Course 8

An Introduction to the Kalman Filter

SIGGRAPH 2001
EXPLORE INTERACTION AND DIGITAL IMAGES

Speakers

Greg Welch

Gary Bishop

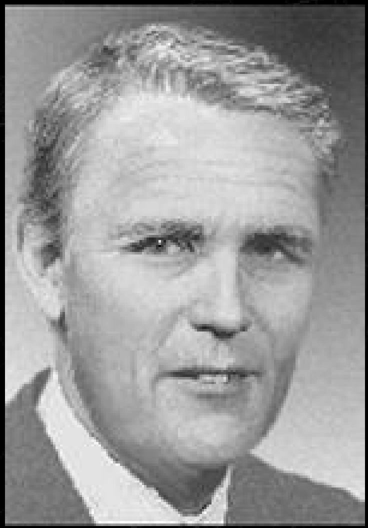
SIGGRAPH 2001 EXPLORE INTERACTION AND DIGITAL IMAGES

Kalman Filters in 2 hours?

- Hah!
- No magic.
- Pretty simple to apply.
- Tolerant of abuse.
- Notes are a standalone reference.
- These slides are online at
  http://www.cs.unc.edu/~tracker/ref/s2001/kalman/

Rudolf Emil Kalman

- Born 1930 in Hungary
- BS and MS from MIT
- PhD 1957 from Columbia
- Filter developed in 1960-61
- Now retired



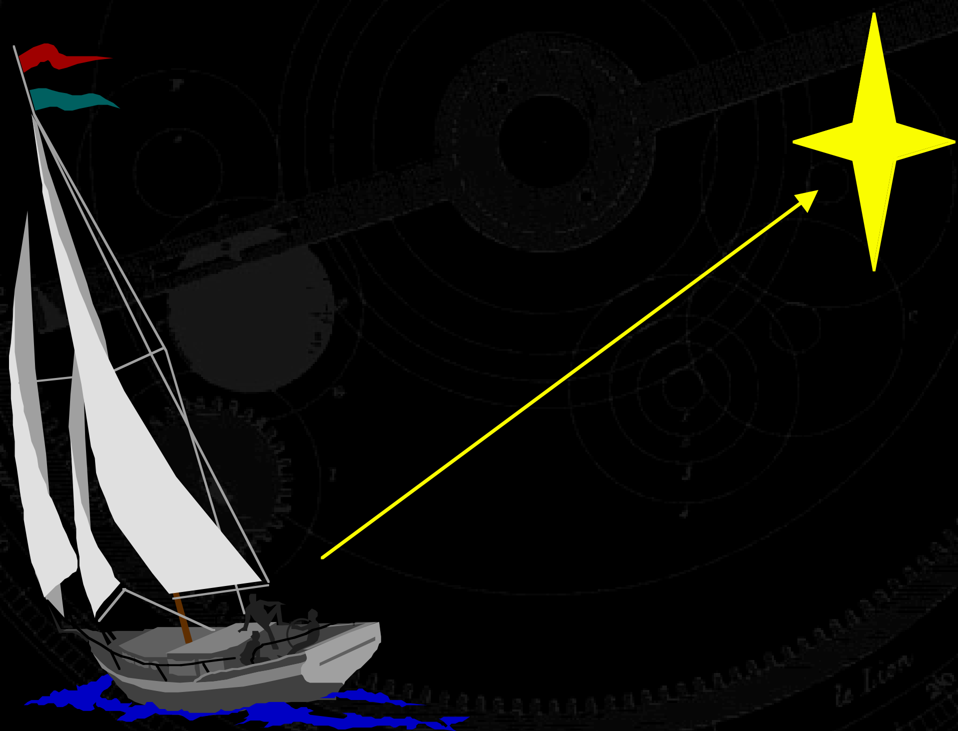
What is a Kalman Filter?

- Just some applied math.
- A linear system: f(a+b) = f(a) + f(b).
- Noisy data in → hopefully less noisy out.
- But delay is the price for filtering...
- Pure KF does not even adapt to the data.

What is it used for?

- Tracking missiles
- Tracking heads/hands/drumsticks
- Extracting lip motion from video
- Fitting Bezier patches to point data
- Lots of computer vision applications
- Economics
- Navigation

A *really* simple example



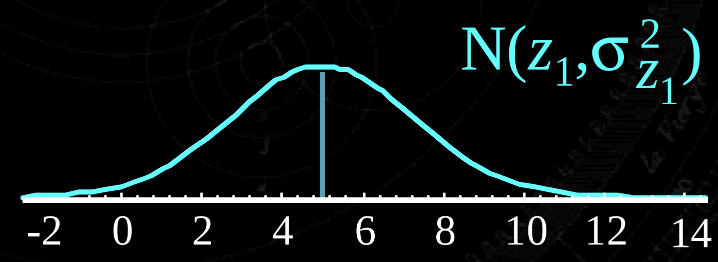
Gary makes a measurement

$$z_1, \sigma_{z_1}^2$$

$$\hat{x}_1 = z_1$$

$$\hat{\sigma}_1^2 = \sigma_{z_1}^2$$

Conditional Density Function

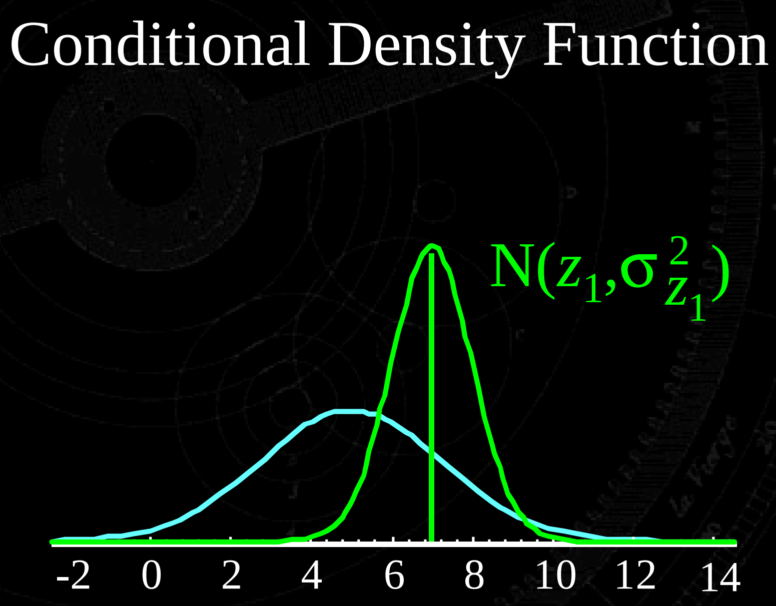


Greg makes a measurement

$$z_2, \sigma_{z_2}^2$$

$$\hat{x}_2 = \ldots ?$$

$$\hat{\sigma}_2^2 = \ldots ?$$



Combine estimates

$$\hat{x}_2 = \hat{x}_1 + K_2\left(z_2 - \hat{x}_1\right)$$

$$K_2 = \frac{\sigma_1^2}{\sigma_1^2 + \sigma_{z_2}^2}$$

Combine variances

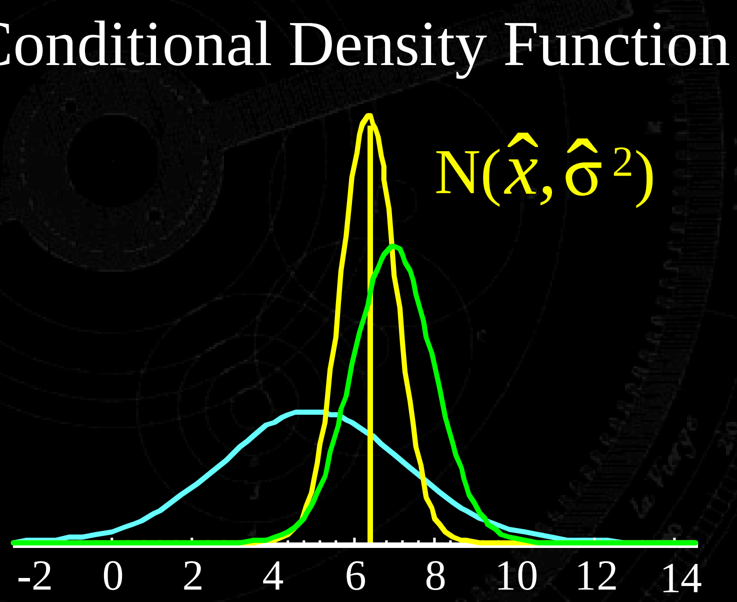
$$\frac{1}{\sigma_2^2} = \frac{1}{\sigma_1^2} + \frac{1}{\sigma_{z_2}^2}$$

SIGGRAPH 2001 EXPLORE INTERACTION AND DIGITAL IMAGES

Combined Estimates

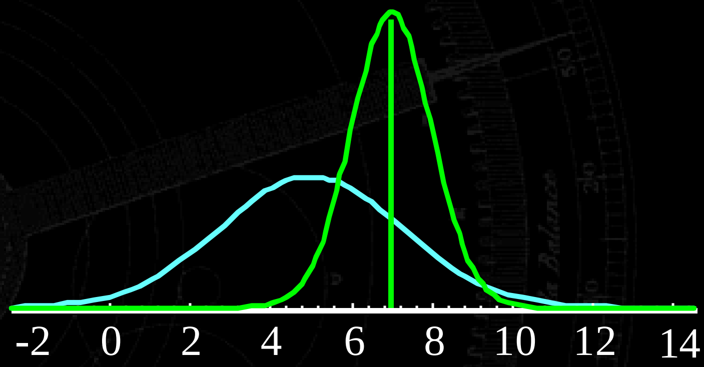
$$\hat{x} = \hat{x}_2$$

$$\hat{\sigma}^2 = \sigma_2^2$$



Online weighted average!

But suppose we're moving



- Not *all* the difference is error
- Some may be motion
- KF can include a motion model
- Estimate velocity and position

Process Model

- Describes how the *state* changes over time
- The *state* for the first example was scalar
- The *process model* was “nothing changes”

A better model might be

- State is a 2-vector [position, velocity]
- $\text{position}_{n+1} = \text{position}_n + \text{velocity}_n * \text{time}$
- $\text{velocity}_{n+1} = \text{velocity}_n$

Measurement Model

“What you see from where you are”

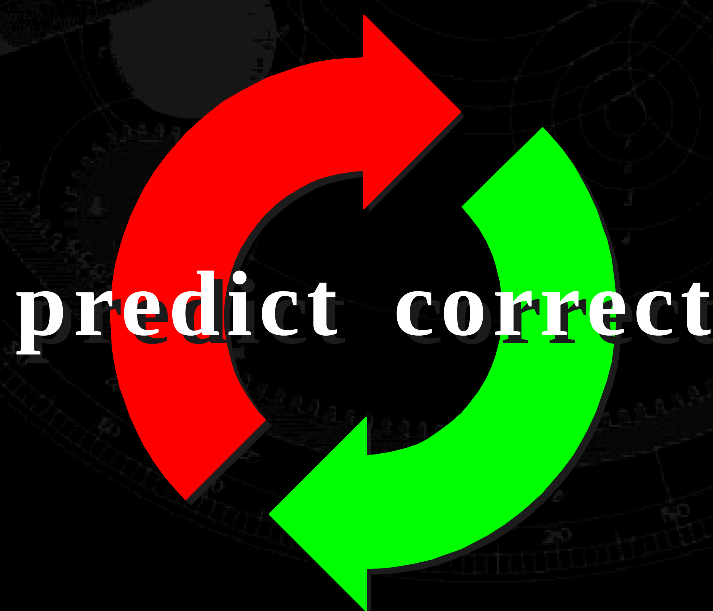
not

“Where you are from what you see”

Predict → Correct

KF operates by

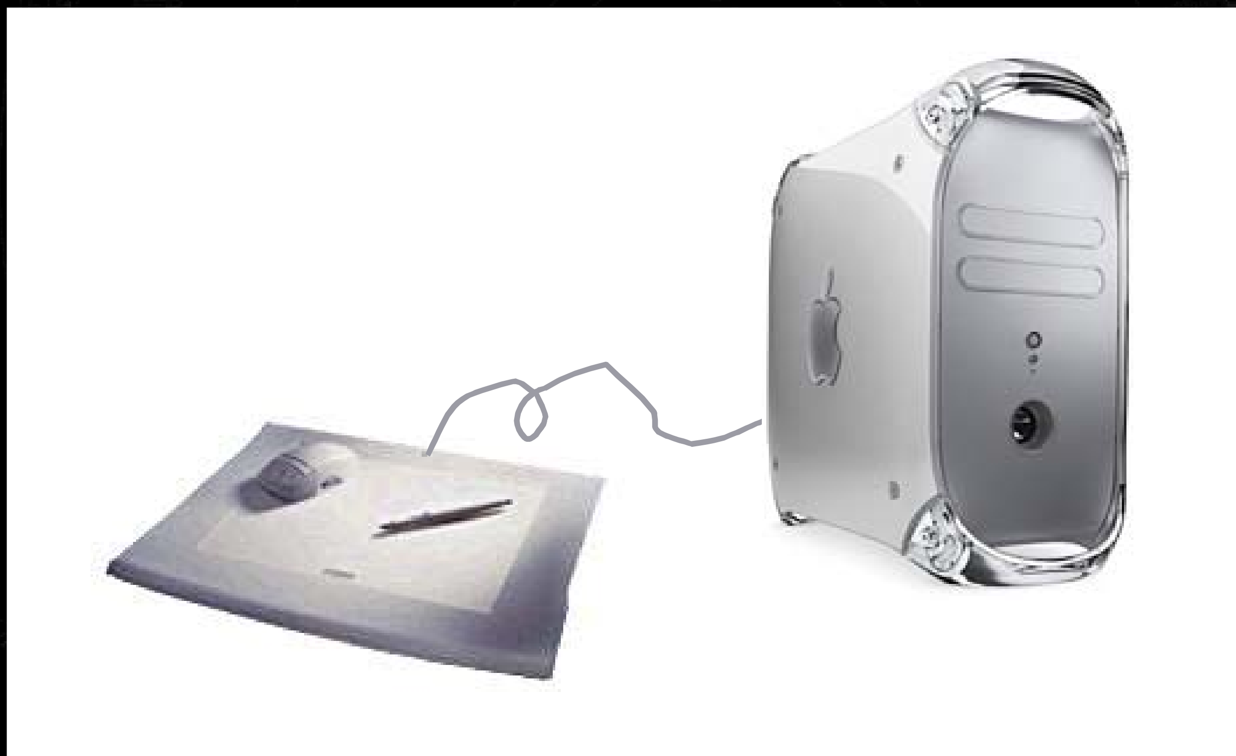
- Predicting the new state and its uncertainty
- Correcting with the new measurement



Example: 2D Position-Only

(Greg Welch)

Apparatus: 2D Tablet



SIGGRAPH 2001 EXPLORE INTERACTION AND DIGITAL IMAGES

Process Model

$$\begin{bmatrix} x_k \\ y_k \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_{k-1} \\ y_{k-1} \end{bmatrix} + \begin{bmatrix} \sim x_{k-1} \\ \sim y_{k-1} \end{bmatrix}$$

$\bar{x}_k$ *state* | A *state transition* | $\bar{x}_{k-1}$ *state* | $\bar{w}_{k-1}$ *noise*

$$\bar{x}_k = A\bar{x}_{k-1} + \bar{w}_{k-1}$$

SIGGRAPH 2001 EXPLORE INTERACTION AND DIGITAL IMAGES

Measurement Model

$$\begin{bmatrix} u_k \\ v_k \end{bmatrix} = \begin{bmatrix} H_x & 0 \\ 0 & H_y \end{bmatrix} \begin{bmatrix} x_k \\ y_k \end{bmatrix} + \begin{bmatrix} \sim u_k \\ \sim v_k \end{bmatrix}$$

measurement $\bar{z}_k$ | *measurement matrix* H | *state* $\bar{x}_k$ | *noise* $\bar{v}_k$

$$\bar{z}_k = H\bar{x}_k + \bar{v}_k$$

Preparation

$$A = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

State Transition

$$Q = E\{\overline{w} * \overline{w}^T\} = \begin{bmatrix} Q_{xx} & 0 \\ 0 & Q_{yy} \end{bmatrix}$$

Process
Noise
Covariance

$$R = E\{\overline{v} * \overline{v}^T\} = \begin{bmatrix} R_{xx} & 0 \\ 0 & R_{yy} \end{bmatrix}$$

Measurement
Noise
Covariance

SIGGRAPH 2001 EXPLORE INTERACTION AND DIGITAL IMAGES

Initialization

$$\bar{x}_0 = H\bar{z}_0$$

$$P_0 = \begin{bmatrix} \varepsilon & 0 \\ 0 & \varepsilon \end{bmatrix}$$

SIGGRAPH 2001 EXPLORE INTERACTION AND DIGITAL IMAGES

PREDICT

$$\bar{x}_k^- = A\bar{x}_{k-1}$$

$$P_k^- = AP_{k-1}A^T + Q$$

transition

uncertainty

CORRECT

$$\bar{x}_k = \bar{x}_k^- + K\left(\underbrace{\bar{z}_k}_{\text{actual}} - \underbrace{H\bar{x}_k^-}_{\text{predicted}}\right)$$

$$P_k = (I - KH)P_k^-$$

$$K = P_k^- H^T \left(\underbrace{HP_k^- H^T + R}_{\text{“denominator” (measurement space)}}\right)^{-1}$$

Summary

$$\bar{x}_k^- = A\bar{x}_{k-1}$$

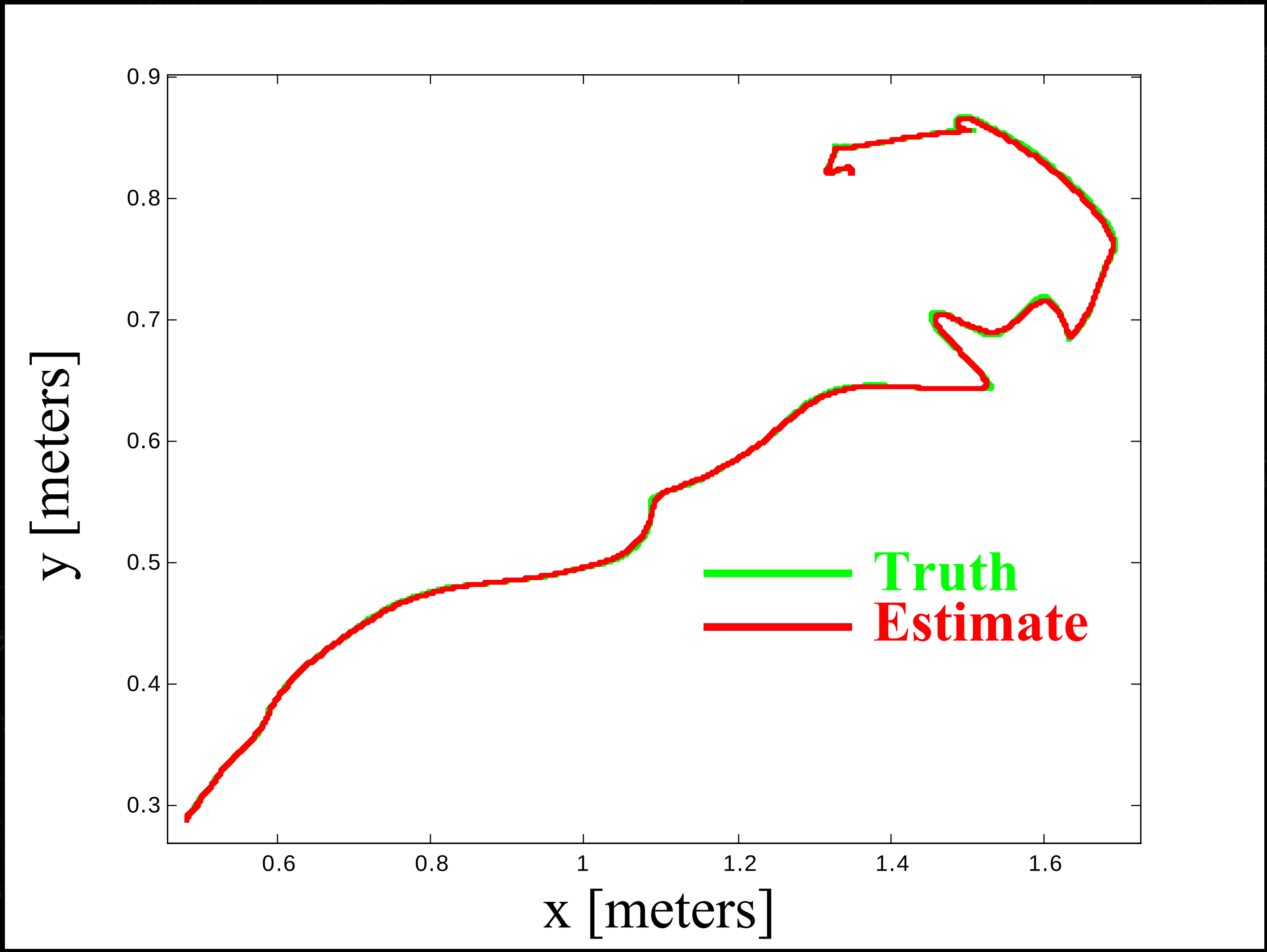
$$P_k^- = AP_{k-1}A^T + Q$$

$$K = P_k^- H^T \left(HP_k^- H^T + R\right)^{-1}$$

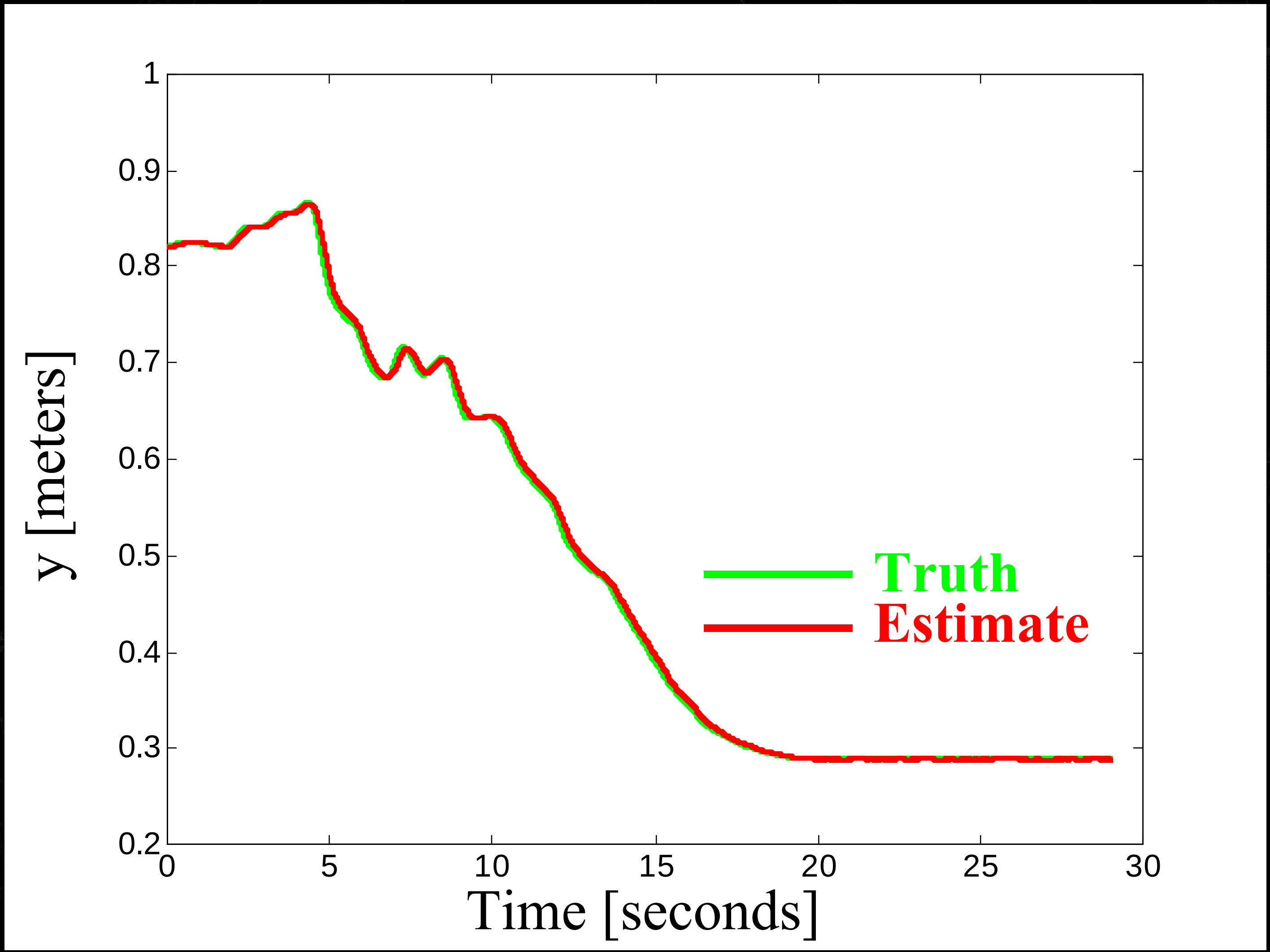
$$\bar{x}_k = \bar{x}_k^- + K\left(\bar{z}_k - H\bar{x}_k^-\right)$$

$$P_k = \left(I - KH\right)P_k^-$$

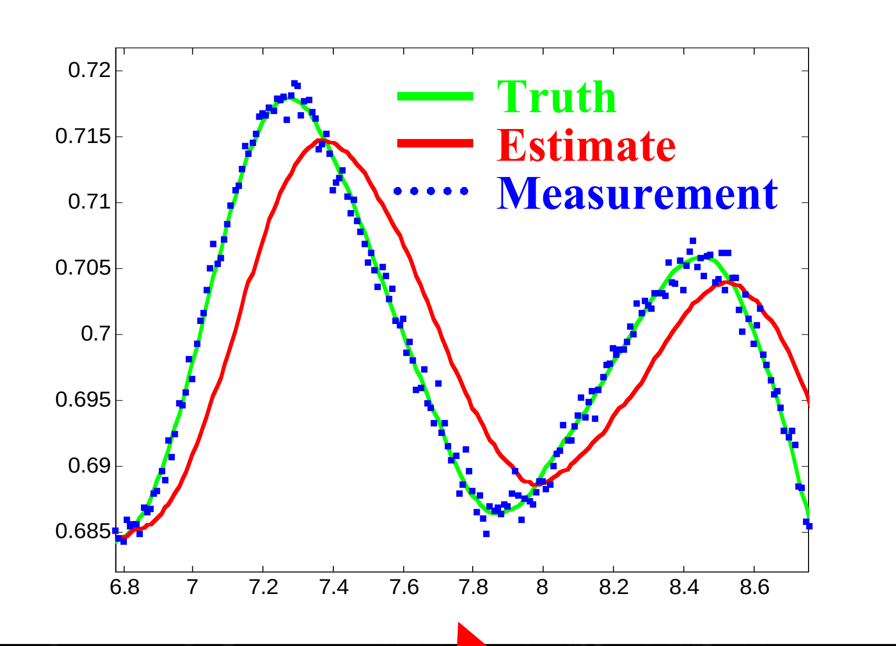
Results: XY Track

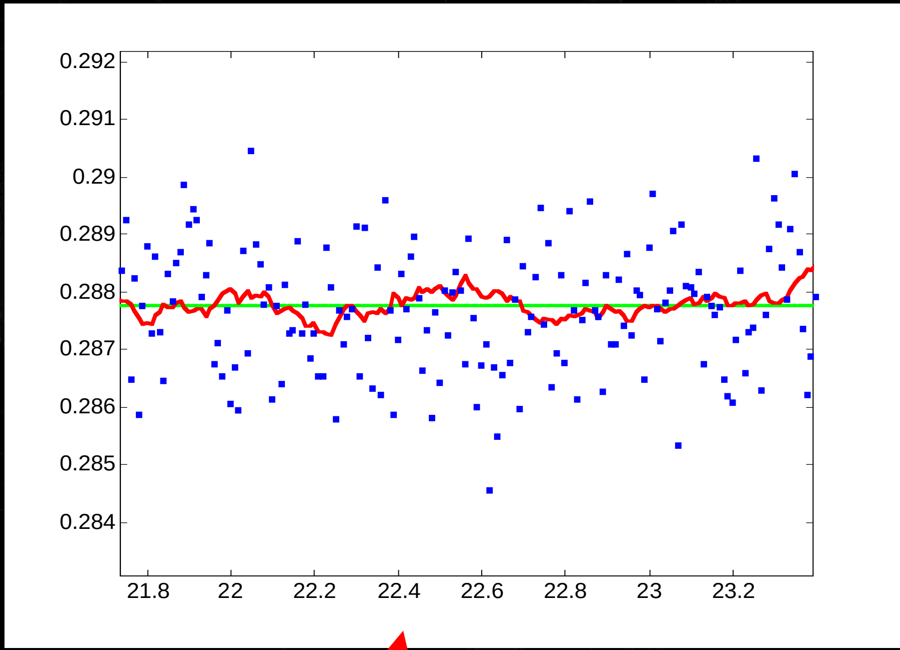


Y Track: Moving then Still

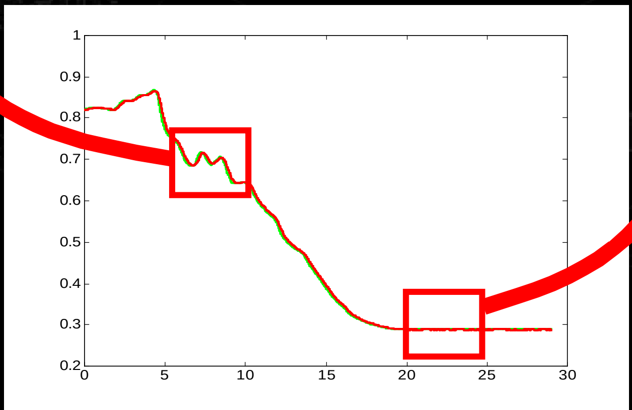


Motion-Dependent Performance





significant *latency* when moving…



…relatively *smooth* when not

Example: 2D Position-Velocity

(PV Model)

Process Model (PV)

state transition state

$$
\begin{bmatrix} 1 & 0 & dt & 0 \\ 0 & 1 & 0 & dt \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\begin{bmatrix} x \\ y \\ dx/dt \\ dy/dt \end{bmatrix}
$$

SIGGRAPH 2001 EXPLORE INTERACTION AND DIGITAL IMAGES

Measurement Model (Same)

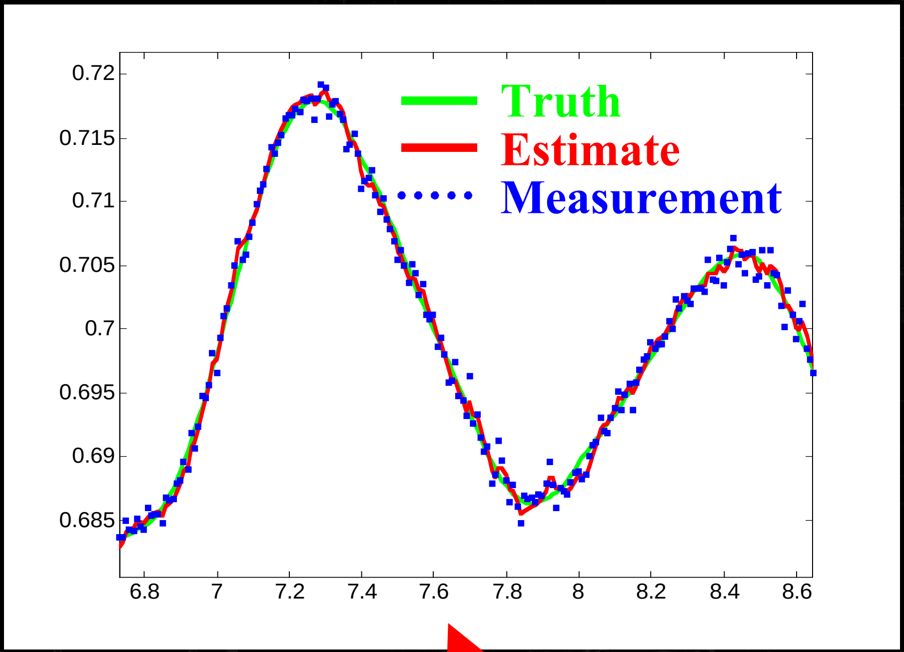
measurement matrix

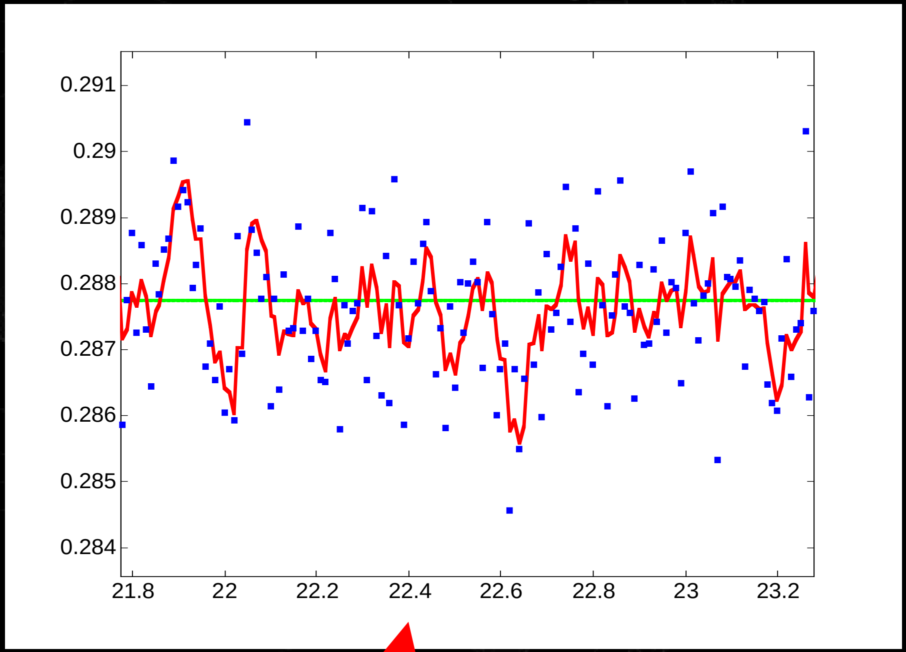
$$\begin{bmatrix} H_x & 0 & 0 & 0 \\ 0 & H_y & 0 & 0 \end{bmatrix}$$

state

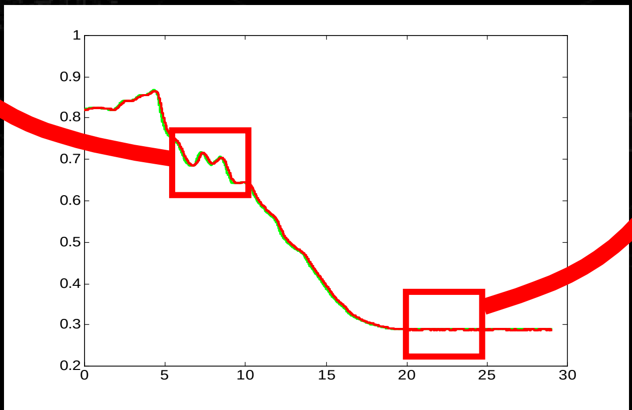
$$\begin{bmatrix} x \\ y \\ dx/dt \\ dy/dt \end{bmatrix}$$

Different Performance





improved latency when moving…



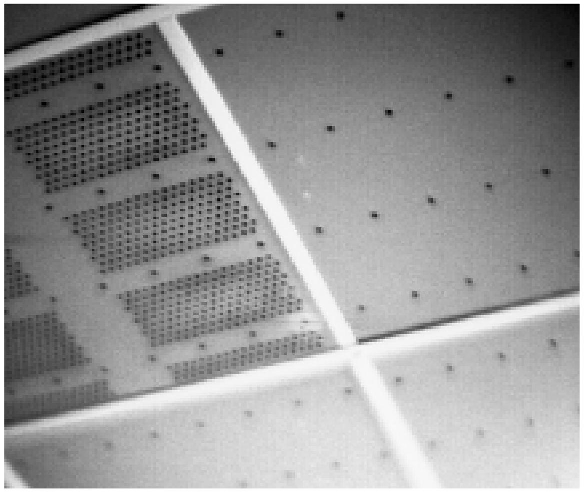
…relatively *noisy* when not

Example: 6D HiBall Tracker

(x, y, z, roll, pitch, yaw)

Apparatus

HiBall with six optical sensors



Ceiling panel with LEDs

State Vector (PV)

$$\bar{x} = \left[\bar{\tau} \quad \bar{\rho} \quad d\bar{\tau}/dt \quad d\bar{\rho}/dt \quad \bar{\lambda}\right]^T$$

$\bar{\tau} =$ translation (3D)

$\bar{\rho} =$ rotation (3D)

$d\bar{\tau}/dt =$ linear velocity (3D)

$d\bar{\rho}/dt =$ angular velocity (3D)

$\bar{\lambda} =$ LED position (3D)

Non-Linear Measurement Model

$$\begin{bmatrix} c_x \\ c_y \\ c_z \end{bmatrix} = V \cdot \mathrm{rotate}(\overline{\rho}) \cdot (\overline{\lambda} - \overline{\tau})$$

view matrix

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} c_x / c_z \\ c_y / c_z \end{bmatrix}$$

SIGGRAPH 2001 EXPLORE INTERACTION AND DIGITAL IMAGES

SCAAT vs. MCAAT

- *Single or Multiple Constraint(s) at a Time*
- **Dimension of the measurement**
  - Nothing about KF mathematics restricts it
  - Can process in "batch" or sequential mode
- **SCAAT**
  - Estimate 15 parameters with 2D measurements
  - Temporal improvements
  - Autocalibration of LED positions

HiBall Initialization

- Initialize pose using a brute-force (relatively slow) MCAAT approach
- Initial velocities = 0
- Initial process covariance P_0 = ~cm/degrees
- Transition to SCAAT Kalman filter

Nonlinear Systems

(Gary Bishop)

SIGGRAPH 2001 EXPLORE INTERACTION AND DIGITAL IMAGES

Kalman Filter assumes linearity

- Only matrix operations allowed
- Measurement is a linear function of state
- Next state is linear function of previous state
- Can't estimate gain
- Can't handle rotations (angles in state)
- Can't handle projection

Extended Kalman Filter

Nonlinear Process (Model)

- Process dynamics: A becomes $a(x)$
- Measurement: H becomes $h(x)$

Filter Reformulation

- Use functions instead of matrices
- Use Jacobians to project forward, and to relate measurement to state

Jacobian?

- Partial derivative of measurement with respect to state
- If measurement is a vector of length M
- And state has length N
- Jacobian of measurement function will be MxN matrix of numbers (not equations)
- Often evaluating h(x) and Jacobian(h(x)) at the same time cost only a little extra

Tips

- Don't compute giant symbolic Jacobian with a symbolic algebra package
- Do use an automatic method during development
- Check out tools from optimization packages
- Differentiating your function line-by-line is usually pretty easy

SIGGRAPH 2001 EXPLORE INTERACTION AND DIGITAL IMAGES

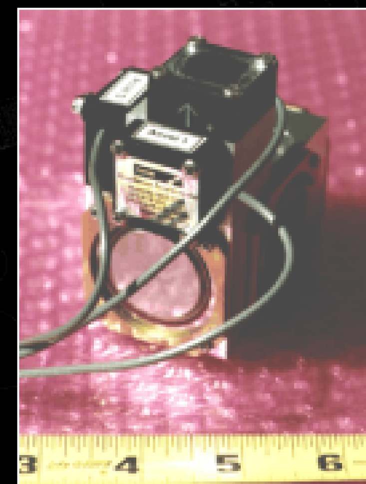
New Approaches

Several extensions are available that work better than the EKF in some circumstances

System Identification

Model Form and Parameters
(Greg Welch)

Measurement Noise (R)



Sampled Process Noise (Q)

For continuous model

$$\frac{d\overline{x}}{dt} = F\overline{x} + Q_c$$

The sampled (discrete) Q is

$$Q_d = \int_0^{dt} e^{F\tau} Q_c e^{F^T \tau} d\tau$$

Example: 2D PV Model

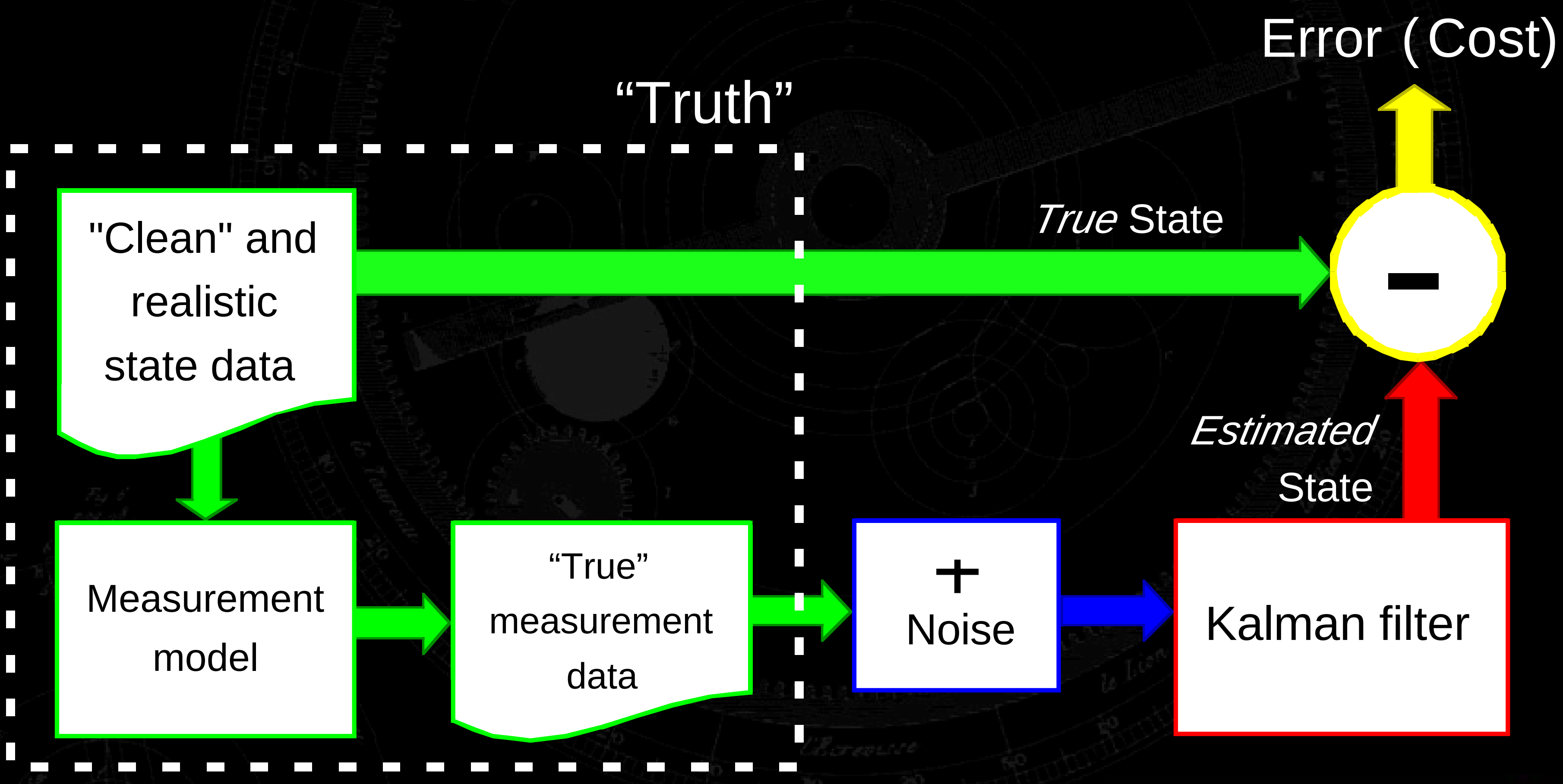
For continuous model

$$\frac{d\bar{x}}{dt} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \bar{x} + \begin{bmatrix} 0 & 0 \\ 0 & q \end{bmatrix}$$

The sampled (discrete) Q is

$$Q_d = \begin{bmatrix} dt^3 q/3 & dt^2 q/2 \\ dt^2 q/2 & dt\, q \end{bmatrix}$$

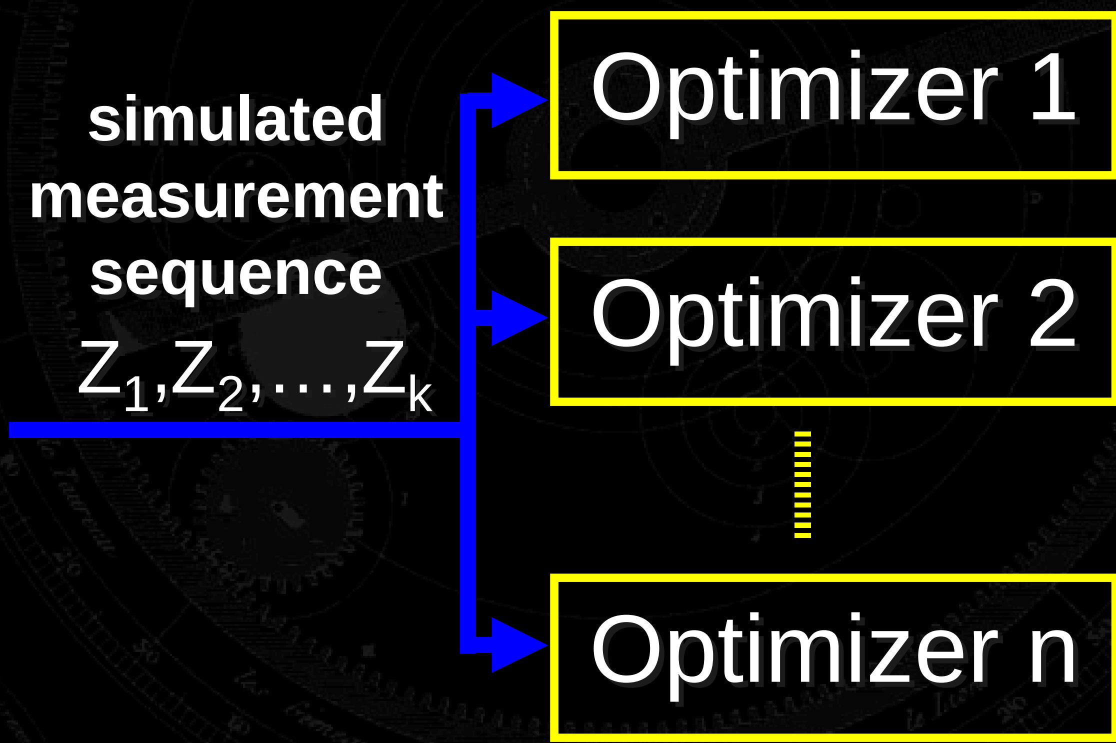
Parameter Optimization



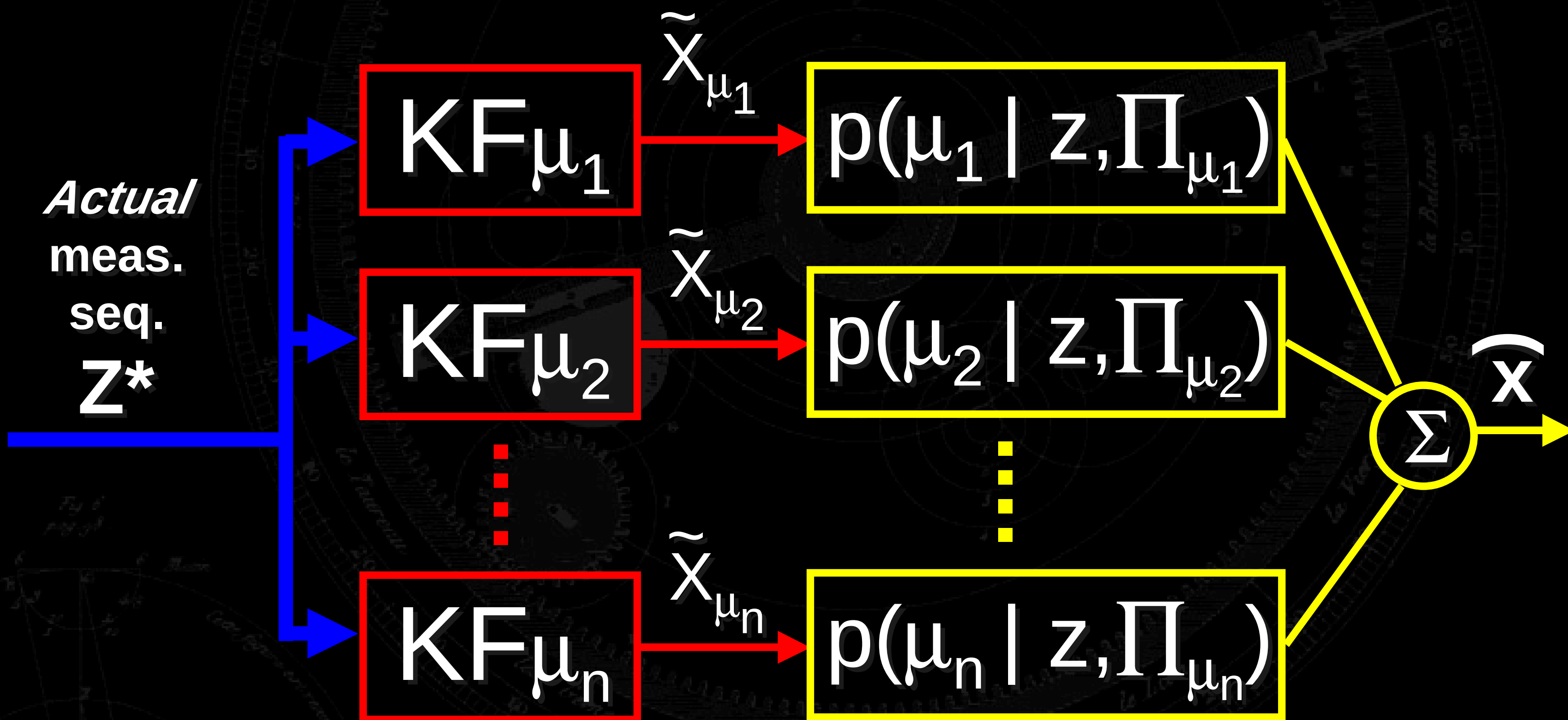
Multiple-Model Configurations

Off or On-Line Model Selection

Off-Line Model Selection



On-Line Multiple-Model Estimation



$$\Pi_{\mu_n} = \{ \bar{x}_k, P_k, H, R \}$$

Probability of Model μ

For model μ with $\Pi_\mu = \{x, P, H, R\}$

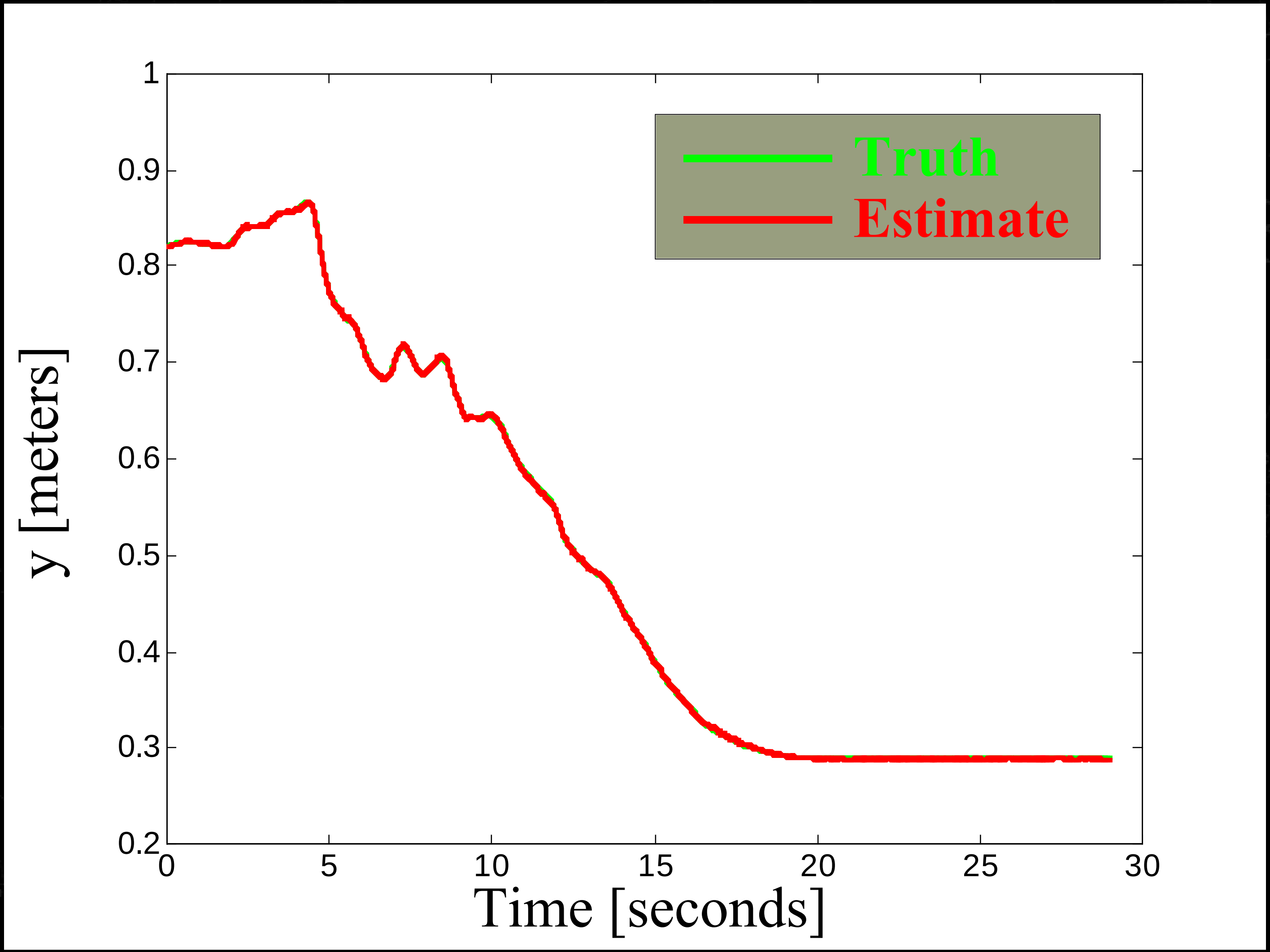
$$p(\mu | z, \Pi_\mu) = \frac{1}{(2\pi|C|)^{\frac{n}{2}}} e^{-\frac{1}{2}(z-Hx)^T C^{-1} (z-Hx)}$$

where $C = HPH^T + R$

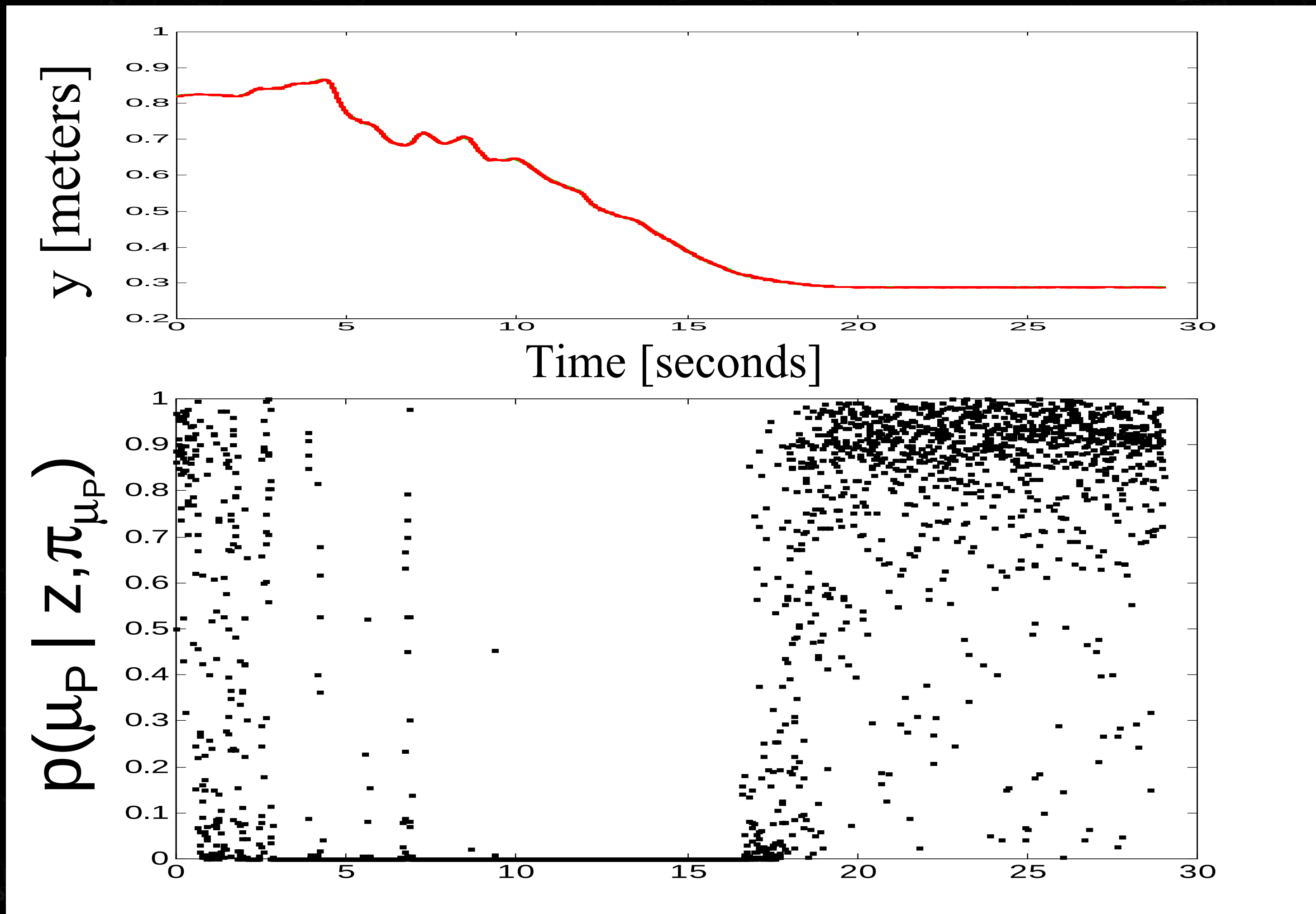
Final Combined Estimate

$$\hat{x} = \sum_{\mu} \hat{x}_{\mu} \frac{p\left(\mu \middle| z, \Pi_{\mu}\right)}{\sum_{\nu} p\left(\nu \middle| z, \Pi_{\nu}\right)}$$

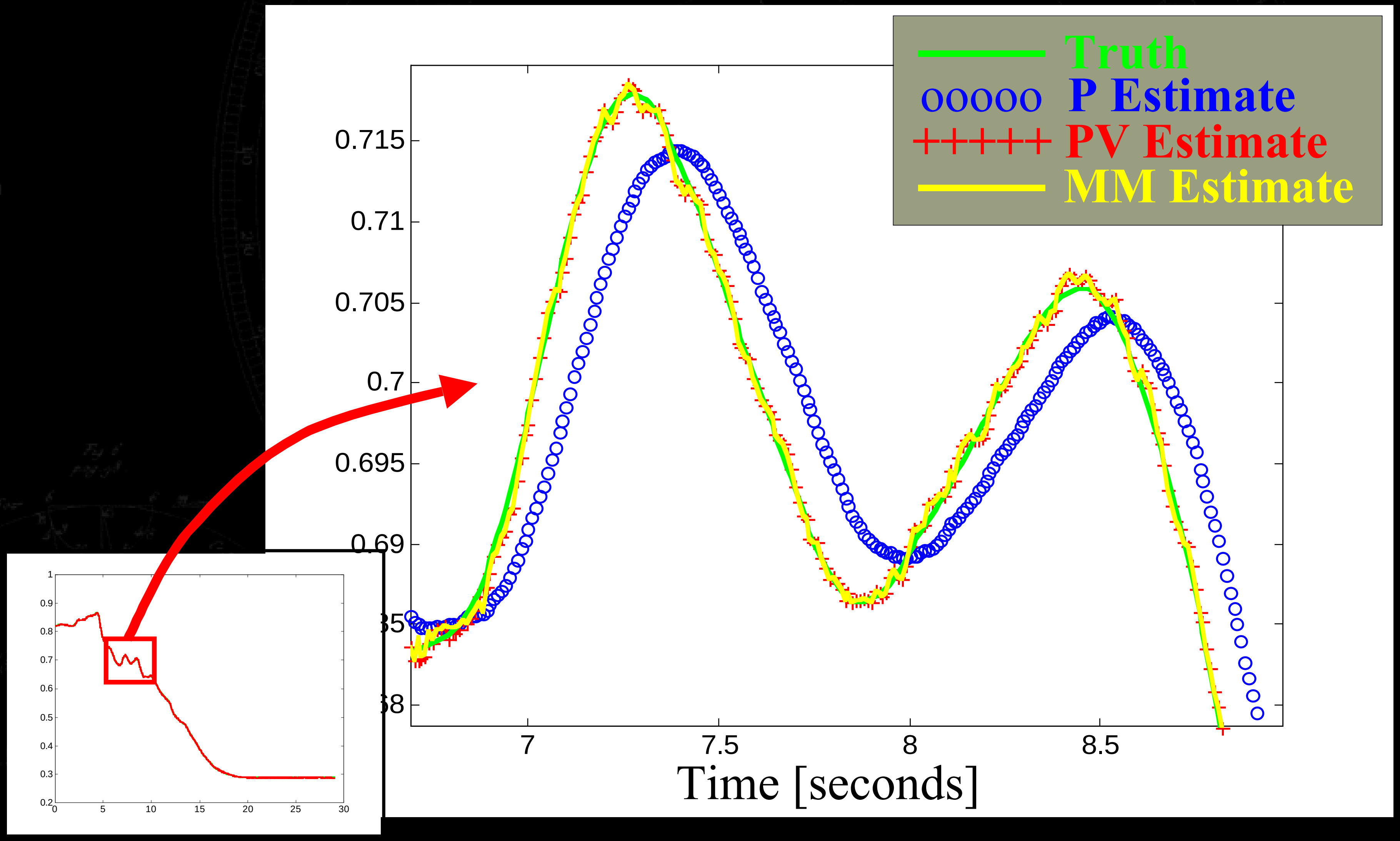
Example: P/PV Multiple-Model



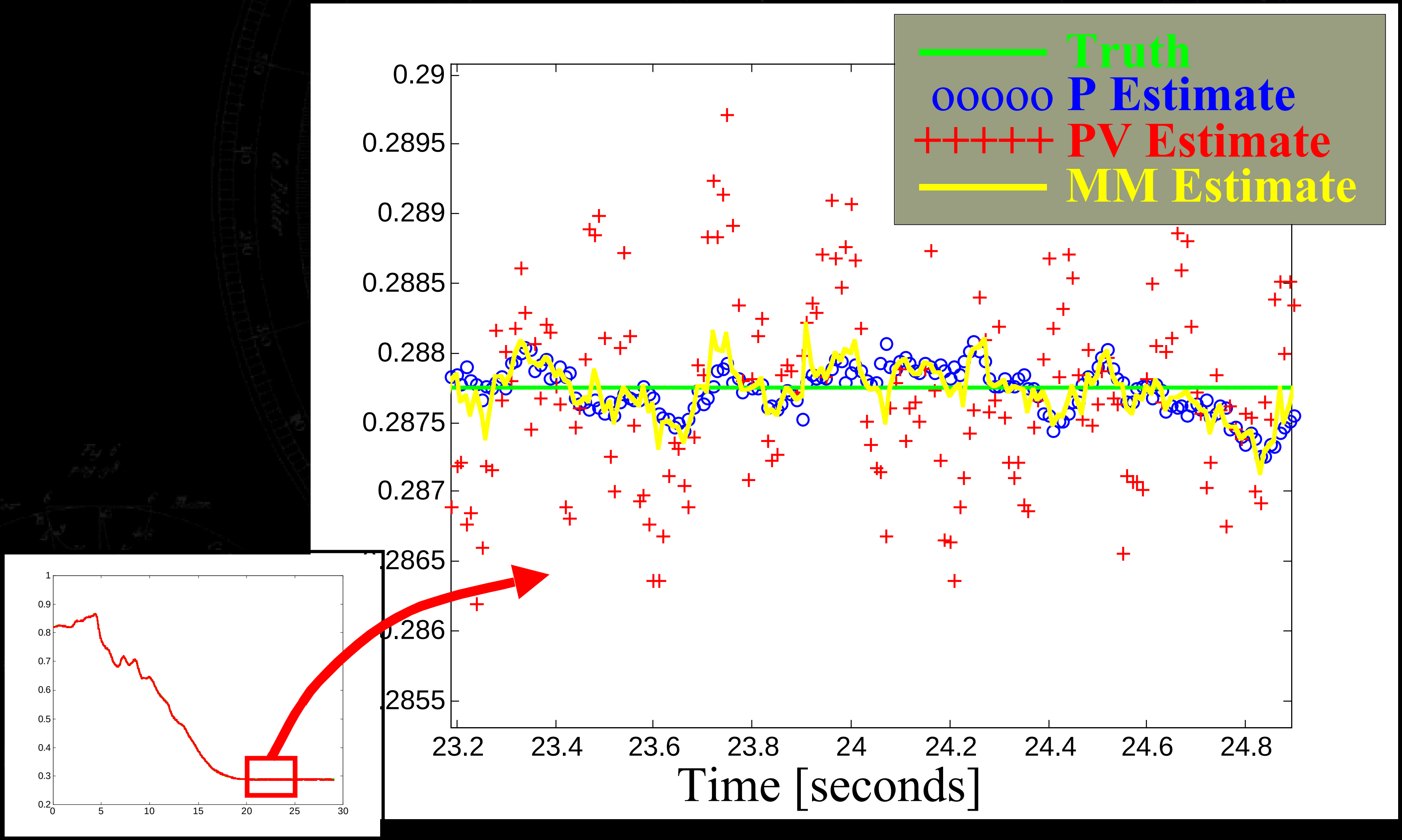
MME Weighting



Low-Latency During Motion



Smooth When Still



Conclusions

Suggestions and Resources
(Greg Welch)

Many Applications (Examples)

- **Engineering**
  - Robotics, spacecraft, aircraft, automobiles
- **Computer**
  - Tracking, real-time graphics, computer vision
- **Economics**
  - Forecasting economic indicators
- **Other**
  - Telephone and electricity loads

SIGGRAPH 2001 EXPLORE INTERACTION AND DIGITAL IMAGES

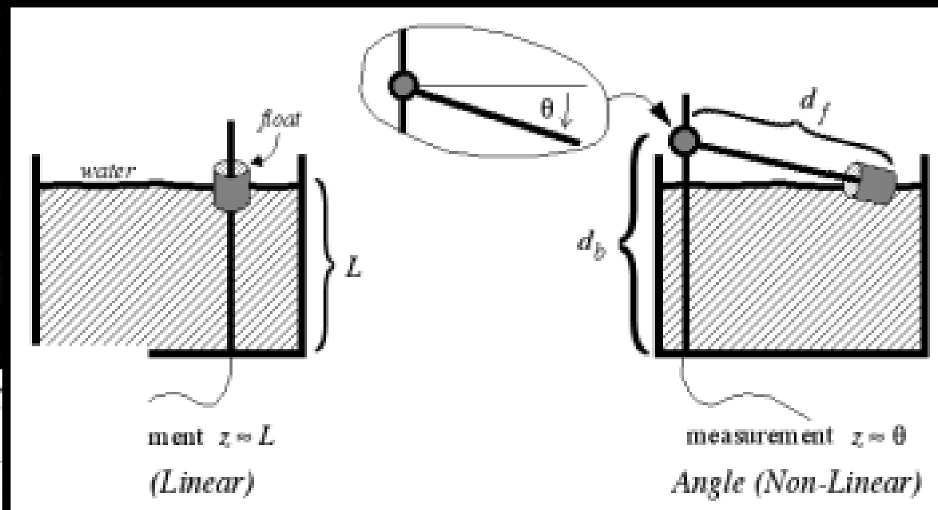
Kalman Filter Web Site

http://www.cs.unc.edu/~welch/kalman/

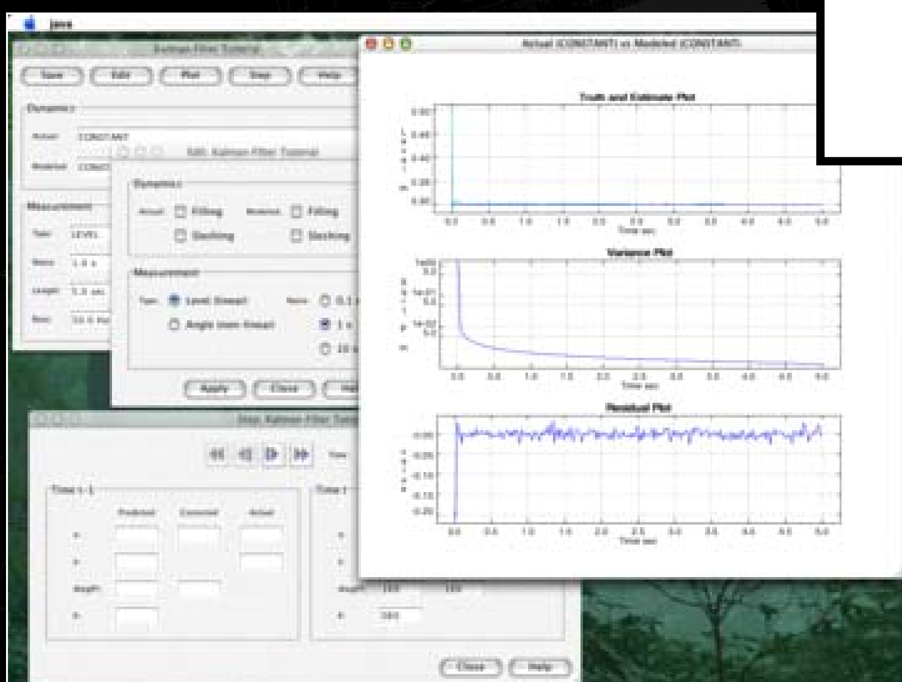
- **Electronic and printed references**
  - Book lists and recommendations
  - Research papers
  - Links to other sites
  - Some software
- **News**

Java-Based KF Learning Tool

- On-line 1D simulation
- Linear and non-linear
- Variable dynamics



http://www.cs.unc.edu/~welch/kalman/



SIGGRAPH 2001 EXPLORE INTERACTION AND DIGITAL IMAGES

KF Course Web Page

http://www.cs.unc.edu/~tracker/ref/s2001/kalman/index.html

(http://www.cs.unc.edu/~tracker/)

- Electronic version of course pack (updated)
- Java-Based KF Learning Tool
- KF web page

- *See also notes for Course 11 (Tracking)*

SIGGRAPH 2001 EXPLORE INTERACTION AND DIGITAL IMAGES

Closing Remarks

- **Try it!**
  - Not too hard to understand or program
- **Start simple**
  - Experiment in 1D
  - Make your own filter in Matlab, etc.
- **Note: the Kalman filter "wants to work"**
  - Debugging can be difficult
  - Errors can go un-noticed

End