CDS 110b: Lecture 1.3
Course Project Information Session

Richard M. Murray

6 January 2006

Goals:

- Provide enough information on the course project for people to decide whether or not they want to pursue it
- Give a high level introduction to Alice, so that you know what the project is intended to accomplish

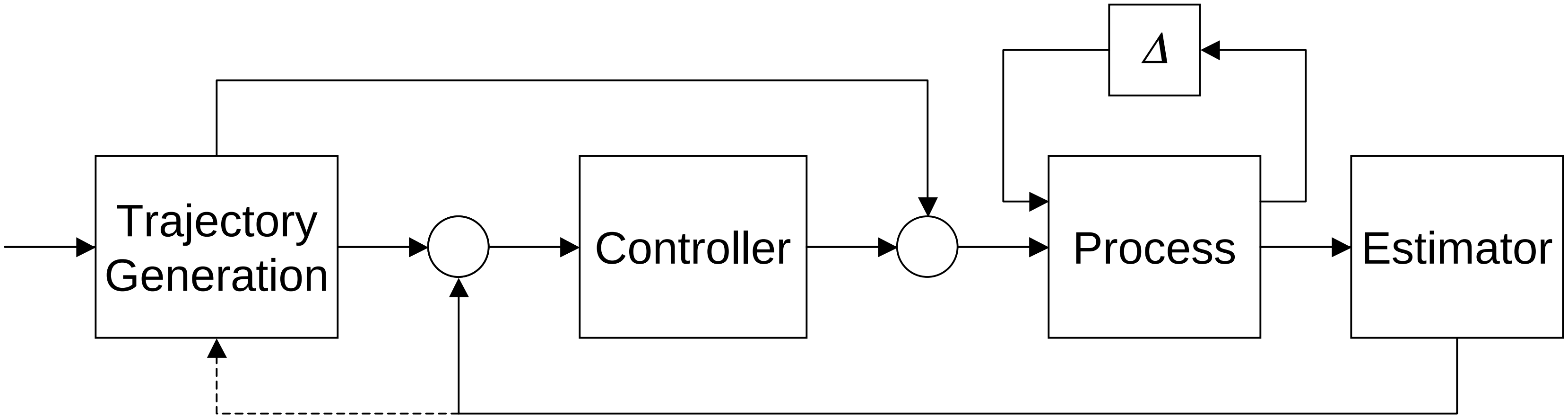
Reading:

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

Course Project: Alice

Control System *Implementation*

- Course work focuses on design techniques, analysis, simulation
- Project will focus on implementation of controllers on Alice →
- SURF opportunities available building on project experience (see SURF web page)



Course Project Administration

1. **Attend three lectures on control implementation - Fridays, 2-3 pm**
   - Week 1 (this lecture): high level project description
   - Week 2 (13 Jan): vehicle dynamics + control specification
   - Week 3 (20 Jan): hardware and software interfacing

2. **Do all course homework**
   - At least one problem on each set will be tuned for use in the project

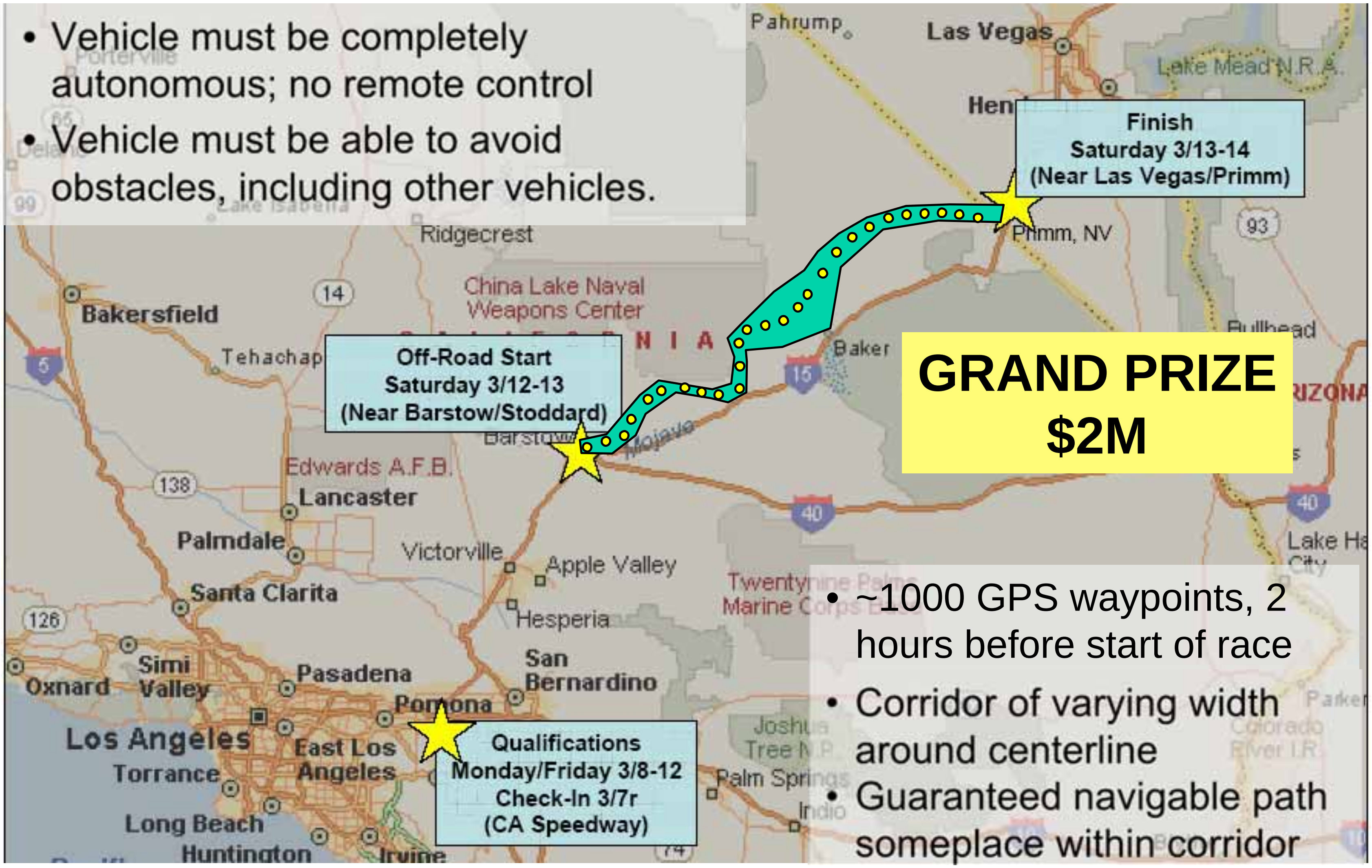
3. **Spend time implementing your controllers on Alice**
   - Recommend spending 3-4 hrs/wk, working in small groups
   - Jeremy will coordinate “test days” when we can take Alice out

4. **Project reports (written and oral) in lieu of midterm and final**
   - Midterm:
     - 3-5 page report describing implementation of an LQR controller on Alice (must include experimental results)
   - Final:
     - 5-10 page report describing full controller implementation (trajectory generation, estimation, robust control)
     - 15-20 minute presentation of controller design and results

Total time required (est): about 30-40 hours (over 10 weeks)

DARPA Grand Challenge: 150 Miles in 10 Hours or Less, No Humans Allowed

- Vehicle must be completely autonomous; no remote control
- Vehicle must be able to avoid obstacles, including other vehicles.



- ~1000 GPS waypoints, 2 hours before start of race
- Corridor of varying width around centerline
- Guaranteed navigable path someplace within corridor

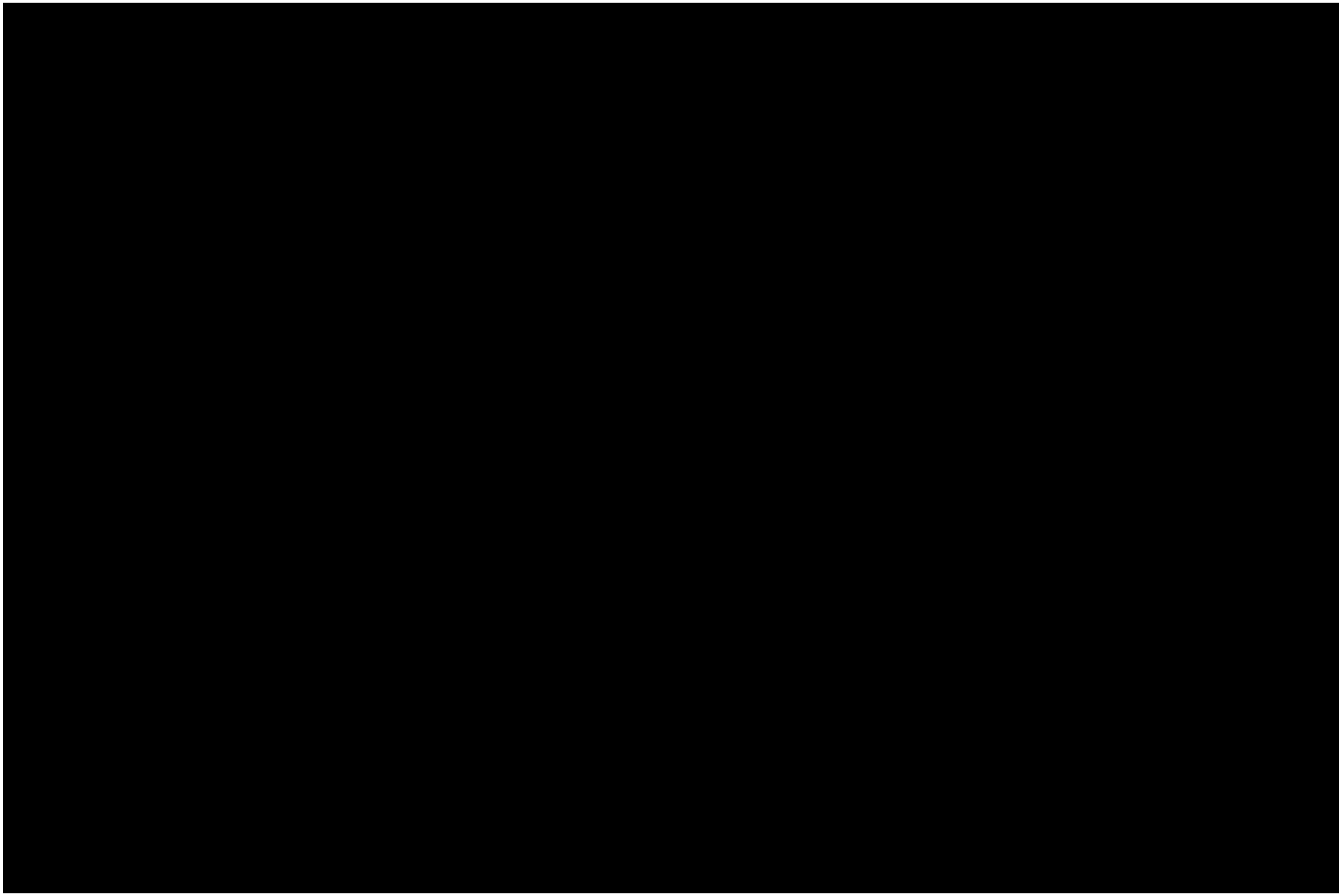
Route: Desert Racing Environment







Alice



Alice Overview

Team Caltech

- 50 students worked on Alice over 1 year
- Course credit through CS/EE/ME 75
- Summer team: 20 SURF students + 6 graduated seniors + 4 work study + 4 grads + 2 faculty + 6 volunteers (= ~40)

Alice

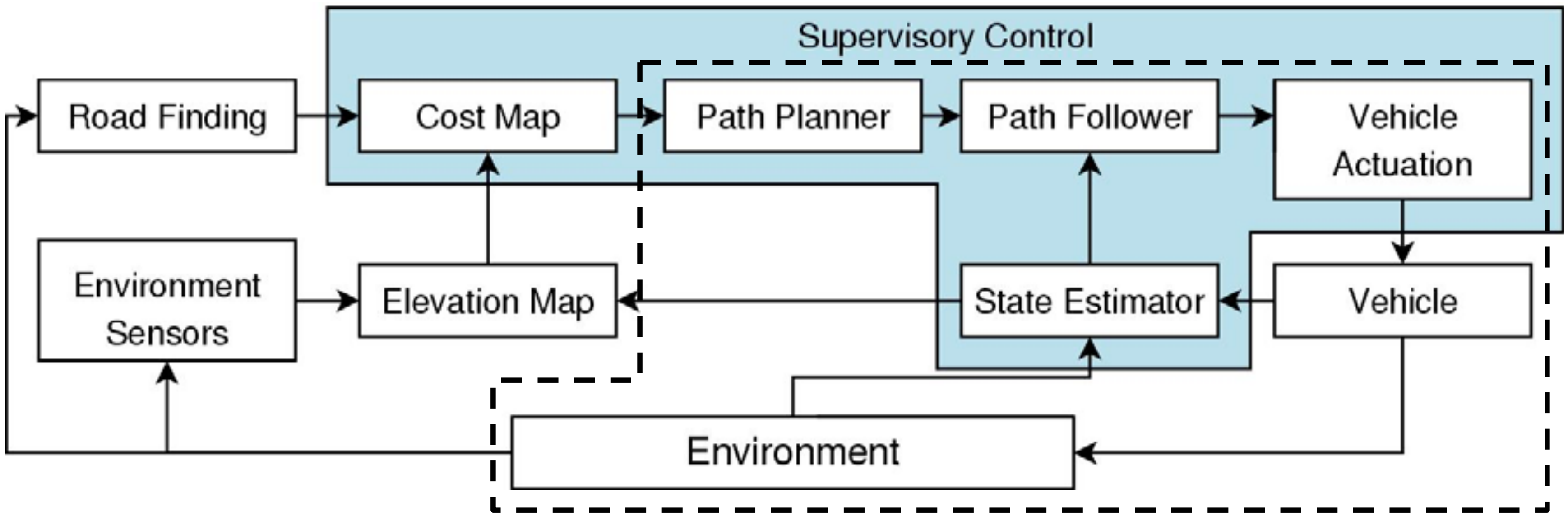
- 2005 Ford E-350 Van
- Sportsmobile 4x4 offroad package
- 5 cameras: 2 stereo pairs + roadfinding
- 5 LADAR units: long, medium*2, short, bumper
- 2 GPS units + 1 IMU (LN 200)
- 6 Dell 750 PowerEdge Servers (P4, 3GHz, gentoo linux)
- 1 IBM Quad Core AMD64 (fast!)
- 1 Gb/s switched ethernet

Software

- 15 individual programs with ~50 threads of execution
- FusionMapper: integrate all sensor data into a speed map for planning
- PlannerModule: optimization-based planning over a 10-20 second horizon

Cremean *et al*, JFR 05 (s)

Alice's Architecture



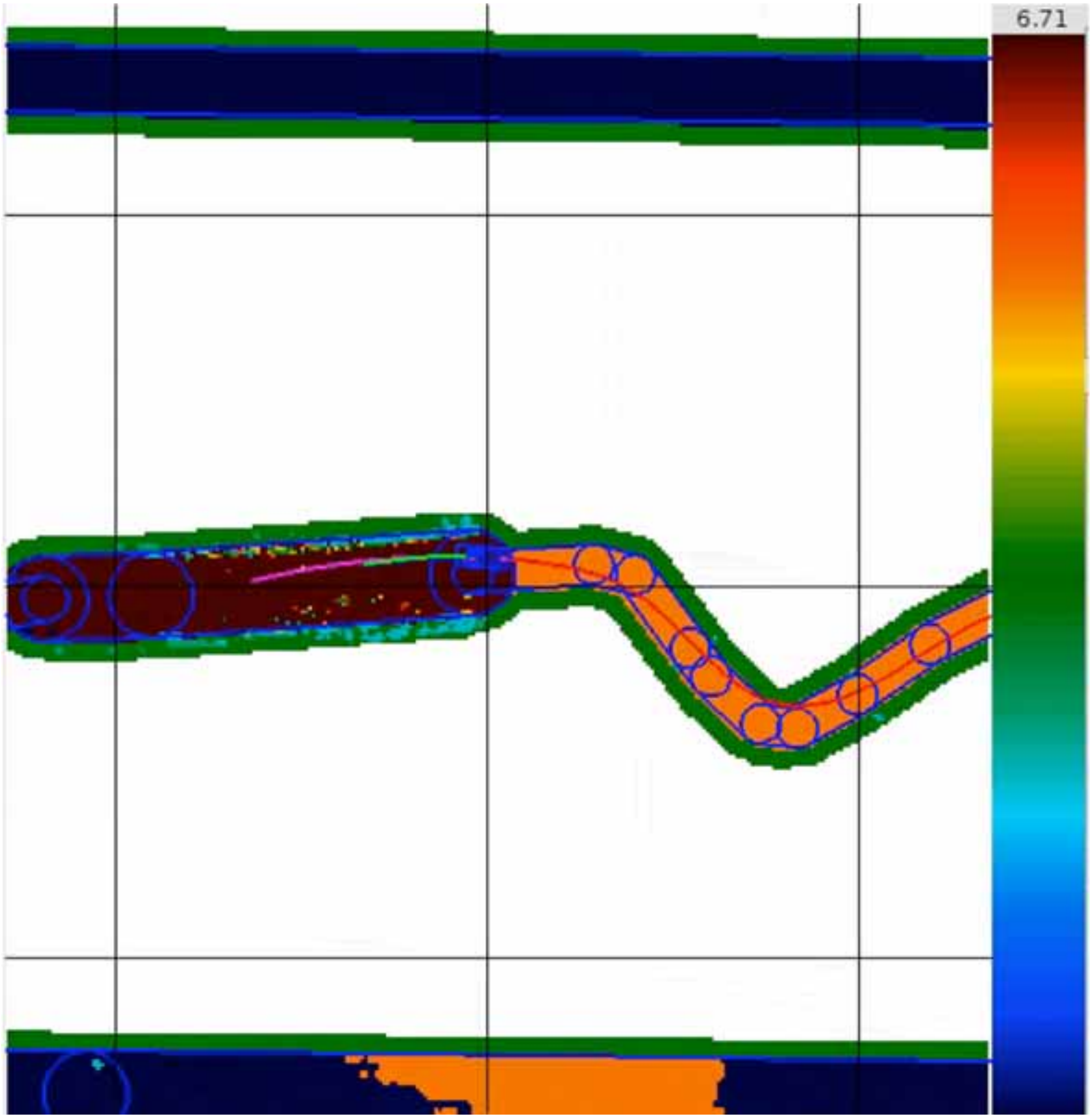
Optimization-Based Control (focus of CDS 110b)

- Real-time optimal trajectory generation
- Kalman filter based state estimation

Additional elements

- Sensor fusion: creating speed maps from estimates of terrain
- Supervisory control: fault and contingency management

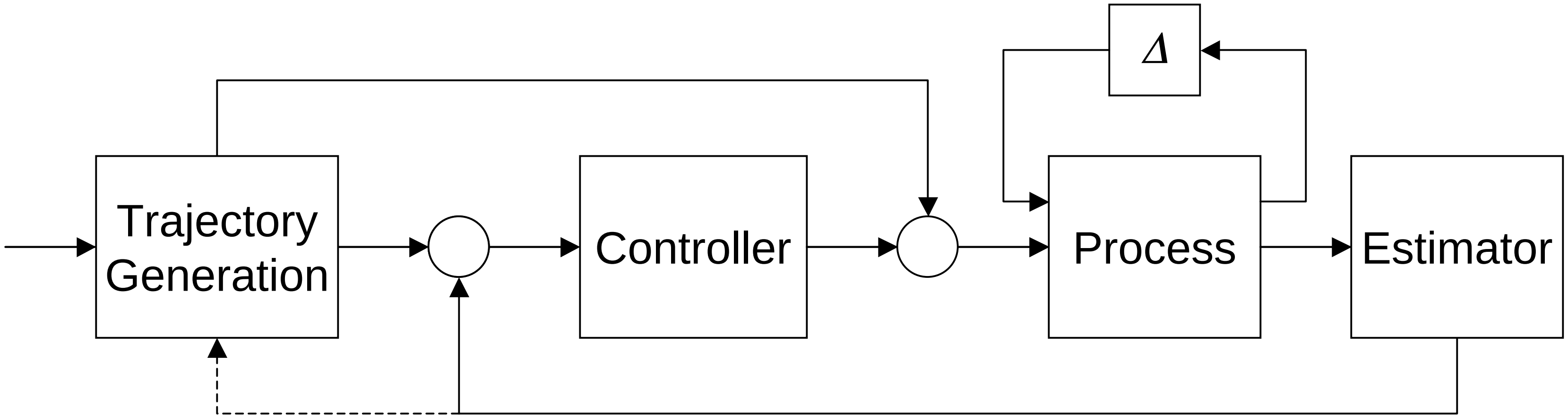
Alice's View of the World

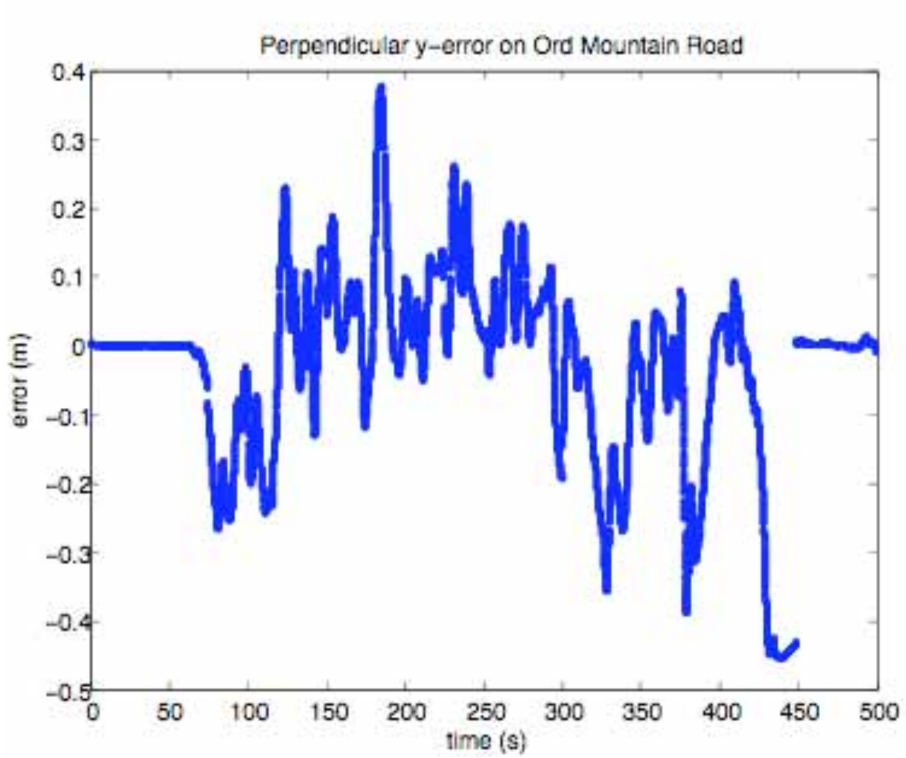


Sensor-Based Navigation

- Sensors create digital elevation map (one per sensor)
  - Riegl LADAR: 35 m
  - Mid LADARS: 20m, 10m
  - Stereo: 5-20m
  - Short LADAR: 3 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

Vehicle Control





Trajectory Generation

- Optimize speed along course
- Will simplify for CDS 110b

Estimator

- Estimation position and heading given GPS and IMU measurements

Controller

- LQR, H_∞ or PID design w/ robustness analysis

← Sample results

Control Design Process

Frequency Domain:

$$u = C(s)(r - y)$$

$$\dot z = Az + B(r - y)$$
$$u = Cz$$

$$w_{k+1} = \tilde{A} w_k + \tilde{B}(r_k - y_k)$$
$$u = \tilde{C} w_k$$

State Space:

$$u_{[t,t+\Delta T]} = \arg\min \int_t^{t+T} L(x(\tau), u(\tau))\, d\tau + V(x(t+T))$$

$$x_0 = x(t) \quad x_f = x_d(t+T)$$

$$\dot x = f(x,u) \quad g(x,u) \le 0$$

$$u = K(x - x_d) + u_d$$

Estimator:

- Construct estimate of current state based on measurement of input/output

$$\hat{x}_{k+1} = \tilde{F}\hat{x}_k + \tilde{G}u_k + L(\hat{H}\hat{x}_k - y_k)$$

Follow (SW)

Testing