Cooperative Active Perception using POMDPs

Matthijs T.J. Spaan
Institute for Systems and Robotics
Instituto Superior Técnico
Av. Rovisco Pais, 1, 1049-001
Lisbon, Portugal

Abstract

This paper studies active perception in an urban scenario, focusing on the cooperation between a set of surveillance cameras and mobile robots. The fixed cameras provide a global but incomplete and possibly inaccurate view of the environment, which can be enhanced by a robot's local sensors. Active perception means that the robot considers the effects of its actions on its sensory capabilities. In particular, it tries to improve its sensors' performance, for instance by pointing a pan-and-tilt camera. In this paper, we present a decision-theoretic approach to cooperative active perception, by formalizing the problem as a Partially Observable Markov Decision Process (POMDP). POMDPs provide an elegant way to model the interaction of an active sensor with its environment. The goal of this paper is to provide first steps towards an integrated decision-theoretic approach of cooperative active perception.

Introduction

Robots are leaving the research labs and operating more often in human-inhabited environments, such as urban pedestrian areas. The scenario we consider in our work is a group of robots assisting humans in a car-free area (Sanfeliu and Andrade-Cetto 2006). The primary task of the robots is to identify persons in need of assistance, and subsequently help them, for instance by guiding to a desired location. Additional tasks could involve transportation of goods as well as performing monitoring and security duties. The pedestrian area in which the robots operate is equipped with surveillance cameras providing the robot with more information. Implementing such a system requires addressing many scientific and technological challenges such as cooperative localization and navigation, map building, human-robot interaction, and wireless networking, to name but a few (Sanfeliu and Andrade-Cetto 2006). In this paper, we focus on one particular problem, namely *cooperative active perception*.

In our context, cooperative perception refers to the fusion of sensory information between the fixed surveillance cameras and each robot, with as goal maximizing the amount and quality of perceptual information available to the system. This information can be used by a robot to choose its actions, as well as providing a global picture for monitoring the system. In general, incorporating information from spatially distributed sensors will raise the level of situational awareness.

Active perception means that an agent considers the effects of its actions on its sensors, and in particular it tries to improve their performance. This can mean selecting sensory actions, for instance pointing a pan-and-tilt camera or choosing to execute an expensive vision algorithm; or to influence a robot's path planning, e.g., given two routes to get to a desired location, take the more informative one. Performance can be measured by trading off the costs of executing actions with how much we improve the quality of the information available to the system, and should be derived from the system's task. Combining the two concepts, cooperative active perception is the problem of active perception involving multiple sensors and multiple cooperating decision makers.

In this paper, we present a decision-theoretic approach to cooperative active perception. In particular, we propose to use Partially Observable Markov Decision Processes (POMDPs) (Kaelbling, Littman, and Cassandra 1998) as a framework for active cooperative perception. POMDPs provide an elegant way to model the interaction of an active sensor with its environment. Based on prior knowledge of the sensor's model and the environment dynamics, we can compute policies that tell the active sensor how to act, based on the observations it receives. As we are essentially dealing with multiple decision makers, it could also be beneficial to consider modeling (a subset of) sensors as a decentralized POMDP (Dec-POMDP) (Bernstein et al. 2002).

In a cooperative perception framework, an important task encoded by the (Dec-)POMDP could be to reduce the uncertainty in its view of the environment as much as possible. Entropy can be used as a suitable measure for uncertainty. However, using a POMDP solution, we can tackle more elaborate scenarios, for instance in which we prioritize the tracking of certain objects. In particular, POMDPs inherently trade off task completion and information gathering. Sensory actions might also include other sensors, as we can reason explicitly about communicating with other sensors. For instance, a fixed sensor could ask a mobile sen-

Copyright © 2008, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

sor to examine a certain location. Regardless of whether we consider a Dec-POMDP or single-agent POMDPs, we will need to tackle two issues: modeling and solving. In this paper we address these issues, to provide first steps towards an integrated decision-theoretic approach of cooperative active perception.

The rest of this paper is organized as follows. First we will start by providing an overview of related literature, considering real-world applications of POMDPs as well as decision-theoretic approaches to active perception. Next we formally introduce the POMDP model, and describe how it can be applied to an active perception task. We continue by detailing the application scenario we are considering, followed by some preliminary experiments. Finally, we discuss our work and avenues of future research.

Related work

We can identify two bodies of literature directly related to our study. First are applications of planning under uncertainty methodology to real-world systems. Second, we will discuss decision-theoretic approaches to active perception.

Techniques for single-agent decision-theoretic planning under uncertainty such as POMDPs are being applied more and more to robotics (Vlassis, Gordon, and Pineau 2006). Over the years, there have been numerous examples demonstrating how POMDPs can be used for robot localization and navigation, see for example work by Simmons and Koenig; Roy, Gordon, and Thrun (1995; 2005). Emery-Montemerlo et al. (2005) demonstrated the viability of approximate Dec-POMDP techniques for controlling a small group of robots. A relevant body of work exists on systems interacting with humans driven by POMDP-based controllers. Fern et al. (2007) propose a POMDP model for providing assistance to users, in which the goal of the user is a hidden variable which needs to be inferred. Boger et al. (2005) apply POMDPs in a real-world task for assisting people with dementia, in which users receive verbal assistance while washing their hands. POMDP models have also been applied to high-level control of a robotic assistant designed to interact with elderly people (Pineau et al. 2003; Roy, Gordon, and Thrun 2003).

There have also been applications of decision-theoretic techniques to active sensing, which is highly related to the problem we are tackling. Although not explicitly modeled as POMDPs, methods for active robot localization using information gain have been proposed, see e.g., (Stone et al. 2006). Darrell and Pentland (1996) propose a visual gesture recognition system, in which a POMDP controller steers the focus of the camera to regions in the image which are most likely to improve recognition performance. Along similar lines, Vogel and Murphy (2007) locate objects in large images of office environments, while exploiting spatial relationships between the objects. Guo (2003) describes a POMDP framework for active sensing in which the actions are using a particular sensor (with an associated cost) or, when enough information has been gathered, outputting a particular classification label. Ji and Carin (2007) consider a similar setting, but couple it with the training of HMM classifiers. Also related to our scenario are decision-theoretic approaches to multi-modal sensor scheduling (Ji, Parr, and Carin 2007). In a multiagent setting, Varakantham et al. (2007) consider a distributed sensor network in which each sensor has to choose its gaze direction, in order to track targets.

POMDPs for active perception

We will discuss POMDP models and solution methods, briefly introducing some general background but focusing on their application to active perception.

Models

We will briefly introduce the POMDP model, a more elaborate description is provided by Kaelbling, Littman, and Cassandra (1998), for instance. A POMDP models the interaction of an agent with a stochastic and partially observable environment, and it provides a rich mathematical framework for acting optimally in such environments.

A POMDP assumes that at any time step the environment is in a state $s \in S$, the agent takes an action $a \in A$ and receives a reward $r(s,a)$ from the environment as a result of this action, while the environment switches to a new state s' according to a known stochastic transition model $p(s'|s,a)$. The agent's task is defined by the reward it receives at each time step and its goal is to maximize its long-term reward. After transitioning to a new state, the agent perceives an observation $o \in O$, that may be conditional on its action, which provides information about the state s' through a known stochastic observation model $p(o|s',a)$.

Given the transition and observation model the POMDP can be transformed to a belief-state MDP: the agent summarizes all information about its past using a belief vector $b(s)$. The belief b is a probability distribution over S, which forms a Markovian signal for the planning task. The initial state of the system is drawn from the initial belief b_0, which is typically part of a POMDP's problem definition. Every time the agent takes an action a and observes o, its belief is updated by Bayes' rule; for the discrete case:

$$b_a^o(s') = \frac{p(o|s',a)}{p(o|a,b)} \sum_{s \in S} p(s'|s,a)b(s), \tag{1}$$

where $p(o|a,b) = \sum_{s' \in S} p(o|s',a) \sum_{s \in S} p(s'|s,a)b(s)$ is a normalizing constant. For the general case, the sums become integrals and we will need to choose a model representation from a family of functions for which integration is defined. A suitable representation can be to represent models as (mixtures of) Gaussians, for which POMDP solution techniques have been developed (Porta et al. 2006). The choice of belief representation is rather orthogonal to the POMDP techniques used in this paper, and we consider the discrete case for simplicity.

When multiple independent decision makers are present in the environment, the problem can be modeled as a decentralized POMDP (Dec-POMDP) (Bernstein et al. 2002; Seuken and Zilberstein 2008; Oliehoek, Spaan, and Vlassis 2008). We will return to this point in the discussion, assuming for the moment that only one decision maker exists, namely a robot. Note that the robot could take into account actions that involve other entities, for instance instruct a surveillance camera to run a particular vision algorithm. Another requirement for treating (parts of) the system is a POMDP is fast and reliable communication, as cameras and robots need to share local observations. Cameras are expected to do local processing, and sharing the resulting observations will require only low bandwidth.

Beliefs for active perception

In general, a belief update scheme is the backbone of many robot localization techniques, in which case the state is the robot's location (and heading). In our case however, the state will also be used to describe location of persons or events in the environment, as well as some of their properties. From each sensor we will need to extract a probabilistic sensor model to be plugged in the observation model. Furthermore, we need to construct the transition model based on the robot's available actions. Both models can either be defined by hand, or can be obtained using machine learning techniques, see for instance work by Stone et al. (2006).

From the perspective of active perception, as the belief is a probability distribution over the state space, it is natural to define the quality of information based on it. We can use the belief to define a measurement of the expected information gain when executing an action. For instance, a common technique is to compare the entropy of a belief b_t at time step t with the entropy of future beliefs, for instance at $t+1$. If the entropy of a future belief b_{t+1} is lower than b_t, the robot has less uncertainty regarding the true state of the environment. Assuming that the observation models are correct (unbiased etc), this would mean we gained information. Given the models, we can predict the set of beliefs $\{b_{t+1}\}$ we could have at $t+1$, conditional on the robot's action a. Each b_{t+1} has a probability of occurrence which is equal to the probability $p(o|a, b_t)$ of receiving the observation o that generated it.

If we adjust the POMDP model to allow for reward models that define rewards based on beliefs instead of states, i.e., $r(b, a)$, we can define a reward model based on the belief entropy. A natural interpretation would be to give higher reward to low-entropy beliefs. This way the robot can be guided to choose actions that lower the entropy of its belief, traded off by the cost of executing an action. However, a reward model defined over beliefs does significantly raise the complexity of planning, as the value function will no longer be piecewise linear and convex. Such a compact representation is being exploited by many optimal and approximate POMDP solvers.

Solution methods

In POMDP literature, a plan is called a policy $\pi(b)$ and maps beliefs to actions. A policy π can be characterized by a value function V^π which is defined as the expected future discounted reward $V^\pi(b)$ the agent can gather by following π starting from belief b:

$$V^\pi(b) = E_\pi\Big[\sum_{t=0}^{h} \gamma^t r(b_t, \pi(b_t))\Big| b_0 = b\Big], \tag{2}$$

where $r(b_t, \pi(b_t)) = \sum_{s\in S} r(s, \pi(b_t))b_t(s)$ following the POMDP model as defined before, h is the planning horizon, and γ is a discount rate, $0 \leq \gamma \leq 1$.

As solving POMDPs optimally is very hard, we will have to consider approximate algorithms. Recent years have seen much progress in approximate POMDP solving which we can leverage, see for instance (Hauskrecht 2000; Spaan and Vlassis 2005) and references therein. Furthermore, when a policy has been computed off-line, executing it on-line does not require much computational requirements. On the other hand, such policies are computed for a particular POMDP model, while in our case we are dealing with a very dynamic environment. In this case, it might not be feasible to construct one POMDP model that serves for all situations, but a better solution might be to construct POMDP models on-the-fly. Such a model would for instance only consider sensors physically close to the robot.

Solving POMDP models approximately off-line can be implemented by computing a value function over the belief space, which defines a policy. Executing such a policy is computationally cheap, but computing the value function can be expensive (depending on the solution method and the level of approximation used). On the other hand, on-line POMDP methods such as (Ross and Chaib-draa 2007; Satia and Lave 1973) construct the POMDP's belief tree and do an on-line search (e.g., branch and bound) for the best action to execute, given the robot's current belief. In this case the off-line cost might be low, but every time we need to choose an action we have to search the belief tree. Hence, an interesting research issue is whether to employ off-line or on-line methods, or a combination of both.

Application scenario

We will now detail the types of cooperative active perception scenarios we are are addressing. Figure 1 shows a map of a part of a university campus, to be used in the URUS project (Sanfeliu and Andrade-Cetto 2006). The project's focus is on designing a network of robots that interact with humans in urban areas, and whose tasks include providing assistance, transportation of goods, and surveillance. All sensors and robots are connected using a wired or wireless network. The environment is car-free, and will be equipped with surveillance cameras, potential locations of which are indicated by arrow heads in Figure 1. Several types of robots with different sensor suites will be employed, but as we are dealing with the problem on a high level, we will not go into details. Such a scenario provides many opportunities and challenges for POMDP-based active perception, as we discuss next.

We will focus on the assistance and surveillance tasks of the robots. More specific, assistance means guiding a human to a desired location, as indicated by the subject.[1] The

[1] How exactly the human subject interacts with the robot, for in-

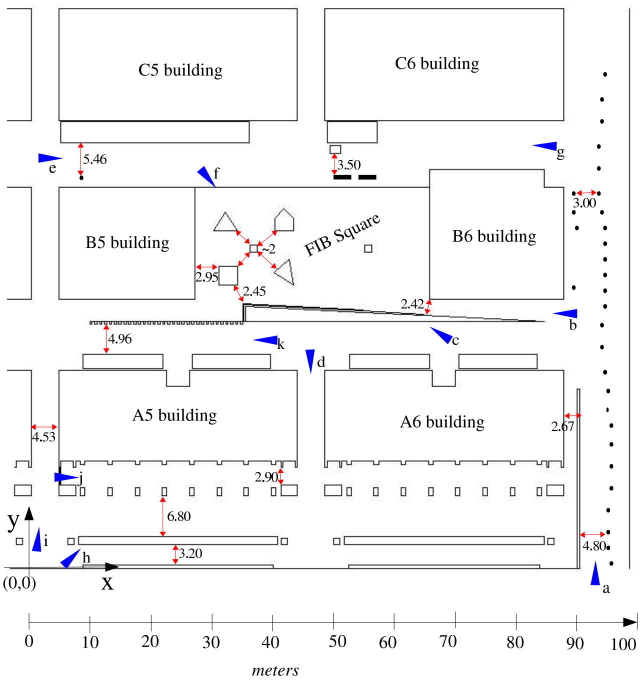

Figure 1: Map of the application scenario, a part of the UPC campus, Barcelona, Spain. Arrow heads indicate potential camera locations. There are 6 buildings grouped around a central square with several trees. Because of the many obstacles (buildings, trees, etc), full high-resolution coverage by surveillance cameras is hard to achieve.

surveillance task can be formulated as verifying the existence of certain events of interest. Events can include people waving, emergencies, persons lying on the floor, fires, and can have different priorities. The surveillance cameras will run a set of event detection algorithms, but will have a limited view and accuracy. In particular, the environment might contain blind spots that are not observed by any fixed camera. Furthermore, other areas might be observed by a camera, but not with sufficient resolution for accurate event detection. One of the fixed (camera) sensors might notice a possible event, and the robot could decide to investigate. Or, the robot could instruct the camera to run a computationally expensive detection algorithm to improve perception.

When a robot is navigating through the environment, while executing a certain task such as guiding a human subject, it could influence its trajectory to improve perception. The goal could be to improve the accuracy of its localization, for instance by guiding it along paths in which its sensors are likely to perform well. Also, a robot's path could be influenced to improve the perception of certain features of the environment, for instance blind spots not covered by

stance using a touch screen or a voice interface, is beyond the scope of this paper. Note that such human-robot interaction problems have also been tackled in a POMDP framework (Pineau et al. 2003; Roy, Gordon, and Thrun 2003).

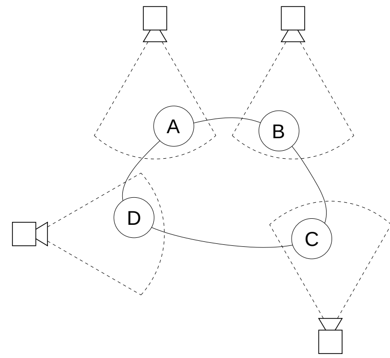

Figure 2: A sensor network with 4 sensors and 4 possible robot locations: A, B, C, and D. The dashed lines indicate each sensor's field of view, and the graph connecting the four locations indicates the robot's movement options.

fixed cameras. An important issue here will be to trade off accomplishing the robot's task (reaching a certain location) with the expected information gain.

Preliminary experiments

We performed some preliminary experiments in a simplified scenario, which considers the event detection problem modeled on a high level. We assume a setup consisting of a network of n sensors and a single robot. Each sensor has a non-overlapping field of view (FOV) and the robot can move from one sensor's FOV to another one. Graphs can be used to represent topological maps with potentially stochastic transitions. A graph illustrating the possible transitions for the case of $n = 4$ sensors is depicted in Figure 2. In the more general case, we would expect the robot to navigate much larger graphs, in which not all nodes lie inside a camera's FOV.

Each sensor is considered to be a camera running an advanced feature-detection algorithm. A sensor can detect persons and fires in its FOV, but only with a limited accuracy. If someone is present, the sensor will detect him with probability $p_p = 0.5$, and flames are detected with probability $p_f = 0.8$. We are interested in detecting whether in any of the FOVs a person or a fire is present. The robot receives the combined observation vector of the sensor network, based on which it selects its next action. The robot's task is to report whether fires or persons are present at a certain location. Basically, this assumes that when a robot is at a location, it can detect events with full certainty. Reporting fires has a higher priority, and consequently correctly reporting a fire receives a higher reward ($r = 100$) than reporting a person ($r = 10$). However, reporting an event which is not present is penalized, as resources are wasted. Finally, the prior probability of a fire $p_F = 0.01$ starting at a location is much lower than the probability of a person being present ($p_P = 0.2$).

We created a POMDP model for this task which has 324 states, as for $n = 4$, $n \cdot 3^n = 324$. There are three states and observations per sensor: nothing, person, or fire present. The problem has 6 actions (go to A, B, C, or D, and report person or fire), and $3^n = 81$ observations. Note that the robot can only move along the graph (i.e., executing "go to A" from location C has no effect). Solving such a model exactly is infeasible, and we performed some preliminary experiments with two approximate off-line methods. In particular, we applied the well-known Q_{MDP} method (Littman, Cassandra, and Kaelbling 1995), as well as PERSEUS (Spaan and Vlassis 2005), a point-based POMDP method. Both techniques compute policies that successfully report persons where they appear. However, when a fire appears, they switch to that location, and report it, as reporting fires has a higher priority. The PERSEUS solution achieves a slightly higher payoff however (57.74 vs. 57.36, $\gamma = 0.95$, $h = 10$), as is to be expected, as it computes better approximations than Q_{MDP}, albeit at a higher computational cost. Note that in a more complicated scenario, in particular in which the robot's sensors are modeled, we would expect Q_{MDP} to perform much worse, as it will not take actions for the purpose of gaining information.

An advantage of point-based methods is that we can influence their run time by varying the size of the belief set (we used 1000 beliefs in this experiment). As discussed before, on-line POMDP solution techniques could be beneficial, as it is likely that a robot will need to create a POMDP model on the fly. In this case, there might not be enough time to run more expensive off-line methods. Furthermore, on-line techniques in general roll out (a part of) the belief tree given the robot's belief, which can be beneficial if we want to reason directly about minimizing belief entropy, instead of only maximizing expected reward.

Discussion and future work

We discussed first steps toward a decision-theoretic approach to cooperative active perception, in which robots and sensors cooperate in an urban scenario. We identified relevant issues, both in modeling the problem as well as regarding solution techniques. An advantage of taking a decision-theoretic approaches using POMDPs is the natural integration of measuring task performance and situational awareness. By considering a robot as a mobile sensor, we also need to take into account the delay in receiving information regarding a possible event, since a robot needs to move to its location. POMDPs allow for modeling such decisions in an integrated way.

Furthermore, many approaches to active sensing in literature focus on minimizing uncertainty per se, without considering other objectives the robot might have. In particular, in some cases certain state features might be irrelevant, given the task definition. For example, if a camera detects a potential fire, we would like the robot to check out that location with high priority. Potential human users asking the robot to guide them would have a lower priority. The POMDP model allows the designer of the system to trade off information gathering with other priorities in a principled manner.

The focus in this paper was on a single decision maker, but essentially we are dealing with multiple decision makers. Dec-POMDPs form a general framework for representing cooperative planning under uncertainty problems. However, as solving a Dec-POMDP in the most general setting

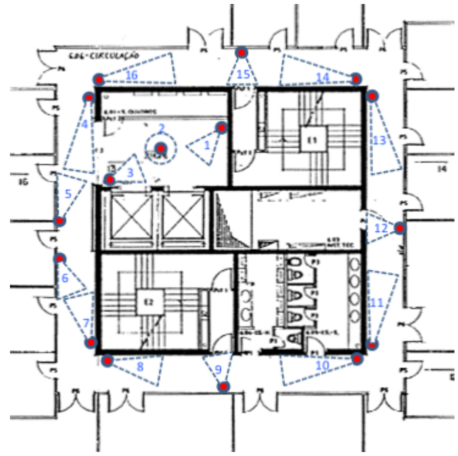

Figure 3: Map of the 6th floor of ISR, Lisbon, Portugal. Shown are the camera locations and their potential field of view (which is adjustable).

is intractable, a large research focus is on identifying and solving restricted but relevant scenarios. Very relevant for our application is that we can exploit the fact that in many domains interaction between agents is a local phenomenon (Oliehoek et al. 2008; Spaan and Melo 2008). Communication can simplify the problem, and an active area is how to successfully incorporate the robot's communication capabilities in Dec-POMDP framework, see for example (Roth, Simmons, and Veloso 2007). Furthermore, issues with unreliable communication have to be considered, as the wireless communication between robots and sensors might fail.

In future work, we will examine the tradeoff between off-line and on-line methods, extending the state of the art if necessary. On-line methods have the benefit of only planning for actually encountered beliefs, which can be beneficial if we define POMDP models on the fly, in which case planning for all or a sampled set of beliefs might be wasteful. On-line methods appear more amendable to reward models based on belief entropy, as they in general do not employ any state-based backup scheme as many off-line methods do, but just search the tree of beliefs, and backup values in tree nodes. However, reward models based on beliefs instead of states preclude the use of piecewise linear and convex value functions, which haven proven very useful for approximate off-line algorithms.

From an experimental point of view, we intend to further develop our simulated experiments, by considering far more complicated scenarios, for instance modeling the robot's sensory capabilities better. In general, designing or learning observation models will be challenging. With respect to real-world experiments, we plan to start by exploring a more controlled indoor environment. The three floors of our institution are being equipped with 16 surveillance cameras each to be used for research purposes. One floor with camera locations is depicted in Figure 3. An indoor experiment will be very valuable to validate and further develop our approach, before moving to a more challenging outdoor environment.

Acknowledgments

This paper has benefited from discussions with Pedro Lima, Luis Montesano, Luis Merino, and Alexandre Bernardino. This work was supported by the European Project FP6-2005-IST-6-045062-URUS, and ISR/IST pluriannual funding through the POS_Conhecimento Program that includes FEDER funds.

References

Bernstein, D. S.; Givan, R.; Immerman, N.; and Zilberstein, S. 2002. The complexity of decentralized control of Markov decision processes. *Mathematics of Operations Research* 27(4):819–840.

Boger, J.; Poupart, P.; Hoey, J.; Boutilier, C.; Fernie, G.; and Mihailidis, A. 2005. A decision-theoretic approach to task assistance for persons with dementia. In *Proc. Int. Joint Conf. on Artificial Intelligence*.

Darrell, T., and Pentland, A. 1996. Active gesture recognition using partially observable Markov decision processes. In *Proc. of the 13th Int. Conf. on Pattern Recognition*.

Emery-Montemerlo, R.; Gordon, G.; Schneider, J.; and Thrun, S. 2005. Game theoretic control for robot teams. In *Proceedings of the IEEE International Conference on Robotics and Automation*.

Fern, A.; Natarajan, S.; Judah, K.; and Tadepalli, P. 2007. A decision-theoretic model of assistance. In *Proc. Int. Joint Conf. on Artificial Intelligence*.

Guo, A. 2003. Decision-theoretic active sensing for autonomous agents. In *Proc. of the Int. Conf. on Computational Intelligence, Robotics and Autonomous Systems*.

Hauskrecht, M. 2000. Value function approximations for partially observable Markov decision processes. *Journal of Artificial Intelligence Research* 13:33–95.

Ji, S., and Carin, L. 2007. Cost-sensitive feature acquisition and classification. *Pattern Recognition* 40(5):1474–1485.

Ji, S.; Parr, R.; and Carin, L. 2007. Non-myopic multi-aspect sensing with partially observable Markov decision processes. *IEEE Trans. Signal Processing* 55(6):2720–2730.

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

Littman, M. L.; Cassandra, A. R.; and Kaelbling, L. P. 1995. Learning policies for partially observable environments: Scaling up. In *International Conference on Machine Learning*.

Oliehoek, F. A.; Spaan, M. T. J.; Whiteson, S.; and Vlassis, N. 2008. Exploiting locality of interaction in factored Dec-POMDPs. In *Proc. of Int. Joint Conference on Autonomous Agents and Multi Agent Systems*.

Oliehoek, F. A.; Spaan, M. T. J.; and Vlassis, N. 2008. Optimal and approximate Q-value functions for decentralized POMDPs. *Journal of Artificial Intelligence Research*.

Pineau, J.; Montemerlo, M.; Pollack, M.; Roy, N.; and Thrun, S. 2003. Towards robotic assistants in nursing homes: Challenges and results. *Robotics and Autonomous Systems* 42(3–4):271–281.

Porta, J. M.; Vlassis, N.; Spaan, M. T. J.; and Poupart, P. 2006. Point-based value iteration for continuous POMDPs. *Journal of Machine Learning Research* 7:2329–2367.

Ross, S., and Chaib-draa, B. 2007. AEMS: An anytime online search algorithm for approximate policy refinement in large POMDPs. In *Proc. Int. Joint Conf. on Artificial Intelligence*.

Roth, M.; Simmons, R.; and Veloso, M. 2007. Exploiting factored representations for decentralized execution in multi-agent teams. In *Proc. of Int. Joint Conference on Autonomous Agents and Multi Agent Systems*.

Roy, N.; Gordon, G.; and Thrun, S. 2003. Planning under uncertainty for reliable health care robotics. In *Proc. of the Int. Conf. on Field and Service Robotics*.

Roy, N.; Gordon, G.; and Thrun, S. 2005. Finding approximate POMDP solutions through belief compression. *Journal of Artificial Intelligence Research* 23:1–40.

Sanfeliu, A., and Andrade-Cetto, J. 2006. Ubiquitous networking robotics in urban settings. In *Proceedings of the IEEE/RSJ IROS Workshop on Network Robot Systems*.

Satia, J. K., and Lave, R. E. 1973. Markovian decision processes with probabilistic observation of states. *Management Science* 20(1):1–13.

Seuken, S., and Zilberstein, S. 2008. Formal models and algorithms for decentralized decision making under uncertainty. *Autonomous Agents and Multi-Agent Systems*.

Simmons, R., and Koenig, S. 1995. Probabilistic robot navigation in partially observable environments. In *Proc. Int. Joint Conf. on Artificial Intelligence*.

Spaan, M. T. J., and Melo, F. S. 2008. Interaction-driven Markov games for decentralized multiagent planning under uncertainty. In *Proc. of Int. Joint Conference on Autonomous Agents and Multi Agent Systems*.

Spaan, M. T. J., and Vlassis, N. 2005. Perseus: Randomized point-based value iteration for POMDPs. *Journal of Artificial Intelligence Research* 24:195–220.

Stone, P.; Sridharan, M.; Stronger, D.; Kuhlmann, G.; Kohl, N.; Fidelman, P.; and Jong, N. K. 2006. From pixels to multi-robot decision-making: A study in uncertainty. *Robotics and Autonomous Systems* 54(11):933–943.

Varakantham, P.; Marecki, J.; Yabu, Y.; Tambe, M.; and Yokoo, M. 2007. Letting loose a SPIDER on a network of POMDPs: Generating quality guaranteed policies. In *Proc. of Int. Joint Conference on Autonomous Agents and Multi Agent Systems*.

Vlassis, N.; Gordon, G.; and Pineau, J. 2006. Planning under uncertainty in robotics. *Robotics and Autonomous Systems* 54(11). Special issue.

Vogel, J., and Murphy, K. 2007. A non-myopic approach to visual search. In *Fourth Canadian Conference on Computer and Robot Vision*.