



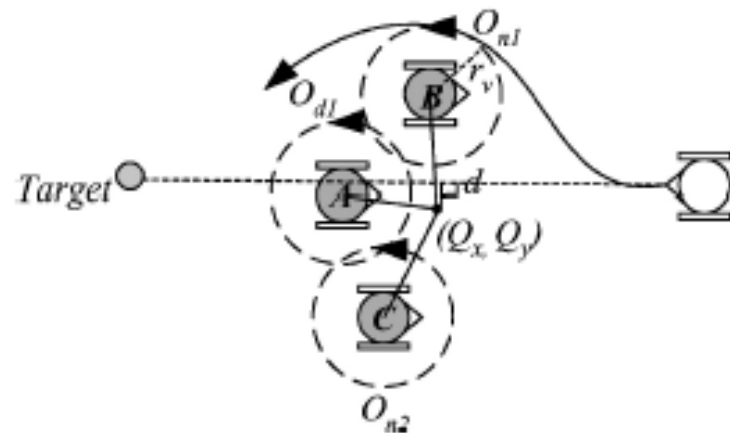


Lessons Learned: Vision

- No benefit to robots that move faster than they can be tracked by the vision system
- Use bounding boxes to maximize field scan efficiency
- Experiment with color block disambiguation and noise elimination; try the dilate and smooth method
- Be prepared to adjust thresholding to fit changes in lighting

Lessons Learned: Control

- Use feed-forward model reference control to achieve superior control of your robots
- Learn the details of the Kalman filter so you can tune it as needed, for a big advantage
- Work on predicting and intercepting a moving ball
- Use limit cycles for obstacle avoidance and virtual obstacles, for dominating ball control



Lessons Learned: Tools

- Tailor your parameter control GUI to meet your needs, helping you be fast and efficient
- Create a parsing tool to read in parameters from files, rather than hardcoding and recompiling
- Use the simulator; figure out what it's good for and what it's not good for
- Coordinate the work on the code, use source control
- Take advantage of global variables shared by many classes

Lessons Learned: Team

- Choose team early and maintain unity
- Have a mix of talents
- Define the roles, but be conversant in all other roles in order to help each other out
- Put in the time early to get ahead, to avoid a late scramble to catch up
- Must try new things; they won't all work, but you need to find an edge
- Decide that you are going to win, then go do it

Lessons Learned: Why Us

- Worked harder than the other teams, put in more time
- We had superior robot control
- We were the quickest to adapt to changes, like new hardware
- We worked hard to become the top team early and spent the following weeks keeping that edge
- We spent more time testing and characterizing the hardware
- Increased our margin of error by correcting every little problem; things went wrong, but we still won