Arbitration

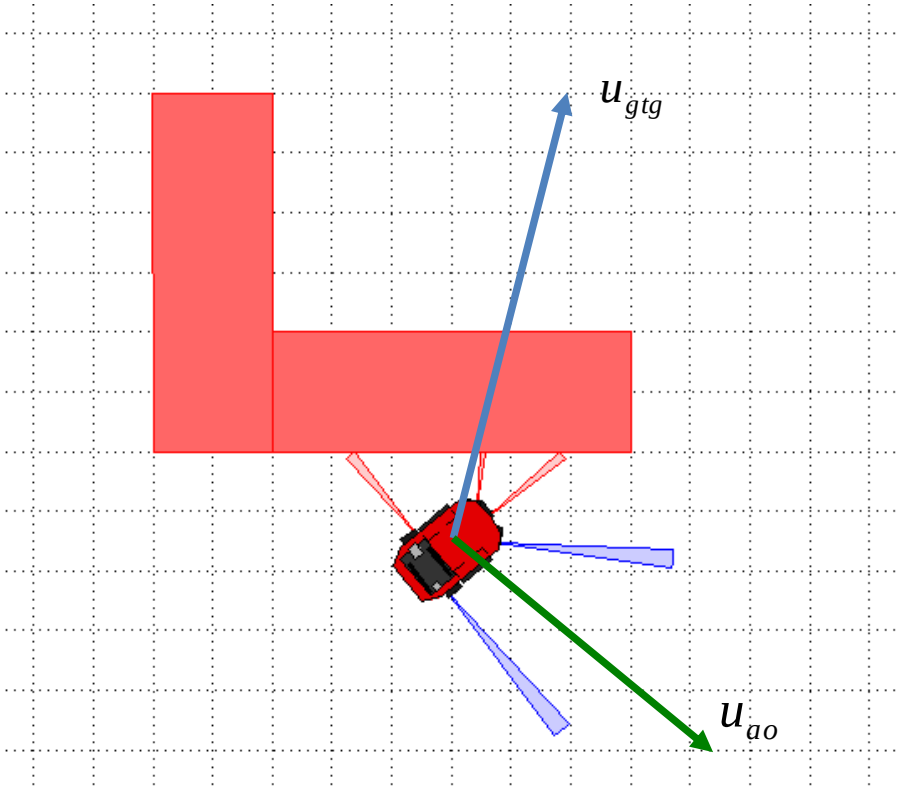
Control of Mobile Robots: Programming & Simulation Week 5

Jean-Pierre de la Croix
ECE Ph.D. Candidate
Georgia Inst. of Technology

Overview

- We will use two arbitration techniques, blending and hard switching, to drive to a goal while avoiding obstacles.
  1. Blend go-to-goal and avoid-obstacle vectors in one controller.
  2. Switch between go-to-goal and avoid-obstacle controllers separately.
  3. Use the blended controller as an intermediary.

Blending



- Two controllers in one.

combination

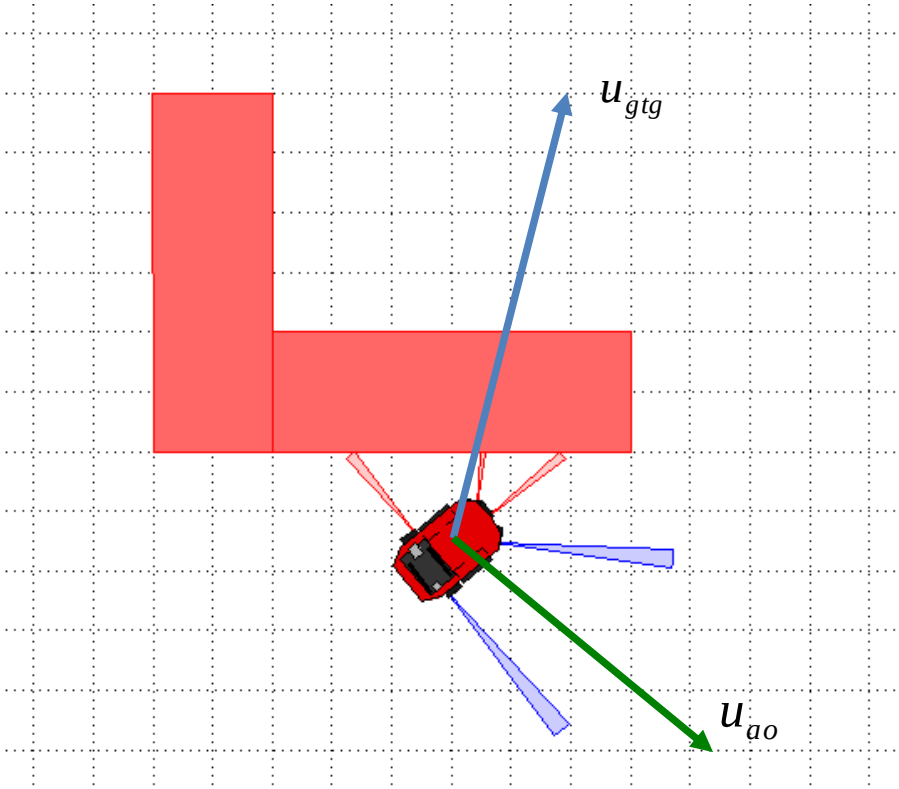
$$u_{ao,gtg} = \alpha u_{ao,n} + (1-\alpha) u_{gtg,n}, 0 \le \alpha \le 1$$

$$u_{ao,n} = \frac{u_{ao}}{\|u_{ao}\|}$$ ← normalized version

α = fraction

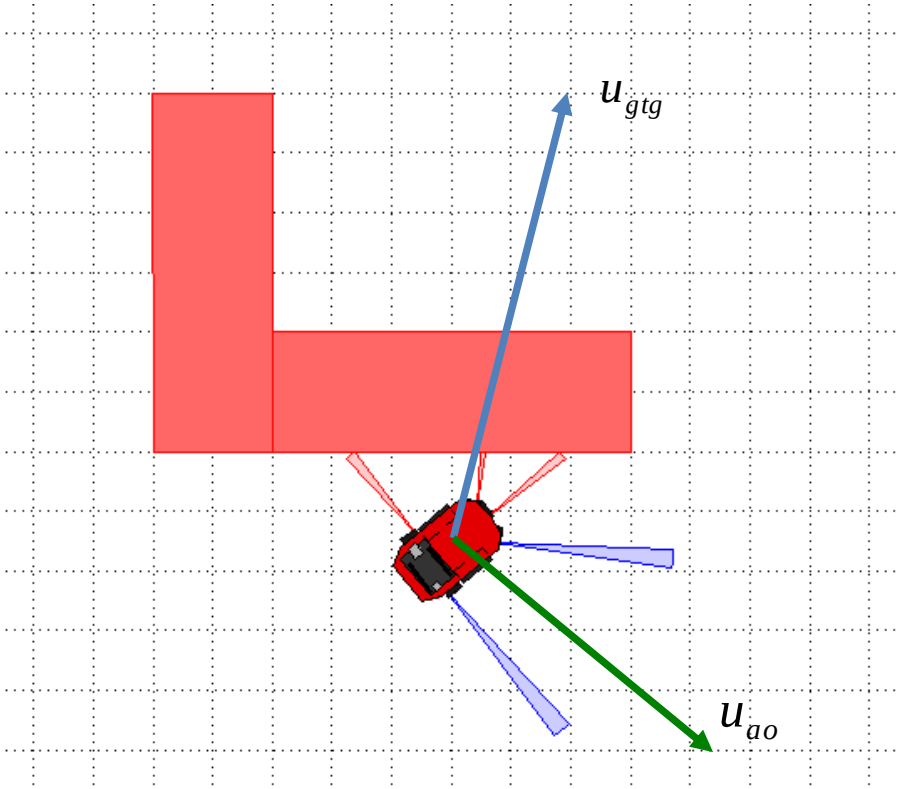
↑ unit length for equal balance

Hard-Switching



- One controller at a time.
- Switch from go-to-goal to avoid-obstacles near any obstacles.

Intermediary



- Avoid chattering by using the blended controller between go-to-goal and avoid obstacles.

Supervisor and State Machine

- Each controller is also a state, and the supervisor can switch between states (controllers).
- For example,

```
obj.switch_to_state('go_to_goal');
```

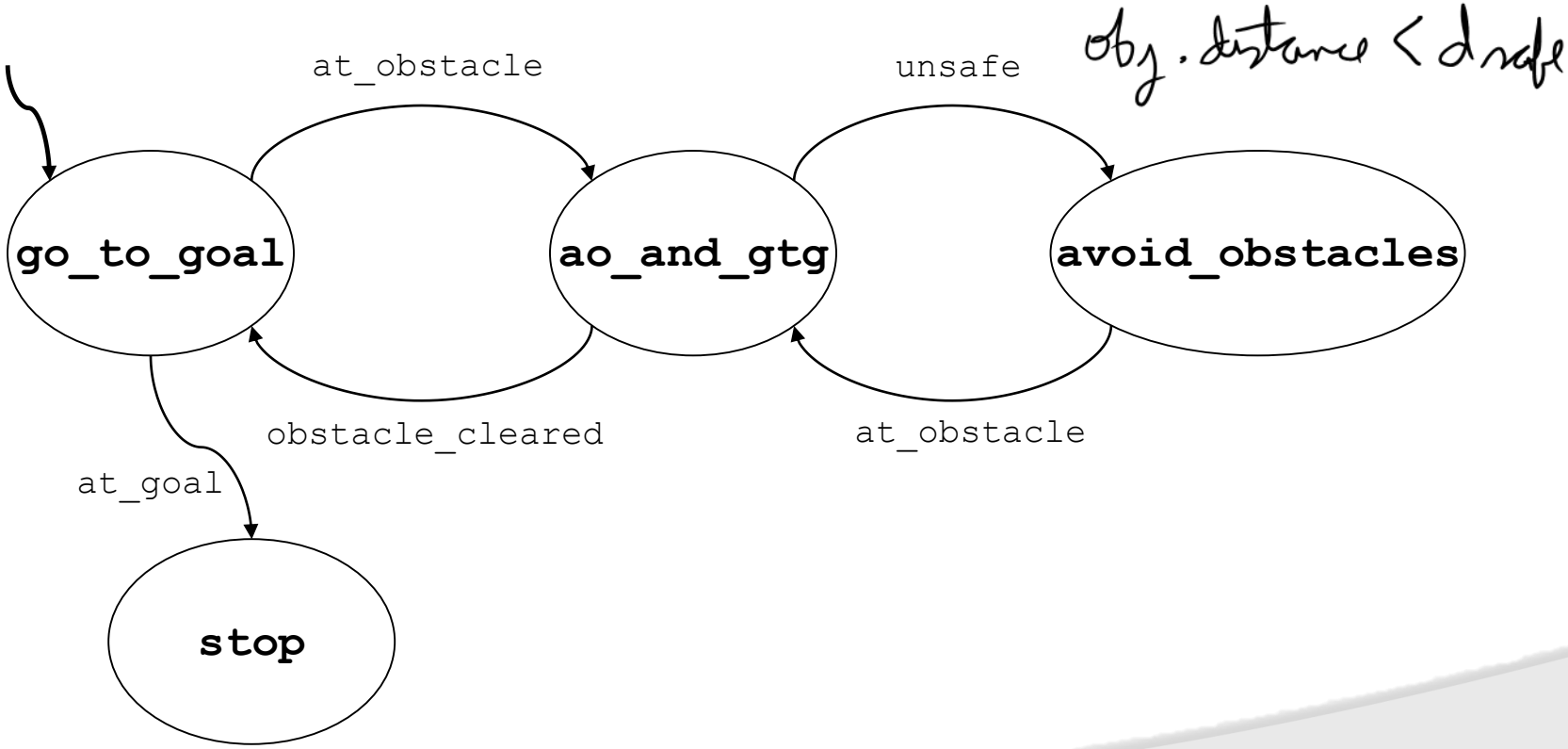
Supervisor and State Machine

- Switching between states (controllers) happens when an event (condition) occurs.
- For example,

```
obj.check_event('at_obstacle');
```

Returns true if any of the IR sensors record a distance less than `obj.d_at_obs`.

A State Machine



Files of Interest

- The state machine will be implemented in the supervisor's execute function.

```
+simiam/+controller/+quickbot/QBSupervisor.m
```

- The blended controller will be implemented as its own class using code from GoToGoal.m and AvoidObstacles.m

```
+simiam/+controller/AOandGTG.m
```

Tips

- Refer to the section for Week 5 in the manual for more details!
- Experiment with different ways of blending go-to-goal and avoid-obstacles.
- Also, experiment with different state machines (ways of stringing states and events together).