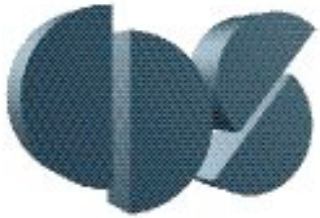
CDS 110a: Lecture I-1
Course Project Introduction



Richard M. Murray

22 October 2006

Goals

- Provide enough information for people to decide whether they want to do the course project
- Give a high level introduction to Alice, so that you know what the project is trying to accomplish

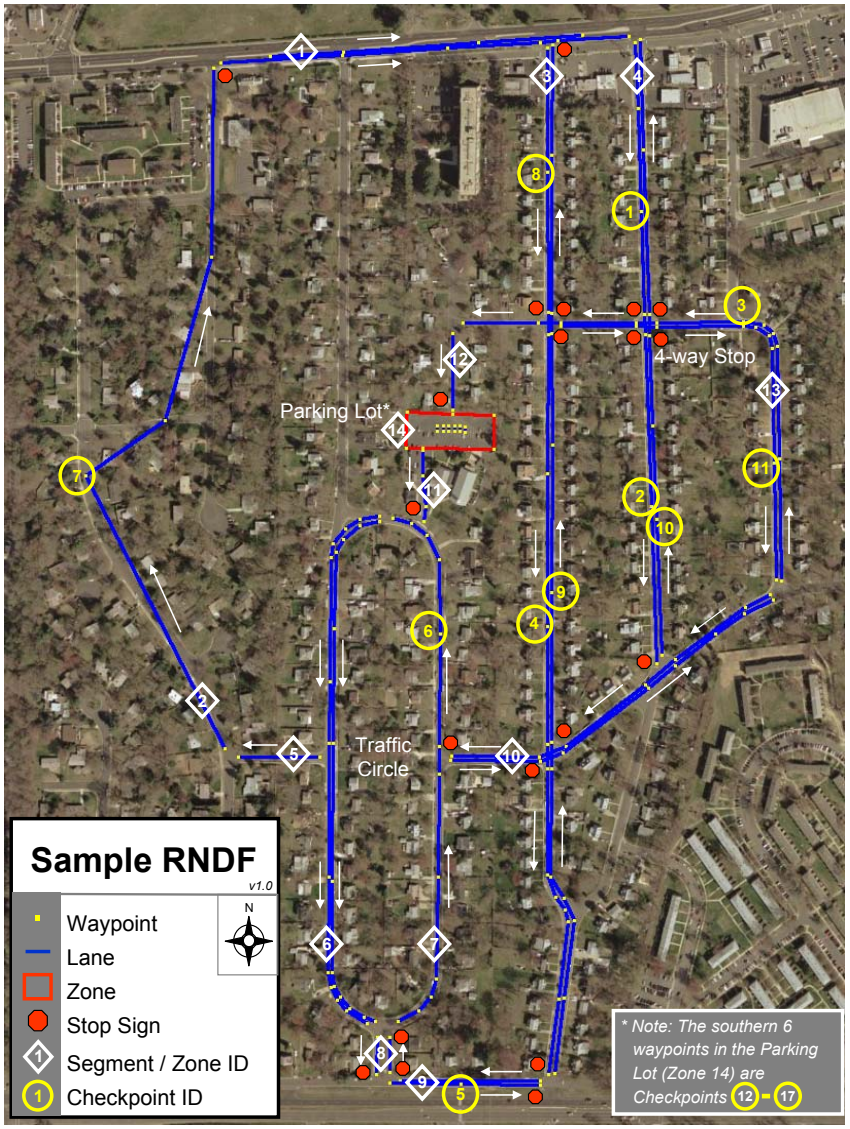
Reading

- Course syllabus (project information section)
- JFR05 paper (for high level overview of Alice) - available on the web

2007 DARPA Grand Challenge (Urban Challenge)

Autonomous Urban Driving

- 60 mile course, less than 6 hours
- City streets, obeying traffic rules
- Follow cars, maintain safe distance
- Pull around stopped, moving vehicles
- Stop and go through intersections
- Navigate in parking lots (w/ other cars)
- U turns, traffic merges, replanning
- Prizes: $2M, $500K, $250K



Urban Driving

Video from 29 Jun 06 field test

- Front and side views from Tosin
- Rendered at 320x240, 15 Hz
- Manually synchronized

- Moving obstacle detection, separation, tracking and prediction
- Decision-making
- Lane markings (w/ shadows)

Some challenges

Team Caltech

Team Caltech

- Started in 2003, for DGC04
- Over 100 undergraduates + grad students, faculty and volunteers

Alice

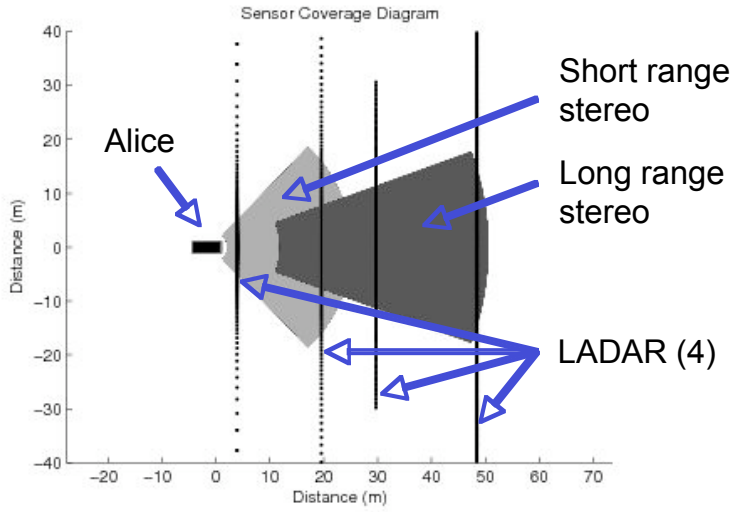
- 2005 Ford E-350 Van
- 5 cameras: 2 stereo pairs, roadfinding
- 5 LADARs: long, med*2, short, bumper
- 2 GPS units + 1 IMU (LN 200)

Computing

- 6 Dell PowerEdge Servers (P4, 3GHz)
- 1 IBM Quad Core AMD64 (fast!)
- 1 Gb/s switched ethernet

Software

- 15 programs with ~100 exec threads
- 100,000+ lines of executable code (good programmer does <100/day)



Course Project Administration

1. **Attend intro lecture on Alice (today) + implementation recitation section (Wed, 8-9)**
   - Will provide the information that you need to know about how Alice works
   - Later in term, recitation sections will provide forum for discussing problems

2. **Work in a group of 2-3 people to implement a specific control function on Alice**
   - Project 1: lateral control in forward and reverse
   - Project 2: lateral control, with gain scheduling
   - Project 3: stopping at a line
   - Project 4: pointing a camera down a road
   - Project 5: reactive obstacle avoidance

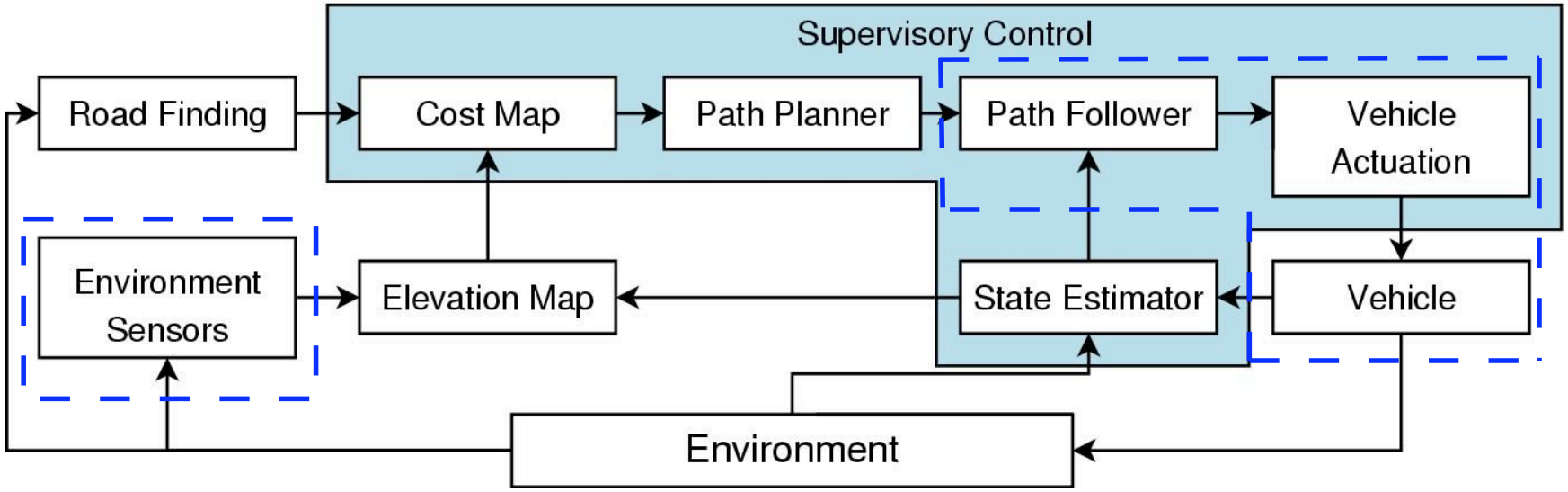
http://www.cds.caltech.edu/~murray/wiki/CDS_110a_Project_Ideas

3. **Continue to do some CDS 110a homework problems**
   - Complete first two problems on each homework set

4. **Project reports in lieu of midterm and final**
   - Midterm: GOTChA chart plus preliminary simulation (building off of homework)
   - Final: 5-10 page report describing implementation (group data, individual writeups) plus 10-15 minute oral presentation of results (team)

Total time required: about 80-100 hours/project (= 30-40 hours/person over 6 weeks)

How Alice Drives

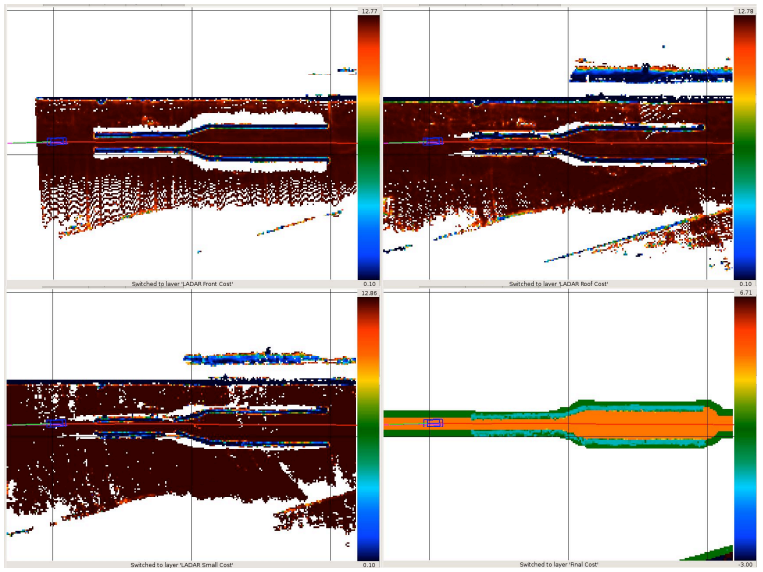


Current architecuture

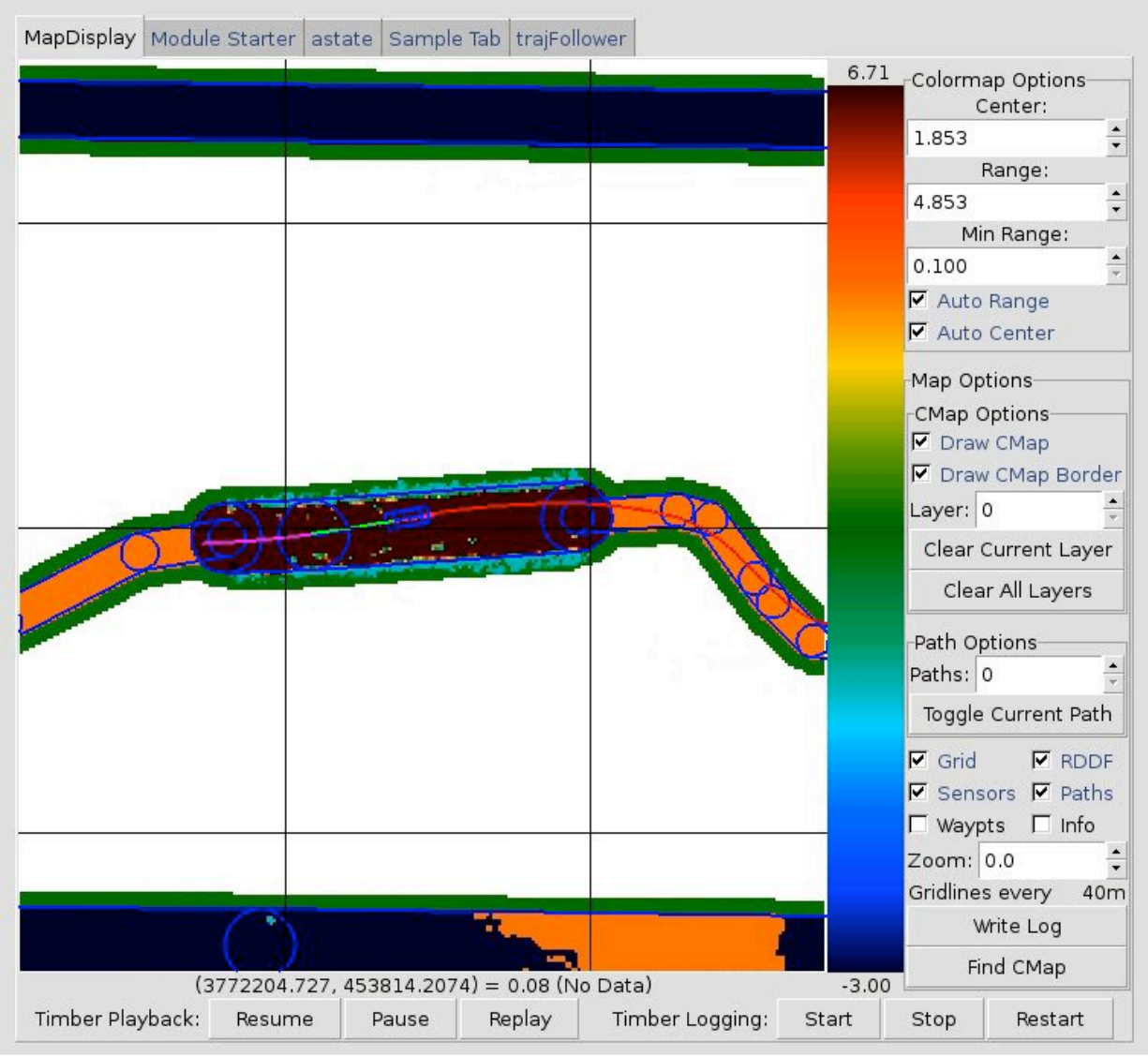
- Reference trajectory generated by planner
- "Inner loop" controls speed, lateral position

What's missing

- No capability for staying in lanes, stopping at stop lines, driving in reverse, pointing cameras, or defensive driving
- All can be implemented w/ tools of CDS 110a



How Alice Drives



Sensor-Based Navigation

- Sensors create individual digital elevation maps
  - Riegl LADAR: 50 m
  - Mid LADARS: 10, 20 m
  - Stereo: 5-20 m
  - Short LADAR: 2 m
- Sensor fusion creates a speed map indicating how quickly a given area can be traversed
  - No obstacle ⇒ RDDF limit
  - Roadfinding bumps speed
- Optimization-based planner computes fastest path for next 20-40 meters
- Supervisor Control used if planner fails or state drifts

Step 0: GOTChA chart for your project

Goals (1-3)

- Describe the goals of your project, in plain English
- Description should be understandable to your engineering friends (avoid jargon)
- Tie to the overall project goal

Objectives (3-6)

- Specific tasks that you need to complete in order to accomplish your goals
- Objectives should be "SMART"
  - Specific - concrete descriptions
  - Measurable - can tell whether you accomplished them or not
  - Attainable - possible to complete in time available
  - Relevant - lined up with system spec
  - Trackable - possible to monitor progress as you go
- Guide: try to include numbers and dates, when possible

Technical Challenges (4-8)

- List of problems that you expect to face in accomplishing your objectives
- Try to list anything that you are not sure about
- OK to include things that you don't yet know (eg, programming in C/C++, implementing a vision algorithm, etc)

Approach (4-8)

- Describe how you are going to tackle the technical challenges that will let you accomplish your objectives to satisfy your goals
- Make sure all of the technical challenges are addressed (otherwise how will you overcome them?)
- Can serve as a work plan for the term - what do you want to do first, next, etc

Step 1: Build a simulation for your project

Default simulator = simulink

Advanced simulator = ROAMS

Step 2: Implement your controller on Alice

Follow

Step 3: Design and implement a demonstration test

Step 4: Document your work

Project Timeline

Week 5: GOTChA chart + simulation

- Get together with your team and put together a GOTChA chart; send to Richard
- As a group, develop a simple simulation that captures the idea of what you will implement (use state space feedback material from L5-1 in CDS 110)

Week 6-7

- Learn how Alice works and implement basic control law (path follower)
- Work on your control design, through simulation and preliminary implementation

Week 8-9

- Finish implementation of controller on Alice, with realistic inputs
- Implement your demo, compare simulation to experiment, refine design

Week 10-11

- Document your results