Goals

• Give an overview of ACM/EE 116: course structure & administration
• Describe some of the types of applications that we will be able to solve using the tools taught in this course

Reading (for the week)

• Grimmett and Stirzaker, Chapters 1 and 2 (24 pp; see web for sections)
• (optional) Grimmett and Stirzaker, Appendices III and IV (history; 4 pp)
• (optional) Gubner, Chapter 1 (applications)
**Piazza**

Q&A forum for students to ask (and answer) questions

- Students can post questions for others to see
  - Student posts can be named or anonymous
  - Can post about homework, lectures or anything else related to the course
- Students can respond to questions by other students, or post followup questions
- TAs and instructor can endorse a student response or provide an instructor response

Important notes

- Your questions and answers are stored on a non-Caltech machine => we have limited control
- Information posted should only be viewable by other students + instructors
- Use of Piazza is optional, but we would like to try it out and get feedback
  - If you use Piazza to post a question or followup, you don't have to sign it at OH

**Honor System**

What is it?

No member of the Caltech community shall take unfair advantage of any other member of the Caltech community

Why is it important?

- Provides a framework for ethical conduct in an academic setting
- Supports a community of scholars, working together to learn and educate
- Allows greater academic freedom through mutual trust and respect

How does it apply to this class?

- Homework: full collaboration allowed, but write up your own results
  - Plan to re-use material, tuned to what we are trying to teach
  - Not allowed to use solutions sets from previous years, other sources
- Tests: take home, open book, limited time, non-proctored
- Violations: student centered – investigated by the BoC or GRB
Application: Autonomous Driving

Computing
- 6 Dell 750 PowerEdge Servers (P4, 3GHz)
- 1 IBM Quad Core AMD64 (fast!)
- 1 Gb/s switched ethernet

Sensing
- 5 cameras: 2 stereo pairs, roadfinding
- 5 LADARs: long, med*2, short, bumper
- 2 GPS units + 1 IMU (LN 200)
- 0.5-1 Gb/s raw data rates

Terrain Estimation

Sensor processing
- Construct local elevation based on measurements and state estimate
- Compute speed based on gradients

Sensor fusion
- Combine individual speed maps
- Process “missing data” cells

Road finding
- Identify regions with road features
- Increase allowable speed along roads

LadarFeeder, StereoFeeder
- HW: LADAR (serial), stereo (firewire)
- In: Vehicle state
- Out: Speed map (deltas)
- Multiple computers to maintain speed

FusionMapper
- In: Sensor speed maps (deltas)
- Output: fused speed map
- Run on quadcore AMD64
Optimal Estimation (Kalman Filtering)

**System description**

\[ x[k + 1] = Ax[k] + Bu[k] + Fv[k] \]
\[ y[k] = Cx[k] + w[k], \]

- Disturbances and noise are multi-variable Gaussians with covariance \( R_v, R_w \)

**Problem statement:** Find the estimate that minimizes the mean square error \( E\{(x[k] - \hat{x}[k])(x[k] - \hat{x}[k])^T\} \)

**Proposition**

- For Gaussian noise, optimal estimate is the expectation of the random process \( x \) given the constraint of the observed output:
  \[ \hat{x}[k] = E\{X[k] | Y[l], l \leq k\} \]

- Can think of this as a least squares problem: given all previous \( y[k] \), find the estimate \( \hat{x}[k] \) that satisfies the dynamics and minimizes the square error with the measured data.

---

**Example: Kalman Filtering for Terrain (Gillula)**

- **Elevation Map:**
  - Individual sensors
  - Fused map

- **Covariance Map:**
  - A: All sensors
  - B: Just LADARs
  - C: LADAR in place
  - D: Long-range LADAR
  - E: LADAR in place
  - F: Sparse measurements
Cell Noise (Elowitz et al, 2002)

Noise in cells
- Put RFP and GFP under identical promoters; should get yellow
- Results: get range of colors

Extrinsic Noise:
- Global to a single cell, but varies from one cell to the next (e.g. cell volume, plasmid copy number)

Intrinsic Noise:
- Inherent stochasticity in gene expression (e.g. what order reactions occur in)

\[ \dot{x}_i = E(t) \cdot f_i(x_i, I_i(t)) \]

System Identification Using Cell Noise

Traditional systems identification
- Engineering: forced response. Difficult to do in vivo (e.g. sinusoids are tricky)
- Biology: gene knockouts; steady state measurements using gene arrays

Idea: use noise as a forcing function
- Steady state distributions are not enough if extrinsic noise is present
- Need to use correlation data instead
System ID of a Synthetic Circuit (Dunlop, Elowitz & M)

E. coli

Plasmid
(7,500 base pairs)
~10 copies per cell

Chromosome
(4,600,000 base pairs)
1 copy per cell

Course Outline

<table>
<thead>
<tr>
<th>Week</th>
<th>Topic</th>
<th>Reading</th>
<th>HW</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Introduction; probability spaces</td>
<td>GS, Ch 1,2</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>Discrete random variables</td>
<td>GS, Ch 3</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>Continuous random variables</td>
<td>GS, Ch 4</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>Generating functions</td>
<td>GS, Ch 5</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>Convergence of sequences of random variables</td>
<td>GS, Chapter 7</td>
<td>5</td>
</tr>
<tr>
<td>6</td>
<td>Introduction to random processes</td>
<td>GS, Ch 8 + notes</td>
<td>6</td>
</tr>
<tr>
<td>7</td>
<td>Discrete time random processes</td>
<td>GS, Ch 9 + notes</td>
<td>7</td>
</tr>
<tr>
<td>8</td>
<td>Continuous time random processes</td>
<td>GS, Ch 9 + notes</td>
<td>8</td>
</tr>
<tr>
<td>9</td>
<td>Advanced topics: diffusion processes, Itô’s formula</td>
<td>GS, Ch 13</td>
<td>9</td>
</tr>
<tr>
<td>10</td>
<td>Review for final</td>
<td></td>
<td>F</td>
</tr>
</tbody>
</table>

Goal: Balance between theory and applications
- Build on a rigorous mathematical basis (sigma fields)
- Develop useful techniques and show how these can be applied to real problems
- Challenge: broad set of backgrounds and interests => will try to use various feedback mechanisms to insure that we are covering the distribution well