POMDP Approach to Robotized Clothes Separation

Pol Monsó, Guillem Alenyà, Carme Torras

Abstract— Rigid object manipulation with robots has mainly relied on precise, expensive models and deterministic sequences. Given the great complexity of accurately modeling deformable objects, their manipulation seems to call for a rather different approach. This paper proposes a probabilistic planner, based on a Partially Observable Markov Decision Process (POMDP), targeted at reducing the inherent uncertainty of deformable object sorting. It is shown that a small set of unreliable actions and inaccurate perceptions suffices to accomplish the task, provided faithful statistics on both of them are collected beforehand. The planner has been applied to a clothes sorting task in a real case context with a depth and color sensor and a robotic arm. Experimental results show the promise of the approach since more than 95% certainty of having isolated a piece of clothing is reached in an average of four steps for quite entangled initial clothing configurations.

I. INTRODUCTION

This paper presents a planning approach to solve the task of handling deformable objects with uncertain perceptions and inaccurate actions. In particular, the approach is applied to a pile of pieces of clothing, from where a robot has to successfully remove one piece of clothing at a time. Manipulation of deformables, like textiles, is a challenging problem as the shape of the objects can change dramatically after the execution of manipulation actions, occlusions and self-occlusions commonly appear, and entanglements lead to undesired manipulation of multiple objects. Perception of deformables is an open problem [1] and can be computationally very expensive [2].

The main contribution is a planning framework that does not explicitly work with deformable object descriptions but instead follows an action policy that takes into account the uncertainty of the underlying manipulation and perception. The main idea is that the initial state is unknown (i.e. the number of objects on the pile or their distribution), and by performing manipulation actions this uncertainty is reduced (i.e. one piece is isolated on the table). Once a certain degree of confidence is reached the textile can be removed from the table, but not before. Specifically, the goal of the robot task is to remove each textile from the table with the minimum number of actions possible while minimizing the errors. Specially, removing more than one piece of clothing at once is a failure. Note that separation is a necessary previous step on a variety of tasks, like identification [1], [3] and folding [4].

II. PREVIOUS WORK

Research on robotic laundry manipulation in relatively unrestricted scenarios is rather new. For socks manipulation, Wang et al. [5] propose a complex vision system to identify sock type and if the sock is inside-out. The manipulation system relies on deterministic motions using very specialized tools. Maitin-Shepard et al. [6] use a robot to take towels from a pile and fold them. Grasping is performed at the central point and vision detects corners through background segmentation and stereo. Finally, a pre-defined set of actions is used to fold the towels.

Alternatively, a more generic approach is to recognize and classify different pieces of cloth. Willimon et al. [3] propose a system where the grasping operation is repeated as many times as necessary to ensure a correct grasp, and then four basic visual features are extracted to classify a previously learned object. Cusumano et al. [4] use a robot to bring a piece of clothing to a specific configuration using predefined grasping actions by an arbitrary edge, and a series of low-hanging-point re-grasps.

We propose here to use a planning framework to select actions. Note that we consider that perceptions can be very noisy and therefore methods assuming full observability are insufficient. This will be demonstrated in the experimental section (Sec. IV) through the comparison of our approach with a classical MDP and the characterization of the noise in the perceptions.

Problems where perceptions are hidden or inaccurate can be handled in two ways: without knowing the world model beforehand, that is, by learning the model and/or model parameters (Hidden Markov Models (HMM), Predictive State Representations (PSR), ...), or otherwise relying on a model as in the partially observable MDP (POMDP) approach used in this paper.

Nonetheless, techniques exist that learn the underlying world POMDP model and its transition probabilities from data. Chrisman [7] addressed this problem with a variation of the Baum-Welsh algorithm and later, based on HMM modelling, successfully retrieved and applied a partially observable Markov model to speech recognition [8]. Other techniques have also been proposed to derive the model from a data set, Roy et al. [9], for example, use Principal Component Analysis (PCA) to reduce the dimensionality of the state space and identify the valuable features. In our case, the model is clear enough to be designed manually, and its transition probabilities are derived empirically based on a ground truth. Note that the accuracy of the model is a required key point of the formulation, as we will see when

| (a) Grasping | (b) Lifting | (c) Leaving on the other side | (d) Leaving on the basket |

Fig. 1. Experimental setup. The robot provides a set of actions to pick up and remove pieces of clothing by combining 3D color information from the Kinect camera with basic robot movements. (a)-(b)-(c) Objects are moved from one side to the other side of the table using different actions to gather information and minimize the uncertainty about the state. Actions possibly fail when no object or several objects are grasped at the same time. (d) When the uncertainty about the isolation of one object in one side of the table is below a certain threshold the *remove* action takes place.

we evaluate the results.

As we have pointed out, our approach is based on a Partially Observable Markov Decision Process formulation. This formulation has been successfully applied to many robotic applications ranging from robot navigation [9], [10] to perception [11], among other applications such as guidance of patients suffering from Alzheimer [12]. Moreover, the recent performance results of point-based solvers over high-dimensional POMDPs are encouraging [13], [14], given that they relax the dimensionality constraints up to several thousands of states. In particular, for this paper, the experimental computation of the POMDP solution has been derived with the SARSOP point-based solver [15].

Nonetheless, to our knowledge, POMDP planning has never been applied to object manipulation other than with a rigid object and tactile perceptions [16].

III. PROBLEM FORMULATION

The standard formulation of POMDPs defines the problem by the tuple $\{S, A, Z, T, \Omega, R, \gamma, b_0\}$, where S is a set of discrete states modelling the world, A is a set of possible actions, and Z is a set of possible observations providing partial state information. The probability distribution $T(s, a, s')$ establishes the probability of reaching state s' from state s by performing action a, while $\Omega(s, a, z)$ establishes the probability of observing z when reaching state s after action a. $R(s, a, s')$ represents the reward received when reaching state s' by performing action a from state s. Finally, γ is the discount factor applied to the rewards and b_0 the initial probability distribution over the states, also known as *belief*, which is updated with the Bellman equation. The goal of POMDP solvers is to obtain the optimal policy $\pi^*(s, a)$ that maximizes the reward.

For further insights, Kaelbling et al. [17] provides a thorough descriptive review of the POMDP formulation.

In our particular problem, the chosen definition of the POMDP tuple is as follows:

- S: The number of objects in each region of the table, with a *catch-all* state >3.
- A: Two different sets of actions: *remove* actions take an object out of the table; *move* actions take objects from one side to the other side of the table. To obtain a rich action set it is desirable that *move* actions perform differently depending on the number of objects and the amount of deformation on the objects.

- Z: The number of objects seen in each area. Note that, for the same state perceptions can be very different due to occlusions and perception errors.
- R : -100 for removing several objects at the same time, $+1$ for removing a single object, $+5$ for removing the last object, and -1 otherwise.
- γ: The discount factor is set to 0.95.
- Ω : The probability distribution of the number of objects seen given a number of objects present in each area.
- T: The probability of reaching one state from another given an action.

We next discuss the most important aspects of this formulation in detail.

A. State definition (S)

We consider 2 zones: the table and the basket where clothes should be transferred one by one. The table is divided in two different areas {left,right}, where there are an unknown number of pieces of cloth. The state is defined as a tuple $< N_x, N_y >$ that codifies the number of pieces of clothing in each side with $N_x, N_Y \in \{0, 1, 2, 3, >3\}$. As a simplification, we consider that in the initial state there are pieces of clothing only in the right side.

Given the proposed state definition, the planning is executed at a symbolic level, delegating part of the intelligence to the actions and consequently permitting this reduction of the dimension of the state space. In the deformable object context, this reduction is specially important given the great variability of the world. By transferring responsibility to the actions, the dimensionality of the POMDP becomes tractable. We will show that, in spite of this simplification, the planning is successful, since the consequences of such simplification are handled by the POMDP through the action success uncertainty.

B. Actions (A)

Actions have been defined in two different groups: 2 specialized actions to *remove* objects from the table and 20 actions to *move* objects from one side of the table to the other side with the goal of isolating one piece of clothing in one side. Both action groups involve 3 motions of the robot: grasping, lifting and leaving (See Fig. 1).

Grasp in *remove* actions is a specialized grasp with high probability of success when only one piece of clothing is

(a) Original Image   (b) Highest point grasp   (c) Most Wrinkled grasping point   (d) Height modification
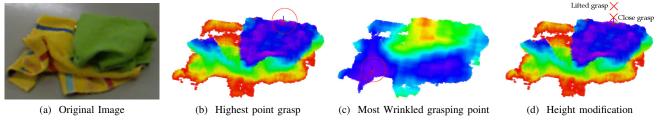
Fig. 2. The actions apply two different grasping point selection criteria to the original point cloud image. (2a) original image. (2b) the highest point relative to the table. (2c) the most wrinkled point. (2d) The grasp point is selected to be close or lifted, which leads to coarse or precise actions.

(a) Finger configuration 1   (b) Finger configuration 2   (c) Finger configuration 3   (d) Finger configuration 4
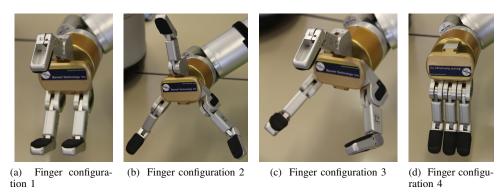
Fig. 3. Finger configurations used in the actions. The configurations are combined with open-wide or slightly closed joint positions that provide a variety of rough and accurate grasps.

present. However, grasp in *move* actions is more complex and involves: grasping point detection, hand configuration, and a height modifier. To compute the desired grasping point we use 2 different perception algorithms (Fig. 2): simple height and wrinkledness detector [18]. Do not confuse this "perception for the actions" with the perceptions for the POMDP observation model. We use a 3-fingered hand, so different configurations of the fingers are possible (Fig. 3). Finally, the height modifier is used to obtain specialized actions (Fig. 2d): going deep to perform a grasp is expected to take probably one object, but also can take more than one object; contrarily, a shallow grasp probably takes one object, but also can fail and take no objects (See Table I and the transition model discussion).

To summarize, the 20 *move* actions depend on the direction of motion ({Movement}), the kind of perception used to determine a grasping point ({Point analysis}), and a combination of finger configuration and height with respect to the defined grasping point ({Point modifier {Finger configuration}}). This leads to the following action set

$$
\left\{ \begin{array}{l} \text{Left} \rightarrow \text{Right} \\ \text{Left} \leftarrow \text{Right} \end{array} \right\} \left\{ \begin{array}{l} \text{height} \\ \text{wrinkle} \end{array} \right\} \left\{ \begin{array}{l} \text{low} \left\{ \begin{array}{l} \text{finger 1} \\ \text{finger 2} \\ \text{finger 3} \end{array} \right\} \\ \text{high} \left\{ \begin{array}{l} \text{finger 1} \\ \text{finger 4} \end{array} \right\} \end{array} \right\}
$$

$$
\left\{ \begin{array}{l} \text{Remove Left} \\ \text{Remove Right} \end{array} \right\}
$$

Not all the actions have been used in the experimental execution. In Sec. V it is shown that the policy selects the actions that are more suited for the task given its model and discards the other ones.

Action id:		Number of objects present				
9		0	1	2	3	4
	0	100	20	20	30	30
Number	1	0	80	50	30	50
of objects	2	0	0	30	30	20
taken	3	0	0	0	10	0
	4	0	0	0	0	0

TABLE I

EXAMPLE OF THE PROBABILITY DISTRIBUTION OF AN ACTION'S
GRASPING SUCCESS.

C. Rewards

The proportion between rewards and punishments determines the amount of risk the policy is allowed to take. An error performing the *remove* action is fatal as removing 2 pieces of clothing is an error. We choose the reward to be conservative and only allow *remove* actions when certainty of having isolated one piece of clothing raises to 95%.

D. Observation and Transition Model $(\Omega, \mathcal{T})$

Both observation and transition models are derived from the characterization of the perceptions and actions. Observations, the number of pieces of clothing present at each side of the table, are obtained with a simple color histogram algorithm. The observation model is derived from values in Table II which are obtained experimentally.

The transition model is more elaborated to obtain, and requires to compute for each different action (22 in our case) the probabilities of grasping success. One example corresponding to the action {wrinkle}{lifted{finger 3}} is presented at Table I.

Observations Distribution	Number of objects present				
	0	1	2	3	>3
0	100	0	0	0	0
Number 1	0	79	26	10	3.5
of objects 2	0	9	71	31	7.5
detected 3	0	6	3	48	30.5
>3	0	6	0	11	58.5

TABLE II

OBSERVATIONS PROBABILITY DENSITY DISTRIBUTION OF OBJECTS DETECTED OVER OBJECTS REALLY PRESENT (IN PERCENTAGE). THE POMDP OBSERVATIONS ARE DERIVED FROM THIS MODEL.

3 objects	Goal reached	Number of actions	% actions used	First object removed in:
POMDP partial	90%	12.1	36%	3.8
MDP partial	80%	7.8	32%	2.9
POMDP occluded	70%	15.3	41%	5.0
MDP occluded	20%	6.0	32%	3.0
5 objects				
POMDP partial	40%	14.25	36%	3.75

TABLE III

COMPARISON WITH THE FULLY OBSERVABLE CASE WITH AND WITHOUT TOTAL OCCLUSIONS AND SCALABILITY OF THE METHOD.

IV. EXPERIMENTAL VALIDATION

Through experimentation, we tested the performance of the algorithm for different levels of complexity of deformable objects with partial occlusions and total occlusions. Then, we compare the performance of the POMDP representation against a MDP representation in terms of task completion and steps required [19]. The scalability of the approach is assessed by increasing the initial number of objects on the table.

The experiments have been conducted with a BarrettHand WAM robot and a Kinect camera. The POMDP observations are provided through a simple color histogram analysis, and a performance test is carried out to characterize the perception uncertainty (see Table II). Each experiment started with objects manually spread at random on the right-hand side of the robot with different difficulties. Figure 6 shows a typical belief evolution through execution and is discussed on the following section.

The summary of the experiments is depicted in Table III. For each set of experiments we evaluate the percentage of goals successfully completed, the average number of actions required and the diversity of actions used. A goal is completed when the robot has successfully extracted separately every piece of clothing, and it is failed as soon as one *remove* action removes more than one piece of clothing from the table. Increasing the number of objects increases the chances of error, and the reliability of the action and observation models becomes critical (Second part of Table III), as we will discuss in the following section.

Regarding the average number of actions required (column 3 and 5 of Table III), in average, the first *removal* action is executed within the 4 and 5 first planned actions. Therefore, through action planning, two task objectives are reached concurrently and quite rapidly: achieving more than a 95% confidence about the real state in conjunction with the isolation of a piece of clothing in one pile.

Table III's results also show that, as one would presume, the occluded scenario is slightly more difficult than the partially occluded scenario.

A. MDP comparison

An MDP solution for the problem has been implemented. It considers the same action model used in the POMDP solution, but the observation model is deterministic, that is, errors in perceptions are not taken into account. As expected, the proposed method outperforms the MDP version of the problem (Table III). In the easy scenario, where all the pieces of clothing are only partially occluded, MDP almost performs with the same success as POMDP. The difference comes because actions sometimes lead to occlusions of previously seen pieces of cloth, that MDP cannot handle. In the scenario with occlusions the performance of MDP drops dramatically. Indeed, as soon as the full observability condition is broken through interaction, the robot fails to complete the task. In contrast, because the POMDP also values the retrieval of information, it is more prone to exploit the diversity of actions provided to fulfill the task successfully.

B. Results evaluation

The comparison of the results between POMDP executions in partially and completely occluded scenarios are encouraging (Row 1 and 3, Table III). In the partial occluded scenario the goal is reached 90% of times, while, when one object completely occludes one or various objects, the goal is reached 70% of times. The average total number of actions is between 12 and 15 (the minimum number of actions is 5: 2 *move* and 3 *remove*). Using the POMDP the first object is removed within the first 4 or 5 actions, and the rest of actions are used to isolate and remove the rest.

Failures are due to the removal of several pieces of clothing in the same action: after some manipulation actions, a complete occlusion remained because grasping could not split the two objects, and the planner was certain about the existence of one unique object.

Another source of error involves perception systematic errors. If not coded explicitly, the POMDP planning has the naive assumption that two observations over the same state are independent, which is not always true. Perceptions can systematically estimate a wrong number of objects, e.g. one object is always estimated as two. Obviously, manipulations will never provide the observation of one object.

Thus, the models have to be a representative sample of the possible inputs, and the actions have to ensure that new observations of the scene are gathered, asymptotically reaching the true state of the scene. Otherwise it is difficult to scale up to a great variety of objects because increasing the number of objects also increases the occurrence of the
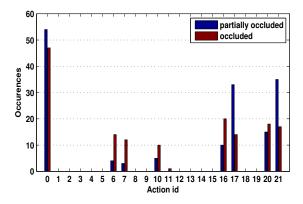
Fig. 4. Comparison of the diversity of applied actions between the scenarios partially occluded and totally occluded. When the perception uncertainty increases and more *surprises* occur, the planning is more prone to exploit the actions that increase the certainty.

Fig. 6. Belief evolution through execution. The labels show the (rounded) probability of each state. The bold state is the most probable. We can see how fast the certainty increases.
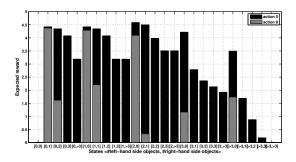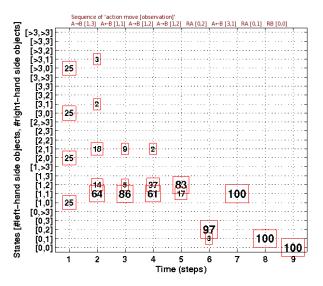

Fig. 5. Action value domination. The action 0 obtained a better value than action 8 in all states from the task perspective, and, therefore, the later can be discarded.

identified sources of error, given that the average number of actions required to fulfill the task also increases (second part of Table III).

Concerning the different actions taken in each scenario (4th column in Table III), Fig. 4 shows that the preferred actions change depending on the type of complexity of the scenario: as perceptions are more uncertain in the presence of occlusions, a higher confidence state is difficult to reach and the set of actions used by the POMDP also changes.

V. Discussion

A. Action value domination

One of the advantages of the proposed method is that the value of each action is assessed from the task perspective. Inspecting the policy, we can rapidly determine the useless actions by comparing their value in the multidimensional belief space. The actions a_2 that given another action a_1 fulfill the inequality $V(s, a_1) > V(s, a_2) \ \forall s$ can be automatically discarded. In our case, 8 out of the 22 actions were dominated by the action 0 or not used at all by the policy. Figure 5 provides an example of such domination. In particular, the actions discarded are: 4, 5, 8, 9, 13, 14, 15, 18 and 19, which correspond to {height}{lifted{finger 3}}, {wrinkle}{*{finger 1}}, {wrinkle}{lifted{finger 3}} and the equivalent actions starting from the other zone.

Plus, since the problem is not symmetrical because the task starts with clothes only on the right-hand side of the robot, {height}{lifted{finger 1}} is useful on that side, but not on the other one.

Additionally to the action selection, the value function also provides an action diversity assessment if a certain degree of improvement were required over the dominant actions' value.

B. Belief evolution through manipulation

Figure 6 shows the belief evolution of a selected execution. At the problem initialization, states representing objects in the right-hand side of the robot are equally probable. Then, the POMDP policy chooses an action that is likely to take one object to the left-hand side, but after running it, the perception informs that the robot has moved three objects. However, the probability associated to this outcome after performing such action is very low, so the belief gets correctly updated biased to a high number of objects. Additionally, although the robot sees only one object on its right-hand side, there is still a 23% probability of having more than one. In this situation, the full observability procedure would have failed, but the proposed method runs a couple of additional actions, moving a piece of clothing from one side to the other, to increase the certainty of having a single object to $83 + 17 = 100\%$ at step 5. After that, the doubts over the other pile of clothing have been dissipated (only a 3% chance of having one object) and the objects are successfully separated and removed with the three following actions.

Note that the policy, aware of its acting and perception failure probabilities, correctly accumulates the information of the observations received after each action. We can see that behaviour between steps three and four. Knowing that the grasp might have failed, the next observation improves the knowledge of the scene instead of assuming that the state has changed. In fact, in some executions, the policy

repeats actions in order to increase the certainty of object isolation. The trade-off between the accepted risk and the cost of performing additional manipulation actions can be tuned with the reward mechanism of the POMDP.

VI. CONCLUSIONS AND FUTURE WORK

The paper describes probabilistic planning of an isolation/extraction task. The POMDP design, modeling and implementation of the task have been applied to deformable object extraction with a real robot, showing the appropriateness of the POMDP approach to achieve the robot goals under unreliable actions and perceptions. The relevant results are two fold: First, the approach effectively exploits and reduces the inherent uncertainty through planning and interaction, which relaxes the precision requirements of robot vision and manipulation, and, second, the value function provides an estimate of the relevance of actions within the task planning context.

Regarding the inherent uncertainty of the perception and manipulation of deformable objects, we have shown that it is rapidly reduced through planning: the POMDP planning effectively combines the certainty distributed amongst several states. Through manipulation, the uncertainty about the state is reduced. Indeed, in the experimental context, the state confidence raises to 95% in less than five actions. It is important to note that the state definition was deliberately kept low-dimensional to show that dealing explicitly with uncertainty helps the robustness of the planning, in spite of acting in a simplified world model. In addition, that constraint makes possible the addition of supplementary deformable object features to the state encoding, exploitable from the planning perspective.

Regarding the value function, we have seen that it is a useful tool for task-driven action selection, a measure to assess the importance of each action over the state space. This is important because being able to identify the valuable actions is not obvious with the raw information, while it is self-evident once the policy is computed. Moreover, in order to take advantage of the action repertoire, the planning requires a sufficient rich state definition.

One focus of future research will be the specialization of the manipulation actions. Further reducing the uncertainty would increase the planning policy efficiency and effectiveness. In this sense, thanks to the policy action evaluation, we could use an over-dimensioned action space to automatically select the actually relevant ones for the particular scenario. Moreover, the current image processing techniques are constrained to colour segmentation, which sets an unnecessary strong constraint on the environment. Therefore, in the future, a combination of robot vision techniques can provide the estimation of the number of objects.

ACKNOWLEDGMENT

REFERENCES

[1] A. Ramisa, G. Alenya, F. Moreno-Noguer, and C. Torras, "Using Depth and Appearance Features for Informed Robot Grasping of Highly Wrinkled Clothes," in *International Conference on Robotics and Automation (ICRA)*, 2012.

[2] P. Jiménez, "Survey on Model-based Manipulation Planning of Deformable Objects," *Robotics and Computer-Integrated Manufacturing*, vol. 28, pp. 154 – 163, 2012.

[3] B. Willimon, S. Birchfield, and I. Walker, "Classification of Clothing using Interactive Perception," in *International Conference on Robotics and Automation (ICRA)*, 2011, pp. 1862–1868.

[4] M. Cusumano-Towner, A. Singh, S. Miller, J. O'Brien, and P. Abbeel, "Bringing clothing into desired configurations with limited perception," in *International Conference on Robotics and Automation (ICRA)*, 2011, pp. 3893–3900.

[5] P. C. Wang, S. Miller, M. Fritz, T. Darrell, and P. Abbeel, "Perception for the manipulation of socks," *International Conference on Intelligent Robots and Systems (ICIRS)*, pp. 4877–4884, Sep. 2011.

[6] J. Maitin-Shepard, M. Cusumano-Towner, J. Lei, and P. Abbeel, "Cloth Grasp Point Detection based on Multiple-View Geometric Cues with Application to Robotic Towel Folding," in *International Conference on Robotics and Automation (ICRA)*, 2010, pp. 2308 – 2315.

[7] L. Chrisman, "Reinforcement learning with perceptual aliasing: The perceptual distinctions approach," *National Conference on Artificial Intelligence (AAAI)*, pp. 183 – 188, 1992.

[8] A. Stolcke and S. Omohundro, "Hidden Markov Model Induction by Bayesian Model Merging," in *Advances in Neural Information Processing Systems (ANIPS)*, vol. 5, no. Ml, 1993, pp. 11–18.

[9] N. Roy, G. Gordon, and S. Thrun, "Planning under Uncertainty for Reliable Health Care Robotics," *International Conference on Field and Service Robotics (ICFSR)*, pp. 417 – 426, 2003.

[10] R. Martinez-Cantin, N. Freitas, E. Brochu, J. Castellanos, and A. Doucet, "A Bayesian exploration-exploitation approach for optimal online sensing and planning with a visually guided mobile robot," *Autonomous Robots*, vol. 27, no. 2, pp. 93 – 103, Aug. 2009.

[11] M. Sridharan, J. Wyatt, and R. Dearden, "HiPPo: Hierarchical POMDPs for Planning Information Processing and Sensing Actions on a Robot," *International Conference on Automated Planning and Scheduling (ICAPS)*, pp. 346–354, 2008.

[12] J. Hoey, P. Poupart, C. Boutilier, and A. Mihailidis, "POMDP Models for Assistive Technology," in *AAAI Fall Symposium on Caring Machines: AI in Eldercare*. University of Toronto, 2005, pp. 65 – 72.

[13] T. Smith and R. Simmons, "Point-Based POMDP Algorithms: Improved Analysis and Implementation," in *Conference on Uncertainty in Artificial Intelligence (CUAI)*, 2005, pp. 542–547.

[14] J. Pineau, G. Gordon, and S. Thrun, "Anytime Point-Based Approximations for Large POMDPs," *Journal of Artificial Intelligence Research*, vol. 27, pp. 335–380, 2006.

[15] H. Kurniawati, D. Hsu, and W. S. Lee, "SARSOP: Efficient Point-Based POMDP Planning by Approximating Optimally Reachable Belief Spaces," in *Robotics: Science & Systems (RSS)*, 2008.

[16] K. Hsiao, L. P. Kaelbling, and T. Lozano-Perez, "Grasping POMDPs," *International Conference on Robotics and Automation (ICRA)*, pp. 4685–4692, Apr. 2007.

[17] L. P. Kaelbling, M. Littman, and A. Cassandra, "Planning and acting in partially observable stochastic domains," *Artificial Intelligence*, vol. 101, no. 1-2, pp. 99–134, 1998.

[18] A. Ramisa, G. Alenya, F. Moreno, and C. Torras, "Determining Where to Grasp Cloth Using Depth Information," in *Catalan Conference on Artificial Intelligence (CCAI)*, 2011, pp. 199 – 207.

[19] J. Pineau, N. Roy, and S. Thrun, "A hierarchical approach to POMDP planning and execution," in *Workshop on Hierarchy and Memory in Reinforcement Learning (ICML)*, vol. 65, no. 66, 2001, p. 51.