Abstracting Away Low-Level Details in Service Robotics with Fuzzy Fluents

Stefan Schiffer[1], Alexander Ferrein[2], and Gerhard Lakemeyer[1]

[1] Knowledge Based Systems Group (KBSG)
RWTH Aachen University, Aachen, Germany
{schiffer,gerhard}@cs.rwth-aachen.de

[2] Mobile Autonomous Systems & Cognitive Robotics Institute (MASCOR)
FH Aachen University of Applied Sciences, Aachen, Germany
ferrein@fh-aachen.de

Abstract. In domestic service robotic applications, complex tasks have to be fulfilled in close collaboration with humans. We try to integrate qualitative reasoning and human-robot interaction by bridging the gap in human and robot representations and by enabling the seamless integration of human notions in the robot's high-level control. The developed methods can also be used to abstract away low-level details of specific robot platforms. These low-level details often pose a problem in re-using software components and applying the same programs and methods in different contexts. When combined with methods for self-maintenance developed earlier these abstractions also allow for seamlessly increasing the robustness and resilience of different robotic systems with only little effort.

1 Introduction

In our previous work we focused on the design of intelligent behaviours in domestic service robotics applications. We developed the logic-based high-level robot programming and plan language Readylog [2, 1], which allows for specifying complex behaviours in a quite intuitive fashion. Readylog supports, for instance, decision-theoretic planning, online-passive sensing and is able to deal with uncertainty. It is a member of the GOLOG language family and is based on the Situation Calculus [5, 6]. We successfully applied Readylog in dynamic real-time domains such as robotic soccer and also on a domestic service robot platform; in particular we participated successfully at RoboCup@Home competitions in the past [8, 12].RoboCup@Home [15, 16, 14] is a robot competition for domestic service robots under the roof of the RoboCup Federation. The robots have to fulfill tasks ranging from rather simple functions such as guiding a human through an apartment over serving drinks to complex missions like cooking meals or acting as a host at a party. Robots in @home are fully autonomous and they must be accessible and usable by non-expert users; this is most often realized via natural language-based control.

2 Qualitative Notions and Self-Maintenance

When operating in human-populated environments in collaboration with humans, it becomes obvious that numeric representation of the robot cannot quite be understood by a human. Humans rather tend to use qualitative representations of space than numerical ones. While humans are very good at interpreting qualitative distances such as "far" or "close" or orientations such as "the cup is left on the table behind the coffee pot" robots are not. One possibility (for robots) to deal with such qualitative notions is to use fuzzy predicates. A fuzzy predicate associates a number of quantitative values from a given domain (e.g. distances $1, 2, \ldots$) to a linguistic term (e.g. "far" or "close"). The association of quantitative and qualitative values is specified with a so-called membership function. Each quantitative value has a membership value, which defines, to what degree a value falls into a certain fuzzy category. An example of a membership function for qualitative unit distance is depicted in Figure 1. We extended the language Readylog with qualitative notions to be able to instruct the robot with command such as "get me the left cup on kitchen table". To this end, we developed fuzzy representation in the Situation Calculus [3, 11, 10] and integrated a control strategies into Readylog [9, 4].

Another challenge for robots working in human environments is that they need to operate for extended periods of time. Therefore, they need to be robust against internal shortcomings and they should continue to be functional even in the face of failures. More precisely, robots should be able to handle and deal with as many errors by themselves as possible. For that purpose, we developed a method to realize a limited form of self-maintenance [13]. It uses explicitly formulated constraints that associate actions with desired internal states of the robot. Before executing an action the robot checks whether the constraints are met and it schedules appropriate counter measures if this is not the case. For example, one could specify that the robot should only drive around if the collision avoidance module is running and in working condition. If then, the robot's next action is to move to a certain position but the collision avoidance component is non-functional, the system would restart it before attempting to execute the move action.

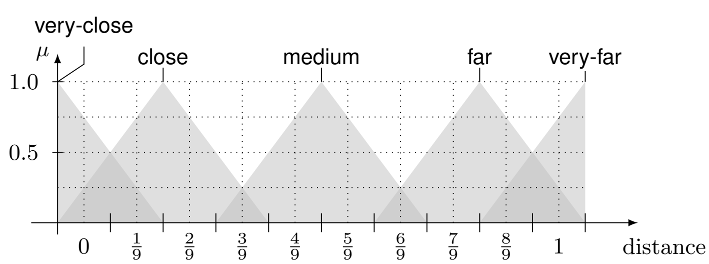

Fig. 1. Membership function for qualitative (unit) distance

3 Fuzzy Fluents for Abstracting Away Low-level Details

The basic motivation behind using fuzzy fluents in the high-level controller of our domestic service robots was to express qualitative facts about the world and reason with them. For translating qualitative notions “left of” or “close” into a quantitative representation that the low-level robot control system can use to perform its actions, we use fuzzy predicates for dividing continuous spaces (e.g. distance and orientation) into a number of equivalence classes. An additional nice feature of fuzzy predicates is that certain quantitative values can belong to more that one class at a time and that complex queries can be evaluated (see [3, 11, 10] for technical details). The linguistic classes and the membership functions can also be used to model different contexts. For instance, “fast” may mean 100 km/h in a self-driving car, while for a humanoid robot it means “100 cm/s”. With modelling the membership function differently in different contexts, these details can he hidden away from the high-level reasoning system. This allows for using the same high-level controller in different contexts and on different platforms by simply providing an appropriate membership function of the qualitative terms used in the program. Extending this concept of “abstracting away” quantitative details of the low-level system can also be used to model self-properties of the robot or to allow a form of hardware abstraction of the low-level hardware system.

As an example application, consider a delivery robot scenario. Two different robot platforms might be equipped with different drive concepts, different payload capabilities, and different energy components. As a result, these two robots will have different maximum speed and they will be able to travel different total distances before they need to recharge. With our qualitative abstractions in place, both robots could use the same control program to conduct their delivery service. The only difference would be the individual membership functions for relating numerical values to their qualitative counterparts that are being used in the high-level program. Also, different payloads would automatically be accounted for when the control programs and constraints mentioned above would be using notions such as “heavy” by appropriate membership functions for different hardware platforms.

4 Discussion

In this paper we illustrated the use of qualitative notions for abstracting away low-level details in service robotic applications. Qualitative notions are realized by fuzzy fluents using membership functions that specify how much a numerical value belongs to a qualitative category. By using only qualitative terms in high-level control programs and also for formulating constraints that exist between actions and internal states of the robot, differences for varying platforms can be accounted for by just using different membership functions. This facilitates creating high-level control that is easily transferable from one robot platform to another. The proposed method could also be used with learning individual qualitative abstractions for different persons [7] or platforms.

References

1. Ferrein, A.: Robot controllers for highly dynamic environments with real-time constraints. Künstliche Intelligenz 24(2), 175–178 (2010)
2. Ferrein, A., Lakemeyer, G.: Logic-based robot control in highly dynamic domains. Robotics and Autonomous Systems, Special Issue on Semantic Knowledge in Robotics 56(11), 980–991 (2008)
3. Ferrein, A., Schiffer, S., Lakemeyer, G.: A fuzzy set semantics for qualitative fluents in the situation calculus. In: Proceedings of the International Conference on Intelligent Robotics and Applications (ICIRA'08), LNCS, vol. 5314, pp. 498–509. Springer (October 15-17 2008)
4. Ferrein, A., Schiffer, S., Lakemeyer, G.: Embedding fuzzy controllers into golog. In: Proceedings of the IEEE International Conference on Fuzzy Systems (FUZZ-IEEE-09). pp. 498–509. IEEE Press (August 20-24 2009)
5. McCarthy, J.: Situations, actions and causal laws. TR, Stanford University (1963)
6. Reiter, R.: Knowledge in Action. Logical Foundations for Specifying and Implementing Dynamical Systems. MIT Press (2001)
7. Robinson, V.B.: Individual and multipersonal fuzzy spatial relations acquired using human-machine interaction. Fuzzy Sets and Systems 113(1), 133–145 (2000)
8. Schiffer, S., Ferrein, A., Lakemeyer, G.: Football is coming home. In: Proceedings of the 2006 International Symposium on Practical Cognitive Agents and Robots (PCAR'06). pp. 39–50. ACM, New York, NY, USA (November 27-28 2006)
9. Schiffer, S., Ferrein, A., Lakemeyer, G.: Fuzzy representations and control for domestic service robots in golog. In: Jeschke, S., Liu, H., Schilberg, D. (eds.) Proceedings of the 4th International Conference on Intelligent Robotics and Applications (ICIRA 2011). LNCS, vol. 7102, pp. 241–250. Springer (2011)
10. Schiffer, S., Ferrein, A., Lakemeyer, G.: Reasoning with qualitative positional information for domestic domains in the situation calculus. Journal of Intelligent and Robotic Systems 66(1-2), 273–300 (2012)
11. Schiffer, S., Ferrein, A., Lakemeyer, G.: Caesar: An Intelligent domestic Service Robot. Intelligent Service Robotics 5(4), 259–273 (2012)
12. Schiffer, S., Niemüller, T., Doostdar, M., Lakemeyer, G.: Allemaniacs@home 2009 team description. In: Proceedings CD RoboCup 2009. Graz, Austria (2009)
13. Schiffer, S., Wortmann, A., Lakemeyer, G.: Self-Maintenance for Autonomous Robots controlled by ReadyLog. In: Ingrand, F., Guiochet, J. (eds.) Proceedings of the 7th IARP Workshop on Technical Challenges for Dependable Robots in Human Environments. pp. 101–107. Toulouse, France (June 16-17 2010)
14. Wisspeintner, T., van der Zant, T., Iocchi, L., Schiffer, S.: Robocup@home: Scientific Competition and Benchmarking for Domestic Service Robots. Interaction Studies. Special Issue on Robots in the Wild 10(3), 392–426 (2009)
15. van der Zant, T., Wisspeintner, T.: RoboCup X: A Proposal for a New League Where RoboCup Goes Real World. In: Bredenfeld, A., Jacoff, A., Noda, I., Takahashi, Y. (eds.) RoboCup 2005: Robot Soccer World Cup IX. LNCS, vol. 4020, pp. 166–172. Springer (2006)
16. van der Zant, T., Wisspeintner, T.: Robotic soccer. In: Lima, P. (ed.) Robotic Soccer, chap. RoboCup@Home: Creating and Benchmarking Tomorrows Service Robot Applications, pp. 521–528. I-Tech Education and Publishing (2007)