Numerical Computation

EE 782
M. Sami Fadali
Professor of Electrical Engineering
University of Nevada, Reno

Outline

• Floating point numbers and arithmetic.
• Roundoff errors and their effects.
• Modeling errors
• Optimal error covariance analysis.
• Examples.

Floating Point Numbers

• Stored using a word of finite length.
• Characterized by four integers:
  ▪ base ($\beta$), precision ($t$), exponent range $e \in [L, U]$
  ▪ Typical values: $(2, 56, -64, 64)$
• Floating Point Numbers $F$: zero and
  \[ f = \pm. d_1 d_2 \ldots d_t \beta^e \]
  \[ 0 < d_i < \beta, d_1 \neq 0, \quad L \leq e \leq U \]
  \[ m \leq |f| \leq M \]
  \[ m = 0.1\beta^L, \quad M = 0.11 \ldots 1\beta^U = \beta^U (1 - \beta^{-t}) \]

Floating Point Arithmetic

Define the set of numbers $G$ and the floating point operator $fl$
\[ G = \{ x \in \mathcal{R} : m \leq |x| \leq M \} \cup \{0\} \]
\[ fl : G \rightarrow F \]
\[ fl(x) = \begin{cases} 
  \text{rounded arithmetic: nearest } c \in F \text{ to } x \\
  \text{If tie, round away from zero.} \\
  \text{chopped arithmetic: nearest } c \in F \text{ to } x, |c| \leq |x| 
\end{cases} \]
Floating Point Operations

- $Op = $ Any of the operations $\{+,-,\times,\div\}$
- $|a \, Op \, b| \not\in G$ an arithmetic fault occurs
  - Overflow $|a \, Op \, b| > M$
  - Underflow $|a \, Op \, b| < m$
- $|a \, Op \, b| \in G$ the computer uses $fl(a \, Op \, b)$
- Errors due to representation and computation.

Roundoff Error

$$f = \pm d_1 \, d_2 \ldots \, d_t \beta^e, \quad \text{step } \beta^{1-t}$$

$$fl(x) = x(1 + \epsilon), |\epsilon| < u$$

Unit roundoff $u = \begin{cases} 
\frac{1}{2} \beta^{1-t}, \text{rounded arithmetic} \\
\beta^{1-t}, \text{chopped arithmetic} 
\end{cases}$

$$fl(a \, Op \, b) = (a \, Op \, b)(1 + \epsilon), |\epsilon| < u$$

Relative error $\frac{fl(a \, Op \, b) - (a \, Op \, b)}{|a \, Op \, b|} = \epsilon, |\epsilon| < u$

Numerical Computation

- Algorithms that are equivalent mathematically often behave very differently numerically.
- Example (Van Loan, 1997): Two equal forms behave very differently with larger errors for the expansion!

$$(x - 1)^6 = x^6 - 6x^5 + 15x^4 - 20x^3 + 15x^2 - 6x + 1$$

$x \in [0.998, 1.002]$

Plots for Two Expressions

$$(x - 1)^6 = x^6 - 6x^5 + 15x^4 - 20x^3 + 15x^2 - 6x + 1$$
Kalman Filter Computation

- Important issue in practice.
- Theoretical tests (stabilizable, detectable) can guarantee the convergence of the KF.
- Kalman filter operation over an extended period can result in divergence due to **computational errors**.
- Must use approaches less sensitive to roundoff errors.

Effect of Roundoff Errors

- Errors may result in an asymmetric error covariance matrix $P$.
- If the perturbed $P$ matrix is positive definite it tends to return to the steady-state solution.
- Remedies for $P$ problems.

Reducing Roundoff Problems

1. Use high-precision arithmetic (off-line).
2. Choose step carefully: for sparse data (slow data sampling) avoid excessive $P$ propagation between measurements (small step size).
3. Avoid deterministic (random) processes as they result in an almost semidefinite $P$ and hence to divergence. Adding small noise components $Q = diag\{\epsilon_i\}$ prevents divergence (suboptimal but converges).

Reducing Roundoff Effects (cont.)

4. Propagate the upper triangle of $P^+$ and $P^-$ $n(n + 1)/2$ terms only (to avoid asymmetry).
5. Numerical problems with large uncertainty (large $P^-_k$) followed by an accurate measurement (small $P^+_k$)

$$P^+_k = (I - K_k H_k) \times P^-_k$$

With

- **small** $P^+_k$
- **large** $P^-_k$
Reducing Roundoff Effects (cont.)

Solutions for (5)

a) Use $P_0^-$ with reduced diagonal values (suboptimal filter).

b) Use Joseph’s form: has a natural symmetry

$$P_k^+ = (I - K_k H_k) P_k^- (I - K_k H_k)^T + K_k R_k K_k^T$$

6. Use array algorithms (e.g. propagate square root of $P$): Difficult program but better numerical performance (see Section 5.7 B&H, Simon, or Kailath et al.).

Modeling Errors

- Problem for both deterministic and stochastic constants.
- Errors in system model (state-space matrices).
- Errors in noise model ($Q, R$).

Optimal Error Covariance Analysis

- Kalman filter allows us to compare the relative accuracy of competing designs.
- $P$ can be propagated without the state estimate and without the measurements.
- Off-line analysis.
- Assume a perfect model.
- Assume perfect computation.

Suboptimal Error Covariance Analysis

- Errors in the process model ($\phi, H$) and noise statistics ($Q, R$) result in a suboptimal filter.
- Assess the degree of suboptimality due to modeling errors using $P$.
- Compare $P$ for the optimal and the suboptimal filters.
### Error Propagation Loop

Enter error covariance $P_0$

- **Compute Kalman Gain**
  
  $K_k = P_k^T H_k^T (H_k P_k^T H_k^T + R_k)^{-1}$

- **Project Ahead**
  
  $P_{k+1} = \phi_k P_k \phi_k^T + Q_k$

- **Compute error covariance**
  
  $P_k^* = (I - K_k H_k) P_k$

**Recall:**

Equivalent to Riccati equation for optimal, Lyapunov equation for suboptimal filter.

---

### Three Filters

1. **Optimal Filter:** True $\phi, H, Q, R$, optimal $K$.
2. **Suboptimal Filter 1:** Errors in $\phi, H, Q, R$, suboptimal $K$ give suboptimal state estimates and $P$ (short form) is not meaningful (does not include the effect of modeling errors).
3. **Suboptimal Filter 2:** True $\phi, H, Q, R$, suboptimal $K$ give suboptimal state estimates but $P$ (Joseph’s form) is meaningful.

---

### Implemented Filter

- Errors in $\phi, H, Q, R$, are inevitable.
- Suboptimal $K$ due to computational errors.
- **Suboptimal Filter 1:** suboptimal state estimates and $P$ is not useful (based on wrong data).
- Consider the effect of errors in the noise characterization $Q, R$.

---

### Incorrect $R_k$: Suboptimal Filter 1

- Assume the perturbed matrix $R_k = R_k^T > 0$ with all other parameters correct.
- Errors in a posteriori covariance matrix $P^+$, a priori covariance matrix $P^-$, gain matrix $K$, and state estimates.
- Use gain matrices from suboptimal filter 1 (wrong model) for suboptimal filter 2.
- Filter 2 gives a meaningful error covariance estimate for assessing the effect of $R_k$ errors.
Example: Incorrect R

- First-order Gauss Markov
  \[ x_{k+1} = e^{-\beta \Delta t} x_k + w_k \]
  \[ z_k = x_k + v_k, \quad H = 1 \]
- Initialization (process mean, variance)
  \[ \hat{x}_0^- = 0, \quad P_0^- = \sigma^2 \]
  \[ K_0 = \frac{P_0^-}{P_0^- + R} = \frac{1}{1 + R/\sigma^2}, \quad 1 - K_0 = \frac{R/\sigma^2}{1 + R/\sigma^2} \]
  \[ P_0^+ = (1 - K_0)P_0^- = \frac{R}{1 + R/\sigma^2} \]

Joseph Form

\[ P_k^+ = (I_n - K_k H_k) P_k^- (I_n - K_k H_k)^T + K_k R_k K_k^T \]
\[ P_0^+ = (1 - K_0)^2 P_0^- + R K_0^2 \]
\[ = \left( \frac{R/\sigma^2}{1 + R/\sigma^2} \right)^2 \sigma^2 + \left( \frac{1}{1 + R/\sigma^2} \right)^2 R \]
\[ P_0^+ = \frac{R}{1 + R/\sigma^2}, \quad K_0 = \frac{1}{1 + R/\sigma^2} \]
- Same answer assuming no computational errors, correct \( K_0 \).

Wrong R

- Nominal \( R = \sigma^2 \) \( \Rightarrow \) \( K_0 = \frac{1}{1 + R/\sigma^2} = 0.5 \)
  \[ P_0^+ = \frac{R}{1 + R/\sigma^2} = \frac{\sigma^2}{2} \]
- True \( R = \alpha \sigma^2 \) \( \Rightarrow \) \( K_0 = \frac{1}{1 + R/\sigma^2} = \frac{1}{1 + \alpha} \)
  \[ P_0^+ = \frac{R}{1 + R/\sigma^2} = \frac{\alpha \sigma^2}{1 + \alpha} \]

Calculations

\[
\begin{array}{|c|c|c|}
\hline
R & K_0 & P_0^+ \\
\hline
\sigma^2 & 0.5 & 0.5 \sigma^2 \\
2 \sigma^2 & 1/3 & 2 \sigma^2 /3 \\
\hline
\end{array}
\]
\[ P_0^+ = (1 - K_0)^2 P_0^- + R K_0^2, \quad K_0 = \frac{1}{1 + R/\sigma^2} \]
- For nominal \( R = \sigma^2 \), actual \( R = 2 \sigma^2 \)
  \[ P_0^+ = 0.5 \sigma^2, \quad \text{actual} \ P_0^+ = 0.75 \sigma^2 \]
- Significant difference in \( P_0^+ \) (wrong for filter 1)
Incorrect $Q_k$: Suboptimal Filter 2

- Assume the perturbed matrix $Q_k = Q_k^T > 0$.
- Errors in all equations.
- Use gain matrices from suboptimal filter 1 for suboptimal filter 2.
- Filter 2 gives a meaningful error covariance estimate for assessing the effect of $Q_k$ errors.

Example: $Q$ Incorrect

**Truth Model**
\[
\dot{x}(t) = u(t) \\
x_{k+1} = x_k + w_k, Q = 1 \\
z_k = x_k + v_k, R = 0.1
\]

**KF Model**
\[
x_{k+1} = x_k, Q = 0 \\
z_k = x_k + v_k, R = 0.1
\]

- **Random walk**: white Gaussian process & measurement noise $\phi = 1, H = 1, \Delta t = 1, P_0^- = 1, R = 0.1 m^2$.
- Incorrectly modeled as a random constant $Q = 0$.
- Error in $Q$ only.

## Simulation Results

- Run DKF with $Q = 0, H = 1, R = 0.1$.
- Filter does not follow the correct model (no noise)
  \[
x(k + 1) = x(k)
\]

**Kalman Gain**
\[
K = \frac{P^+}{P^+ + R} = \frac{10P^-}{10P^- + 1} \\
P^+ = (1 - K)P^- = \frac{10P^-}{10P^- + 1}
\]

**Steady State**
\[
P = \frac{P}{10P + 1} \Rightarrow P = 0, \quad K = 0
\]

## B & H Results: Random Walk

![Graph illustrating random walk results](image_url)
Lessons

1. Fixed models are risky if used over an extended period.
2. Instrument biases change with time and often need calibration.
3. Add noise to each state variable to avoid zero noise covariance $Q$: solution will be suboptimal but will converge.

Example: Error Analysis

- One run shows divergence (not proof).
- Can verify results with repeated runs and average but this is hard!
- Assess error using suboptimal filter 2 (correct model) with gains from suboptimal filter 1.
- R.M.S. error approaches a ramp with slope proportional to $(\text{number of steps})^{1/2}$.
- Results verify that divergence occurs and shows rate of growth.

Example: GPS Predictor

\[
\ddot{x}(t) = u(t) \\
\dot{x}(t) = F_{\text{sub}} x(t) + G u(t)
\]

\[
F_{\text{sub}} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad G = \begin{bmatrix} 0 \\ 1 \end{bmatrix}
\]

\[
F_{\text{SA}} = \begin{bmatrix} 0 & 1 \\ -\omega_0^2 & -\sqrt{2}\omega_0 \end{bmatrix}
\]

- Compare optimal to suboptimal predictor with incorrect state matrix (both stationary).
- Cannot recycle gains from suboptimal filter 1 in suboptimal filter 2 (different state-transition matrix).
Discrete Filter Error

\[ x(k + N) = \phi_{SA}(N)x(k) + w(k + N, k) \]

\[ w(k + N, k) \text{ includes } N \text{ terms.} \]

Predictor \[ \hat{x}(k + N|k) = \phi_{sub}(N)\hat{x}(k|k) \]
\[ \phi_{sub}(N) = \begin{bmatrix} 1 & N\Delta t \\ 0 & 1 \end{bmatrix} \]

\[ e(N) = x_{SA}(k + N) - \hat{x}(k + N|k) \]
\[ = [\phi_{SA}(N) - \phi_{sub}(N)]x(k) + w(k + N, k) \]

- Two orthogonal terms.

Error Covariance

- Initialize prediction with \[ \hat{x}(k|k) = x(k) \]
\[ e(N) = [\phi_{SA}(N) - \phi_{sub}(N)]x(k) + w(k + N, k) \]
\[ P_{sub}(N) = E\{e(N)e^T(N)\} \]
\[ = [\phi_{SA}(N) - \phi_{sub}(N)]R_{xx}(0)[\phi_{SA}(N) - \phi_{sub}(N)]^T + Q_{SA}(k + N|k) \]
\[ R_{xx}(0) = E\{x(k)x^T(k)\} \]
\[ = diag\{(30m)^2, (0.36m/sec)^2\} \]

- Use one-step formula iteratively

Comparison of R.M.S. Errors

Comments

- Simulation with MATLAB/SIMULINK is easy.
- Initial stage: difference between optimal and suboptimal predictors insignificant.
- For prediction time = 50 s, R.M.S. error≈10m for optimal predictor, error≈12.5 m for suboptimal.
- Difference becomes large with time.
- Can use a simpler model to reduce computation for small prediction time.