

Designing Reactive Robot Controllers with LTLMoP*

Cameron Finucane and Gangyuan Jing and Hadas Kress-Gazit

{cpf37, gj56, hadaskg}@cornell.edu
Cornell University, Ithaca, NY 14853

Abstract

This paper shows an example application of the LTLMoP mission planning toolkit, in which an Aldebaran Nao and an iRobot Create each play a very basic game of hide-and-seek.

Introduction

Most robots currently available to the consumer, medical, and industrial markets are single-purpose, pre-programmed devices. However, as hardware technology matures, we will see more and more general-purpose robots become commonplace. This development, however, will require an accompanying advance in human interfaces, so that end-users can have the robots perform desired tasks on the fly, without needing specialized technical knowledge or being limited to a repertoire of pre-specified behaviors.

The Linear Temporal Logic Mission Planning (LTLMoP) toolkit¹ presents one possible solution to this problem, avoiding the tedious and error-prone process of manual controller design by automatically synthesizing controllers from task specifications written by a user in structured English. In other words, it is only necessary for users to input *what* they want the robot to do, not *how* the robot should specifically accomplish it.

In this paper, we show an example application of the toolkit in which we have an Aldebaran Nao² and an iRobot Create³ each play a very basic game of hide-and-seek with a person.

Technical Approach

In LTLMoP, a task specification consists of behavioral requirements for the robot (i.e. goals and constraints) as well as any assumptions that can be made about the behavior of the environment. The structured English specification is transformed into a formula in Linear Temporal Logic (LTL), specified over binary propositions that correspond to abstracted sensor states (e.g. “see-player”), atomic

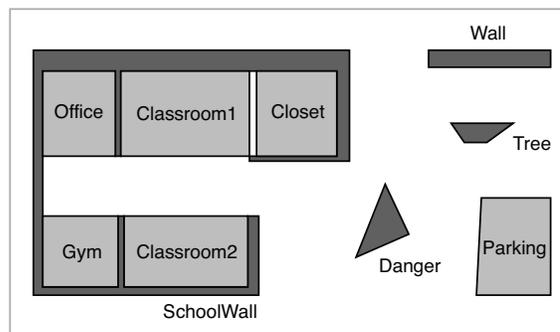


Figure 1: Map of school

robot behaviors (e.g. “whistle”), and regions in the partitioned workspace.

Based on region adjacency information from the map and behavioral requirements from the specification, a high-level discrete controller is automatically synthesized using the technique in (Kress-Gazit, Fainekos, and Pappas 2009).

The synthesized automaton is guaranteed by construction to satisfy the specification from which it was generated; if the specification asks the robot to do something that it cannot (i.e. it is unsynthesizable), then no controller will be generated. Also, because of the structure of the LTL formula, guarantees about the robot’s behavior only hold as long as no environment assumptions are falsified.

Based on the choice of robot, this high-level controller is then composed with appropriate low-level continuous controllers to provide real-world sensing, movement, and action. Note that this means that the same high-level controller can be used for multiple robots, as long as they provide equivalent capabilities (e.g., they both have a way to sense the presence of a person); this feature will be demonstrated in the example in this paper.

Task Specification

The game of hide-and-seek will take place on the map shown in Figure 1.

The rules for the game are as follows:

- Start in the parking lot.
- Avoid obstacle regions.

*This work is supported by ARO MURI (SUBTLE) W911NF-07-1-0216.

Copyright © 2011, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

¹Available from <http://ltlmoop.github.com>

²<http://www.aldebaran-robotics.com/>

³<http://www.irobot.com>

- If you are the seeker:
 - Go to the parking lot and count until you hear a ready whistle.
 - Search all the places someone might hide.
 - Once you’ve found someone, you are now a hider.
- If you are a hider:
 - Go to the parking lot and wait for counting to start.
 - Go hide in a hiding spot and whistle.
 - Once found, you are now the seeker.

We define the hiding spots for the robot to be Closet, Office, and between Tree and Wall. The spots to search are those hiding spots plus both Classrooms and the Gym. Obstacles are the Wall, SchoolWall, Tree, and Danger.

Listing 1 shows how a subset of this specification would be represented in the structured English used by LTLMoP.

Listing 1 Excerpt of structured English specification for hide-and-seek example

```

### Game start and end conditions ###
playing is set on ((seeker and Parking and hear_whistle)
                  or (not seeker and Parking and hear_counting))
                  and reset on see_player
If you are not activating playing then go to Parking and
stay there
### Hiding behavior ###
If you are activating playing and not seeker then go to
Closet or Office or between Tree and Wall and stay
there
Do whistle if and only if you were not activating hide
and you are activating hide

```

Controller Implementation

After inputting the map and specification into LTLMoP, a controller can be built by clicking “compile.” This controller can then be run with either a simulated or real robot. For details, see (Finucane, Jing, and Kress-Gazit 2010).

Simulation

A simulation environment can be used to check that the behavior you specified is in fact what you intended. Since in this case we are only interested in testing the high-level controller, sensing is accomplished by using toggle buttons for each proposition, and actuators simply print their name to the log when activated.

Physical Experiment

For this demonstration, a physical experiment was run with a Vicon motion capture system providing pose information. The mappings between physical sensors/actuators and specification propositions are shown in Figures 2 and 3. A basic potential-field method was used for motion control between regions. A still image from the experiment can be seen in Figure 4.

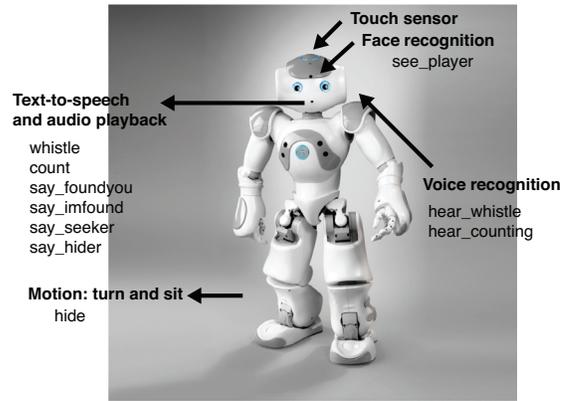


Figure 2: Sensor and actuator mapping (Nao)

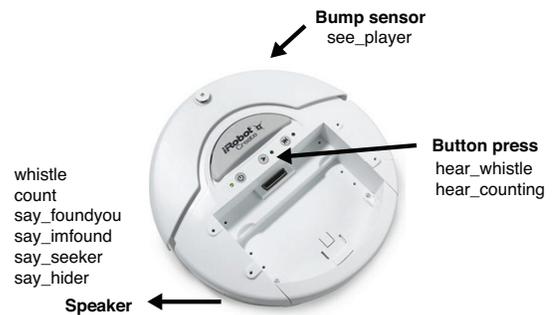


Figure 3: Sensor and actuator mapping (Create)

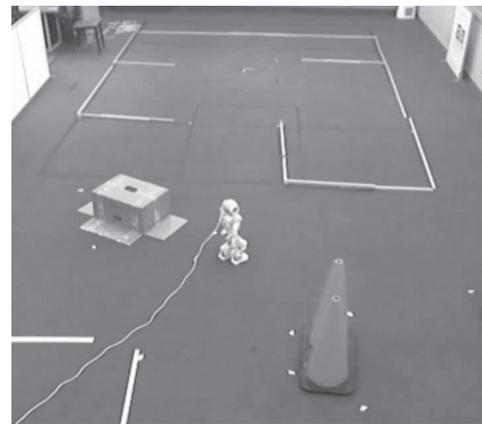


Figure 4: Nao heading to a hiding spot

References

Finucane, C.; Jing, G.; and Kress-Gazit, H. 2010. LTLMoP: Experimenting with language, temporal logic and robot control. In *IROS*.

Kress-Gazit, H.; Fainekos, G. E.; and Pappas, G. J. 2009. Temporal-logic-based reactive mission and motion planning. *IEEE Transactions on Robotics* 25(6):1370–1381.