Policy Activation for Open-Ended Dialogue Management

Pierre Lison and **Geert-Jan M. Kruijff**
German Research Centre for Artificial Intelligence (DFKI GmbH)
Saarbrücken, Germany – {plison,gj}@dfki.de

Abstract

An important difficulty in developing spoken dialogue systems for robots is the open-ended nature of most interactions. Robotic agents must typically operate in complex, continuously changing environments which are difficult to model and do not provide any clear, predefined goal. Directly capturing this complexity in a single, large dialogue policy is thus inadequate. This paper presents a new approach which tackles the complexity of open-ended interactions by breaking it into a set of small, independent policies, which can be activated and deactivated at runtime by a dedicated mechanism. The approach is currently being implemented in a spoken dialogue system for autonomous robots.

Introduction

Human-robot interactions (HRI) often have a distinctly open-ended character. In many applications, the robot does not know in advance which goals needs to be achieved, but must discover these during the interaction itself. The user might communicate new requests, clarify or modify existing ones, ask questions, or provide the robot with new information at any time. The robotic agent must therefore be capable of handling a wide variety of tasks, some being purely reactive (such as answering a question), some being more deliberative in nature (such as planning a complex sequence of actions towards a long-term goal).

The interaction dynamics are also significantly more difficult to predict in HRI. In classical, slot-filling dialogue applications, the domain provides strong, predefined constraints on how the dialogue is likely to unfold. Interactive robots, on the other hand, usually operate in rich, dynamic environments which can evolve in unpredictable ways. The interaction is therefore much more difficult to model and depends on numerous parameters. (Bohus and Horvitz 2009) provide a review of important technical challenges to address in such kind of open-ended interactions.

Previous work on this issue mostly focussed on techniques for enlarging the state and action spaces to directly capture this complexity. These techniques are usually coupled with mechanisms for factoring (Bui et al. 2010) or abstracting (Young et al. 2010) these large spaces to retain tractability. Applied to human-robot interactions, these approaches unfortunately suffer from two shortcomings: first, the complexity of the planning problem increases exponentially with the size of the state space, making these approaches difficult to scale. Second, from the dialogue developer viewpoint, maintaining and adapting dialogue policies over very large spaces is far from trivial.

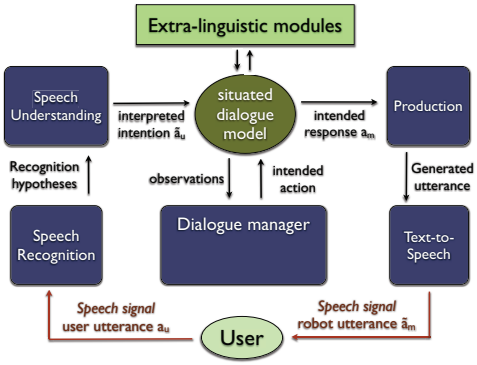
This paper sketches a new approach specifically tailored to open-ended interactions. Instead of using one single policy operating over large spaces, the idea is to break up this complexity into a set of shorter, more predictable interactions, which can be activated and deactivated at runtime. The dialogue manager contains a repository of potential policies, and decides which policies to use at a given time via a dedicated *policy activation* mechanism. Several policies can be activated in parallel, and the dialogue manager is responsible for the trade-offs between the activated policies.

Architecture

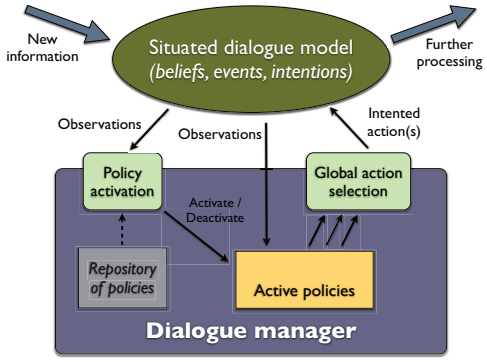
The general architecture of the dialogue system is illustrated in Figure 1. The architecture revolves around a *situated dialogue model*, which stores various epistemic objects such as beliefs, events and intentions. These epistemic objects are generic representations of the agent's knowledge (e.g. the dialogue history as well as relevant perceptual information), and are expressed as probabilistic relational structures – see (Lison, Ehrler, and Kruijff 2010) for details. The dialogue manager continuously monitors this dialogue model, and reacts upon changes by triggering new observations. These observations can in turn influence the policy activation mechanism (by activating or deactivating policies), or provide direct input to the active policies.

Approach

Instead of designing each dialogue policy by hand – a tedious task given the high levels of noise and uncertainty encountered in HRI –, we define each interaction as a *Partially Observable Markov Decision Process* (POMDP), and apply optimisation algorithms to extract a near-optimal policy for it. POMDPs are a principled mathematical framework for control problems featuring partial observability, stochastic action effects, decision-making over arbitrary horizons, incomplete knowledge of the environment dynamics, and mul-



(a) Global schema of the spoken dialogue system.



(b) Detailed schema of the dialogue management module.

Figure 1: Architectural schema, illustrating the dialogue system as a whole (left), and the dialogue management module (right).

tiple, conflicting objectives. As such, they provide an ideal modelling tool to develop dialogue policies for HRI.

A POMDP is a tuple $\langle \mathcal{S}, \mathcal{A}, \mathcal{Z}, T, \Omega, R \rangle$, with $\mathcal{S}$ the state space; $\mathcal{A}$ the action space; $\mathcal{Z}$ the observation space; $T(s, a, s')$ the transition function from state s to state s' via action a; $\Omega(z, a, s')$ the observation function for observing z in state s' after performing a; and $R(s, a)$ is the reward function encoding the utility of executing action a in state s.

A central idea of POMDP is the assumption that the state is not directly accessible and can only be inferred from observation. Such uncertainty is expressed in the *belief state* b, which is a probability distribution $b : \mathcal{S} \rightarrow [0, 1]$ over possible states. A POMDP policy is then defined over this belief space as a function $\pi : \mathcal{B} \rightarrow \mathcal{A}$ determining the action to perform for each point of the belief space.

Each interaction is modelled in our approach as a separate POMDP. Since these POMDPs have a small state space, a well-defined purpose and a more predictable transition function, they are much easier to model than a single, monolithic POMDP. Furthemore, the policies of these small POMDP can be easily learned via reinforcement learning techniques (Sutton and Barto 1998), using a user simulator.

Policy activation

The policy activation is based on a repository of policies. Each policy is associated with a set of *triggers*. These triggers are reactive to particular changes in the dialogue model – a dialogue policy dealing with replies to user questions will for instance be made reactive to the appearance of a new question onto the dialogue model. The triggers can be viewed as a hierarchical POMDP with abstract actions to activate or deactivate specific subpolicies.

Action selection with multiple policies

Several dialogue policies can be activated in parallel in the dialogue manager. The agent must therefore be capable of setting the right trade-offs between the various policies.

To this end, we maintain a separate belief point b_i for each activated policy p_i. We define the vector $\mathbf{b}$ as the set of these belief points. Assuming each policy also provides us directly a Q-value function $Q_i(b_i, a)$, we can then compute the best global strategy $\pi(\mathbf{b})$ by maximising the sum of Q-values over the set of activated policies:

$$\pi(\mathbf{b}) = \underset{a \in \mathcal{A}}{\operatorname{argmax}} \sum_{b_i \in \mathbf{b}} Q(b_i, a) \qquad (1)$$

The global action space $\mathcal{A}$ in Eq. (1) is defined as $\cup_i \mathcal{A}_i$. This enables us to select the action which is globally optimal with respect to the set of activated policies.

Conclusion

In this paper, we presented a first sketch of an POMDP-based approach to dialogue management which explicitly handles open-ended interactions by activating and deactivating policies at runtime. Future work will focus on implementing and evaluating the outlined approach in a real-world dialogue system for autonomous robots.

Acknowledgements

This work was supported by the EU projects "*ALIZ-E: Adaptive Strategies for Sustainable Long-Term Social Interaction*" (FP7-ICT-248116) and "*CogX: Cognitive Systems that Self-Understand and Self-Extend*" (FP7-ICT- 215181).

References

Bohus, D., and Horvitz, E. 2009. Dialog in the open world: Platform and applications. In *Proceedings of ICMI'09*.

Bui, T. H.; Zwiers, J.; Poel, M.; and Nijholt, A. 2010. Affective dialogue management using factored pomdps. In *Interactive Collaborative Information Systems*. 209–238.

Lison, P.; Ehrler, C.; and Kruijff, G. 2010. Belief modelling for situation awareness in human-robot interaction. In *Proceedings of the 19th International Symposium on Robot and Human Interactive Communication (RO-MAN 2010)*.

Sutton, R., and Barto, A. 1998. *Reinforcement Learning: An Introduction*. The MIT Press.

Young, S.; Gašić, M.; Keizer, S.; Mairesse, F.; Schatzmann, J.; Thomson, B.; and Yu, K. 2010. The hidden information state model: A practical framework for pomdp-based spoken dialogue management. *Computer Speech & Language* 24(2):150–174.