observation that the user gave at each location as described in Table I where the path is represented by numbers corresponding to the states' numbers and the letters corresponding to the observation in each state (L=Left, R=Right, U=Up and D=Down).

The user in many cases might be unable to give a proper joystick input due to disability or a disease causing a shaky hand for example. To best customize the POMDP model for this user, a joystick calibration is required to determine the uncertainties in the user's inputs. This uncertainty will be a set of n probabilities describing the user's inability to give the right joystick input, where n is the number of $JoystickInputs = \{Up, Down, Right, Left, NoInput\}$.

Having obtained the training data and the uncertainty, the observation model is then generated by adding the uncertainty to the frequency probability (the probability of obtaining a certain observation in a state).

VI. ON-LINE ASSISTANCE

Once the planning problem is formulated, we solve the POMDP to get the optimal policy π^*. While predicting on-line, we start with an initial belief state b_t. Since we

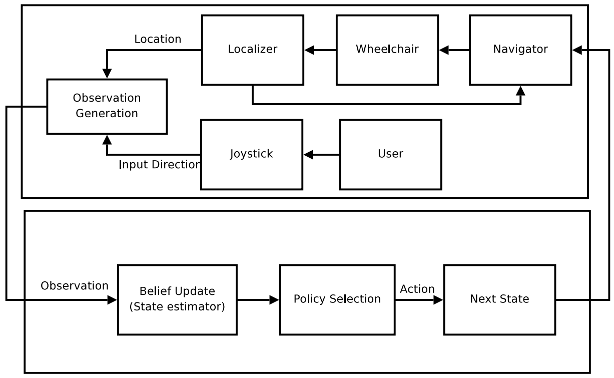

Fig. 4. The POMDP driver assistance architecture. The user's input together with the current location generate an observation that helps in updating the belief in the destination. The appropriate action will be selected based on that belief, and the next state will then be determined and given to the navigator to drive the wheelchair to the next state.

Step	1	2	3	4	5	6
Wheelchair Location	S22	S24	S27	S28	S29	S30
Predicted Destination						
d1	0.725%	0.957%	0.971%	0.027%	0%	0%
d2	0.725%	0.027%	0%	0%	0%	0%
d3	24.63%	13.56%	0.019%	0%	0%	0%
d4	24.63%	32.55%	33%	33.32%	92.3%	100%
d5	24.63%	32.55%	33%	33.32%	3.85%	0%
d6	24.63%	32.55%	33%	33.32%	3.85%	0%
Observation	Right	Up	Up	Up	Up	NoInput
Action Selected	East▶	North▲	North▲	North▲	North▲	Stop■
Translate to State	S24	S27	S28	S29	S30	S30

Fig. 5. The result of a real experiment. The wheelchair starts in state 22 and tries to predict where the user is going to based on his joystick inputs (observations). The wheelchair in this case successfully takes the user's joystick inputs and decides on the correct actions that take the user to state 30.

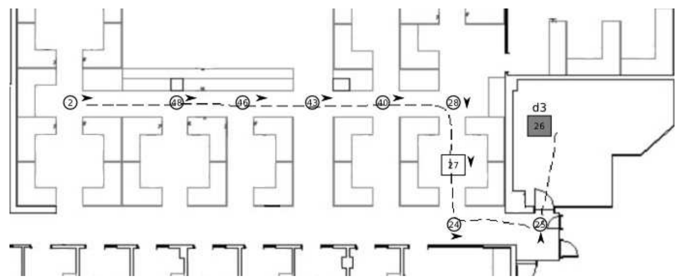

Fig. 6. The result of a real Wheelchair experiment showing the path (dashed line) and observations (arrows). The wheelchair starts in state 2 and drives the user successfully to state 26 by updating the belief at each step from the obtained observation.

know our current location from our localizer, the initial belief is limited to those states in the $StateSpace$ with our current location, and we will end up with a belief set size equivalent to the available destinations. For example, if our $Destination = Kitchen, Bathroom, T.V Room$ and we know where we are, then our initial belief is distributed among these destinations and is equal to $1/3$. Based on our initial belief, we execute the optimal policy action for that belief state $\pi^*(s_t)$, calculate the reward r_t for taking that action, get an observation z_{t+1} and update our belief b_{t+1}, then repeat the procedure. This is described in Procedure 1 and illustrated in Fig. 4.

Procedure 1 On-line Navigation

1. Initial belief: b_t.
2. Execute the action from the optimal policy: $\pi^*(s_t)$.
3. Calculate the reward: r_t.
4. Get an observation : z_{t+1}.
5. Update the belief: b_{t+1}.
6. Repeat until destination reached.

VII. Experimental Results

To validate the proposed intention recognition architecture we simulated a training data that represents the activities of a user in the environment shown in Fig. 3. The destinations are represented by the gray shaded squares and they form the set $Destination = \{s1d1, s6d2, s26d3, s30d4, s31d5, s38d6\}$. The POMDP was generated using a simulated training data with uncertainty added to the observations to represent the user's ability to control the joystick (in this example uncertainty on Up=10%, Down=5%, Right=15%, Left=10% and Nothing=20%). The generated POMDP problem was then solved using zmdpSolver [18] and the optimal policy was obtained.

The generated policy and model were tested against the tasks in the training data. For each task in the training data we start with a known location (the first state in the task) but unknown destination (equal belief among destinations) then we take observations from that task one by one, update the belief based on these observations, select an action based on the optimal policy and execute that action to get to the next state. This procedure is repeated until we reach the end of the observations in the task. If the end state reached after the last observation is the same as the intended destination (the last state in the task), then the test is considered successful, otherwise it fails. The test was successful in all of the 289 tasks in this experiment producing a 100% success rate.

An example of a navigation task on a real wheelchair platform is shown in Fig. 5. The wheelchair used was the one described in [19] and it measures 1.2x0.7m. The wheelchair's size is considered large compared to the environment and driving it in such a constrained environment can be a challenging task for inexperienced users or users with severe tremors. In this example, the user was giving observations at each state to indicate where he wants to go. Initially, the user can be going to any of the pre-determined destinations, therefore the belief is uniformly distributed among them. With the first observation, the belief is updated and the next state is determined based on the appropriate selected action and the wheelchair navigates to that state autonomously. This is repeated until the user reaches his destination.

A longer real wheelchair navigation example can be seen in Fig. 7. The path followed and the observations obtained

Step	1	2	3	4	5	6	7	8	9	10
Wheelchair Location	S2	S48	S46	S43	S40	S28	S27	S24	S25	S26
Predicted Destination										
d1	0.727%	0.022%	0%	0%	0%	0%	0%	0%	0%	0%
d2	0.364%	0.011%	0%	0%	0%	0%	0%	0%	0%	0%
d3	24.72%	24.99%	25%	25%	25%	92.68%	99.79%	99.99%	100%	100%
d4	24.72%	24.99%	25%	25%	25%	2.43%	0.069%	0.002%	0%	0%
d5	24.72%	24.99%	25%	25%	25%	2.43%	0.069%	0.002%	0%	0%
d6	24.72%	24.99%	25%	25%	25%	2.43%	0.069%	0.002%	0%	0%
Observation	Right	Right	Right	Right	Right	Down	Down	Right	Up	NoInput
Action Selected	East▶	East▶	East▶	East▶	East▶	South▼	South▼	East▶	North▲	Stop■
Translate to State	S48	S46	S43	S40	S28	S27	S24	S25	S26	S26

Fig. 7. The results of the navigation experiment depicted in Fig. 6.

are those illustrated in Fig. 6. The same procedure as the one described above is used and again the sequence of observations help the system to successful drive the user to his/her destination.

VIII. Conclusion

In this paper we have presented a new method for wheelchair assistance that considers the wheelchair as a smart robotic agent, interacting with the user with the aid of a sequential decision making algorithm (POMDP). Unlike most of the currently available assistive methods that are based on semi-autonomous systems which merge wheelchair's perception and user's control with some added heuristics, our method tries to predict where the wheelchair's user is trying to go, and takes him there without any extra mode or behavioural selection. POMDP was chosen because it provides a good platform for planning and predicting under uncertainty for human-robot interaction, as we have shown in this paper. The results we have obtained so far from the simulated and real platform tests are promising and they validate our method. Our efforts in the future will be devoted to further enhance the capabilities and the intelligence of the system through automated activity monitoring and tasks extraction.

References

[1] M. Alwan, P. J. Rajendran, A. Ledoux, C. Huang, G. Wasson, and P. Sheth. Stability margin monitoring in steering-controlled intelligent walkers for the elderly. In *Proceedings of AAAI Symposium*, 2005.

[2] S. Kang, Y. Ho, and I. Hyuk Moon. Development of an intelligent guide-stick for the blind. *Proceedings of IEEE International Conference on Robotics and Automation*, 4:3208–3213, 2001.

[3] D. Vanhooydonck, E. Demeester, M. Nuttin, and H. V. Brussel. Shared control for intelligent wheelchairs: an implicit estimation of the user intention. In *Proceedings of the International Workshop on Advances in Service Robotics*, 2003.

[4] C. Mandel, K. Huebner, and T. Vierhuff. Towards an autonomous wheelchair: Cognitive aspects in service robotics. In *Proceedings of Towards Autonomous Robotic Systems*, pages 165–172, 2005.

[5] G. Pires, R. Araujo, U. Nunes, and A. T. de Almeida. Robchair-a powered wheelchair using a behaviour-based navigation. *Proceedings of International Workshop on Advanced Motion Control*, pages 536–541, 1998.

[6] P. Beattie and J. M. Bishop. Localization of the senario wheelchair. In *Proceedings of MobiNet Symposium Mobile Robotics Technology for Health Care Services*, pages 287–293, 1997.

[7] A. Pruski, M. Ennaji, and Y. Morere. Vahm: a user adapted intelligent wheelchair. *Proceedings of the International Conference on Control Applications*, 2:784–789, 2002.

[8] H. A. Yanco. Development and testing of a robotic wheelchair system for outdoor navigation. In *Proceedings of the Conference of the Rehabilitation Engineering and Assistive Technology Society of North America*, 2001.

[9] K. A. Tahboub. Intelligent human-machine interaction based on dynamic bayesian networks probabilistic intention recognition. *Journal of Intelligent and Robotic Systems*, 45(1):31–52, 2006.

[10] E. Demeester, A. Huntemann, D. Vanhooydonck, G. Vanacker, A. Degeest, H. V. Brussel, and M. Nuttin. Bayesian estimation of wheelchair driver intents: Modeling intents as geometric paths tracked by the driver. *International IEEE/RSJ Conference on Intelligent Robots and Systems*, pages 5775–5780, Oct. 2006.

[11] M. Lopez, L. M. Bergasa, R. Barea, and M. S. Escudero. A navigation system for assistant robots using visually augmented pomdps. *Autonomous Robots*, 19(1):67–87, 2005.

[12] J. Hoey, A. V. Bertoldi, P. Poupart, and A. Mihailidis. Assisting persons with dementia duing handwashing using a partially observable markov decision process. In *Proceedings of the International Conference on Vision Systems (ICVS)*, Biefeld, Germany, 2007.

[13] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra. Planning and acting in partially observable stochastic domains. *Artificial Intelligence*, 101:99–134, 1998.

[14] J. Pineau and G. Gordon. Pomdp planning for robust robot control. In *Proceedings of International Symposium on Robotics Research (ISRR)*, 2005.

[15] J. Pineau, G. Gordon, and S. Thrun. Point-based value iteration: An anytime algorithm for pomdps. In *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1025–1032, August 2003.

[16] T. Spexard, S. Li, B. Wrede, J. Fritsch, G. Sagerer, O. Booij, Z. Zivkovic, B. Terwijn, and B. Krose. Biron, where are you? enabling a robot to learn new places in a real home environment by integrating spoken dialog and visual localization. *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, pages 934–940, Oct. 2006.

[17] M. Heerink, B. J. A. Krose, B. J. Wielinga, and V. Evers. Human-robot user studies in eldercare: Lessons learned. June 2006.

[18] T. Smith. zmdpsolver. [http://www.cs.cmu.edu/˜trey/zmdp/].

[19] T. Taha, J. V. Miro, and D. Lui. An efficient path planner for large mobile platforms in cluttered environments. In *Proceedings of IEEE Conference on Robotics, Automation and Mechatronics*, pages 225–230, 2006.