Real-Time Systems

Lab 4: Adding Guidance and Planning

Due: March 8, 2018 (Before Class)

Introduction

Now that you've written a controller for Ringo we can use it, together with a reactive planning task and a guidance task, to be able to have better control and accomplish a mission. The mission will be to navigate through a maze to a goal location avoiding obstacles along the way. As a result you will need to implement goal seeking behavior and obstacle avoidance into your planner. I suggest using a "reactive" planning strategy wherein your Ringo responds to sensed obstacles and moves in the direction of a goal via a potential field. Feel free to use online resources to come up with ideas for a guidance law and planner, though you really can do it on your own. This is another chance for you to be creative and "design" something.

Write any support tasks that might be required (e.g., a sensing task). At the end of this lab you should probably have around 3-4 tasks: sensing, planning, guidance, and control. These can be subdivided into more tasks if needed. If you have too many tasks you will likely run into memory space constraints resulting in odd behavior that is difficult to debug.

Finally, provide an estimate for the WCET for each task in the system. This will be needed for the next lab.

Intro to Guidance and Planning

"Guidance," "planning," and "control" are really just abstractions for algorithmically and/or architecturally breaking up the task of moving a robot from point to point in an environment. As such, it is difficult to provide a precise definition of each of them. "Control" usually means low-level control where an input to an actuator is computed to achieve some step response. As an example, in Ringo this might mean moving forward some amount or executing a single rotation. A controller, on its own, is typically insufficient to accomplish a mission.

"Planning" is often divided into "deliberative" and "reactive" strategies. Deliberative planning usually means requiring full "world" knowledge to devise a plan containing high level motions and/or tasks that the robot must do to accomplish a mission. As an example, for Ringo, this could mean generating the entire path to navigate through an environment that has obstacles provided you know where the goal and obstacles are. In that scenario, a valid plan would break the mission down into waypoints that avoid obstacles and perhaps meets some performance objectives (like not taking an entire day to get there).

"Reactive" planning strategies typically only respond in the moment to stimuli and make a decision only about the next action (as opposed to a full plan to accomplish the mission). As a result they tend to consist of a goal and a set of condition-action rules. These strategies are typically much simpler but exhibit suboptimal behavior. Most successful planners utilize both of these strategies; high level plans for optimal behavior coupled with low-level reactive actions capable of dealing with a dynamic environment.

"Guidance" often refers to a middle layer that bridges higher level plans from a planning algorithm with the small motions the controller is capable of achieving. If you wrote a controller capable of moving in a straight line or heading, and a controller capable of rotations, then the guidance law should take the plan of waypoints and break this plan down into sequential steps that can be achieved by your controllers.

Please note that the division of a mission into plans, guidance laws, and controllers is arbitrary and depends on mission requirements and your design. The final exam will ask you to navigate one or more polygon shapes in some reasonable amount of time. You should design a planner and guidance law, that, together with your controllers is capable of accomplishing that type of "mission." Once you have a guidance law and planner, be sure to interface it with the controller task(s) written before to demonstrate your Ringo can achieve mission objectives.

NOTE: There are MANY resources for learning about robot planning online. Feel free to use them. But be warned that much of it is overly complex for our purposes. We're focused on RTS principles, not so much on guidance, planning, and control. For additional resources, and learning, have a look at [2, 3, 1, 4, 5] for a few of my favorites (you should be able to look these up online and/or get the book from the library).

Worst Case Execution Time (WCET)

You can use a static analysis tool or a measurement based technique to estimate WCET of each task. A reasonable list of WCET tools can be found at `https://www.rapitasystems.com/WCET-Tools`. Alternatively, you can do a statistical analysis by running your task enough times and estimating WCET. Here "enough times" is ill-defined as discussed in class. Whichever method you use, remember to add a ~20% cushion to the time for a margin of "safety."

What To Do

The final exam will consist of navigating your Ringo through a maze with obstacles to a goal location. To enable this, I recommend you design a reactive planning strategy for goal seeking behavior and obstacle avoidance using Ringo's sensors. The easiest goal seeking behavior would be to calculate the direction of the goal and simply move in that direction at all times unless confronted with an obstacle. Upon encountering an obstacle you'll need to make a decision about what to do. Be sure to utilize the controller you wrote in the previous lab.

Finally, once your tasks are all working estimate their WCET using one of the methods (or some combination thereof) we discussed in class. I suggest setting up some kind of infrastructure to do this since you will need to redo it in the next lab and compare the two sets of results. Your infrastructure might include some timing code, a mechanism for collecting timestamps, and a method for extracting them and estimating the WCET.

What to submit

1. **(35 points)** Zip the entire Arduino project so that we can just unzip and execute the source code.
   - Please include a diff of the source code between this lab and previous one. Please use a "diff" tool of some kind that highlights the changes from previous files (e.g. `https://www.diffchecker.com/`, meld, github, etc.). If you did not build on previous code at all, state such in your document and do not provide a "diff."
2. **(30 points)** Documentation of the task(s) you have implemented for this lab. (no more than 1 page)
   - Provide an algorithmic overview of the guidance and planner laws you've written
   - Provide a WCET estimate for each task you have
   - Describe each implemented task briefly and how you decided how to break up the behaviors into different tasks (or why you chose to use a single task). Classify any tasks into periodic, aperiodic, or sporadic, and provide justifications on why a task is of a certain type and what were the thoughts while deciding the frequency of the tasks.
3. **(35 points)** Record a short video of your Ringo navigating a very simple maze to get to a goal location. Demonstrate your Ringo avoiding at least two obstacles and getting to a goal state at least 3 feet from the starting point. This will be good preparation for the final exam.

- Be sure to describe or narrate the test that Ringo is doing that demonstrates that you have accomplished the objectives of the lab.
- Please upload the video to YouTube, Vimeo, or something similar and then provide me with a link. Just put the link somewhere in the writeup you did in #2 above.

4. Upload all of this into "handin" at `https://cse-apps.unl.edu/handin`

References

[1] On three-layer architectures. In *Artificial intelligence and mobile robots*, pages 195–210. AAAI press Cambridge, MA, 1998.

[2] Jim Blythe and W. Scott Reilly. Integrating reactive and deliberative planning for agents. Technical report, DTIC Document, 1993.

[3] R Peter Bonasso, R James Firby, Erann Gat, David Kortenkamp, David P Miller, and Mark G Slack. Experiences with an architecture for intelligent, reactive agents. *Journal of Experimental & Theoretical Artificial Intelligence*, 9(2-3):237âĂŞ256, 1997.

[4] C Goerzen, Zhaodan Kong, and Bernard Mettler. A survey of motion planning algorithms from the perspective of autonomous uav guidance. *Journal of Intelligent and Robotic Systems*, 57(1-4):65âĂŞ100, 2010.

[5] S J Russell and P Norvig. *Artificial Intelligence: A Modern Approach*. Prentice Hall, 3 edition, 2010.