Lecture 3: Batch Linear-Gaussian Estimation

AER1513: State Estimation

Timothy D. Barfoot

University of Toronto

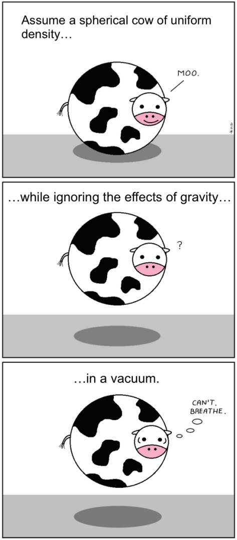
Copyright © 2023

Outline

Lecture 3: Batch Linear-Gaussian Estimation

- Problem Setup
- Bayesian Inference
- Maximum A Posteriori
- Sparsity
- Existence and Uniqueness
- Uncertainty

Why linear-Gaussian?


Assume a spherical cow of uniform density…
MOO.
…while ignoring the effects of gravity…
?
…in a vacuum.
CAN'T. BREATHE.

- most of the classical results are for linear-Gaussian systems
- we can often find results without approximation
- we can get some intuition for real systems by studying them

> *"Equation (1.2-9) is a second order, nonlinear, vector, differential equation which has defied solution in its present form. It is here therefore we depart from the realities of nature to make some simplifying assumptions…"*
>
> – *Bate et al., Fundamentals of Astrodynamics (1971)*

> *"Linear systems are very important because we can solve them exactly!"*
>
> – *Sami Mikhail (heard third hand via Raja Mukherji)*

System

- we define our system using the following linear, time-varying models:

$$\text{motion model:} \quad \mathbf{x}_k = \mathbf{A}_{k-1}\mathbf{x}_{k-1} + \mathbf{v}_k + \mathbf{w}_k, \quad k = 1 \ldots K \tag{1a}$$

$$\text{observation model:} \quad \mathbf{y}_k = \mathbf{C}_k\mathbf{x}_k + \mathbf{n}_k, \quad k = 0 \ldots K \tag{1b}$$

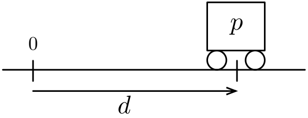
  where k is the discrete-time index and K its maximum

- the variables have the following meanings:

system state :	$\mathbf{x}_k \in \mathbb{R}^N$
initial state :	$\mathbf{x}_0 \in \mathbb{R}^N \sim \mathcal{N}\left(\check{\mathbf{x}}_0, \check{\mathbf{P}}_0\right)$
input :	$\mathbf{v}_k \in \mathbb{R}^N$
process noise :	$\mathbf{w}_k \in \mathbb{R}^N \sim \mathcal{N}\left(\mathbf{0}, \mathbf{Q}_k\right)$
measurement :	$\mathbf{y}_k \in \mathbb{R}^M$
measurement noise :	$\mathbf{n}_k \in \mathbb{R}^M \sim \mathcal{N}\left(\mathbf{0}, \mathbf{R}_k\right)$

UNIVERSITY OF TORONTO

System example



$$\ddot{p}(t) = a(t) + w(t)$$

$$\dot{p}(t) = v(t)$$
$$\dot{v}(t) = a(t) + w(t)$$

$$p_k \approx p_{k-1} + T v_{k-1} + \frac{1}{2} T^2 (a_k + w_k)$$
$$v_k \approx v_{k-1} + T\,(a_k + w_k)$$

$$d_k = p_k + n_k$$

$$\underbrace{d_k}_{\mathbf{y}_k} = \underbrace{\begin{bmatrix} 1 & 0 \end{bmatrix}}_{\mathbf{C}_k} \underbrace{\begin{bmatrix} p_k \\ v_k \end{bmatrix}}_{\mathbf{x}_k} + \underbrace{n_k}_{\mathbf{n}_k}$$

$$\underbrace{\begin{bmatrix} p_k \\ v_k \end{bmatrix}}_{\mathbf{x}_k} = \underbrace{\begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix}}_{\mathbf{A}_{k-1}} \underbrace{\begin{bmatrix} p_{k-1} \\ v_{k-1} \end{bmatrix}}_{\mathbf{x}_{k-1}} + \underbrace{\begin{bmatrix} \frac{1}{2}T^2 \\ T \end{bmatrix} a_k}_{\mathbf{v}_k} + \underbrace{\begin{bmatrix} \frac{1}{2}T^2 \\ T \end{bmatrix} w_k}_{\mathbf{w}_k}$$

observation model

motion model

UNIVERSITY OF TORONTO

What do we know?

- although we want to know the state of the system (at all times), we only have access to the following quantities, and must base our estimate, $\hat{\mathbf{x}}_k$, on just this information:
  (i) the initial state knowledge, $\check{\mathbf{x}}_0$, and the associated covariance matrix, $\check{\mathbf{P}}_0$; sometimes we do not have this piece of information and must do without
  (ii) the inputs, $\mathbf{v}_k$, which typically come from the output of our controller and so are known; we also have the associated process noise covariance, $\mathbf{Q}_k$
  (iii) the measurements, $\mathbf{y}_{k,\text{meas}}$, which are realizations of the associated random variables, $\mathbf{y}_k$, and the associated covariance matrix, $\mathbf{R}_k$
- we will use $\hat{(\cdot)}$ to indicate posterior estimates (incorporating measurements) and $\check{(\cdot)}$ to indicate prior estimates (not incorporating measurements)

Problem statement

- we define the state estimation problem as follows:

Definition

The problem of state estimation is to come up with an estimate, $\hat{\mathbf{x}}_k$, of the true *state* of a system, at one or more timesteps, k, given knowledge of the initial state, $\check{\mathbf{x}}_0$, a sequence of measurements, $\mathbf{y}_{0:K,\text{meas}}$, a sequence of inputs, $\mathbf{v}_{1:K}$, as well as knowledge of the system's motion and observation models.

- the rest of the linear-Gaussian section will investigate a suite of techniques for addressing this state estimation problem
- our approach will always be to attempt to come up with not only a state estimate, but also to quantify the uncertainty in that estimate

Batch is best

- we will begin by investigating batch linear-Gaussian techniques (sometimes called smoothers)
- the batch solution is very useful for computing state estimates after the fact because it uses all the measurements in the estimation of all the states at once (hence the usage of 'batch')
- a batch method cannot be used online since we cannot employ future measurements to estimate past states
- in the next lecture we'll look at recursive state estimators (called filters), which can be used online

smoothers use all available information to estimate states

$$\overbrace{\underbrace{\check{\mathbf{x}}_0, \mathbf{y}_0, \mathbf{v}_1, \mathbf{y}_1, \mathbf{v}_2, \mathbf{y}_2, \ldots, \mathbf{v}_{k-1}, \mathbf{y}_{k-1}, \mathbf{v}_k, \mathbf{y}_k}_{\hat{\mathbf{x}}_{k,f}}, \mathbf{v}_{k+1}, \mathbf{y}_{k+1}, \ldots, \mathbf{v}_K, \mathbf{y}_K}^{\hat{\mathbf{x}}_k}$$

filters only use past/current information to estimate states

All roads lead to Rome

- we will set up the batch linear-Gaussian estimation problem using two different paradigms:
  (i) Bayesian inference; here we update a prior density over states (based on the initial state knowledge, inputs, and motion model) with our measurements, to produce a posterior (Gaussian) density over states
  (ii) maximum a posteriori; here we employ optimization to find the most likely posterior state given the information we have (initial state knowledge, measurements, inputs)
- while these approaches are somewhat different in nature, it turns out that we arrive at the exact same answer for the linear-Gaussian problem
- this is because the full Bayesian posterior is exactly Gaussian and so the optimization approach will find the maximum (i.e., mode) of a Gaussian, and this is the same as the mean

A marriage of convenience

- we will combine the initial state knowledge and inputs in the following way:

$$\color{red}{\mathbf{v}} = (\check{\mathbf{x}}_0, \mathbf{v}_{1:K}) = (\check{\mathbf{x}}_0, \mathbf{v}_1, \ldots, \mathbf{v}_K) \tag{2}$$

  since they are both related to our prior knowledge of the state
- we will also define

$$\color{red}{\mathbf{y}} = \mathbf{y}_{0:K} = (\mathbf{y}_0, \ldots, \mathbf{y}_K) \tag{3}$$

  to denote all of our measurements and

$$\color{red}{\mathbf{x}} = \mathbf{x}_{0:K} = (\mathbf{x}_0, \ldots, \mathbf{x}_K) \tag{4}$$

  for our entire state (all timesteps)

UNIVERSITY OF TORONTO

Approach (i): Bayesian inference

- we would like to compute the full Bayesian posterior:

$$p(\mathbf{x}|\mathbf{y},\mathbf{v}) = \frac{p(\mathbf{y}|\mathbf{x},\mathbf{v})p(\mathbf{x}|\mathbf{v})}{p(\mathbf{y}|\mathbf{v})} \tag{5}$$

- we will take the two-step approach of building a joint density over states, inputs, measurements,

$$p(\mathbf{x},\mathbf{y}|\mathbf{v}) = \underbrace{p(\mathbf{y}|\mathbf{x},\mathbf{v})}_{\text{observations}} \underbrace{p(\mathbf{x}|\mathbf{v})}_{\text{prior}} \tag{6}$$

  and then factor it the other way, keeping only the bit we want:

$$p(\mathbf{x},\mathbf{y}|\mathbf{v}) = \underbrace{p(\mathbf{x}|\mathbf{y},\mathbf{v})}_{\text{posterior}}\underbrace{p(\mathbf{y}|\mathbf{v})}_{\text{discard}} \tag{7}$$

UNIVERSITY OF TORONTO

Prior

- recall our motion model,

$$\mathbf{x}_k = \mathbf{A}_{k-1}\mathbf{x}_{k-1} + \mathbf{v}_k + \mathbf{w}_k \tag{8}$$

- in lifted matrix form (i.e., whole trajectory), we can write this as

$$\mathbf{x} = \mathbf{A}(\mathbf{v} + \mathbf{w}) \tag{9}$$

where $\mathbf{w}$ is the lifted form of the initial state and process noise, and

$$\mathbf{A} = \begin{bmatrix} \mathbf{1} & & & & & \\ \mathbf{A}_0 & \mathbf{1} & & & & \\ \mathbf{A}_1\mathbf{A}_0 & \mathbf{A}_1 & \mathbf{1} & & & \\ \vdots & \vdots & \vdots & \ddots & & \\ \mathbf{A}_{K-2}\cdots\mathbf{A}_0 & \mathbf{A}_{K-2}\cdots\mathbf{A}_1 & \mathbf{A}_{K-2}\cdots\mathbf{A}_2 & \cdots & \mathbf{1} & \\ \mathbf{A}_{K-1}\cdots\mathbf{A}_0 & \mathbf{A}_{K-1}\cdots\mathbf{A}_1 & \mathbf{A}_{K-1}\cdots\mathbf{A}_2 & \cdots & \mathbf{A}_{K-1} & \mathbf{1} \end{bmatrix} \tag{10}$$

is the lifted transition matrix, which we see is lower-triangular

Prior

- the lifted mean is then

$$\check{\mathbf{x}} = E[\mathbf{x}] = E[\mathbf{A}(\mathbf{v} + \mathbf{w})] = \mathbf{A}\mathbf{v} \tag{11}$$

- the lifted covariance is

$$\check{\mathbf{P}} = E\left[(\mathbf{x} - E[\mathbf{x}])(\mathbf{x} - E[\mathbf{x}])^T\right] = \mathbf{A}\mathbf{Q}\mathbf{A}^T \tag{12}$$

  where $\mathbf{Q} = E[\mathbf{w}\mathbf{w}^T] = \text{diag}(\check{\mathbf{P}}_0, \mathbf{Q}_1, \ldots, \mathbf{Q}_K)$

- our prior can then be neatly expressed as

$$p(\mathbf{x}|\mathbf{v}) = \mathcal{N}\left(\check{\mathbf{x}}, \check{\mathbf{P}}\right) = \mathcal{N}\left(\mathbf{A}\mathbf{v}, \mathbf{A}\mathbf{Q}\mathbf{A}^T\right) \tag{13}$$

UNIVERSITY OF TORONTO

Measurements

- we now want to compute $p(\mathbf{x}, \mathbf{y} | \mathbf{v}) = p(\mathbf{y}|\mathbf{x}, \mathbf{v}) p(\mathbf{x}|\mathbf{v})$
- we will use the measurement model

$$\mathbf{y}_k = \mathbf{C}_k \mathbf{x}_k + \mathbf{n}_k \qquad (14)$$

- this can also be written in lifted form as

$$\mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{n} \qquad (15)$$

where $\mathbf{n}$ is the lifted form of the measurement noise and

$$\mathbf{C} = \text{diag}\left(\mathbf{C}_0, \mathbf{C}_1, \ldots, \mathbf{C}_K\right) \qquad (16)$$

is the lifted observation matrix

Joint Gaussian

- the joint likelihood of the prior lifted state and the measurements can now be written as

$$p(\mathbf{x}, \mathbf{y}|\mathbf{v}) = \mathcal{N}\left(\begin{bmatrix} \check{\mathbf{x}} \\ \mathbf{C}\check{\mathbf{x}} \end{bmatrix}, \begin{bmatrix} \check{\mathbf{P}} & \check{\mathbf{P}}\mathbf{C}^T \\ \mathbf{C}\check{\mathbf{P}} & \mathbf{C}\check{\mathbf{P}}\mathbf{C}^T + \mathbf{R} \end{bmatrix} \right) \tag{17}$$

  where $\mathbf{R} = E[\mathbf{n}\mathbf{n}^T] = \text{diag}(\mathbf{R}_0, \mathbf{R}_1, \ldots, \mathbf{R}_K)$
- we can factor this according to

$$p(\mathbf{x}, \mathbf{y}|\mathbf{v}) = p(\mathbf{x}|\mathbf{y}, \mathbf{v})p(\mathbf{y}|\mathbf{v}) \tag{18}$$

- we only care about the first factor, which is the full Bayesian posterior

UNIVERSITY OF TORONTO

Posterior

- this can be written, using the Gaussian inference approach from the lecture on probability, as

$$p(\mathbf{x}|\mathbf{v},\mathbf{y}) = \mathcal{N}\Big(\check{\mathbf{x}} + \check{\mathbf{P}}\mathbf{C}^T(\mathbf{C}\check{\mathbf{P}}\mathbf{C}^T + \mathbf{R})^{-1}(\mathbf{y} - \mathbf{C}\check{\mathbf{x}}), \\ \check{\mathbf{P}} - \check{\mathbf{P}}\mathbf{C}^T(\mathbf{C}\check{\mathbf{P}}\mathbf{C}^T + \mathbf{R})^{-1}\mathbf{C}\check{\mathbf{P}}\Big) \tag{19}$$

- using the Sherman-Morrison-Woodbury identity, this can be manipulated into the following form:

$$p(\mathbf{x}|\mathbf{v},\mathbf{y}) = \mathcal{N}\Bigg(\underbrace{\left(\check{\mathbf{P}}^{-1} + \mathbf{C}^T\mathbf{R}^{-1}\mathbf{C}\right)^{-1}\left(\check{\mathbf{P}}^{-1}\check{\mathbf{x}} + \mathbf{C}^T\mathbf{R}^{-1}\mathbf{y}\right)}_{\hat{\mathbf{x}},\ \text{mean}}, \\ \underbrace{\left(\check{\mathbf{P}}^{-1} + \mathbf{C}^T\mathbf{R}^{-1}\mathbf{C}\right)^{-1}}_{\hat{\mathbf{P}},\ \text{covariance}}\Bigg) \tag{20}$$

- we could use this as is, but there is a better way

Posterior

- we can rearrange the mean expression to arrive at

$$\underbrace{\left(\check{\mathbf{P}}^{-1} + \mathbf{C}^T\mathbf{R}^{-1}\mathbf{C}\right)}_{\hat{\mathbf{P}}^{-1}} \hat{\mathbf{x}} = \check{\mathbf{P}}^{-1}\check{\mathbf{x}} + \mathbf{C}^T\mathbf{R}^{-1}\mathbf{y} \tag{21}$$

  and we see the inverse covariance appearing on the left-hand side

- substituting in $\check{\mathbf{x}} = \mathbf{A}\mathbf{v}$ and $\check{\mathbf{P}}^{-1} = \left(\mathbf{A}\mathbf{Q}\mathbf{A}^T\right)^{-1} = \mathbf{A}^{-T}\mathbf{Q}^{-1}\mathbf{A}^{-1}$ we can rewrite this as

$$\underbrace{\left(\mathbf{A}^{-T}\mathbf{Q}^{-1}\mathbf{A}^{-1} + \mathbf{C}^T\mathbf{R}^{-1}\mathbf{C}\right)}_{\hat{\mathbf{P}}^{-1}} \hat{\mathbf{x}} = \mathbf{A}^{-T}\mathbf{Q}^{-1}\mathbf{v} + \mathbf{C}^T\mathbf{R}^{-1}\mathbf{y} \tag{22}$$

- or more compactly,

$$\left(\mathbf{H}^T\mathbf{W}^{-1}\mathbf{H}\right) \hat{\mathbf{x}} = \mathbf{H}^T\mathbf{W}^{-1}\mathbf{z} \tag{23}$$

$$\mathbf{z} = \begin{bmatrix} \mathbf{v} \\ \mathbf{y} \end{bmatrix}, \quad \mathbf{H} = \begin{bmatrix} \mathbf{A}^{-1} \\ \mathbf{C} \end{bmatrix}, \quad \mathbf{W} = \begin{bmatrix} \mathbf{Q} & \\ & \mathbf{R} \end{bmatrix} \tag{24}$$

Approach (ii): maximum a posteriori (MAP)

- instead of computing the full Bayesian posterior, what if we just tried to find the most likely state of our system given the initial state knowledge, inputs, and measurements?
- to accomplish this, we could try to solve the following optimization problem:

$$\hat{\mathbf{x}} = \arg\max_{\mathbf{x}} p(\mathbf{x}|\mathbf{y}, \mathbf{v}) \tag{25}$$

  which is to say that we want to find the best single estimate for the state of the system (at all timesteps), $\hat{\mathbf{x}}$, given the prior information, $\mathbf{v}$, and measurements, $\mathbf{y}$

MAP

– we begin by rewriting the MAP estimate using Bayes' rule:

$$\hat{\mathbf{x}} = \arg\max_{\mathbf{x}} p(\mathbf{x}|\mathbf{y}, \mathbf{v}) = \arg\max_{\mathbf{x}} \frac{p(\mathbf{y}|\mathbf{x}, \mathbf{v})p(\mathbf{x}|\mathbf{v})}{p(\mathbf{y}|\mathbf{v})}$$
$$= \arg\max_{\mathbf{x}} p(\mathbf{y}|\mathbf{x})p(\mathbf{x}|\mathbf{v}) \qquad (26)$$

where we drop the denominator because it does not depend on $\mathbf{x}$

– we also drop $\mathbf{v}$ in $p(\mathbf{y}|\mathbf{x}, \mathbf{v})$ since it does not affect $\mathbf{y}$ in our system if $\mathbf{x}$ is known (see observation model)

MAP

- if we assume that all of the noise variables, $\mathbf{w}_k$ and $\mathbf{n}_k$ for $k = 0 \ldots K$, are uncorrelated, we can use Bayes' rule to factor $p(\mathbf{y}|\mathbf{x})$ in the following way:

$$p(\mathbf{y}|\mathbf{x}) = \prod_{k=0}^{K} p(\mathbf{y}_k \,|\, \mathbf{x}_k) \tag{27}$$

- furthermore, Bayes' rule allows us to factor $p(\mathbf{x}|\mathbf{v})$ as

$$p(\mathbf{x}|\mathbf{v}) = p(\mathbf{x}_0 \,|\, \check{\mathbf{x}}_0) \prod_{k=1}^{K} p(\mathbf{x}_k \,|\, \mathbf{x}_{k-1}, \mathbf{v}_k) \tag{28}$$

Negative log likelihood

- to make the optimization easier, we insert the (negative) logarithm function, which is monotonically decreasing

$$\begin{aligned}\hat{\mathbf{x}} &= \arg\min_{\mathbf{x}} \left(-\ln(p(\mathbf{y}|\mathbf{x})p(\mathbf{x}\,|\mathbf{v}))\right) \\ &= \arg\min_{\mathbf{x}} \Bigg(-\ln p(\mathbf{x}_0 \,|\, \check{\mathbf{x}}_0) - \sum_{k=1}^{K} \ln p(\mathbf{x}_k \,|\, \mathbf{x}_{k-1}, \mathbf{v}_k) \\ &\qquad - \sum_{k=0}^{K} \ln p(\mathbf{y}_k \,|\, \mathbf{x}_k) \Bigg) \end{aligned} \tag{29}$$

- note, we are now minimizing due to the negative sign, which is more common in optimization

MAP

- the component Gaussian densities are given by

$$p(\mathbf{x}_0|\check{\mathbf{x}}_0) = \frac{1}{\sqrt{(2\pi)^N \det \check{\mathbf{P}}_0}} \exp\left(-\frac{1}{2}(\mathbf{x}_0 - \check{\mathbf{x}}_0)^T \check{\mathbf{P}}_0^{-1} (\mathbf{x}_0 - \check{\mathbf{x}}_0)\right) \tag{30a}$$

$$\begin{aligned} p(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{v}_k) = & \frac{1}{\sqrt{(2\pi)^N \det \mathbf{Q}_k}} \exp\left(-\frac{1}{2}(\mathbf{x}_k - \mathbf{A}_{k-1}\mathbf{x}_{k-1} - \mathbf{v}_k)^T\right. \\ & \left. \times \mathbf{Q}_k^{-1} (\mathbf{x}_k - \mathbf{A}_{k-1}\mathbf{x}_{k-1} - \mathbf{v}_k)\right) \end{aligned} \tag{30b}$$

$$p(\mathbf{y}_k|\mathbf{x}_k) = \frac{1}{\sqrt{(2\pi)^M \det \mathbf{R}_k}} \exp\left(-\frac{1}{2}(\mathbf{y}_k - \mathbf{C}_k\mathbf{x}_k)^T \mathbf{R}_k^{-1} (\mathbf{y}_k - \mathbf{C}_k\mathbf{x}_k)\right) \tag{30c}$$

MAP

- the component negative-log densities are given by

$$-\ln p(\mathbf{x}_0|\check{\mathbf{x}}_0) = \frac{1}{2}\left(\mathbf{x}_0-\check{\mathbf{x}}_0\right)^T \check{\mathbf{P}}_0^{-1}\left(\mathbf{x}_0-\check{\mathbf{x}}_0\right) + \underbrace{\frac{1}{2}\ln\left((2\pi)^N \det \check{\mathbf{P}}_0\right)}_{\text{independent of } \mathbf{x}} \quad (31a)$$

$$-\ln p(\mathbf{x}_k|\mathbf{x}_{k-1},\mathbf{v}_k) = \frac{1}{2}\left(\mathbf{x}_k-\mathbf{A}_{k-1}\mathbf{x}_{k-1}-\mathbf{v}_k\right)^T \mathbf{Q}_k^{-1}\left(\mathbf{x}_k-\mathbf{A}_{k-1}\mathbf{x}_{k-1}-\mathbf{v}_k\right) + \underbrace{\frac{1}{2}\ln\left((2\pi)^N \det \mathbf{Q}_k\right)}_{\text{independent of } \mathbf{x}} \quad (31b)$$

$$-\ln p(\mathbf{y}_k|\mathbf{x}_k) = \frac{1}{2}\left(\mathbf{y}_k-\mathbf{C}_k\mathbf{x}_k\right)^T \mathbf{R}_k^{-1}\left(\mathbf{y}_k-\mathbf{C}_k\mathbf{x}_k\right) + \underbrace{\frac{1}{2}\ln\left((2\pi)^M \det \mathbf{R}_k\right)}_{\text{independent of } \mathbf{x}} \quad (31c)$$

MAP as optimization

- from an optimization perspective, we want to solve the following problem:

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} J(\mathbf{x}) \tag{32}$$

- our cost function is

$$J(\mathbf{x}) = \sum_{k=0}^{K} \left(J_{v,k}(\mathbf{x}) + J_{y,k}(\mathbf{x}) \right) \tag{33}$$

with

$$J_{v,k}(\mathbf{x}) = \begin{cases} \frac{1}{2}\left(\mathbf{x}_0 - \check{\mathbf{x}}_0\right)^T \check{\mathbf{P}}_0^{-1}\left(\mathbf{x}_0 - \check{\mathbf{x}}_0\right) & k = 0 \\ \frac{1}{2}\left(\mathbf{x}_k - \mathbf{A}_{k-1}\mathbf{x}_{k-1} - \mathbf{v}_k\right)^T & \\ \quad \times \mathbf{Q}_k^{-1}\left(\mathbf{x}_k - \mathbf{A}_{k-1}\mathbf{x}_{k-1} - \mathbf{v}_k\right) & k = 1 \ldots K \end{cases} \tag{34a}$$

$$J_{y,k}(\mathbf{x}) = \frac{1}{2}\left(\mathbf{y}_k - \mathbf{C}_k\mathbf{x}_k\right)^T \mathbf{R}_k^{-1}\left(\mathbf{y}_k - \mathbf{C}_k\mathbf{x}_k\right) \qquad k = 0 \ldots K \tag{34b}$$

MAP as optimization

- we can express our cost function more compactly as

$$J(\mathbf{x}) = \frac{1}{2} (\mathbf{z} - \mathbf{H}\mathbf{x})^T \mathbf{W}^{-1} (\mathbf{z} - \mathbf{H}\mathbf{x}) \tag{35}$$

  recalling our definitions from the full Bayesian approach:

$$\mathbf{z} = \begin{bmatrix} \mathbf{v} \\ \mathbf{y} \end{bmatrix}, \quad \mathbf{H} = \begin{bmatrix} \mathbf{A}^{-1} \\ \mathbf{C} \end{bmatrix}, \quad \mathbf{W} = \begin{bmatrix} \mathbf{Q} & \\ & \mathbf{R} \end{bmatrix} \tag{36}$$

- we see that $J(\mathbf{x})$ is exactly quadratic in $\mathbf{x}$

UNIVERSITY OF TORONTO

MAP as optimization

- since $J(\mathbf{x})$ is exactly a paraboloid, we can find its minimum in closed form
- simply set the partial derivative with respect to the design variable, $\mathbf{x}$, to zero:

$$\left.\frac{\partial J(\mathbf{x})}{\partial \mathbf{x}^T}\right|_{\hat{\mathbf{x}}} = -\mathbf{H}^T\mathbf{W}^{-1}\left(\mathbf{z} - \mathbf{H}\hat{\mathbf{x}}\right) = \mathbf{0} \tag{37a}$$

- rearranging we have

$$\left(\mathbf{H}^T\mathbf{W}^{-1}\mathbf{H}\right)\hat{\mathbf{x}} = \mathbf{H}^T\mathbf{W}^{-1}\mathbf{z} \tag{37b}$$

  which is the identical linear system we arrived at in the full Bayesian approach

What gives?

- why do the full Bayesian and MAP approaches result in the same linear system if they are trying to do different things?

$$\left(\mathbf{H}^T\mathbf{W}^{-1}\mathbf{H}\right)\hat{\mathbf{x}} = \mathbf{H}^T\mathbf{W}^{-1}\mathbf{z} \tag{38}$$

- the explanation lies in the fact that the full Bayesian posterior is exactly Gaussian and the mean and mode (i.e., maximum) of a Gaussian are one and the same
- when we later turn to nonlinear systems, these two approaches will diverge, because the posterior will not be exactly Gaussian

Sparsity

- both our approaches, full Bayesian and MAP, require us to compute $\mathbf{A}^{-1}$
- it turns out this has a beautifully simple form:

$$\mathbf{A}^{-1} = \begin{bmatrix} \mathbf{1} & & & & & \\ -\mathbf{A}_0 & \mathbf{1} & & & & \\ & -\mathbf{A}_1 & \mathbf{1} & & & \\ & & -\mathbf{A}_2 & \ddots & & \\ & & & \ddots & \mathbf{1} & \\ & & & & -\mathbf{A}_{K-1} & \mathbf{1} \end{bmatrix} \tag{39}$$

- the sparsity comes from the fact that the system model obeys the Markov property

Sparsity

- when we plug $\mathbf{A}^{-1}$ into the linear system (via $\mathbf{H}$)

$$\left(\mathbf{H}^T\mathbf{W}^{-1}\mathbf{H}\right)\hat{\mathbf{x}} = \mathbf{H}^T\mathbf{W}^{-1}\mathbf{z} \tag{40}$$

  the left-hand side looks like this:

$$\mathbf{H}^T\mathbf{W}^{-1}\mathbf{H} = \begin{bmatrix} * & * & & & & \\ * & * & * & & & \\ & * & * & * & & \\ & & \ddots & \ddots & \ddots & \\ & & & * & * & * \\ & & & & * & * \end{bmatrix} \tag{41}$$

  where $*$ indicates a non-zero block

- the fact that the LHS is exactly block-tridiagonal means we can solve for $\hat{\mathbf{x}}$ in $O(K)$ time instead of the usual $O(K^3)$ time for solving linear systems

Sparsity

– it turns out we can efficiently factor $\mathbf{H}^T\mathbf{W}^{-1}\mathbf{H}$ using a sparse Cholesky factorization into

$$\mathbf{H}^T\mathbf{W}^{-1}\mathbf{H} = \mathbf{L}\mathbf{L}^T \tag{42}$$

where $\mathbf{L}$ is a block-lower-triangular matrix called the Cholesky factor:

$$\mathbf{L} = \begin{bmatrix} \mathbf{L}_0 & & & & & \\ \mathbf{L}_{10} & \mathbf{L}_1 & & & & \\ & \mathbf{L}_{21} & \mathbf{L}_2 & & & \\ & & \ddots & \ddots & & \\ & & & \mathbf{L}_{K-1,K-2} & \mathbf{L}_{K-1} & \\ & & & & \mathbf{L}_{K,K-1} & \mathbf{L}_K \end{bmatrix} \tag{43}$$

– the decomposition can be computed in $O(N^3(K+1))$ time

Cholesky solution

- once we have the Cholesky factorization, the solution of our linear system is straightforward
- forward pass: solve
$$\mathbf{L}\mathbf{d} = \mathbf{H}^T\mathbf{W}^{-1}\mathbf{z} \tag{44}$$
for $\mathbf{d}$
- backward pass: solve
$$\mathbf{L}^T\hat{\mathbf{x}} = \mathbf{d} \tag{45}$$
for $\hat{\mathbf{x}}$
- both passes can be done in $O(N^3(K+1))$ time through forward/backward substitution owing to the sparse lower-triangular form of $\mathbf{L}$
- thus, the batch equations can be solved in computation time that scales linearly with the size of the state

Cholesky solution

- at the block level, the forwards pass, $k = 1 \ldots K$, is

$$\begin{aligned}
\mathbf{L}_{k-1}\mathbf{L}_{k-1}^T &= \mathbf{I}_{k-1} + \mathbf{A}_{k-1}^T\mathbf{Q}_k^{-1}\mathbf{A}_{k-1} && \text{(46a)}\\
\mathbf{L}_{k-1}\mathbf{d}_{k-1} &= \mathbf{q}_{k-1} - \mathbf{A}_{k-1}^T\mathbf{Q}_k^{-1}\mathbf{v}_k && \text{(46b)}\\
\mathbf{L}_{k,k-1}\mathbf{L}_{k-1}^T &= -\mathbf{Q}_k^{-1}\mathbf{A}_{k-1} && \text{(46c)}\\
\mathbf{I}_k &= -\mathbf{L}_{k,k-1}\mathbf{L}_{k,k-1}^T + \mathbf{Q}_k^{-1} + \mathbf{C}_k^T\mathbf{R}_k^{-1}\mathbf{C}_k && \text{(46d)}\\
\mathbf{q}_k &= -\mathbf{L}_{k,k-1}\mathbf{d}_{k-1} + \mathbf{Q}_k^{-1}\mathbf{v}_k + \mathbf{C}_k^T\mathbf{R}_k^{-1}\mathbf{y}_k && \text{(46e)}
\end{aligned}$$

- the backwards pass, $k = K \ldots 1$, is

$$\mathbf{L}_{k-1}^T\hat{\mathbf{x}}_{k-1} = -\mathbf{L}_{k,k-1}^T\hat{\mathbf{x}}_k + \mathbf{d}_{k-1} \qquad \text{(47)}$$

- these are initialized with

$$\begin{aligned}
\mathbf{I}_0 &= \check{\mathbf{P}}_0^{-1} + \mathbf{C}_0^T\mathbf{R}_0^{-1}\mathbf{C}_0 && \text{(48a)}\\
\mathbf{q}_0 &= \check{\mathbf{P}}_0^{-1}\check{\mathbf{x}}_0 + \mathbf{C}_0^T\mathbf{R}_0^{-1}\mathbf{y}_0 && \text{(48b)}\\
\hat{\mathbf{x}}_K &= \mathbf{I}_K^{-1}\mathbf{q}_K && \text{(48c)}
\end{aligned}$$

Cholesky solution $\mapsto$ Rauch-Tung-Striebel smoother

- our Cholesky approach will certainly work, and is appealing because it follows easily from the batch setup and the block-tridiagonal sparsity of the LHS
- historically, however, the Rauch-Tung-Striebel (RTS) smoother equations constitute the canonical form
- the forwards pass of the RTS smoother is called the Kalman filter, which we will discuss in more detail later
- the Cholesky and RTS equations are algebraically equivalent, but there is quite a lot of algebra needed to show this (see the book)

RTS smoother

- forwards pass (a.k.a., Kalman filter), $k = 1 \ldots K$:

$$\check{\mathbf{P}}_{k,f} = \mathbf{A}_{k-1}\hat{\mathbf{P}}_{k-1,f}\mathbf{A}_{k-1}^T + \mathbf{Q}_k \tag{49a}$$

$$\check{\mathbf{x}}_{k,f} = \mathbf{A}_{k-1}\hat{\mathbf{x}}_{k-1,f} + \mathbf{v}_k \tag{49b}$$

$$\mathbf{K}_k = \check{\mathbf{P}}_{k,f}\mathbf{C}_k^T\left(\mathbf{C}_k\check{\mathbf{P}}_{k,f}\mathbf{C}_k^T + \mathbf{R}_k\right)^{-1} \tag{49c}$$

$$\hat{\mathbf{P}}_{k,f} = \left(\mathbf{1} - \mathbf{K}_k\mathbf{C}_k\right)\check{\mathbf{P}}_{k,f} \tag{49d}$$

$$\hat{\mathbf{x}}_{k,f} = \check{\mathbf{x}}_{k,f} + \mathbf{K}_k\left(\mathbf{y}_k - \mathbf{C}_k\check{\mathbf{x}}_{k,f}\right) \tag{49e}$$

- backwards pass, $k = K \ldots 1$:

$$\hat{\mathbf{x}}_{k-1} = \hat{\mathbf{x}}_{k-1,f} + \left(\hat{\mathbf{P}}_{k-1,f}\mathbf{A}_{k-1}^T\check{\mathbf{P}}_{k,f}^{-1}\right)\left(\hat{\mathbf{x}}_k - \check{\mathbf{x}}_{k,f}\right) \tag{50a}$$

$$\hat{\mathbf{P}}_{k-1} = \hat{\mathbf{P}}_{k-1,f} + \left(\hat{\mathbf{P}}_{k-1,f}\mathbf{A}_{k-1}^T\check{\mathbf{P}}_{k,f}^{-1}\right)\left(\hat{\mathbf{P}}_k - \check{\mathbf{P}}_{k,f}\right) \times \left(\hat{\mathbf{P}}_{k-1,f}\mathbf{A}_{k-1}^T\check{\mathbf{P}}_{k,f}^{-1}\right)^T \tag{50b}$$

- these are initialized with

$$\hat{\mathbf{P}}_{0,f} = (\mathbf{1} - \mathbf{K}_0\mathbf{C}_0)\check{\mathbf{P}}_0, \quad \hat{\mathbf{x}}_{0,f} = \check{\mathbf{x}}_0 + \mathbf{K}_0(\mathbf{y}_0 - \mathbf{C}_0\check{\mathbf{x}}_0) \tag{51a}$$

$$\hat{\mathbf{P}}_K = \hat{\mathbf{P}}_{K,f}, \quad \hat{\mathbf{x}}_K = \hat{\mathbf{x}}_{K,f} \tag{51b}$$

UNIVERSITY OF TORONTO

Take-home message

The full Bayesian and MAP approaches result in the same system of linear equations because the Bayesian posterior is exactly Gaussian and the mean and mode (i.e., maximum) of a Gaussian are one and the same.

$$\left(\mathbf{H}^T \mathbf{W}^{-1} \mathbf{H}\right) \hat{\mathbf{x}} = \mathbf{H}^T \mathbf{W}^{-1} \mathbf{z} \qquad (52)$$

Either the Cholesky or RTS smoother equations can be used to solve this system of equations exactly and efficiently.

Existence and uniqueness

- how do we know there will be a solution to our batch equations?

$$\left(\mathbf{H}^T\mathbf{W}^{-1}\mathbf{H}\right)\hat{\mathbf{x}} = \mathbf{H}^T\mathbf{W}^{-1}\mathbf{z} \tag{53}$$

- in general, linear systems have 0, 1, or infinitely many solutions
- for a unique solution the LHS might be invertible or equivalently

$$\text{rank}\left(\mathbf{H}^T\mathbf{W}^{-1}\mathbf{H}\right) = N(K+1) \tag{54}$$

  because we have dim $\mathbf{x} = N(K+1)$

- we can drop $\mathbf{W}^{-1}$ as long as it's invertible – it is, it's positive definite by assumption – the test becomes

$$\text{rank}\left(\mathbf{H}^T\right) = N(K+1) \tag{55}$$

Case (i): we have knowledge of the initial state

- we see that

$$
\begin{aligned}
&\text{rank } \mathbf{H}^T \\
&= \text{rank} \left[\begin{array}{ccccc|ccccc}
\mathbf{1} & -\mathbf{A}_0^T & & & & \mathbf{C}_0^T & & & & \\
 & \mathbf{1} & -\mathbf{A}_1^T & & & & \mathbf{C}_1^T & & & \\
 & & \mathbf{1} & \ddots & & & & \mathbf{C}_2^T & & \\
 & & & \ddots & -\mathbf{A}_{K-1}^T & & & & \ddots & \\
 & & & & \mathbf{1} & & & & & \mathbf{C}_K^T
\end{array} \right]
\end{aligned}
\tag{56}
$$

- is exactly in block-row-echelon form, meaning there are $N(K+1)$ 'leading ones'
- this means all the rows are linearly independent so the matrix is full rank and our solution is unique
- intuitively, the prior already provides a complete solution and the measurements nudge it

UNIVERSITY OF TORONTO

Case (ii): no knowledge of the initial state

- it might be that we have no idea about the initial state of the system
- this removes the first column in $\mathbf{H}^T$ and our condition becomes

$$
\begin{aligned}
&\text{rank } \mathbf{H}^T \\
&= \text{rank} \left[\begin{array}{cccc|ccccc}
-\mathbf{A}_0^T & & & & \mathbf{C}_0^T & & & & \\
\hline
\mathbf{1} & -\mathbf{A}_1^T & & & & \mathbf{C}_1^T & & & \\
 & \mathbf{1} & \ddots & & & & \mathbf{C}_2^T & & \\
 & & \ddots & -\mathbf{A}_{K-1}^T & & & & \ddots & \\
 & & & \mathbf{1} & & & & & \mathbf{C}_K^T
\end{array} \right]
\end{aligned}
\qquad (57)
$$

- it's no longer obvious that the solution is unique

UNIVERSITY OF TORONTO

Case (ii): no knowledge of the initial state

- moving the top block-row to the bottom does not change the rank

$$
\begin{aligned}
&\text{rank } \mathbf{H}^T \\
&= \text{rank} \left[\begin{array}{cccc|cccc}
\mathbf{1} & -\mathbf{A}_1^T & & & \mathbf{C}_1^T & & & \\
 & \mathbf{1} & \ddots & & & \mathbf{C}_2^T & & \\
 & & \ddots & -\mathbf{A}_{K-1}^T & & & \ddots & \\
 & & & \mathbf{1} & & & & \mathbf{C}_K^T \\
\hline
-\mathbf{A}_0^T & & & & \mathbf{C}_0^T & & &
\end{array}\right] \qquad (58)
\end{aligned}
$$

- again without changing the rank, we can add to the bottom block-row, $\mathbf{A}_0^T$ times the first block-row, $\mathbf{A}_0^T \mathbf{A}_1^T$ times the second block-row, . . . , and $\mathbf{A}_0^T \cdots \mathbf{A}_{K-1}^T$ times the Kth block-row

UNIVERSITY OF TORONTO

Case (ii): no knowledge of the initial state

- the result of this last step is

$$
\begin{aligned}
&\text{rank } \mathbf{H}^T \\
&= \text{rank} \left[\begin{array}{cccc|ccccc}
\mathbf{1} & -\mathbf{A}_1^T & & & & \mathbf{C}_1^T & & & \\
 & \mathbf{1} & \ddots & & & & \mathbf{C}_1^T & & \\
 & & \ddots & -\mathbf{A}_{K-1}^T & & & & \ddots & \\
 & & & \mathbf{1} & & & & & \mathbf{C}_K^T \\
\hline
 & & & & \mathbf{C}_0^T & \mathbf{A}_0^T \mathbf{C}_1^T & \mathbf{A}_0^T \mathbf{A}_1^T \mathbf{C}_2^T & \cdots & \mathbf{A}_0^T \cdots \mathbf{A}_{K-1}^T \mathbf{C}_K^T
\end{array} \right]
\end{aligned}
\tag{59}
$$

- the upper-left block is full rank, NK, since every row has a ‘leading one’
- all that remains is to determine if the bottom-right block is full rank:

$$
\text{rank } \begin{bmatrix} \mathbf{C}_0^T & \mathbf{A}_0^T \mathbf{C}_1^T & \mathbf{A}_0^T \mathbf{A}_1^T \mathbf{C}_2^T & \cdots & \mathbf{A}_0^T \cdots \mathbf{A}_{K-1}^T \mathbf{C}_K^T \end{bmatrix} = N
\tag{60}
$$

UNIVERSITY OF TORONTO

Case (ii): no knowledge of the initial state – time-invariant

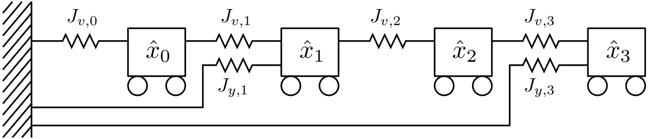
- if we further assume the system is time-invariant such that for all k we have $\mathbf{A}_k = \boldsymbol{A}$ and $\mathbf{C}_k = \boldsymbol{C}$ and we make the not-too-restrictive assumption that $K \gg N$, we may further simplify this condition
- the Cayley-Hamilton theoreom says square matrices satisfy their own characteristic equation, so powers of $\mathbf{A}$ greater than or equal to N can be written as linear combinations of the lower powers
- this let's us keep only the first N block-rows, reducing the test to

$$\operatorname{rank} \begin{bmatrix} \boldsymbol{C} \\ \boldsymbol{C}\boldsymbol{A} \\ \vdots \\ \boldsymbol{C}\boldsymbol{A}^{(N-1)} \end{bmatrix} = N \qquad (61)$$

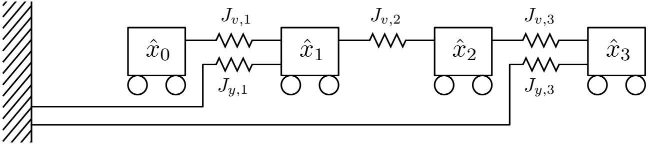
- this is precisely the test for observability – that we can reconstruct the initial state from a finite number of measurements

Mass-spring analogy

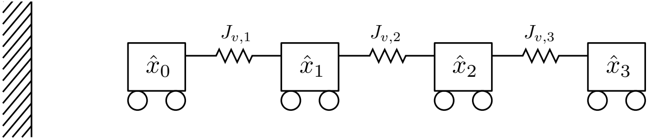
initial state knowledge plus inputs ensures observability



no initial state knowledge can be observable



no initial state knowledge can also be unobservable (in 1D this only happens with no measurements)



Knowing when we don't know

- we said at the beginning of the lecture that we'd like to bookkeep all of the uncertainties in our estimate, but the linear system just tells us the mean, right?
- wrong, we can interpret it in the following way:

$$\underbrace{\left(\mathbf{H}^T\mathbf{W}^{-1}\mathbf{H}\right)}_{\text{inverse covariance}}\underbrace{\hat{\mathbf{x}}}_{\text{mean}} = \underbrace{\mathbf{H}^T\mathbf{W}^{-1}\mathbf{z}}_{\text{information vector}} \qquad (62)$$

- so the covariance of our estimate is

$$\hat{\mathbf{P}} = \left(\mathbf{H}^T\mathbf{W}^{-1}\mathbf{H}\right)^{-1} \qquad (63)$$

- our (Bayesian) estimate is

$$p(\mathbf{x}|\mathbf{z}) = \mathcal{N}\left(\hat{\mathbf{x}}, \hat{\mathbf{P}}\right) \qquad (64)$$

UNIVERSITY OF TORONTO

Knowing how to know when we don't know

- to see this, we can directly take the expectation of the estimate:

$$\mathbf{x} - \underbrace{\left(\mathbf{H}^T\mathbf{W}^{-1}\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{W}^{-1}\mathbf{z}}_{E[\mathbf{x}]} = \left(\mathbf{H}^T\mathbf{W}^{-1}\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{W}^{-1}\underbrace{\left(\mathbf{H}\mathbf{x} - \mathbf{z}\right)}_{\mathbf{s}} \tag{65}$$

where $\mathbf{s} = \begin{bmatrix} \mathbf{w} \\ \mathbf{n} \end{bmatrix}$

- in this case we have

$$\begin{aligned} \hat{\mathbf{P}} &= E\left[\left(\mathbf{x} - E[\mathbf{x}]\right)\left(\mathbf{x} - E[\mathbf{x}]\right)^T\right] && \text{(66a)} \\ &= \left(\mathbf{H}^T\mathbf{W}^{-1}\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{W}^{-1}\underbrace{E\left[\mathbf{s}\,\mathbf{s}^T\right]}_{\mathbf{W}}\mathbf{W}^{-1}\mathbf{H}\left(\mathbf{H}^T\mathbf{W}^{-1}\mathbf{H}\right)^{-1} && \text{(66b)} \\ &= \left(\mathbf{H}^T\mathbf{W}^{-1}\mathbf{H}\right)^{-1} && \text{(66c)} \end{aligned}$$

Summary

- we have looked at how to estimate a whole trajectory using initial state knowledge, a sequence of inputs, and a sequence of measurements for linear systems corrupted by Gaussian noise
- our Bayesian method involves solving a linear system of equations for the mean and also building the covariance
- we know how to solve this linear system very efficiently by exploiting the sparsity of the matrices
- we have a test for when the solution is unique
- the approach can only be used offline since it is acausal (it uses future data to estimate past states)
- next, we'll look at recursive techniques for use in online situations – we've seen a preview of this in the forwards pass of the RTS smoother equations

UNIVERSITY OF TORONTO