ASEN 5044, Fall 2024

Statistical Estimation for Dynamical Systems

Lecture 31:
DT Linearized Kalman Filter

Prof. Nisar Ahmed (Nisar.Ahmed@Colorado.edu)

Tuesday 11/19/2024

Ann and H.J. Smead
Aerospace Engineering Sciences
UNIVERSITY OF COLORADO **BOULDER**

Announcements

- Midterm 2 being graded…will be ready over Fall Break (along with HW 6-8)

- **Final project partners signup: see instructions + project descriptions + signup link on Canvas – please complete if you have not yet!!!**
  - **Project task + Progress report #1 assignments POSTED (validation data for Part 1 also provided)**

- **HW 8 out, due Fri 11/22 (final hw)**
  - **Office hours at usual time/places this week**

- **Quiz 8 to be released after Fall Break (final quiz)**

- **No lecture/live class this Thurs 11/21 – Prof. Ahmed will hold extra office hours online (Zoom) and in Aero 265 from 11-12 pm**

Last Time: (Semi-)formal problem statement…

- How to define an "optimal" state estimator for a nonlinear dynamical system?

$x(t) \in \mathbb{R}^n$

$y(t) \in \mathbb{R}^p$

Given:

$$\dot{x}(t) = \mathcal{F}[x(t), u(t), \tilde{w}(t)]$$

$$y(t) = \mathcal{H}[x(t), \tilde{v}(t)]$$

$\mathcal{F}$: nonlinear CT dyn fxn (ODE vector fxn)

$\mathcal{H}$: " " meas. fxn (algebraic)

state trans. fxn [e.g. integrate $\mathcal{F}$ from time $k \to k+1$, i.e. via ODE45/Runge-Kutta]

$$x(k+1) = f[x(k), u(k), w(k)], \qquad w(k) = \mathcal{N}(0, Q) \text{ (AWGN)}$$

$$y(k+1) = h[x(k+1), v(k+1)], \qquad v(k) = \mathcal{N}(0, R) \text{ (AWGN)}$$

nonlinear DT fxns

- Follow same logic as before with linear systems to set up a cost fxn J(K)in DT:

$$\text{let} \quad e_k^+ = x_k - \hat{x}_k^+,$$

$$J(k) = E[e_k^{+T} e_k^+] = (E[\text{trace}(e_k^+ e_k^{+T})] = \text{trace}(E[e_k^+ e_k^{+T}]) = \text{trace}(P_k^+)$$

FACT: it is possibly to show that, generally, $J(k)$ is minimized by:

$$\hat{x}_k^+ = E[x_k | y_{1:k}]_{p(x_k|y_{1:k})} \text{ (conditional mean of } x_k \text{ given all data } y_1, \ldots, y_k)$$

→ Can easily show: true for any pdf $p(x_k|y_{1:k})$, i.e. whether Gaussian or not!

Issues with the "Exact" Optimal Estimator

- Theoretically works for <u>any</u> set of dynamics/measurements models → only need to get the posterior pdf $p(x_k \mid y_{1:k})$ and read off its mean and covariance!!

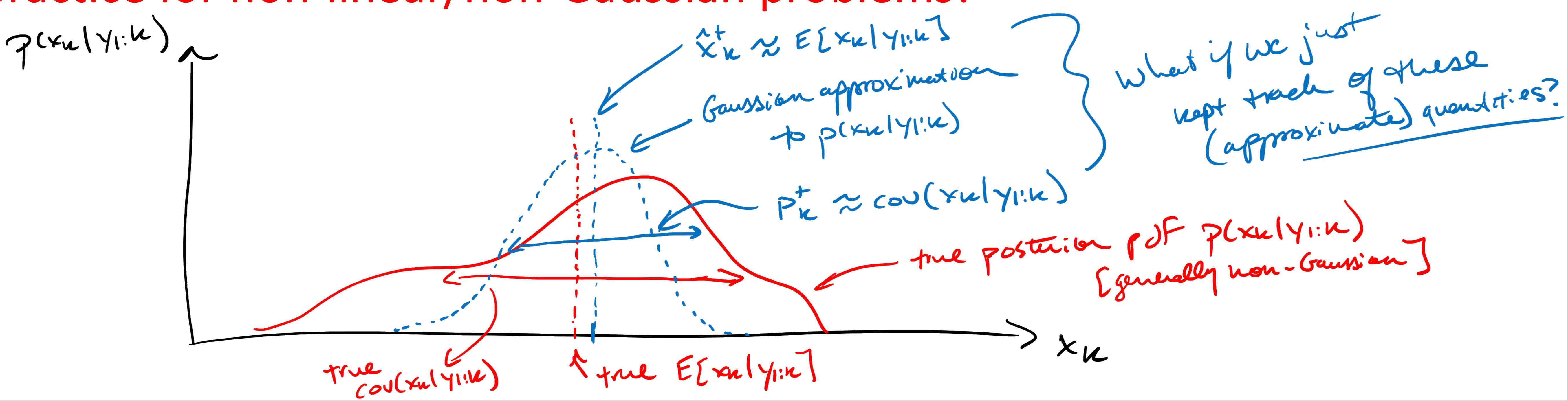
→ The KF for linear-Gaussian systems computes exactly

The posterior mean ↔ $\hat{x}_k^+ = E[x_k | y_{1:k}]$

& " " covariance ↔ $P_k^+ = cov(x_k | y_{1:k})$

Proof: See adv. topic/special lectures 25 & 28!

- But practically computing/representing posterior pdf is also <u>very hard</u> in theory and in practice for non-linear/non-Gaussian problems!



Approximating the Optimal NL Estimator

Optimal non-linear state estimation problem → intractable in general!

<u>Most popular workaround</u>:

- if sample time ΔT is not "too big"…
- And nonlinearities "smooth enough"…

→then can use DT linearization to get <u>approx. optimal solutions from a linear KF</u>

→this approximately tracks posterior pdf $p(x_k | y_{1:k})$ mean and covariance

(don't need full posterior pdf -- just recursively update the first two moments!!)

- <u>Key trick:</u> use given nonlinear CT model to get a "proxy" linearized DT model about some nominal state trajectory to define KF updates

$$\dot{x}(t) = \mathcal{F}[x(t), u(t), w(t)] \longrightarrow \quad x(k+1) = f[x(k), u(k), w(k)] \xrightarrow{\text{linearization}} \approx x_{\text{nom},k} + \tilde{F}_k \delta x_k + \tilde{G}_k \delta u_k + \tilde{\Omega}_k w_k$$

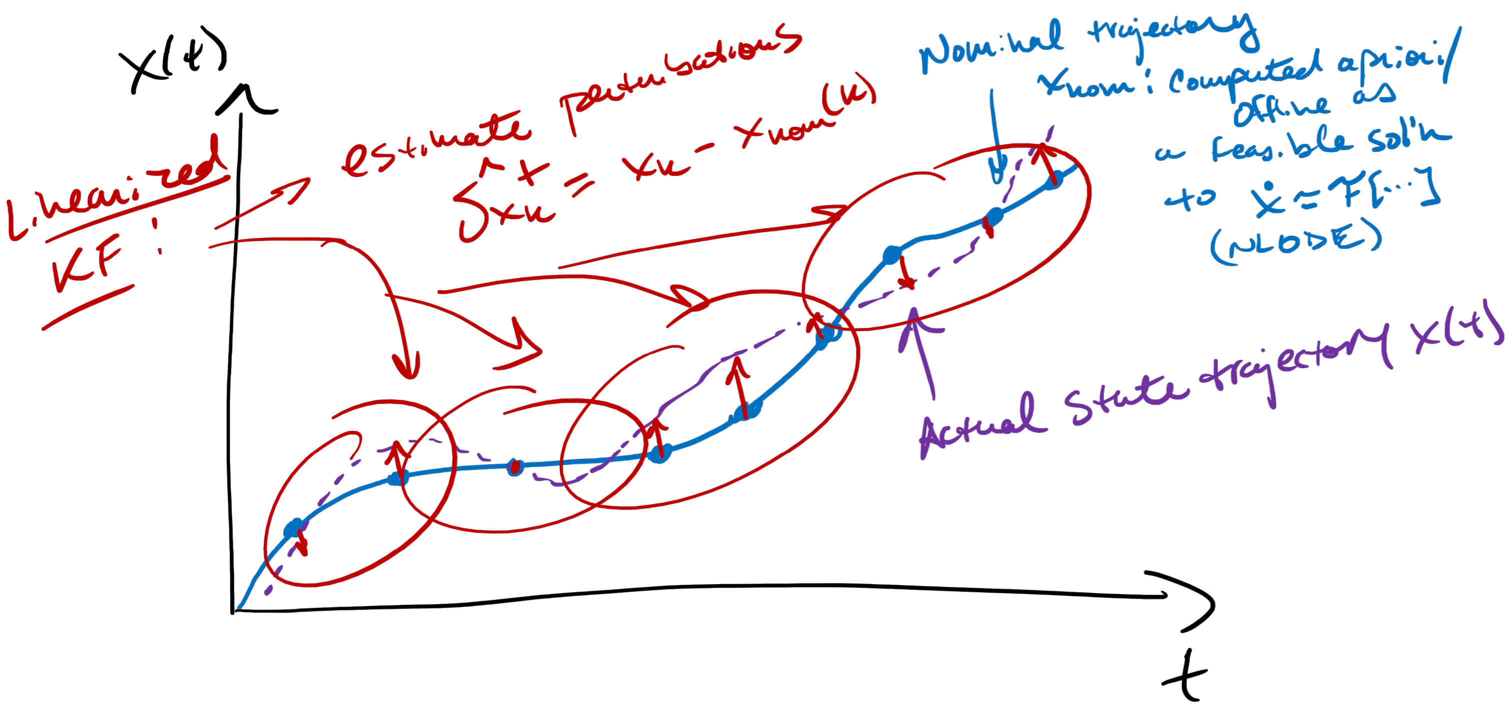
$$y(t) = \mathcal{H}[x(t), v(t)] \xrightarrow{\Delta T} \quad y(k+1) = h[x(k+1), v(k+1)] \xrightarrow[\text{linearization}]{} \approx y_{\text{nom},k+1} + \tilde{H}_{k+1} \delta x_{k+1} + v_{k+1}$$

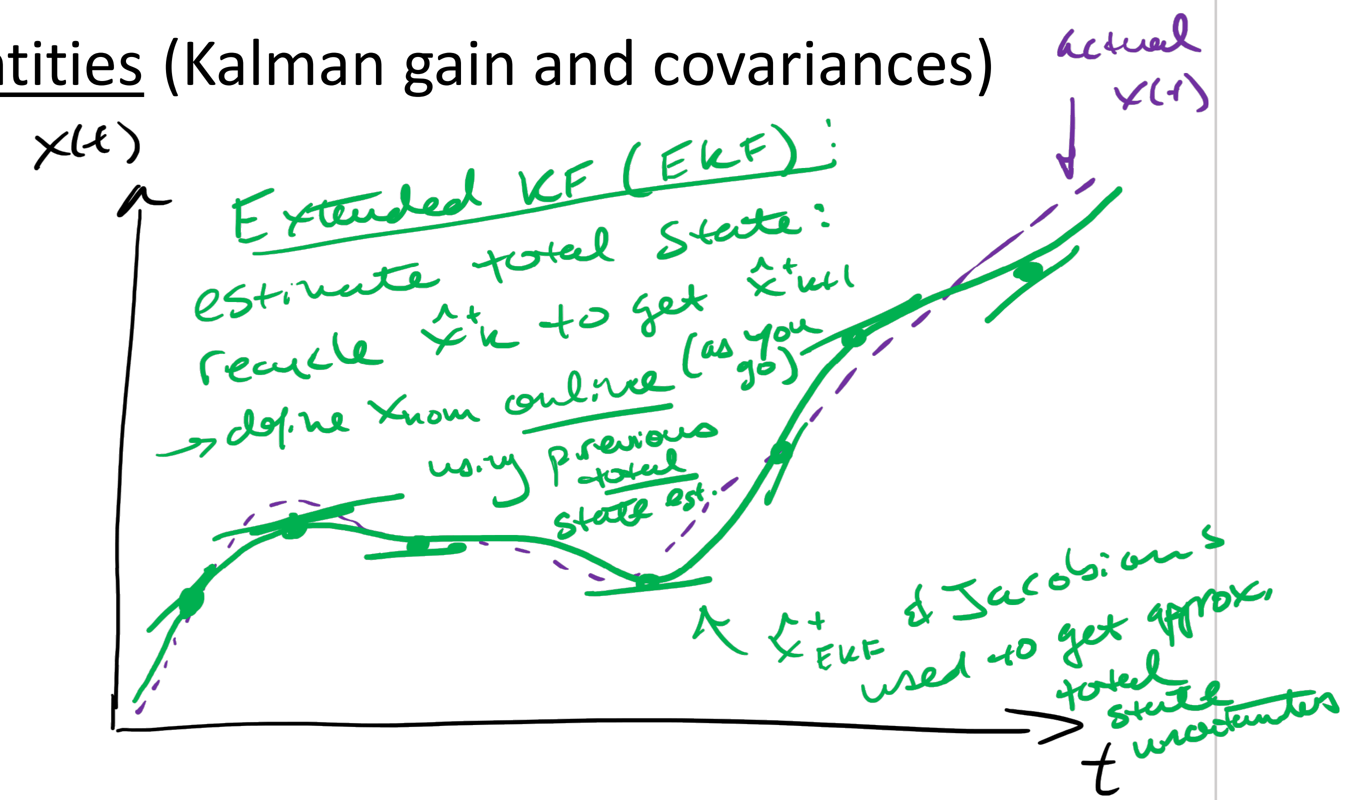
(start w/ NL CT Dyn. & Meas. model) → (DT sample)

Today (and Next Time…)

Two popular sub-optimal "analytic" approximations along these lines:

- **Linearized KF:** estimate *perturbations* around a priori nominal state trajectory:
  - Uses linearization about nominal trajectory for both mean and covariance updates
- **Extended KF (EKF):** estimate *total state* around online estimated trajectory:
  - Uses full nonlinear model for predicted mean and predicted sensor measurements
  - Uses linearization only to approximate matrix quantities (Kalman gain and covariances)





READ SIMON BOOK, CHAPTERS 13.1-13.2 (nonlinear KFs)

Two Subtle Mathematical Issues

- #1: Linearization of CT nonlinear system in general leads to LTV approximation
- Example: consider a 2D robot with a "Dubin's unicycle" model:

$$\left.\begin{aligned} \dot{\zeta}_a &= v_a \cos\theta_a + \tilde{w}_{x,a} \\ \dot{\eta}_a &= v_a \sin\theta_a + \tilde{w}_{y,a} \\ \dot{\theta}_a &= \omega_a + \tilde{w}_{\omega,a} \end{aligned}\right] \rightarrow = \dot{x} = \tilde{\mathcal{F}}[x, u, \tilde{w}],$$

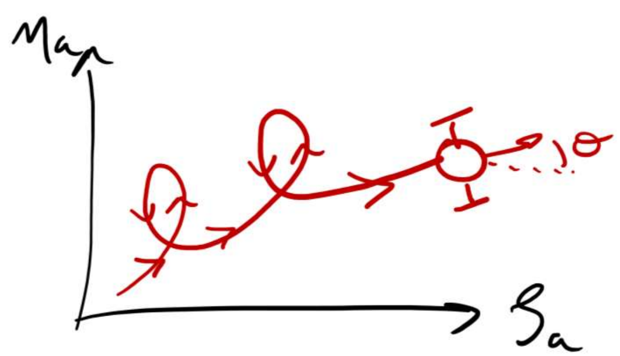
$$x(t) = \begin{bmatrix} \zeta_a \\ \eta_a \\ \theta_a \end{bmatrix}, \quad u(t) = \begin{bmatrix} v_a \\ \omega_a \end{bmatrix}, \quad \tilde{w}(t) = \begin{bmatrix} \tilde{w}_{x,a} \\ \tilde{w}_{y,a} \\ \tilde{w}_{\omega,a} \end{bmatrix}$$

$\Longrightarrow$ CT Jacobian w.r.t. state vars:

$$\frac{\partial \tilde{\mathcal{F}}}{\partial x(t)} = \begin{bmatrix} 0 & 0 & -v_a(t)\sin\theta_a(t) \\ 0 & 0 & v_a(t)\cos\theta_a(t) \\ 0 & 0 & 0 \end{bmatrix}$$

NOT Time Invariant!

Depend on state & input @ time t!



- If we can discretize and then linearize a CT nonlinear model about a time-varying state trajectory, then this generally yields LTV DT model with time-varying $(\tilde{F}_k, \tilde{G}_k, \tilde{\Omega}_k, \tilde{H}_k)$

- BUT: **will linear KF ideas still work for LTV dynamics models???**

Useful Fact #1: The Linear KF for LTV Systems

- The linear KF naturally extends to LTV DT systems as long as the matrices only depend on time (i.e. not also depend on state/inputs)

If *actual* dynamics are truly LTV:

$$x(k+1) = F_k x_k + G_k u_k + \Omega_k w_k$$

$$y(k+1) = H_{k+1} x_{k+1} + v_{k+1}$$

KF Time update/Prediction

$$\hat{x}^-_{k+1} = F_k \hat{x}^+_k + G_k u_k$$

$$P^-_{k+1} = F_k P^+_k F^T_k + \Omega_k Q_k \Omega^T_k$$

KF Meas update/Correction

$$\hat{x}^+_{k+1} = \hat{x}^-_{k+1} + K_{k+1}(y_{k+1} - \hat{y}^-_{k+1})$$

$$P^+_{k+1} = (I - K_{k+1} H_{k+1}) P^-_{k+1}$$

$$K_{k+1} = P^-_{k+1} H^T_{k+1} [S_{k+1}]^{-1}$$

- But, for nonlinear filtering problem, we want to use Jacobians that must be evaluated along state trajectories – so there is a state dependence!
- But we can "cheat" by linearizing around "known nominal trajectory" (solution to nonlinear ODE), so that we can *pretend* it is only a time-varying dependence

$$\Rightarrow (\tilde{F}_k, \tilde{G}_k, \tilde{\Omega}_k, \tilde{H}_k)$$

- So will basically need to cross our fingers and hope that nominal trajectory stays "close enough" to what system actually doing! (hence: no formal guarantees for the linearized KF/EKF…)

Two Subtle Mathematical Issues

- #2: How to get a CT nonlinear model in DT and then linearize it anyway???

➔ tricky/very annoying to exactly find DT Jacobians for $\tilde{F}_k, \tilde{G}_k, \tilde{\Omega}_k$ $\quad (\tilde{H}_k \text{ is easy})$

NL ODEs $\leftrightarrow \mathcal{F} \neq f \leftarrow$ models

(CT) (DT)

$$\dot{x}(t) = \mathcal{F}[x(t), u(t), w(t)] \quad \xrightarrow{\Delta T} \quad x(k+1) = f[x(k), u(k), w(k)] \approx x_{nom}(k+1) + \tilde{F}_k \delta x_k + \tilde{G}_k \delta u_k + \tilde{\Omega}_k w_k$$

$$y(t) = \mathcal{H}[x(t), v(t)] \quad \longrightarrow \quad y(k+1) = h[x(k+1), v(k+1)] \approx y_{nom}(k) + \tilde{H}_{k+1} \delta x_{k+1} + v_{k+1}$$

where: $\tilde{F}_k = \frac{\partial f}{\partial x_k}\Big|_{nom[k]}$ $\quad \tilde{G}_k = \frac{\partial f}{\partial u_k}\Big|_{nom[k]}$ $\quad \tilde{\Omega}_k = \frac{\partial f}{\partial w_k}\Big|_{nom[k]}$ $\quad \tilde{H}_k = \frac{\partial h}{\partial x_k}\Big|_{nom[k]}$ ✓

$(\rightarrow \text{easy since } h = \mathcal{H}!)$

$f[x_k, u_k, w_k]$ generally not closed form $\rightarrow$ DT f Jacobians not closed form !!!

$\rightarrow$ **DT Jacobians must be computed numerically**

(this is one reason some people don't like using linearized KFs/EKFs at all!)

➔ Fortunately, a simple approximation procedure <u>based on CT Jacobians</u> works reasonably well for linearized KF/EKF calculations when ΔT sufficiently small…

Useful Fact #2: “Eulerized” DT Jacobians

- Use Euler integration to approximate DT state transition fxn (f) for small ΔT
- Then take partial derivatives of this to approximate required DT Jacobians
- Naturally get to use CT Jacobians as part of result

Start with (mild) assumption that the CT nonlinear model can be generally written as

$$\dot{x}(t) = \mathcal{F}[x(t), u(t)] + \Gamma(t) \cdot \tilde{w}(t)$$

Euler approx.: $x(t_{k+1}) = x(t_k) + \Delta T \cdot \dot{x}(t)\big|_{t=t_k}$

$\rightarrow x_{k+1} = x_k + \Delta t\, \dot{x}(t=t_k)$

$= x_k + \Delta t \cdot \{ \mathcal{F}[x(t_k), u(t_k)] + \Gamma(t_k)\tilde{w}(t_k) \} \approx f(x_k, u_k, \tilde{w}_k)$

$\rightarrow$ then $\underset{[n \times n]}{\tilde{F}_k} = \frac{\partial f}{\partial x_k}\Big|_{nom[k]} = \frac{\partial x_{k+1}}{\partial x_k}\Big|_{nom[k]} = \frac{\partial}{\partial x_k}\left(x_k + \Delta T \cdot \{ \mathcal{F}[x(t_k), u(t_k)] + \Gamma(t_k)\tilde{w}(t_k) \} \right)\Big|_{nom[k]}$

$= \frac{\partial}{\partial x_k}(x_k) + \frac{\partial}{\partial x_k}\{ \Delta T \cdot \mathcal{F}[\dots] + \Gamma(t_k)\tilde{w}(t_k) \}$

$= I + \Delta T \frac{\partial \mathcal{F}[\dots]}{\partial x(t)}\Big|_{t=t_k, nom[k]} + 0 \rightarrow \boxed{\tilde{F}_k = I + \Delta T \cdot \underbrace{\tilde{A}_{nom[k]}}_{\text{CT Jacobian!}}}$

$= \frac{\partial \mathcal{F}}{\partial x}\Big|_{t=t_k, nom[k]}$

Useful Fact #2 (cont'd): "Eulerized" DT Jacobians

- We can get approximations to remaining DT Jacobians in a similar fashion:

$$\tilde{G}_k = \frac{\partial F}{\partial u_k}\Big|_{nom[k]} = \frac{\partial x_{k+1}}{\partial u_k}\Big|_{nom[k]} \Longrightarrow \Delta T \cdot \underbrace{\frac{\partial \mathcal{F}}{\partial u(t)}}_{\to \tilde{B}|_{nom[k]}}\Big|_{\substack{t=t_k \\ nom[k]}}$$

(use Eulerized Jacobian as on prev. slide) $\to$ $\boxed{\tilde{G}_k = \Delta T \cdot \tilde{B}|_{nom[k]}}$

$$\tilde{\Omega}_k = \frac{\partial F}{\partial \tilde{w}_k}\Big|_{nom[k]} = \frac{\partial x_{k+1}}{\partial \tilde{w}_k}\Big|_{nom[k]} \Rightarrow \boxed{\tilde{\Omega}_k = \Delta T \cdot \Gamma(t=t_k)|_{nom[k]}}$$

$$\Big(\text{Recall Lecture 5 \& 6 definitions of } \underbrace{\tilde{A}|_{nom[k]}, \tilde{B}|_{nom[k]}}_{\text{CT Jacobians.}}, \ldots\Big)$$

The Linearized KF

- Suppose nonlinear system stays near a nominal trajectory x*(t) for some u*(t) with 0 process noise input (desired equilibrium, or offline-calculated nonlinear ODE solution)

$$\dot{x}(t) = \mathcal{F}(x,u) + \Gamma(t)\tilde{w}(t), \text{ where } \tilde{w}, \tilde{v} \text{ are AWGN}$$

$$y(t) = h(x) + \tilde{v}(t),$$

x^* is idealized feasible solution

$\tilde{w}_{nom}(t) = 0 \; \forall t$

$\rightarrow$ Nominal state satisfies: $\dot{x}^*(t) = \mathcal{F}(x^*(t), u^*(t))$ (deterministic solution **with no** process noise)

$\rightarrow$ Now consider actual state evolution **with** process noise present:

$$x(t) = x^*(t) + \delta x(t), \qquad \delta x(t) = x(t) - x^*(t) \text{ (perturbation from } x^*(t))$$
$$u(t) = u^*(t) + \delta u(t), \qquad \delta u(t) = u(t) - u^*(t) \text{ (perturbation from } u^*(t))$$

$\rightarrow$ Plug into dynamics and measurement equation:

$$(\dot{x}^* + \dot{\delta x}) = \mathcal{F}(x^* + \delta x, u^* + \delta u) + \Gamma(t)\tilde{w}(t),$$

$$y(t) = h(x^* + \delta x) + \tilde{v}(t),$$

Linearization via Vector Taylor Series

- Now consider Taylor Series expansion of dynamics and measurement models near x*
- Using results of CT linearization from beginning of the course, we have that

for **small** δx and δu perturbations,

$$(\dot{x}^* + \dot{\delta x}) \approx \mathcal{F}(x^*, u^*) + \frac{\partial \mathcal{F}}{\partial x}|_{(x^*,u^*)}\delta x(t) \;+\; \frac{\partial \mathcal{F}}{\partial u}|_{(x^*,u^*)}\delta u(t) \;+\; \Gamma(t)\tilde{w}(t),$$

$$y(t) \approx h(x^*(t)) + \frac{\partial h}{\partial x}|_{(x^*,u^*)}\delta x(t) \;+\; \tilde{v}(t),$$

$\rightarrow$ simplify using fact that $\dot{x}^*(t) = \mathcal{F}(x^*, u^*)$ and $\delta y(t) = y(t) - h(x^*)$:

$$\dot{\delta x}(t) \approx \frac{\partial \mathcal{F}}{\partial x}|_{(x^*,u^*)}\delta x(t) \;+\; \frac{\partial \mathcal{F}}{\partial u}|_{(x^*,u^*)}\delta u(t) \;+\; \Gamma(t)\tilde{w}(t)$$

$$\delta y(t) \approx \frac{\partial h}{\partial x}|_{(x^*,u^*)}\delta x(t) + \tilde{v}(t)$$

$$\rightarrow \dot{\delta x}(t) \approx \tilde{A}|_{(x^*,u^*)}\delta x(t) + \tilde{B}|_{(x^*,u^*)}\delta u(t) + \Gamma(t)\tilde{w}(t),$$

$$\rightarrow \delta y(t) \approx \tilde{C}|_{(x^*,u^*)}\delta x(t) + \tilde{v}(t)$$

The Linearized KF Model

- **Thus:**

$$\left.\begin{aligned}\dot{\delta x}(t) &\approx \tilde{A}|_{(x^*,u^*)}\delta x(t) + \tilde{B}|_{(x^*,u^*)}\delta u(t) + \Gamma(t)\tilde{w}(t),\\ \delta y(t) &\approx \tilde{C}|_{(x^*,u^*)}\delta x(t) + \tilde{v}(t),\end{aligned}\right\}$$

(stochastic)
CT perturbation
dynamics model

where $\tilde{A}, \tilde{B}, \tilde{C}$ are the CT Jacobian matrices evaluated at (x^*, u^*)

$\rightarrow$ now convert CT perturbation model into DT model:

$$\delta x(k+1) \approx \tilde{F}_k|_{nom[k]}\delta x(k) \ + \ \tilde{G}_k|_{nom[k]}\delta u(k) \ + \ \tilde{\Omega}_k w(k),$$

$$\delta y(k+1) \approx \tilde{H}_{k+1}|_{nom[k+1]}\delta x(k+1) + v(k+1)$$

where we already showed earlier that: (for sufficiently small ΔT):

$$\tilde{F}_k|_{nom[k]} \approx I + \Delta T \cdot \tilde{A}|_{(x^*,u^*,t=t_k)}, \qquad \tilde{G}_k|_{nom[k]} \approx \Delta T \cdot \tilde{B}|_{(x^*,u^*,t=t_k)},$$

$$\tilde{\Omega}_k|_{nom[k]} \approx \Delta T \cdot \Gamma(t)|_{(t=t_k)}, \qquad \tilde{H}_{k+1}|_{nom[k+1]} = \tilde{C}|_{(x^*,u^*,t=t_{k+1})} = \frac{\partial h}{\partial x}|_{(x^*,u^*,t=t_{k+1})}$$

The Linearized KF Algorithm

- So now we can estimate the total state as follows:

Computed offline → deterministic

$$\hat{x}^+_{k+1} \approx x^*_{k+1} + \hat{\delta x}^+_{k+1}$$

→ random correction computed online

where $x^*_{k+1} = x^*(t = t_{k+1})$ and $\hat{\delta x}_{k+1}$ is estimated using LTV KF for δx_{k+1} and δy_{k+1}:

Time update/prediction step for time k+1:

$$\hat{\delta x}^-_{k+1} = \tilde{F}_k \hat{\delta x}^+_k + \tilde{G}_k \delta u_k$$

$$P^-_{k+1} = \tilde{F}_k P^+_k \tilde{F}^T_k + \tilde{\Omega}_k Q_k \tilde{\Omega}^T_k$$

$$\delta u_{k+1} = u_{k+1} - u^*_{k+1}$$

Measurement update/correction step for time k+1:

$$\hat{\delta x}^+_{k+1} = \hat{\delta x}^-_{k+1} + K_{k+1}(\underline{\delta y_{k+1}} - \tilde{H}_{k+1} \hat{\delta x}^-_{k+1})$$

$$P^+_{k+1} = (I - K_{k+1}\tilde{H}_{k+1})P^-_{k+1}$$

$$K_{k+1} = P^-_{k+1}\tilde{H}^T_{k+1}[\tilde{H}_{k+1}P^-_{k+1}\tilde{H}^T_{k+1} + R_{k+1}]^{-1}$$

$$\delta y_{k+1} = \underbrace{y_{k+1}}_{\text{Actual received sensor measurement at time k+1}} - \underbrace{y^*_{k+1}}_{\text{Computed nominal sensor measurement at time k+1}} = y_{k+1} - \underbrace{h(x^*_{k+1})}_{\text{Computed nominal sensor measurement at time k+1}}$$

where $\tilde{F}_k, \tilde{G}_k, \tilde{\Omega}_k, \tilde{H}_k$ eval'd along (x^*, u^*) nom. sol'n at each time step k