Solving Navigation Tasks with Learned Teleo-Reactive Programs

Blanca Vargas and Eduardo F. Morales
Department of Computer Science, National Institute of Astrophysics, Optics and Electronics - Puebla, México
{blanca,emorales}@ccc.inaoep.mx

I. Introduction

Programming a robot to perform tasks in dynamic environments is a complex process. Teleo-Reactive Programs (TRPs) [1] have proved to be an effective framework to continuously perform a set of actions to achieve particular goals and react in the presence of unexpected events, however, their definition is a difficult and time-consuming process. In this paper, it is shown how a robot can learn TRPs from human guided traces. A user guides a robot to perform a task and the robot learns how to perform such task in similar dynamic environments. Our approach follows three steps:

- **Transforms traces** with low-level sensor information into high-level traces based on natural landmarks. A natural landmark identification process [3] is applied to produce a smaller set of more meaningful information consisting of discontinuities, corners and walls (see Fig. 2(a)).
- **Learns basic TRPs**. High-level traces and background knowledge are given to the ILP system ALEPH [4]. The output is a set of TRPs that express when to perform an action to achieve simple tasks, e.g., *avoid* (see Fig. 2(c)). Learned concepts (e.g. *front-zone, orientation-zone*) are used as background knowledge.
- **Learns hierarchical TRPs**. We introduce FOSeq, an algorithm to learn TRPs that is able to produce particular sequences of actions satisfying a specific goal e.g., goto (see Fig 2 (c)).

The learned TRPs were used to solve navigation tasks in different unknown and dynamic environments, both in simulation and in a service robot called Markovito. Figure 1 shows the process for learning basic and complex TRPs.

II. FOSeq: Learning Complex TRPs

The user controls the robot to achieve a particular goal, e.g., go to a specific area. The low-level information from the sensors is transformed into a high-level trace of state-action pairs from which applicable TRPs, previously learned, are identified and replaced in the sequence. FOSeq is then used to learn how to produce similar sequences. Given a set of sequences of TRPs, FOSeq:

- Learns a grammar for each sequence. FOSeq identifies repeated elements in the sequence to construct a grammar.
- Evaluates the sequences with each learned grammar. The evaluation is based on the percentage of the sequence that can be generated by the grammar.
- Selects the best evaluated grammar and a list of candidates to be generalized. The candidates are those that are likely to improve the evaluation of the best grammar.
- Applies a generalization process to the selected grammars. It uses adaptation rules to modify/add a new rule.
- Transforms the grammar into a logic program.

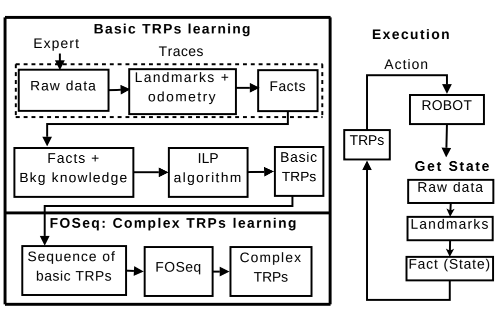
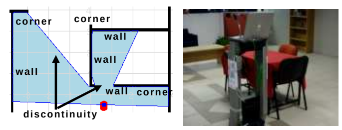


Fig. 1. An overview of how to learn basic and complex TRPs



	Conditions	Actions
goto	in-goal	nil
goto	obstacle	avoid
...	...	...
avoid	not-obstacle	nil
avoid	right-obstacle	turn-left
avoid	left-obstacle	turn-right

(a) (b) (c)

Fig. 2. (a) The landmark identification process recognizes walls, corners and discontinuities (b) Markovito visiting several places using a learned TRP (c) Some conditions and actions for the goto and avoid TRPs

III. Experiments

The learning process was performed using simulation in the Player/Stage robot platform [2]. The TRPs were tested in simulation and as the navigation module of a PeopleBot ActivMedia robot called Markovito. To perform the navigation tasks it used a sonar ring and a Laser SICK LMS200. The tasks to accomplish were: (i) following a human under user commands, (ii) navigating to several places in the environment designated semantically (see Fig. 2(b)), (iii) finding one of a set of different objects in a house and (iv) delivering messages and/or objects between different people. The first three tasks are part of the RoboCup@Home challenge.

References

[1] S. Benson and N. J. Nilsson, Reacting, Planning, and Learning in an Autonomous Agent, *in Machine Intelligence*, vol. 14, 1995, pp. 29-62.
[2] R. Vaughan and B. Gerkey and A. Howard, On device abstractions for portable, reusable robot code, *Proceedings of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS) http://playerstage.sourceforge.net*, Las Vegas, Nevada, 2003.
[3] S. Hernández and E. Morales, Global localization of mobile robots for indoor environments using natural landmarks, *IEEE International Conference on Robotics, Automation and Mechatronics (RAM)*, 2006.
[4] A. Srinivasan, *The Aleph 5 Manual*, 2005.