Sensor selection for Kalman filtering of linear dynamical systems: Complexity, limitations and greedy algorithms

Written by: Zhang, Ayoub, Sundaram

Presented by: Bradley Hedges

University of Waterloo

April 3, 2019

Overview

Introduction

- A current problem in the design of control systems is to decide where to place sensors to achieve some sort of objective such as maintaining robustness, or optimizing certain system parameters.

- This paper looked at sensor selection to minimize the Error covariance when applying a Kalman Filter to a system (henceforth referred to as KFSS). This is the typical metric for determining how 'good' a Kalman Filter is.

Problem Formulation

We begin with the typical Dynamical system:

$$x_{k+1} = Ax_k + w_k$$

$$y_k = Cx_k + v_k$$

Where,

- $x_k \in \mathbb{R}^n$ is the state,
- $w_k \in \mathbb{R}^n$ is a zero-mean white noise with Covariance W
- A is such that $(A, W^{\frac{1}{2}})$ is stabilizable

The only difference is in how we view the second equation. Here,

- $C \in \mathbb{R}^{pxn}$ is such that each row i of C is the measurement made by sensor i
- v_k is zero-mean white noise with Covariance V

As always, we require that (A, C) is detectable

Problem Formulation

We will now introduce the problem of sensor placement.

Suppose we have a set of sensors Q (with length q), where each sensor $i \in Q$ has an associated cost b_i, and we have a sensor budget of B. That is, B is the maximum total cost that can be spent on sensors.

For the sake of simplicity in my implementation, I have assumed all sensors cost the same, but it is easy to extend it to a variable cost.

Definition: Selection Matrix

We let $z_i = 1$ iff sensor i has been installed in our system, then

$$Z = diag(z_1, ..., z_q)$$

is the Selection Matrix.

Problem Formulation

Now that we have our Z, we write:
$\widetilde{C} = ZC$ and $\widetilde{V} = ZVZ^T$
We define a *priori error covariance* $\Sigma_{k|k-1}(z)$. For constant A and C, this $\Sigma_{k|k-1}(z)$ converges to a matrix called $\Sigma(z)$, that is the solution to:

$$\Sigma(z) = A\Sigma(z)A^T + W - A\Sigma(z)\widetilde{C}^T(\widetilde{C}\Sigma(z)\widetilde{C}^T + \widetilde{V})^{-1}\widetilde{C}\Sigma(z)A^T$$

This is the *discrete algebraic Riccati equation (DARE)* that we've seen many times.
We can also define a *posteriori error covariance* $\Sigma_{k|k}(z)$, and solve a similar equation to the DARE with slightly different conditions. For the sake of time, we will only look at the priori problem.
Note: There is a pseudo-invertible transformation between the priori and posteriori solutions

Problem Formulation

We finally have enough information to pose our *Priori KFSS Problem*

Problem: Priori KFSS

Given a dynamics matrix $A \in \mathbb{R}^{nxn}$, a measurement matrix $C \in \mathbb{R}^{sxn}$, system noise covariance W and sensor noise covariance V, cost b and budget B, the priori KFSS is to find a sensor selection z that solves:

$$\min_{z} trace(\Sigma(z))$$

$$\text{s.t. } b^T z \leq B$$

$$z \in \{0, 1\}^q$$

Where $\Sigma(z)$ solves the DARE from before.

Theorem

Theorem

The priori KFSS problem and the posteriori KFSS problem are NP-hard even under the assumption that A is stable

As such, we do not have an obvious algorithm to find the optimal solution to our priori KFSS problem. The paper instead chooses to use a greedy algorithm.

A Priori Covariance based Greedy Algorithm

Algorithm 1: Priori Covariance based Greedy Algorithm

Input : Matrix A, the set of all sensors Q, noise covariance W and V, desired number of sensors p

Output: A set S of chosen sensors

$k \leftarrow 0,\ S \leftarrow \emptyset$

for $k \leq p$ **do**

  for $i \in Q \cap \overline{S}$ **do**

    Calculate $trace(\Sigma_{i,S}) \triangleq trace(\Sigma(S \cup \{i\}))$

  end

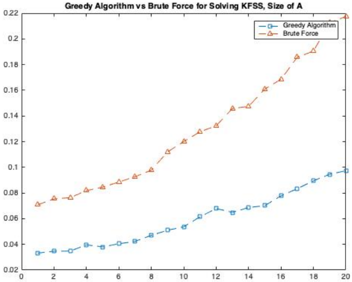
  Choose j with $trace(\Sigma_{j,S}) = min_i trace(\Sigma_{i,S})$

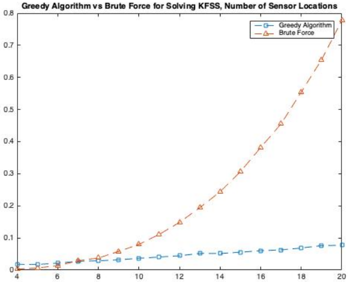
  $S \leftarrow S \cup \{j\},\ Q \leftarrow Q/\{j\},\ k \leftarrow k+1$

end

Analysis of Algorithm

One might wonder why it is better to use this Greedy Algorithm instead of finding the actual optimal sensor arrangement through a brute force approach.

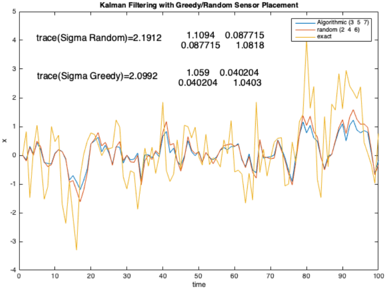


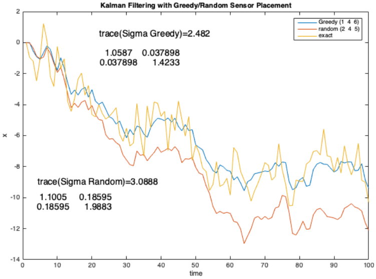


These plots show us that the runtimes for a brute-force approach grow so much quicker than the greedy algorithm that it would not be possible to use brute-force for large systems with many sensors.

Implementation of Kalman Filter

The final aspect to the sensor placement problem that we will address is the implementation of the Kalman Filter with different sensor placements.





Our metric for determining the accuracy of a Kalman Filter is the trace of our Error Covariance matrix. Thus, the matrices on this slide do not give that much information. Rather, they just highlight the fact that different sensor selections give different estimations of the system.

References

Zhang, H., Ayoub, R., Sundaram, S. (2017)
Sensor selection for Kalman filtering of linear dynamical systems: Complexity, limitations and greedy algorithms
Automatica 78(2017), 202-210.