ROB 501 Handouts
RLS, BLUE and MVE
J.W. Grizzle

Recursive Least Squares, Best Linear Unbiased Estimator, and Minimum Variance Estimators

Sources:

- Recursive Least Squares
  `http://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-241j-dynamic`
  `readings/MIT6_241JS11_chap02.pdf`

- Simplified Presentation of Various Estimators
  `http://home.engineering.iastate.edu/~namrata/EE527_Spring08/l3.pdf`
  (See also) `http://home.engineering.iastate.edu/~namrata/EE527_Spring12/`

- Higher Level Summary of Various Estimators. Heavily Based on the Projection Theorem
  `http://www4.ncsu.edu/~mtchu/Teaching/Lectures/MA719/chapter4.pdf`

Recursive Least Squares (See Section 2.6).

Notational changes we make in lecture with respect to the handout:

$$\bar{S} \rightarrow R$$

$$\bar{y} \rightarrow Y$$

$$\bar{A} \rightarrow A$$

$$A_i \rightarrow C_i$$

Lectures on Dynamic Systems and Control

Mohammed Dahleh   Munther A. Dahleh   George Verghese

Department of Electrical Engineering and Computer Science

Massachuasetts Institute of Technology[1]

Chapter 2

Least Squares Estimation

2.1   Introduction

If the criterion used to measure the error $e = y - Ax$ in the case of inconsistent system of equations is the sum of squared magnitudes of the error components, i.e. $e'e$, or equivalently the square root of this, which is the usual Euclidean norm or 2-norm $\|e\|_2$, then the problem is called a *least squares* problem. Formally it can be written as

$$\min_x \|y - Ax\|_2. \tag{2.1}$$

The x that minimizes this criterion is called the least square error estimate, or more simply, the *least squares estimate*. The choice of this criterion and the solution of the problem go back to Legendre (1805) and Gauss (around the same time).

Example 2.1   Suppose we make some measurements y_i of an unknown function $f(t)$ at discrete points t_i, $i = 1, \ldots, N$:

$$y_i = f(t_i), \quad i = 1, \ldots, N.$$

We want to find the function $g(t)$ in the space χ of polynomials of order $m - 1 < N - 1$ that best approximates $f(t)$ at the measured points t_i, where

$$\chi = \left\{ g(t) = \sum_{i=0}^{m-1} \alpha_i t^i, \ \alpha_i \ \text{real} \right\}$$

For any $g(t) \in \chi$, we will have $y_i = g(t_i) + e_i$ for $i = 1, \ldots, N$. Writing this in

matrix form for the available data, we have

$$
\underbrace{\begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix}}_{y} = \underbrace{\begin{bmatrix} 1 & t_1 & t_1^2 & \cdots & t_1^{m-1} \\ \vdots & & & & \vdots \\ 1 & t_N & t_N^2 & \cdots & t_N^{m-1} \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} \alpha_0 \\ \vdots \\ \alpha_{m-1} \end{bmatrix}}_{x} + \underbrace{\begin{bmatrix} e_1 \\ \vdots \\ e_N \end{bmatrix}}_{e}
$$

The problem is to find $\alpha_0, \ldots, \alpha_{m-1}$ such that $e'e = \sum_{i-1}^{N} e_i^2$ is minimized.

2.2   Computing the Estimate

The solution, $\hat{x}$, of Equation 2.1 is characterized by:

$$(y - A\hat{x}) \perp \mathcal{R}(A).$$

All elements in a basis of $\mathcal{R}(A)$ must be orthogonal to $(y - A\hat{x})$. Equivalently this is true for the set of columns of A, $[a_1, \ldots, a_n]$. Thus

$$(y - A\hat{x}) \perp \mathcal{R}(A) \quad \Leftrightarrow a_i'(y - A\hat{x}) = 0 \qquad \text{for } i = 1, \ldots, n$$

$$\Leftrightarrow A'(y - A\hat{x}) = 0$$

$$\Leftrightarrow A'A\hat{x} = A'y$$

This system of m equations in the m unknowns of interest is referred to as the **normal equations**. We can solve for the unique $\hat{x}$ iff $A'A$ is invertible. Conditions for this will be derived shortly. In the sequel, we will present the generalization of the above ideas for infinite dimensional vector spaces.

2.3   Preliminary: The Gram Product

Given the array of n_A vectors $A = [a_1 \mid \cdots \mid a_{n_A}]$ and the array of n_B vectors $B = [b_1 \mid \cdots \mid b_{n_B}]$ from a given inner product space, let $\prec A, B \succ$ denote the $n_A \times n_B$ *matrix* whose (i, j)-th element is $< a_i, b_j >$. We shall refer to this object as the *Gram product* (but note that this terminology is not standard!).

If the vector space under consideration is $\mathbf{R}^m$ or $\mathbf{C}^m$, then both A and B are matrices with m rows, but our definition of $\prec A, B \succ$ can actually handle more general A, B. In fact, the vector space can be infinite dimensional, as long as we are only examining finite collections of vectors from this space. For instance, we could use the same notation to treat finite collections of vectors chosen from the infinite-dimensional vector space $\mathcal{L}^2$ of *square*

integrable functions, i.e. functions $a(t)$ for which $\int_{-\infty}^{\infty} a^2(t)\, dt < \infty$. The inner product in $\mathcal{L}^2$ is $< a(t), b(t) > = \int_{-\infty}^{\infty} a^*(t)b(t)\, dt$. (The space $\mathcal{L}^2$ is an example of an infinite dimensional *Hilbert space*, and most of what we know for finite dimensional spaces — which are also Hilbert spaces! — has natural generalizations to infinite dimensional Hilbert spaces. Many of these generalizations involve introducing notions of topology and measure, so we shall *not* venture too far there. It is worth also mentioning here another important infinite dimensional Hilbert space that is central to the *probabilistic* treatment of least squares estimation: the space of zero-mean *random variables*, with the expected value $E(ab)$ serving as the inner product $< a, b >$.)

For the usual Euclidean inner product in an m-dimensional space, where $< a_i, b_j > = a_i' b_j$, we simply have $\prec A, B \succ = A'B$. For the inner product defined by $< a_i, b_j > = a_i' S b_j$ for a positive definite, Hermitian matrix S, we have $\prec A, B \succ = A'SB$.

- Verify that the symmetry and linearity of the inner product imply the same for the Gram product, so $\prec AF, BG + CH \succ = F' \prec A, B \succ G + F' \prec A, C \succ H$, for any constant matrices F, G, H (a constant matrix is a matrix of scalars), with A, B, C denoting arrays whose columns are vectors.

2.4   The Least Squares Estimation Problem

The problem of interest is to find the least square error (LSE) estimate of the parameter vector x that arises in the linear model $y \approx Ax$, where A is an array of n vectors, $A = [a_1, \cdots, a_n]$. Defining the *error* e by

$$e = y - Ax$$

what we want to determine is

$$\widehat{x} = \arg\min_x \|e\| = \arg\min_x \|y - Ax\|, \qquad y,\ A \text{ given}$$

(where "$\arg\min_x$" should be read as "the value of the argument x that minimizes"). To state this yet another way, note that as x is varied, Ax ranges over the subspace $\mathcal{R}(A)$, so we are looking for the point

$$\widehat{y} = A\widehat{x}$$

in $\mathcal{R}(A)$ that comes closest to y, as measured by whatever norm we are using.

Rather than restricting the norm in the above expression to be the Euclidean 2-norm used in Lecture 1, we shall now actually permit it to be any norm induced by an inner product, so $\|e\| = \sqrt{< e, e >}$. This will allow us to solve the so-called *weighted* least squares problem in a finite dimensional space with no additional work, because error criteria of the form $e'Se$ for positive definite Hermitian S are thereby included. Also, our problem formulation then applies to infinite dimensional spaces that have an inner product defined on them, with the restriction that our model Ax be confined to a finite dimensional subspace. This actually covers the cases of most interest to us; treatment of the more general case involves introducing further topological notions (*closed* subspaces, etc.), and we avoid doing this.

We shall also assume that the vectors a_i, $i = 1, \ldots, n$ in A are *independent*. This assumption is satisfied by any reasonably parametrized model, for otherwise there would be an infinite number of choices of x that attained any achievable value of the error $y - Ax$. If the vectors in A are discovered to be dependent, then a re-parametrization of the model is needed to yield a well-parametrized model with independent vectors in the new A. (A subtler problem — and one that we shall say something more about in the context of ill-conditioning and the singular value decomposition — is that the vectors in A can be *nearly* dependent, causing practical difficulties in numerical estimation of the parameters.)

Gram Matrix Lemma

An important route to verifying the independence of the vectors that make up the columns of A is a lemma that we shall refer to as the *Gram Matrix Lemma*. This states that the vectors in A are independent iff the associated Gram matrix (or *Gramian*) $\prec A, A \succ = [< a_i, a_j >]$ is invertible; all norms are equivalent, as far as this result is concerned — one can pick any norm. As noted above, for the case of the usual Euclidean inner product, $\prec A, A \succ = A'A$. For an inner product of the form $< a_i, a_j > = a_i' S a_j$, where S is Hermitian and positive definite, we have $\prec A, A \succ = A'SA$. The lemma applies to the infinite dimensional setting as well (e.g. $\mathcal{L}^2$), provided we are only considering the independence of a finite subset of vectors.

Proof: If the vectors in A are dependent, there is some nonzero vector η such that $A\eta = \sum_j a_j \eta_j = 0$. But then $\sum_j < a_i, a_j > \eta_j = 0$, by the linearity of the inner product; in matrix form, we can write $\prec A, A \succ \eta = 0$ — so $\prec A, A \succ$ is not invertible.

Conversely, if $\prec A, A \succ$ is not invertible, then $\prec A, A \succ \eta = 0$ for some nonzero η. But then $\eta' \prec A, A \succ \eta = 0$, so by the linearity of inner products $< \sum \eta_i a_i, \sum a_j \eta_j > = 0$, i.e. the norm of the vector $\sum a_j \eta_j = A\eta$ is zero, so the vectors in A are dependent.

2.5   The Projection Theorem and the Least Squares Estimate

The solution to our least squares problem is now given by the *Projection Theorem*, also referred to as the **Orthogonality Principle**, which states that

$$\widehat{e} = (y - A\widehat{x}) \quad \perp \quad \mathcal{R}(A)$$

from which - — as we shall see — $\widehat{x}$ can be determined. In words, the theorem/"principle" states that the point $\widehat{y} = A\widehat{x}$ in the subspace $\mathcal{R}(A)$ that comes closest to y is characterized by the fact that the associated error $\widehat{e} = y - \widehat{y}$ is orthogonal to $\mathcal{R}(A)$, i.e., orthogonal to the space spanned by the vectors in A. This principle was presented and proved in the previous chapter. We repeat the proof here in the context of the above problem.

Proof: We first show that y has a unique decomposition of the form $y = y_1 + y_2$, where $y_1 \in \mathcal{R}(A)$ and $y_2 \in \mathcal{R}^{\perp}(A)$. We can write any $y_1 \in \mathcal{R}(A)$ in the form $y_1 = A\alpha$ for some vector α.

If we want $(y - y_1) \in \mathcal{R}^\perp(A)$, we must see if there is an α that satisfies

$$< a_i, (y - A\alpha) > = 0 , \quad i = 1, \ldots, n$$

or, using our Gram product notation,

$$\prec A, (y - A\alpha) \succ = 0$$

Rearranging this equation and using the linearity of the Gram product, we get

$$\prec A, A \succ \alpha = \prec A, y \succ$$

which is in the form of the normal equations that we encountered in Lecture 1. Under our assumption that the vectors making up the columns of A are independent, the Gram matrix lemma shows that $\prec A, A \succ$ is invertible, so the unique solution of the preceding equation is

$$\alpha = \prec A, A \succ^{-1} \prec A, y \succ$$

We now have the decomposition that we sought.

To show that the preceding decomposition is unique, let $y = y_{1a} + y_{2a}$ be another such decomposition, with $y_{1a} \in \mathcal{R}(A)$ and $y_{2a} \in \mathcal{R}^\perp(A)$. Then

$$y_1 - y_{1a} = y_2 - y_{2a}$$

and the left side is in $\mathcal{R}(A)$ while the right side is in its orthogonal complement. It is easy to show that the only vector common to a subspace and its orthogonal complement is the zero vector, so $y_1 - y_{1a} = 0$ and $y_2 - y_{2a} = 0$, i.e., the decomposition of y is unique.

To proceed, decompose the error $e = y - Ax$ similarly (and uniquely) into the sum of $e_1 \in \mathcal{R}(A)$ and $e_2 \in \mathcal{R}^\perp(A)$. Note that

$$\|e\|^2 = \|e_1\|^2 + \|e_2\|^2$$

Now we can rewrite $e = y - Ax$ as

$$e_1 + e_2 = y_1 + y_2 - Ax$$

or

$$e_2 - y_2 = y_1 - e_1 - Ax$$

Since the right side of the above equation lies in $\mathcal{R}(A)$ and the left side lies in $\mathcal{R}^\perp(A)$, each side separately must equal 0 — again because this is the only vector common to a subspace and its orthogonal complement. We thus have $e_2 = y_2$, and the choice of x can do nothing to affect e_2. On the other hand, $e_1 = y_1 - Ax = A(\alpha - x)$, and the best we can do as far as minimizing $\|e\|^2$ is to make $e_1 = 0$ by choosing $x = \alpha$, so $\hat{x} = \alpha$, i.e.,

$$\widehat{x} = \prec A, A \succ^{-1} \prec A, y \succ$$

This solves the least squares estimation problem that we have posed.

The above result, though rather abstractly developed, is immediately applicable to many concrete cases of interest.

- Specializing to the case of $\mathbf{R}^m$ or $\mathbf{C}^m$, and choosing x to minimize the usual Euclidean norm,

$$\|e\|^2 = e'e = \sum_{i=1}^{m} |e_i|^2$$

  we have

$$\widehat{x} = (A'A)^{-1}A'y$$

  Note that if the columns of A form a mutually orthogonal set (i.e. an orthogonal basis for $\mathcal{R}(A)$), then $A'A$ is diagonal, and its inversion is trivial.

- If instead we choose to minimize $e'Se$ for some positive definite Hermitian S ($\neq I$), we have a **weighted least squares** problem, with solution given by

$$\widehat{x} = (A'SA)^{-1}A'Sy$$

  For instance, with a *diagonal S*, the criterion that we are trying to minimize becomes

$$\sum_{i=1}^{m} s_{ii}|e_i|^2$$

  where the s_{ii} are all positive. We can thereby preferentially weight those equations in our linear system for which we want a smaller error in the final solution; a larger value of s_{ii} will encourage a smaller e_i.

  Such weighting is important in any practical situation, where different measurements y_i may have been subjected to different levels of noise or uncertainty. One might expect that s_{ii} should be inversely proportional to the noise intensity on the ith equation. In fact, a probabilistic derivation, assuming zero-mean noise on each equation in the system but noise that is uncorrelated across equations, shows that s_{ii} should vary inversely with the *variance* of e_i.

  A full matrix S rather than a diagonal one would make sense if the errors were correlated across measurements. A probabilistic treatment shows that the proper weighting matrix is $S = (E[ee'])^{-1}$, the inverse of the *covariance matrix* of e. In the deterministic setting, one has far less guidance on picking a good S.

- The boxed result also allows us to immediately write down the choice of coefficients x_i that minimizes the integral

$$\int [\, y(t) - a_1(t)x_1 - a_2(t)x_2 - \cdots - a_n(t)x_n \,]^2 \, dt$$

for specified functions $y(t)$ and $a_i(t)$. If, for instance, $y(t)$ is of finite extent (or finite "support") T, and the $a_i(t)$ are sinusoids whose frequencies are integral multiples of $2\pi/T$, then the formulas that we obtain for the x_i are just the familiar Fourier series expressions. A simplification in this example is that the vectors in A are orthogonal, so $\prec A, A \succ$ is diagonal.

2.6   Recursive Least Squares (optional)

What if the data is coming in sequentially? Do we have to recompute everything each time a new data point comes in, or can we write our new, updated estimate in terms of our old estimate?

Consider the model

$$y_i = A_i x + e_i \ , \quad i = 0, 1, \ldots, \tag{2.2}$$

where $y_i \in \mathbf{C}^{m \times 1}$, $A_i \in \mathbf{C}^{m \times n}$, $x \in \mathbf{C}^{n \times 1}$, and $e_i \in \mathbf{C}^{m \times 1}$. The vector e_k represents the mismatch between the measurement y_k and the model for it, $A_k x$, where A_k is known and x is the vector of parameters to be estimated. At each time k, we wish to find

$$\widehat{x}_k = \arg \min_x \left(\sum_{i=1}^k (y_i - A_i x)'_i S_i (y_i - A_i x) \right) = \arg \min_x \left(\sum_{i=1}^k e'_i S_i e_i \right) \ , \tag{2.3}$$

where $S_i \in \mathbf{C}^{m \times m}$ is a positive definite Hermitian matrix of weights, so that we can vary the importance of the e_i's and components of the e_i's in determining $\widehat{x}_k$.

To compute $\widehat{x}_{k+1}$, let:

$$\overline{y}_{k+1} = \begin{bmatrix} y_0 \\ y_1 \\ . \\ . \\ y_{k+1} \end{bmatrix} ; \qquad \overline{A}_{k+1} = \begin{bmatrix} A_0 \\ A_1 \\ . \\ . \\ A_{k+1} \end{bmatrix} ; \qquad \overline{e}_{k+1} = \begin{bmatrix} e_0 \\ e_1 \\ . \\ . \\ e_{k+1} \end{bmatrix} ;$$

and

$$\overline{S}_{k+1} = \operatorname{diag} (S_0 \, , \, S_1 \, , \, \ldots \, , \, S_{k+1})$$

where S_i is the weighting matrix for e_i.

Our problem is then equivalent to

$$\min(\overline{e}'_{k+1}\overline{S}_{k+1}\overline{e}_{k+1})$$

$$\text{subject to:} \quad \overline{y}_{k+1} = \overline{A}_{k+1}x_{k+1} + \overline{e}_{k+1}$$

The solution can thus be written as

$$(\overline{A}'_{k+1}\overline{S}_{k+1}\overline{A}_{k+1})\widehat{x}_{k+1} = \overline{A}'_{k+1}\overline{S}_{k+1}\overline{y}_{k+1}$$

or in summation form as

$$\left(\sum_{i=0}^{k+1} A'_i S_i A_i\right) \widehat{x}_{k+1} = \sum_{i=0}^{k+1} A'_i S_i y_i$$

Defining

$$Q_{k+1} = \sum_{i=0}^{k+1} A'_i S_i A_i.$$

we can write a recursion for Q_{k+1} as follows:

$$Q_{k+1} = Q_k + A'_{k+1}S_{k+1}A_{k+1}.$$

Rearranging the summation form equation for $\widehat{x}_{k+1}$, we get

$$\widehat{x}_{k+1} = Q_{k+1}^{-1}\left[\left(\sum_{i=0}^{k} A'_i S_i A_i\right)\widehat{x}_k + A'_{k+1}S_{k+1}y_{k+1}\right]$$

$$= Q_{k+1}^{-1}\left[Q_k\widehat{x}_k + A'_{k+1}S_{k+1}y_{k+1}\right]$$

This clearly displays the new estimate as a weighted combination of the old estimate and the new data, so we have the desired recursion. Another useful form of this result is obtained by substituting from the recursion for Q_{k+1} above to get

$$\widehat{x}_{k+1} = \widehat{x}_k - Q_{k+1}^{-1}\left(A'_{k+1}S_{k+1}A_{k+1}\widehat{x}_k - A'_{k+1}S_{k+1}y_{k+1}\right) \quad,$$

which finally reduces to

$$\widehat{x}_{k+1} = \widehat{x}_k + \underbrace{Q_{k+1}^{-1}A'_{k+1}S_{k+1}}_{\text{Kalman Filter Gain}} \underbrace{(y_{k+1} - A_{k+1}\widehat{x}_k)}_{\text{innovations}}$$

The quantity $Q_{k+1}^{-1}A'_{k+1}S_{k+1}$ is called the *Kalman gain*, and $y_{k+1} - A_{k+1}\widehat{x}_k$ is called the *innovations*, since it compares the difference between a data update and the prediction given the last estimate.

Unfortunately, as one acquires more and more data, i.e. as k grows large, the Kalman gain goes to zero. One data point cannot make much headway against the mass of previous data which has 'hardened' the estimate. If we leave this estimator as is—without modification—the estimator 'goes to sleep' after a while, and thus doesn't adapt well to parameter changes. The homework investigates the concept of a 'fading memory' so that the estimator doesn't go to sleep.

An Implementation Issue

Another concept which is important in the implementation of the RLS algorithm is the computation of Q_{k+1}^{-1}. If the dimension of Q_k is very large, computation of its inverse can be computationally expensive, so one would like to have a recursion for Q_{k+1}^{-1}.

This recursion is easy to obtain. Applying the handy matrix identity

$$(A + BCD)^{-1} = A^{-1} - A^{-1}B\left(DA^{-1}B + C^{-1}\right)^{-1}DA^{-1}$$

to the recursion for Q_{k+1} yields

$$Q_{k+1}^{-1} = Q_k^{-1} - Q_k^{-1}A_{k+1}'\left(A_{k+1}Q_k^{-1}A_{k+1}' + S_{k+1}^{-1}\right)^{-1}A_{k+1}Q_k^{-1} \ .$$

Upon defining

$$P_{k+1} = Q_{k+1}^{-1} \ ,$$

this becomes

$$P_{k+1} = P_k - P_k A_{k+1}'\left(S_{k+1}^{-1} + A_{k+1}P_k A_{k+1}'\right)^{-1}A_{k+1}P_k \ .$$

which is called the (discrete-time) *Riccati equation*.

Interpretation

We have $\widehat{x}_k$ and y_{k+1} available for computing our updated estimate. Interpreting $\widehat{x}_k$ as a measurement, we see our model becomes

$$\begin{bmatrix} \widehat{x}_k \\ y_{k+1} \end{bmatrix} = \begin{bmatrix} I \\ A_{k+1} \end{bmatrix} x + \begin{bmatrix} e_k \\ e_{k+1} \end{bmatrix}.$$

The criterion, then, by which we choose $\widehat{x}_{k+1}$ is thus

$$\widehat{x}_{k+1} = \operatorname{argmin}\left(e_k' Q_k e_k + e_{k+1}' S_{k+1} e_{k+1}\right) \ .$$

In this context, one interprets Q_k as the weighting factor for the previous estimate.

Exercises

Exercise 2.1 Least Squares Fit of an Ellipse

Suppose a particular object is modeled as moving in an elliptical orbit centered at the origin. Its nominal trajectory is described in rectangular coordinates (r, s) by the constraint equation $x_1 r^2 + x_2 s^2 + x_3 rs = 1$, where x_1, x_2, and x_3 are unknown parameters that specify the orbit. We have available the following noisy measurements of the object's coordinates (r, s) at ten different points on its orbit:

(0.6728, 0.0589) (0.3380, 0.4093) (0.2510, 0.3559) (-0.0684, 0.5449)
(-0.4329, 0.3657) (-0.6921, 0.0252) (-0.3681, -0.2020) (0.0019, -0.3769)
(0.0825, -0.3508) (0.5294, -0.2918)

The ten measurements are believed to be equally reliable. For your convenience, these ten pairs of measured (r, s) values have been stored in column vectors named r and s that you can access through the 6.241 locker on Athena.[*] After `add 6.241`, and once in the directory in which you are running Matlab, you can copy the data using `cp /mit/6.241/Public/fall95/hw1rs.mat hw1rs.mat`. Then, in Matlab, type `load hw1rs` to load the desired data; type `who` to confirm that the vectors r and s are indeed available.

Using the assumed constraint equation, we can arrange the given information in the form of the linear system of (approximate) equations $Ax \approx b$, where A is a known 10×3 matrix, b is a known 10×1 vector, and $x = (x_1, x_2, x_3)^T$. This system of 10 equations in 3 unknowns is inconsistent. We wish to find the solution x that minimizes the Euclidean norm (or length) of the error $Ax - b$. Compare the solutions obtained by using the following four Matlab invocations, each of which in principle gives the desired least-square-error solution:

(a) $x = A \backslash b$

(b) $x = \text{pinv}(A) * b$

(c) $x = \text{inv}(A' * A) * A' * b$

(d) $[q, r] = qr(A)$, followed by implementation of the approach described in Exercise 3.1.

For more information on these commands, try `help slash`, `help qr`, `help pinv`, `help inv`, etc. [Incidentally, the prime, $'$, in Matlab takes the transpose of the *complex conjugate* of a matrix; if you want the ordinary transpose of a complex matrix C, you have to write $C.'$ or `transp(C)`.]

You should include in your solutions a plot the ellipse that corresponds to your estimate of x. If you create the following function file in your Matlab directory, with the name `ellipse.m`, you can obtain the polar coordinates `theta, rho` of n points on the ellipse specified by the parameter vector x. To do this, enter `[theta,rho]=ellipse(x,n);` at the Matlab prompt. You can then plot the ellipse by using the `polar(theta,rho)` command.

```
function [theta,rho]=ellipse(x,n)
% [theta,rho]=ellipse(x,n)
%
% The vector x = [x(1),x(2),x(3)]', defines an ellipse centered at the origin
% via the equation x(1)*r^ 2 + x(2)*s^ 2 +x(3)*r*s = 1.
% This routine generates the polar coordinates of points on the ellipse,
% to send to a plot command. It does this by solving for the radial
% distance in n equally spaced angular directions.
% Use polar(theta,rho) to actually plot the ellipse.
```

[*] Athena is MIT's UNIX-based computing environment. OCW does not provide access to it.

```
theta = 0:(2*pi/n):(2*pi);
a = x(1)*cos(theta).^ 2 + x(2)*sin(theta).^ 2 + x(3)*(cos(theta).*sin(theta));
rho = ones(size(a))./sqrt(a);
```

Exercise 2.2 Approximation by a Polynomial

Let $f(t) = 0.5e^{0.8t}$, $t \in [0, 2]$.

(a) Suppose 16 exact measurements of $f(t)$ are available to you, taken at the times t_i listed in the array T below:

$$T = \begin{array}{llllllll} [2 \cdot 10^{-3}, & 0.136, & 0.268, & 0.402, & 0.536, & 0.668, & 0.802, & 0.936, \\ 1.068, & 1.202, & 1.336, & 1.468, & 1.602, & 1.736, & 1.868, & 2.000] \end{array}$$

Use Matlab to generate these measurements:

$$y_i = f(t_i) \qquad i = 1, \ldots, 16 \qquad t_i \in T$$

Now determine the coefficients of the least square error polynomial approximation of the measurements, for

1. a polynomial of degree 15, $p_{15}(t)$;
2. a polynomial of degree 2, $p_2(t)$.

Compare the quality of the two approximations by plotting $y(t_i)$, $p_{15}(t_i)$ and $p_2(t_i)$ for all t_i in T. To see how well we are approximating the function on the whole interval, also plot $f(t)$, $p_{15}(t)$ and $p_2(t)$ on the interval $[0, 2]$. (Pick a very fine grid for the interval, e.g. t=[0:1000]'/500.) Report your observations and comments.

(b) Now suppose that your measurements are affected by some noise. Generate the measurements using

$$y_i = f(t_i) + e(t_i) \qquad i = 1, \ldots, 16 \qquad t_i \in T$$

where the vector of noise values can be generated in the following way:

$$randn('seed', 0);$$
$$e = randn(size(T));$$

Again determine the coefficients of the least square error polynomial approximation of the measurements for

1. a polynomial of degree 15, $p_{15}(t)$;
2. a polynomial of degree 2, $p_2(t)$.

Compare the two approximations as in part (a). Report your observations and comments. Explain any surprising results.

(**c**) So far we have obtained polynomial approximations of $f(t)$, $t \in [0,2]$, by approximating the measurements at $t_i \in T$. We are now interested in minimizing the square error of the polynomial approximation over the whole interval $[0,2]$:

$$\min \|f(t) - p_n(t)\|_2^2 = \min \int_0^2 |f(t) - p_n(t)|^2 \, dt$$

where $p_n(t)$ is some polynomial of degree n. Find the polynomial $p_2(t)$ of degree 2 that solves the above problem. Are the optimal $p_2(t)$ in this case and the optimal $p_2(t)$ of parts (a) and (b) very different from each other? Elaborate.

Exercise 2.3 Combining Estimates

Suppose $y_1 = C_1 x + e_1$ and $y_2 = C_2 x + e_2$, where x is an n-vector, and C_1, C_2 have full column rank. Let $\hat{x}_1$ denote the value of x that minimizes $e_1^T S_1 e_1$, and $\hat{x}_2$ denote the value that minimizes $e_2^T S_2 e_2$, where S_1 and S_2 are positive definite matrices. Show that the value $\hat{x}$ of x that minimizes $e_1^T S_1 e_1 + e_2^T S_2 e_2$ can be written entirely in terms of $\hat{x}_1$, $\hat{x}_2$, and the $n \times n$ matrices $Q_1 = C_1^T S_1 C_1$ and $Q_2 = C_2^T S_2 C_2$. What is the significance of this result?

Exercise 2.4 Exponentially Windowed Estimates

Suppose we observe the *scalar* measurements

$$y_i = c_i x + e_i \ , \qquad i = 1, 2, \ldots$$

where c_i and x are possibly vectors (row- and column-vectors respectively).

(**a**) Show (by reducing this to a problem that we already know how to solve — don't start from scratch!) that the value $\hat{x}_k$ of x that minimizes the criterion

$$\sum_{i=1}^k f^{k-i} e_i^2, \qquad \text{some fixed } f, \quad 0 < f \le 1$$

is given by

$$\hat{x}_k = \left(\sum_{i=1}^k f^{k-i} c_i^T c_i \right)^{-1} \left(\sum_{i=1}^k f^{k-i} c_i^T y_i \right)$$

The so-called *fade* or *forgetting* factor f allows us to preferentially weight the more recent measurements by picking $0 < f < 1$, so that old data is discounted at an exponential rate. We then say that the data has been subjected to exponential fading or forgetting or weighting or windowing or tapering or ... . This is usually desirable, in order to keep the filter adaptive to changes that may occur in x. Otherwise the filter becomes progressively less attentive to new data and falls asleep, with its gain approaching 0.

(b) Now show that

$$\hat{x}_k = \hat{x}_{k-1} + Q_k^{-1} c_k^T (y_k - c_k \hat{x}_{k-1})$$

where

$$Q_k = f Q_{k-1} + c_k^T c_k, \qquad Q_0 = 0$$

The vector $g_k = Q_k^{-1} c_k^T$ is termed the *gain* of the estimator.

(c) If x and c_i are scalars, and c_i is a constant c, determine g_k as a function of k. What is the *steady-state gain* g_∞? Does g_∞ increase or decrease as f increases — and why do you expect this?

Exercise 2.5 Suppose our model for some waveform $y(t)$ is $y(t) = \alpha \sin(\omega t)$, where α is a scalar, and suppose we have measurements $y(t_1), \ldots, y(t_p)$. Because of modeling errors and the presence of measurement noise, we will generally not find any choice of model parameters that allows us to precisely account for all p measurements.

(a) If ω is known, find the value of α that minimizes

$$\sum_{i=1}^{p} [y(t_i) - \alpha \sin(\omega t_i)]^2$$

(b) Determine this value of α if $\omega = 2$ and if the measured values of $y(t)$ are:

$$y(1) = +2.31 \quad y(2) = -2.01 \quad y(3) = -1.33 \quad y(4) = +3.23$$

$$y(5) = -1.28 \quad y(6) = -1.66 \quad y(7) = +3.28 \quad y(8) = -0.88$$

(I generated this data using the equation $y(t) = 3\sin(2t) + e(t)$ evaluated at the integer values $t = 1, \ldots, 8$, and with $e(t)$ for each t being a random number uniformly distributed in the interval - 0.5 to +0.5.)

(c) Suppose that α *and* ω are unknown, and that we wish to determine the values of these two variables that minimize the above criterion. Assume you are given initial estimates α_0 and ω_0 for the minimizing values of these variables. Using the Gauss-Newton algorithm for this nonlinear least squares problem, i.e. applying LLSE to the problem obtained by linearizing about the initial estimates, determine explicitly the estimates α_1 and ω_1 obtained after one iteration of this algorithm. Use the following notation to help you write out the solution in a condensed form:

$$a = \sum \sin^2(\omega_0 t_i) , \quad b = \sum t_i^2 \cos^2(\omega_0 t_i) , \quad c = \sum t_i [\sin(w_0 t_i)][\cos(w_0 t_i)]$$

(d) What values do you get for α_1 and ω_1 with the data given in (b) above if the initial guesses are $\alpha_0 = 3.2$ and $\omega_0 = 1.8$? Continue the iterative estimation a few more steps. Repeat the procedure when the initial guesses are $\alpha_0 = 3.5$ and $\omega_0 = 2.5$, verifying that the algorithm does not converge.

(e) Since only ω enters the model nonlinearly, we might think of a decomposed algorithm, in which α is estimated using *linear* least squares and ω is estimated via nonlinear least squares. Suppose, for example, that our initial estimate of ω is $\omega_0 = 1.8$. Now obtain an estimate α_1 of α using the linear least squares method that you used in (b). Then obtain an (improved?) estimate ω_1 of ω, using one iteration of a Gauss-Newton algorithm (similar to what is needed in (c), except that now you are only trying to estimate ω). Next obtain the estimate α_2 via linear least squares, and so on. Compare your results with what you obtain via this decomposed procedure when your initial estimate is $\omega_0 = 2.5$ instead of 1.8.

Exercise 2.6 Comparing Different Estimators

This problem asks you to compare the behavior of different parameter estimation algorithms by fitting a model of the type $y(t) = a\sin(2\pi t) + b\cos(4\pi t)$ to noisy data taken at values of t that are .02 apart in the interval (0,2].

First synthesize the data on which you will test the algorithms. Even though your estimation algorithms will assume that a and b are constant, we are interested in seeing how they track parameter changes as well. Accordingly, let $a = 2$, $b = 2$ for the first 50 points, and $a = 1$, $b = 3$ for the next 50 points. To get (approximately) normally distributed random variables, we use the function *randn* to produce variables with mean 0 and variance 1.

An elegant way to generate the data in Matlab, exploiting Matlab's facility with vectors, is to define the vectors $t1 = 0.02 : 0.02 : 1.0$ and $t2 = 1.02 : 0.02 : 2.0$, then set

$$y1 = 2 * \sin(2 * \text{pi} * t1) + 2 * \cos(4 * \text{pi} * t1) + s * randn(size(t1))$$

and

$$y2 = \sin(2 * \text{pi} * t2) + 3 * \cos(4 * \text{pi} * t2) + s * randn(size(t2))$$

where s determines the standard deviation of the noise. Pick $s = 1$ for this problem. Finally, set $y = [y1, y2]$. No loops, no counters, no fuss!!

Now estimate a and b from y using the following algorithms. Assume prior estimates $\hat{a}_0 = 3$ and $\hat{b}_0 = 1$, weighted equally with the measurements (so all weights can be taken as 1 without loss of generality). Plot your results to aid comparison.

(i) Recursive least squares.

(ii) Recursive least squares with exponentially fading memory, as in Problem 3. Use $f = .96$.

(iii) The algorithm in (ii), but with Q_k of Problem 3 replaced by $q_k = (1/n) \times \text{trace}(Q_k)$, where n is the number of parameters, so $n = 2$ in this case. (Recall that the trace of a matrix is the sum of its diagonal elements. Note that q_k itself satisfies a recursion, which you should write down.)

(iv) An algorithm of the form

$$\hat{x}_k = \hat{x}_{k-1} + \frac{.04}{c_k c_k^T} c_k^T \left(y_k - c_k \hat{x}_{k-1} \right)$$

where $c_k = [\sin(2\pi t), \cos(4\pi t)]$ evaluated at the kth sampling instant, so $t = .02k$.

Exercise 2.7 Recursive Estimation of a State Vector

This course will soon begin to consider *state-space models* of the form

$$x_\ell = A x_{\ell-1} \tag{2.4}$$

where x_ℓ is an n-vector denoting the state at time ℓ of our model of some system, and A is a known $n \times n$ matrix. For example, suppose the system of interest is a rotating machine, with angular position d_ℓ and angular velocity ω_ℓ at time $t = \ell T$, where T is some fixed sampling interval. If we believed the machine to be rotating at constant speed, we would be led to the model

$$\begin{pmatrix} d_\ell \\ \omega_\ell \end{pmatrix} = \begin{pmatrix} 1 & T \\ 0 & 1 \end{pmatrix} \begin{pmatrix} d_{\ell-1} \\ \omega_{\ell-1} \end{pmatrix}$$

Assume A to be nonsingular throughout this problem.

For the rotating machine example above, it is often of interest to obtain least-square-error estimates of the position and (constant) velocity, using noisy measurements of the angular position d_j at the sampling instants. More generally, it is of interest to obtain a least-square-error estimate of the state vector x_i in the model (2.4) from noisy p-component measurements y_j that are related to x_j by a linear equation of the form

$$y_j = C x_j + e_j \ , \qquad j = 1, \ldots, i$$

where C is a $p \times n$ matrix. We shall also assume that a prior estimate $\hat{x}_0$ of x_0 is available:

$$\hat{x}_0 = x_0 + e_0$$

Let $\hat{x}_{i|i}$ denote the value of x_i that minimizes

$$\sum_{j=0}^{i} \|e_j\|^2$$

This is the estimate of x_i given the prior estimate and measurements up to time i, or the "filtered estimate" of x_i. Similarly, let $\hat{x}_{i|i-1}$ denote the value of x_i that minimizes

$$\sum_{j=0}^{i-1} \|e_j\|^2$$

This is the least-square-error estimate of x_i given the prior estimate and measurements up to time $i-1$, and is termed the "one-step prediction" of x_i.

a) Set up the linear system of equations whose least square error solution would be $\hat{x}_{i|i}$. Similarly, set up the linear system of equations whose least square error solution would be $\hat{x}_{i|i-1}$.

b) Show that $\hat{x}_{i|i-1} = A\hat{x}_{i-1|i-1}$.

c) Determine a recursion that expresses $\hat{x}_{i|i}$ in terms of $\hat{x}_{i-1|i-1}$ and y_i. This is the prototype of what is known as the *Kalman filter*. A more elaborate version of the Kalman filter would include additive noise driving the state-space model, and other embellishments, all in a stochastic context (rather than the deterministic one given here).

Exercise 2.8 Let $\hat{x}$ denote the value of x that minimizes $\|y - Ax\|^2$, where A has full column rank. Let $\overline{x}$ denote the value of x that minimizes this same criterion, but now subject to the constraint that $z = Dx$, where D has full *row* rank. Show that

$$\overline{x} = \hat{x} + (A^T A)^{-1} D^T \left(D(A^T A)^{-1} D^T \right)^{-1} (z - D\hat{x})$$

(Hint: One approach to solving this is to use our recursive least squares formulation, but modified for the limiting case where one of the measurement sets — namely $z = Dx$ in this case — is known to have no error. You may have to use some of the matrix identities from the previous chapter).

6.241J / 16.338J Dynamic Systems and Control

Spring 2011

Simplified Presentation of BLUE (Best Linear Unbiased Estimators)

and

MVE (Minimum Variance Estimators)

(Ignore material on Maximum Likelihood)

LINEAR MODELS

Polynomial Curve Fitting Example. Continuous signal $x(t)$ is modeled as a polynomial of degree $p - 1$ in additive noise:

$$x(t) = \theta_1 + \theta_2 t + \cdots + \theta_p t^{p-1} + w(t).$$

Suppose that we are given $\{x(t_n)\}_{n=0}^{N-1}$. Define

$$
\begin{aligned}
\boldsymbol{x} &= [x(t_0), \ldots, x(t_{N-1})]^T \\
\boldsymbol{w} &= [w(t_0), \ldots, w(t_{N-1})]^T \\
\boldsymbol{\theta} &= [\theta_1, \ldots, \theta_p]^T \\
\boldsymbol{H} &= \begin{bmatrix} 1 & t_0 & \cdots & t_0^{p-1} \\ 1 & t_1 & \cdots & t_1^{p-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & t_{N-1} & \cdots & t_{N-1}^{p-1} \end{bmatrix} \quad \text{(an } N \times p \text{ matrix).}
\end{aligned}
$$

The data model is then

$$\boldsymbol{x} = \boldsymbol{H}\boldsymbol{\theta} + \boldsymbol{w}$$

where $\boldsymbol{H}$ is known and $\boldsymbol{\theta}$ is the parameter vector to be estimated.

Sinusoidal Amplitude and Phase Estimation Estimation:

Measured signal $x(t)$ is modeled as a superposition of $p/2$ sinusoids (having known frequencies but unknown amplitudes and phases):

$$x(t) = \sum_{k=1}^{p/2} r_k \sin(\omega_k t + \phi_k) + w(t).$$

This model is linear in r_k but nonlinear in ϕ_k. However, we can rewrite it as

$$x(t) = \sum_{k=1}^{p/2} [A_k \cos(\omega_k t) + B_k \sin(\omega_k t)] + w(t).$$

Given $\boldsymbol{x} = [x(t_0), \ldots, x(t_{N-1})]^T$, we get the following model:

$$\boldsymbol{x} = \boldsymbol{H}\boldsymbol{\theta} + \boldsymbol{w}.$$

Linear Models (cont.)

For $p/2 = 2$ sinusoids:

$$\boldsymbol{H} =$$

$$\begin{bmatrix} \cos(\omega_1 t_0) & \cos(\omega_2 t_0) & \sin(\omega_1 t_0) & \sin(\omega_2 t_0) \\ \cos(\omega_1 t_1) & \cos(\omega_2 t_1) & \sin(\omega_1 t_1) & \sin(\omega_2 t_1) \\ \vdots & \vdots & \vdots & \vdots \\ \cos(\omega_1 t_{N-1}) & \cos(\omega_2 t_{N-1}) & \sin(\omega_1 t_1) & \sin(\omega_2 t_{N-1}) \end{bmatrix}$$

and

$$\boldsymbol{\theta} = [A_1, \ldots, A_{p/2}, B_1, \ldots, B_{p/2}]^T.$$

Once we compute an estimate $\widehat{\boldsymbol{\theta}}$ of $\boldsymbol{\theta}$, $\widehat{r}_k$ and $\widehat{\phi}_k$ are obtained using the simple conversion from rectangular to polar coordinates.

Note: Even if $\widehat{\boldsymbol{\theta}}$ is a minimum variance unbiased (MVU) estimator, $\{\widehat{r}_k\}$ and $\{\widehat{\phi}_k\}$ will only be *asymptotically* MVU (for large N), as we will see later.

Consider the model

$$x = H\theta + \overbrace{w}^{\text{noise}}$$

where x is a measured $N \times 1$ vector and H is a known *deterministic* $N \times p$ matrix, with $N \geq p$. We wish to estimate the *unknown* parameter vector θ.

Assume that w is distributed as $w \sim \mathcal{N}(0, \sigma^2 I)$.

Recall the *identifiability condition:*

$$p(x; \theta_1) = p(x; \theta_2) \quad \Leftrightarrow \quad \theta_1 = \theta_2$$

which, in this case, reduces to

$$H\theta_1 = H\theta_2 \quad \Leftrightarrow \quad \theta_1 = \theta_2.$$

To satisfy this condition, we assume that H has full rank p.

Minimum Variance Unbiased Estimator for the Linear Model

Theorem 1. *For the model*

$$x = H\theta + w$$

where $w \sim \mathcal{N}(0, \sigma^2 I)$, the MVU estimator of θ is given by

$$\widehat{\theta} = (H^T H)^{-1} H^T x. \tag{1}$$

The covariance matrix of $\widehat{\theta}$ attains the Cramér-Rao bound (CRB) for all $\theta \in \mathbb{R}^p$ and is given by

$$C_{\widehat{\theta}} = \mathrm{E}\left[(\widehat{\theta} - \theta)(\widehat{\theta} - \theta)^T\right] = \sigma^2 (H^T H)^{-1}.$$

Proof. Verifying the unbiasedness of $\widehat{\theta}$ and the covariance matrix expression $\mathrm{cov}(\widehat{\theta}) = \mathrm{E}\left[(\widehat{\theta} - \theta)(\widehat{\theta} - \theta)^T\right] = \sigma^2 (H^T H)^{-1}$ proves the theorem.

For the above model,

$$\mathbf{CRB}(\theta) = \mathcal{I}(\theta)^{-1}$$

and the Fisher information matrix (FIM) for $\boldsymbol{\theta}$, $\mathcal{I}(\boldsymbol{\theta})$, is computed using the general Gaussian FIM expression in handout # 2:

$$[\mathcal{I}(\boldsymbol{\theta})]_{i,k} = \frac{1}{\sigma^2} \cdot \frac{\partial \boldsymbol{\mu}(\boldsymbol{\theta})^T}{\partial \theta_i} \frac{\partial \boldsymbol{\mu}(\boldsymbol{\theta})}{\partial \theta_k}$$

where $\boldsymbol{\mu}(\boldsymbol{\theta}) = \boldsymbol{H}\boldsymbol{\theta}$. Now

$$\frac{\partial \boldsymbol{\mu}(\boldsymbol{\theta})}{\partial \theta_i} = \frac{\partial (\boldsymbol{H}\boldsymbol{\theta})}{\partial \theta_i} = i\text{th column of } \boldsymbol{H}$$

implying that

$$\mathcal{I}(\boldsymbol{\theta}) = \frac{1}{\sigma^2} \cdot \boldsymbol{H}^T \boldsymbol{H} \quad \Longrightarrow \quad \mathbf{CRB}(\boldsymbol{\theta}) = \sigma^2 \, (\boldsymbol{H}^T \boldsymbol{H})^{-1}. \quad (2)$$

□

Comments:

- Since the joint FIM and CRB for $[\boldsymbol{\theta}^T, \sigma^2]^T$ are block-diagonal matrices, $\boldsymbol{\theta}$ and σ^2 are decoupled $\Longrightarrow \mathbf{CRB}(\boldsymbol{\theta})$ **is the same regardless of whether σ^2 is known or not.** To be more precise, $\mathbf{CRB}(\boldsymbol{\theta})$ in (2) is the *CRB for $\boldsymbol{\theta}$ assuming that σ^2 is known* and here is the full CRB for $\underbrace{\boldsymbol{\theta} \text{ and } \sigma^2}$ for the case

$$\boldsymbol{\rho} = \begin{bmatrix} \boldsymbol{\theta} \\ \sigma^2 \end{bmatrix}$$

where both $\boldsymbol{\theta}$ and σ^2 are unknown:

$$\mathbf{CRB}_{\boldsymbol{\rho},\boldsymbol{\rho}}(\boldsymbol{\theta},\sigma^2) = \begin{bmatrix} \overbrace{\mathbf{CRB}_{\boldsymbol{\theta}\boldsymbol{\theta}}(\boldsymbol{\theta},\sigma^2)}^{\text{same as (2)}} & \mathbf{0} \\ \mathbf{0} & \mathrm{CRB}_{\sigma^2,\sigma^2}(\sigma^2) \end{bmatrix}.$$

Therefore, $\widehat{\boldsymbol{\theta}}$ in (1) is the MVU estimator of $\boldsymbol{\theta}$ regardless of whether σ^2 is known or not.

- $\widehat{\boldsymbol{\theta}}$ in (1) coincides with the *least-squares (LS) estimate* of $\boldsymbol{\theta}$:

$$\widehat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \|\boldsymbol{x} - \boldsymbol{H}\boldsymbol{\theta}\|^2$$

which can be shown by differentiating $\|\boldsymbol{x} - \boldsymbol{H}\boldsymbol{\theta}\|^2$ with respect to $\boldsymbol{\theta}$ and setting the result to zero or by completing the squares. Later in this handout, we will see a geometric interpretation of the LS approach.

The solution from the above theorem is numerically not sound as given. It is better to use a QR factorization, say, briefly outlined below. Suppose that the $N \times p$ matrix $\boldsymbol{H}$ is factored as

$$\boldsymbol{H} = \boldsymbol{Q}\boldsymbol{R} = [\boldsymbol{Q}_1 \ \boldsymbol{Q}_2] \begin{bmatrix} \boldsymbol{R}_1 \\ 0 \end{bmatrix} = \boldsymbol{Q}_1 \boldsymbol{R}_1$$

where $\boldsymbol{Q}$ is orthonormal and $\boldsymbol{R}_1$ is upper triangular $p \times p$ (MATLAB: qr). Then

$$(\boldsymbol{H}^T \boldsymbol{H})^{-1} \boldsymbol{H}^T = \boldsymbol{R}_1^{-1} \boldsymbol{Q}_1^T.$$

Thus, $\widehat{\boldsymbol{\theta}}$ can be obtained by solving the triangular system of equations

$$\boldsymbol{R}_1 \widehat{\boldsymbol{\theta}} = \boldsymbol{Q}_1^T \boldsymbol{x}.$$

MATLAB has the "backslash" command for computing the LS solution:

$$\boldsymbol{\theta} = \boldsymbol{H} \backslash \boldsymbol{x};$$

Minimum Variance Unbiased Estimator for the Linear Model, Colored Noise

Suppose that we have colored noise, so that $\boldsymbol{w} \sim \mathcal{N}(\boldsymbol{0}, \sigma^2 \boldsymbol{C})$, where $\boldsymbol{C} \neq \boldsymbol{I}$ is known and positive definite.

We can use *prewhitening* to get back to the old problem (i.e. the white-noise case). We compute the Cholesky factorization of $\boldsymbol{C}^{-1}$:

$$\boldsymbol{C}^{-1} = \boldsymbol{D}^T \boldsymbol{D} \quad \text{Matlab:} \ \texttt{D = inv(chol(C))';}$$

(Any other square-root factorization could be used as well.)

Now, define the transformed measurement model:

$$\underbrace{\boldsymbol{D}\,\boldsymbol{x}}_{\boldsymbol{x}^{\text{transf}}} = \underbrace{\boldsymbol{D}\,\boldsymbol{H}}_{\boldsymbol{H}^{\text{transf}}}\ \boldsymbol{\theta} + \underbrace{\boldsymbol{D}\,\boldsymbol{w}}_{\boldsymbol{w}^{\text{transf}}}\ .$$

Clearly, $\boldsymbol{w}^{\text{transf}} \sim \mathcal{N}(\boldsymbol{0}, \sigma^2 \boldsymbol{I})$ and the problem is reduced to the white-noise case.

MVU Estimation, Colored Noise (cont.)

Theorem 2. *For colored Gaussian noise with known covariance C, the MVU estimate of $\boldsymbol{\theta}$ is*

$$\widehat{\boldsymbol{\theta}} = (\boldsymbol{H}^T \boldsymbol{C}^{-1} \boldsymbol{H})^{-1} \boldsymbol{H}^T \boldsymbol{C}^{-1} \boldsymbol{x}.$$

The covariance matrix of $\widehat{\boldsymbol{\theta}}$ attains the CRB and is given by

$$C_{\widehat{\boldsymbol{\theta}}} = (\boldsymbol{H}^T \boldsymbol{C}^{-1} \boldsymbol{H})^{-1}.$$

Note: $\widehat{\boldsymbol{\theta}}$ is a weighted LS estimate,

$$
\begin{aligned}
\widehat{\boldsymbol{\theta}} &= \arg\min_{\boldsymbol{\theta}} \|\boldsymbol{x} - \boldsymbol{H}\boldsymbol{\theta}\|_{\boldsymbol{W}}^2 \\
&= \arg\min_{\boldsymbol{\theta}} (\boldsymbol{x} - \boldsymbol{H}\boldsymbol{\theta})^T \boldsymbol{W} (\boldsymbol{x} - \boldsymbol{H}\boldsymbol{\theta}).
\end{aligned}
$$

The "optimal weight matrix," $\boldsymbol{W} = \boldsymbol{C}^{-1}$, prewhitens the residuals.

Best Linear Unbiased Estimator

Given the model

$$\boldsymbol{x} = \boldsymbol{H}\boldsymbol{\theta} + \boldsymbol{w} \qquad\qquad (3)$$

where $\boldsymbol{w}$ has zero mean and covariance matrix $\mathrm{E}\left[\boldsymbol{w}\boldsymbol{w}^T\right] = \boldsymbol{C}$, we look for the *best linear unbiased estimator (BLUE)*. Hence, we restrict our estimator to be

- linear (i.e. of the form $\widehat{\boldsymbol{\theta}} = \boldsymbol{A}^T\boldsymbol{x}$) and

- unbiased

and minimize its variance.

Theorem 3. *(Gauss-Markov) The BLUE of $\boldsymbol{\theta}$ is*

$$\widehat{\boldsymbol{\theta}} = (\boldsymbol{H}^T\boldsymbol{C}^{-1}\boldsymbol{H})^{-1}\boldsymbol{H}^T\boldsymbol{C}^{-1}\boldsymbol{x}$$

and its covariance matrix is

$$\boldsymbol{C}_{\widehat{\boldsymbol{\theta}}} = (\boldsymbol{H}^T\boldsymbol{C}^{-1}\boldsymbol{H})^{-1}.$$

The expression for $\boldsymbol{C}_{\widehat{\boldsymbol{\theta}}}$ holds *independently* of the distribution of $\boldsymbol{w}$ — all we impose on $\boldsymbol{w}$ is that it has known mean vector and covariance matrix, equal to $\boldsymbol{0}$ and $\boldsymbol{C}$ (respectively).

The estimate $\widehat{\boldsymbol{\theta}}$ is (statistically) efficient if $\boldsymbol{w}$ is Gaussian (i.e. it attains the CRB), but it is not efficient in general. For non-Gaussian measurement models, there might be a better nonlinear estimate. (Most likely, there exists a better nonlinear estimate.)

Proof. (of Theorem 3). For simplicity, consider first the case where θ is scalar. Then, our measurement model is

$$x[n] = h[n]\,\theta + w[n] \quad \Longleftrightarrow \quad \boldsymbol{x} = \boldsymbol{h}\,\theta + \boldsymbol{w}.$$

The candidate linear estimates of θ have the following form:

$$\widehat{\theta} = \sum_{n=0}^{N-1} a_n x[n] = \boldsymbol{a}^T \boldsymbol{x}.$$

First, the bias is computed:

$$\mathrm{E}\,[\widehat{\theta}] = \boldsymbol{a}^T \mathrm{E}\,[\boldsymbol{x}] = \boldsymbol{a}^T \boldsymbol{h}\,\theta.$$

Thus, $\widehat{\theta}$ is unbiased if and only if $\boldsymbol{a}^T \boldsymbol{h} = 1$. Next, compute the variance of $\widehat{\theta}$. We have

$$\widehat{\theta} - \theta = \boldsymbol{a}^T (\underbrace{\boldsymbol{h}\,\theta + \boldsymbol{w}}_{\boldsymbol{x}}) - \theta = \boldsymbol{a}^T \boldsymbol{w}$$

where we have used the unbiasedness condition:
Therefore, the variance is

$$\mathrm{E}\left[(\widehat{\theta} - \theta)^2\right] = \mathrm{E}\left[(\boldsymbol{a}^T\boldsymbol{w})^2\right] = \mathrm{E}\left[\boldsymbol{a}^T\boldsymbol{w}\boldsymbol{w}^T\boldsymbol{a}\right] = \boldsymbol{a}^T\boldsymbol{C}\boldsymbol{a}.$$

Note: The variance of $\widehat{\theta}$ depends only on the second-order properties of the noise. This result holds for any noise distribution that has second-order moments.

Thus, the BLUE problem is

$$\min_{\boldsymbol{a}} \boldsymbol{a}^T\boldsymbol{C}\boldsymbol{a} \qquad \text{such that} \quad \boldsymbol{a}^T\boldsymbol{h} = 1.$$

Note the equivalence with MVDR beamforming. To read more about MVDR beamforming, see

H.L. Van Trees, *Detection, Estimation and Modulation Theory*, New York: Wiley, 2002, pt. IV.

Lagrange-multiplier formulation:

$$L(\boldsymbol{a}) = \boldsymbol{a}^T\boldsymbol{C}\boldsymbol{a} + \lambda\cdot(\boldsymbol{a}^T\boldsymbol{h} - 1) \quad \textcolor{red}{\overset{\text{differentiate}}{\Longrightarrow}} \quad 2\,\boldsymbol{C}\boldsymbol{a} + \lambda\,\boldsymbol{h} = \boldsymbol{0}.$$

Hence

$$\boldsymbol{a} = -\frac{\lambda}{2}\cdot\boldsymbol{C}^{-1}\boldsymbol{h}$$

and then

$$\boldsymbol{a}^T\boldsymbol{h} = -\frac{\lambda}{2}\boldsymbol{h}^T\boldsymbol{C}^{-1}\boldsymbol{h} = 1 \quad \Rightarrow \quad \lambda = -\frac{2}{\boldsymbol{h}^T\boldsymbol{C}^{-1}\boldsymbol{h}}$$

and optimal $\boldsymbol{a}$ follows: $\boldsymbol{a} = (\boldsymbol{h}^T \boldsymbol{C}^{-1} \boldsymbol{h})^{-1} \boldsymbol{C}^{-1} \boldsymbol{h}$. Returning to our estimator, we find the BLUE to be

$$\widehat{\theta} = (\boldsymbol{h}^T \boldsymbol{C}^{-1} \boldsymbol{h})^{-1} \boldsymbol{h}^T \boldsymbol{C}^{-1} \boldsymbol{x}.$$

and its variance is given by

$$\mathrm{E}\,[(\widehat{\theta} - \theta)^2] = (\boldsymbol{h}^T \boldsymbol{C}^{-1} \boldsymbol{h})^{-1}.$$

$\square$

Consider the vector case. Linear unbiased estimates of $\boldsymbol{\theta}$:

$$\widehat{\boldsymbol{\theta}} = \boldsymbol{A}^T \boldsymbol{x}, \quad \text{where } \boldsymbol{A} \text{ is independent of } \boldsymbol{x}. \tag{4}$$

Remark: For LS estimate $\widehat{\boldsymbol{\theta}}_{\mathrm{LS}}$, $\boldsymbol{A}^T = (\boldsymbol{H}^T \boldsymbol{H})^{-1} \boldsymbol{H}^T$.

$$\boldsymbol{\theta} = \mathrm{E}\,[\widehat{\boldsymbol{\theta}}] = \mathrm{E}\,[\boldsymbol{A}^T \boldsymbol{x}] = \mathrm{E}\,[\boldsymbol{A}^T(\boldsymbol{H}\boldsymbol{\theta} + \boldsymbol{w})] = \boldsymbol{A}^T \boldsymbol{H} \boldsymbol{\theta}.$$

$$\Rightarrow \quad \boldsymbol{A}^T \boldsymbol{H} = \boldsymbol{I}.$$

Remark: For BLUE $\widehat{\boldsymbol{\theta}}_{\mathrm{BLUE}}$, $\boldsymbol{A}^T_{\mathrm{BLUE}} = (\boldsymbol{H}^T \boldsymbol{C}^{-1} \boldsymbol{H})^{-1} \boldsymbol{H}^T \boldsymbol{C}^{-1}$.
Since $\boldsymbol{A}^T_{\mathrm{BLUE}} \boldsymbol{H} = \boldsymbol{I} \Rightarrow \mathrm{E}\,[\widehat{\boldsymbol{\theta}}_{\mathrm{BLUE}}] = \boldsymbol{\theta}$.

$$\mathrm{cov}(\widehat{\boldsymbol{\theta}}) = \mathrm{E}\,\{[\boldsymbol{A}^T(\underbrace{\boldsymbol{H}\boldsymbol{\theta} + \boldsymbol{w}}_{\boldsymbol{x}}) - \boldsymbol{\theta}][\boldsymbol{A}^T(\boldsymbol{H}\boldsymbol{\theta} + \boldsymbol{w}) - \boldsymbol{\theta}]^T\} = \boldsymbol{A}^T \boldsymbol{C} \boldsymbol{A}.$$

and

$$\mathrm{cov}(\widehat{\boldsymbol{\theta}}_{\mathrm{BLUE}}) \;=\; (\boldsymbol{H}^T \boldsymbol{C}^{-1} \boldsymbol{H})^{-1} \boldsymbol{H}^T \boldsymbol{C}^{-1} \boldsymbol{C} \boldsymbol{C}^{-1} \boldsymbol{H} (\boldsymbol{H}^T \boldsymbol{C}^{-1} \boldsymbol{H})^{-1}$$

$$=\; (\boldsymbol{H}^T \boldsymbol{C}^{-1} \boldsymbol{H})^{-1}.$$

To prove that $\widehat{\boldsymbol{\theta}}_{\mathrm{BLUE}}$ has the smallest variance [within the family of linear unbiased estimators $\widehat{\boldsymbol{\theta}}$ in (4)], we show that

$$\mathrm{cov}(\widehat{\boldsymbol{\theta}}_{\mathrm{BLUE}}) \leq \mathrm{cov}(\widehat{\boldsymbol{\theta}})$$

as follows:

$$\mathrm{cov}(\widehat{\boldsymbol{\theta}}) - \mathrm{cov}(\widehat{\boldsymbol{\theta}}_{\mathrm{BLUE}}) \;=\; \boldsymbol{A}^T \boldsymbol{C} \boldsymbol{A} - (\boldsymbol{H}^T \boldsymbol{C}^{-1} \boldsymbol{H})^{-1}$$

$$\stackrel{\boldsymbol{A}^T \boldsymbol{H} = \boldsymbol{I}}{=} \boldsymbol{A}^T \boldsymbol{C} \boldsymbol{A} - \boldsymbol{A}^T \boldsymbol{H} (\boldsymbol{H}^T \boldsymbol{C}^{-1} \boldsymbol{H})^{-1} \boldsymbol{H}^T \boldsymbol{A}$$

$$=\; \boldsymbol{A}^T [\boldsymbol{C} - \boldsymbol{H} (\boldsymbol{H}^T \boldsymbol{C}^{-1} \boldsymbol{H})^{-1} \boldsymbol{H}^T] \boldsymbol{A}$$

$$=\boldsymbol{A}^T [\boldsymbol{C} - \boldsymbol{H} (\boldsymbol{H}^T \boldsymbol{C}^{-1} \boldsymbol{H})^{-1} \boldsymbol{H}^T] \boldsymbol{C}^{-1} [\boldsymbol{C} - \boldsymbol{H} (\boldsymbol{H}^T \boldsymbol{C}^{-1} \boldsymbol{H})^{-1} \boldsymbol{H}^T] \boldsymbol{A}$$

which is always positive semidefinite.

Based on Chapter 4 of Luenberger, "Optimization by Vector Space Methods". Based on the

Projection Theorem. Has a short probability review as well.

Chapter 4

Linear Estimation Theory

- Virtually all branches of science, engineering, and social science for data analysis, system control subject to random disturbances or for decision making based on incomplete information call for estimation theory.

- Many estimation problem can be formulated as a minimum norm problem in Hilbert space.

- The projection theorem can be applied directly to the area of statistical estimation.

- There are a number of different ways to formulate a statistical estimation.

  ⋄ Least squares.

  ⋄ Maximum likelihood

  ⋄ Bayesian techniques.

- When all variables are Gaussian statistics, these techniques produce linear equations.

Preliminaries

- If x is a real-valued random variable,

  ◇ The *probability distribution* P of the variable x is defined to be

  $$P(\xi) = \text{Prob}(x \leq \xi).$$

  ◇ The "derivative" $p(\xi)$ of the probability distribution $P(\xi)$ is called the *probability density function (pdf)* of the variable x, i.e.,

  $$P(\xi) = \int_{-\infty}^{\xi} p(x)dx.$$

  ▷ Note that
  $$\int_{-\infty}^{\infty} p(x)dx = 1.$$

  ▷ $p(\xi) \geq 0$ for all ξ.

- The expected value of any function g of x is defined to be

  $$\mathcal{E}[g(x)] := \int_{-\infty}^{\infty} g(\xi)p(\xi)d\xi.$$

  ◇ The *expected value* of x is $\mathcal{E}[x]$.
  ◇ The *variance* of x is $\mathcal{E}[(x - \mathcal{E}[x])^2]$.

- For random vector $\mathbf{x} = [x_1, \ldots, x_n]^\top$,

  ◇ There is a *joint probability distribution* P defined by

  $$P(\xi_1, \ldots \xi_n) = \text{Prob}(x_1 \le \xi_1, \ldots, x_n \le \xi_n).$$

  ◇ The *covariance matrix* $\text{cov}(\mathbf{x})$ is defined by

  $$\text{cov}(\mathbf{x}) = \mathcal{E}\left[(\mathbf{x} - \mathcal{E}[\mathbf{x}])(\mathbf{x} - \mathcal{E}[\mathbf{x}])^\top \right].$$

    ▷ Two random variables x_i and x_j are said to be uncorrelated or stochastically independent if

    $$\mathcal{E}[(x_1 - \mathcal{E}[x_1])(x_2 - \mathcal{E}[x_2])] = \mathcal{E}[x_1 - \mathcal{E}[x_1]]\mathcal{E}[x_2 - \mathcal{E}[x_2]].$$

Least Squares Model

- This is a familiar subject as we have seen in many occasions.

- This problem is not a statistical one.

- It amounts to approximating a vector $\mathbf{y} \in \mathbb{R}^m$ by a vector lying in the column space of $W \in \mathbf{R}^{m \times n}$ and $n < m$.

  $\diamond$ We assume a linear model that the response $\mathbf{y}$ is related to the input $\boldsymbol{\beta}$ linearly, i.e.,

  $$\mathbf{y} = W\boldsymbol{\beta}.$$

  $\diamond$ We would like to recover $\boldsymbol{\beta}$ from observed $\mathbf{y}$. (Would it be a linear relationship?)

  $\diamond$ We are not assuming that the observed $\mathbf{y}$ carries errors.

- It would be interesting to compare the least squares setting with those with random noises.

Least Squares Formulation

- Given

    ◇ A known matrix $W \in \mathbb{R}^{m \times n}$, $n < m$.

    ◇ An observation vector $\mathbf{y} \in \mathbb{R}^m$.

  Find $\hat{\boldsymbol{\beta}} \in \mathbb{R}^n$ such that $\|\mathbf{y} - W\hat{\boldsymbol{\beta}}\|$ is minimized over all $\boldsymbol{\beta} \in \mathbf{R}^n$.

- By the projection theorem, the solution exists and is unique.

- The normal equation is given by

$$W^\top(\mathbf{y} - W\hat{\boldsymbol{\beta}}) = 0.$$

- If W has linear independent columns, then

$$\hat{\boldsymbol{\beta}} = \underbrace{(W^\top W)^{-1} W^\top}_{K} \mathbf{y}.$$

    ◇ Note that the optimal solution $\hat{\boldsymbol{\beta}}$ is related to $\mathbf{y}$ linearly.

Gauss-Markov Model

- A more realistic model in an experiment is

$$\mathbf{y} = W\boldsymbol{\beta} + \boldsymbol{\epsilon}.$$

  ◇ $W \in \mathbb{R}^{m \times n}$ is known.

  ◇ $\boldsymbol{\epsilon} \in \mathbb{R}^m$ is a random vector with zero mean and covariance $\mathcal{E}(\boldsymbol{\epsilon}\boldsymbol{\epsilon}^\top) = Q$.

  ◇ $\mathbf{y}$ represents the outcome of inexact measurements in $\mathbb{R}^m$.

- Want to estimate unknown parameter vector $\boldsymbol{\beta} \in \mathbb{R}^n$ from $\mathbf{y} \in \mathbb{R}^m$ using

$$\hat{\boldsymbol{\beta}} := K\mathbf{y}$$

  with K an unknown matrix in $R^{n \times m}$.

- Suppose the approximation is measured by minimizing the expected value of the error, i.e.,

$$\min_{K \in \mathbb{R}^{n \times m}} \mathcal{E}[\|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}\|^2]$$

  ◇ Since $\mathbf{y}$ carries random noise, it is a random vector.

  ◇ Both estimate $\hat{\boldsymbol{\beta}}$ and the difference $\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}$ are random vectors.

  ◇ The statistics of these random vectors are determined by those of $\boldsymbol{\epsilon}$ and K.

Gauss-Markov Estimate

- Observe that

$$
\begin{aligned}
\mathcal{E}[\|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}\|^2] &= \mathcal{E}\left[\langle K(W\boldsymbol{\beta} + \epsilon) - \boldsymbol{\beta}, K(W\boldsymbol{\beta} + \epsilon) - \boldsymbol{\beta}\rangle\right] \\
&= \|KW\boldsymbol{\beta} - \boldsymbol{\beta}\|^2 + \mathcal{E}[\langle K\epsilon, K\epsilon\rangle].
\end{aligned}
$$

- Consider unbiased estimation:

    ⋄ Observe

    $$
    \mathcal{E}[\hat{\boldsymbol{\beta}}] = \mathcal{E}[KW\boldsymbol{\beta} + K\epsilon] = KW\mathcal{E}[\boldsymbol{\beta}].
    $$

    It is expected that $KW = I_n$.

    ⋄ The problem now becomes, given a symmetric and positive definite matrix Q,

    $$
    \begin{aligned}
    &\text{minimize}_{K \in \mathbb{R}^{n \times m}} \quad \text{trace } KQK^\top \\
    &\text{subject to} \qquad KW = I_n.
    \end{aligned}
    $$

    ⋄ This is in the form of a standard minimum norm problem.

- The problem has a closed form solution.

  ◇ The optimal solution is given by

  $$K = \left(W^\top Q^{-1} W\right)^{-1} W^\top Q^{-1}.$$

  ◇ The minimum-variance unbiased estimation of $\boldsymbol{\beta}$ is given by

  $$\hat{\boldsymbol{\beta}} = \left(W^\top Q^{-1} W\right)^{-1} W^\top Q^{-1} \mathbf{y}.$$

  ◇ The special case $Q = I_m$ is the classical least squares problem.

    ▷ The classical least squares solution is providing the unbiased minimum-variance estimate of $\boldsymbol{\beta}$, if the perturbation presented in data is white noise.

- It can be argued that the above solution $\hat{\boldsymbol{\beta}}_i$ is the minimum-variance unbiased estimation of $\boldsymbol{\beta}_i$ for each individual i.

  ◇ This is the true minimum-variance unbiased estimate.

Minimum-Variance Model

- Assume in the linear model
$$\mathbf{y} = W\boldsymbol{\beta} + \boldsymbol{\epsilon}.$$

  ◇ $W \in \mathbb{R}^{m \times n}$ is known.

  ◇ $\boldsymbol{\epsilon} \in \mathbb{R}^m$ is a random vector with zero mean and covariance $\mathcal{E}(\boldsymbol{\epsilon}\boldsymbol{\epsilon}^\top) = Q$.

  ◇ $\boldsymbol{\beta}$ is a random vector in $\mathbb{R}^n$ with known statistical information.

  ◇ $\mathbf{y}$ represents the outcome of inexact measurements in $\mathbb{R}^m$.

- Want to estimate the unknown random vector $\boldsymbol{\beta} \in \mathbb{R}^n$ based on $\mathbf{y} \in \mathbb{R}^m$ using
$$\hat{\boldsymbol{\beta}} := K\mathbf{y}$$

  where K is an unknown matrix in $R^{n \times m}$.

- The best approximation is measured by minimizing the expected value of the random error, i.e.,
$$\min_{K \in \mathbb{R}^{n \times m}} \mathcal{E}[\|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}\|^2]$$

Minimum-Variance Estimate

- Assume $(\mathcal{E}[\mathbf{y}\mathbf{y}^T])^{-1}$ exists. Then the minimum-variance estimator of β is given by

$$\hat{\boldsymbol{\beta}} = \mathcal{E}[\boldsymbol{\beta}\mathbf{y}^T](\mathcal{E}[\mathbf{y}\mathbf{y}^T])^{-1}\mathbf{y}.$$

  ⋄ The estimate is independent of W and ϵ.

- Proof Is Interesting!

  ⋄ Write K in rows, i.e., $K = [\mathbf{k}_1^\top, \dots \mathbf{k}_n^\top]^\top$.
  ⋄ $\mathcal{E}[\|\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}\|^2] = \sum_{i=1}^n \mathcal{E}[(\hat{\beta}_i - \beta_i)^2] = \sum_{i=1}^n \mathcal{E}[(\mathbf{k}_i^\top \mathbf{y} - \beta_i)^2]$.
    ▷ Suffices to consider each individual term.
  ⋄ Let $f(\mathbf{y}, \beta_i)$ denote the (unknown) joint pdf of $\mathbf{y}$ and β_i.
    ▷ Define

$$\begin{aligned} g(\mathbf{k}_i) \ &:= \ \mathcal{E}[(\mathbf{y}^\top \mathbf{k}_i - \beta_i)^2] \\ &= \ \int \int (\mathbf{y}^\top \mathbf{k}_i - \beta_i)^2 f(\mathbf{y}, \beta_i) \, d\mathbf{y} \, d\beta_i. \end{aligned}$$

    ▷ Necessary condition is $\nabla g(\mathbf{k}_i) = 0$.

◇ Easy to see

$$\frac{\partial g}{\partial \mathbf{k}_{i,j}} = \int \int 2(\mathbf{y}^\top \mathbf{k}_i - \beta_i)\mathbf{y}_j f(\mathbf{y}, \beta_i) \, d\mathbf{y} \, d\beta_i$$
$$= 2\mathcal{E}[(\mathbf{y}^\top \mathbf{k}_i - \beta_i)\mathbf{y}_j].$$

◇ Rewrite the necessary condition as

$$\mathcal{E}[\mathbf{y}(\mathbf{y}^\top \mathbf{k}_i - \beta_i)] = 0, \quad \text{(for each } i)$$
$$\mathcal{E}[\mathbf{y}\mathbf{y}^\top]K^\top = \mathcal{E}[\mathbf{y}\beta^\top], \quad \text{(in matrix form)}$$
$$K = \mathcal{E}[\beta\mathbf{y}^\top](\mathcal{E}[\mathbf{y}\mathbf{y}^\top])^{-1}.$$

- The estimate so far is *biased*, unless $\mathcal{E}[\boldsymbol{\beta}] = \mathcal{E}[\mathbf{y}] = 0$.

- In the general case where $\mathcal{E}[\boldsymbol{\beta}] = \boldsymbol{\beta}_0$ and $\mathcal{E}[\mathbf{y}] = \mathbf{y}_0$,

  ◇ The estimate should assume the form

  $$\hat{\boldsymbol{\beta}} := K\mathbf{y} + \mathbf{b}.$$

  ◇ The minimum-variance estimate is given by

  $$\hat{\boldsymbol{\beta}} = \boldsymbol{\beta}_0 + \mathcal{E}[(\boldsymbol{\beta} - \boldsymbol{\beta}_0)(\mathbf{y} - \mathbf{y}_0)^\top](\mathcal{E}[(\mathbf{y} - \mathbf{y}_0)(\mathbf{y} - \mathbf{y}_0)^\top])^{-1}(\mathbf{y} - \mathbf{y}_0).$$

Equivalent Formulas

- Assume that

$$\begin{aligned}
\mathcal{E}[\boldsymbol{\epsilon}\boldsymbol{\epsilon}^\top] &= Q \in \mathbb{R}^{m \times m}, \\
\mathcal{E}[\boldsymbol{\beta}\boldsymbol{\beta}^\top] &= R \in \mathbb{R}^{n \times n}, \\
\mathcal{E}[\boldsymbol{\epsilon}\boldsymbol{\beta}^\top] &= 0 \in \mathbb{R}^{m \times n}.
\end{aligned}$$

- Then the minimum-variance estimate can be written as

$$\begin{aligned}
\hat{\boldsymbol{\beta}} &= \underbrace{RW^\top(WRW^\top + Q)^{-1}}_{K}\mathbf{y} \\
&= \overbrace{(W^\top Q^{-1}W + R^{-1})^{-1}W^\top Q^{-1}}^{K}\mathbf{y}.
\end{aligned}$$

  ⋄ The equivalence can be proved by direct substitution.

  ⋄ Check out the dimension of the inverse matrices involved in the two expressions.

Comparision with Gauss-Markov Estimate

- The Gauss-Markov estimate is

$$\hat{\boldsymbol{\beta}} = \left(W^\top Q^{-1} W\right)^{-1} W^\top Q^{-1} \mathbf{y}.$$

- The more subtle minimum-variance estimate is

$$\hat{\boldsymbol{\beta}} = (W^\top Q^{-1} W + R^{-1})^{-1} W^\top Q^{-1} \mathbf{y}.$$

- If $R^{-1} = 0$, the two estimates are idential.

  ⋄ What is meant by $R^{-1} = 0$?

  ⋄ *Infinite* variance of $\boldsymbol{\beta}$ in the more subtle estimate means that we have *absolutely no a priori knowledge* of $\boldsymbol{\beta}$ at all.

- When $\boldsymbol{\beta}$ is considered as a random variable, the size m of observations $\mathbf{y}$ does not need to be large.

  ⋄ $(WRW^\top + Q)^{-1}$ exists so long as Q is positive positive definite.

  ⋄ Every new measurement simply provides additional information which may modify the original estimate.

Application to Adaptic Optics

- Imaging through the Atmosphere

- Adaptive Optics System

  ◇ Basic Relationships
  ◇ Open-loop Model
  ◇ Closed-loop Model

- Adaptive Optics Control

  ◇ An Ideal Control
  ◇ An Inverse Problem
  ◇ Temporary Latency

- Numerical Illustration

Atmospheric Imaging Computation

- Purpose:

  ⋄ To compensate for the degradation of astronomical image quality caused by the effects of atmospheric turbulence.

- Two stages of approach:

  ⋄ Partially nullify optical distortions by a deformable mirror (DM) operated from a closed-loop adaptive optics (AO) system.

  ⋄ Minimize noise or blur via off-line post-processing deconvolution techniques (not this talk).

- Challenges:

  ⋄ Atmospheric turbulence can only be measured adaptively.

  ⋄ Need theory to pass atmospheric measurements to knowledge of actuating the DM.

  ⋄ Require fast performance of large-scale data processing and computations.

A Simplified AO System

Induced Wave Front

System
Pupil

Collimating
Lens

Wave Front
Sensor (WFS)

Deformable
Mirror (DM)

Beamsplitter

Focusing
Lens

Actuator
Control
Computer

Image Plane Dector
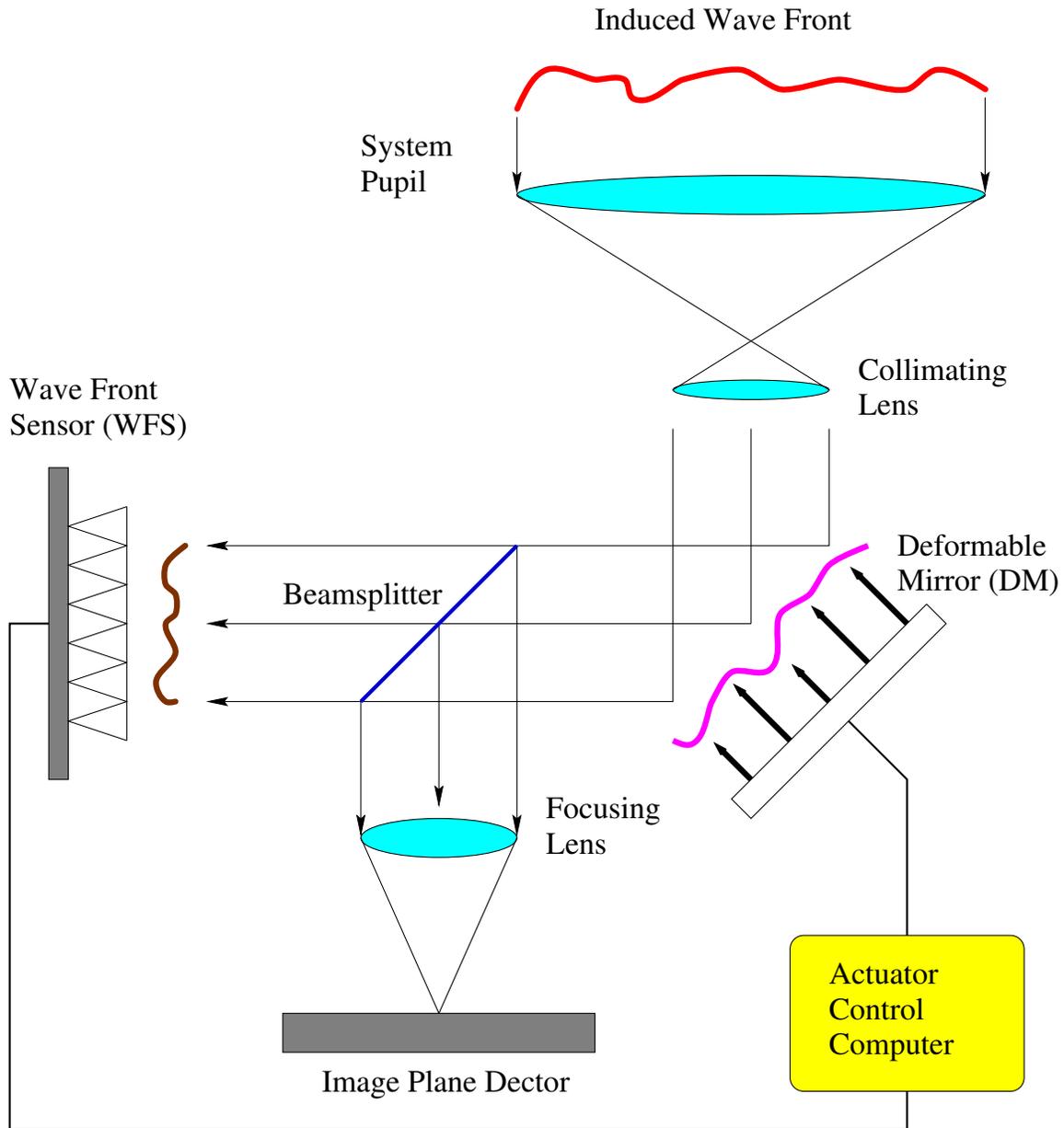
Basic Notation

- Three quantities:

  - $\diamond$ $\phi(t)$ = turbulence-induced phase profile at time t.
  - $\diamond$ $a(t)$ = deformable mirror (DM) actuator command at time t.
  - $\diamond$ $s(t)$ = wavefront slope sensor (WFS) measurement at time t and with no correction.

- Two transformations:

  - $\diamond$ $H :=$ transformation from actuator commands to resulting phase profile adjustments.
  - $\diamond$ $G :=$ transformation from actuator commands to slope sensor measurement adjustments.

From Actuator to DM Surface

- H is used to describe the DM surface change due to the application of actuators.

- $r_i(\vec{x})$ = influence function on the DM surface at position $\vec{x}$ with an unit adjustment to the ith actuator.

- Assuming m actuators and linear response of actuators to the command, model the DM surface by

$$\hat{\phi}(\vec{x}, t) = \sum_{i=1}^{m} a_i(t) r_i(\vec{x}).$$

⋄ Sampled at n DM surface positions, can write

$$\boxed{\hat{\phi}(t) = H a(t)}$$

▷ $H = (r_i(\vec{x}_j)) \in R^{n \times m}$.
▷ $\hat{\phi}(t) = [\hat{\phi}(\vec{x}_1, t), \ldots, \hat{\phi}(\vec{x}_n, t)]^T \in R^n$ = discrete corrected phase profile at time t.

From Actuator to WFS Measurement

- G is used to describe the WFS slope measurement associated with the actuator command a.

- Consider the H-WFS model where

$$s_j(t) := -\int d\vec{x}(\nabla W_j(\vec{x})\phi(\vec{x}, t), \quad j = 1, \ldots, \ell.$$

  ◇ W_j = given specifications of jth subaperture.

- The measurement corresponding to $\hat{\phi}(\vec{x}, t)$ would be

$$\hat{s}_j(t) = \sum_{i=1}^{m} \underbrace{\left(-\int d\vec{x}(\nabla W_j(\vec{x})r_i(\vec{x}) \right)}_{G_{ji}} a_i(t).$$

  ◇ Can write

$$\boxed{\hat{s}(t) = Ga(t)}$$

  where $G = [G_{ij}] \in R^{\ell \times m}$.

  ◇ The DM actuators are *not* capable of producing the exact wavefront phase $\phi(\vec{x}, t)$ due to its finiteness of degrees of freedom. So $\hat{s} = Ga$ is not an exact measurement.

A Closed-loop AO Control Model

Open-loop
Sensor Measurements
s

Σ

Closed-loop
Sensor Measurements
$\Delta s = s - Ga$

Corrected
Sensor
Measurement

$\hat{s} = Ga$

(Estimated Residual
Phase Error)

$e = M(s - Ga)$

Actuator
Command
a

Loop Compensation

Reconstructor

Corrected
Phase Profile

$\hat{\phi} = Ha$

ϕ
Open-loop
Phase Profile

Σ

$\Delta \phi = \phi - Ha$
Closed-loop
Phase Profile
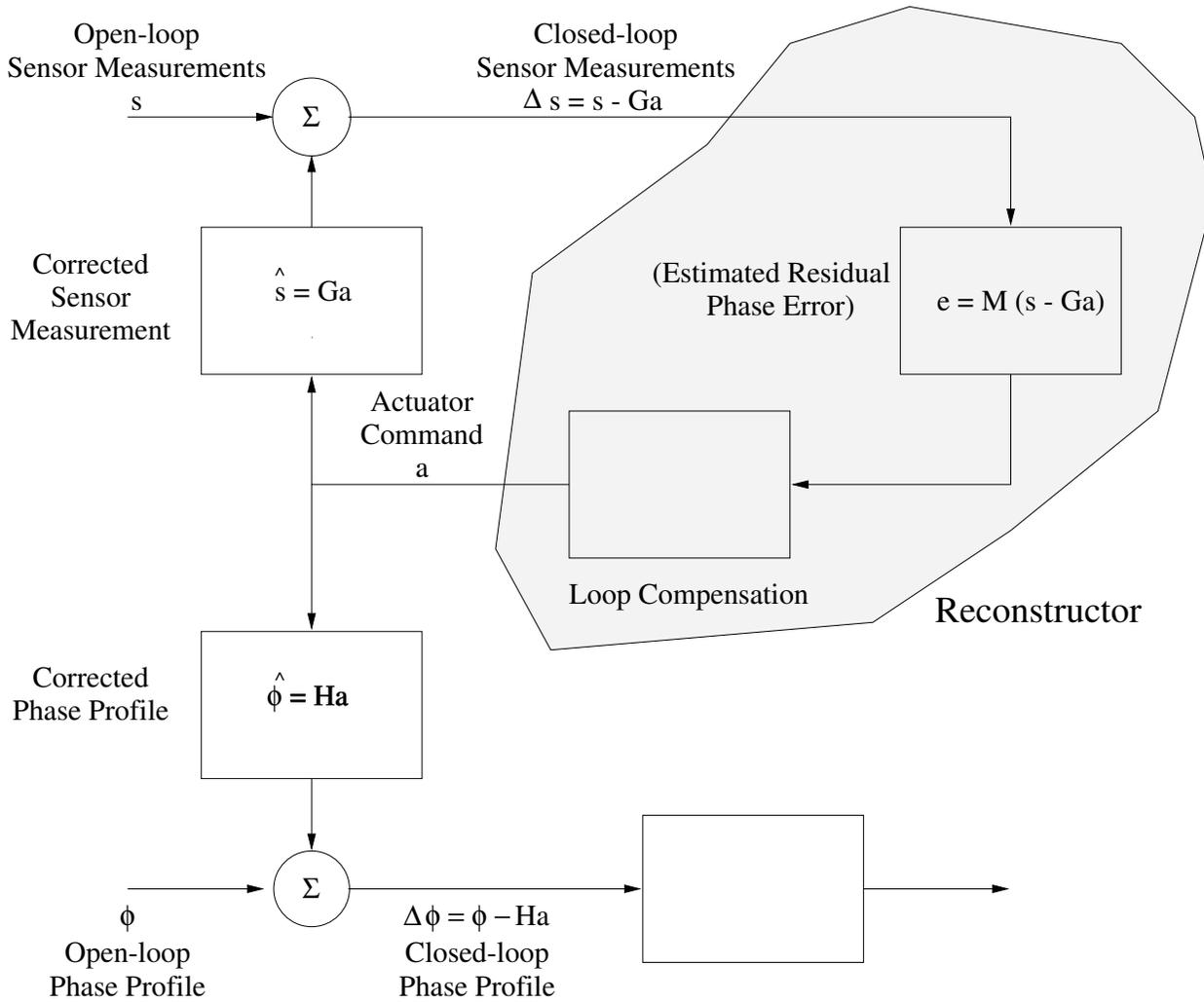
What is Available?

- Two residuals that are available in a *closed-loop* AO system:

    ◇ $\Delta\phi(t) := \phi(t) - Ha(t)$

    ▷ Represents the residual phase error remaining after the AO correction.
    ▷ Also means instantaneous closed-loop wavefront distortion at time t.

    ◇ $\Delta s(t) := s(t) - Ga(t)$

    ▷ Represents feedback applied to $s(t)$ by DM actuator adjustment.
    ▷ Also means *observable* wavefront sensor measurement at time t.

- In practice, there is a servo lag or delay in time Δt, i.e., it is likely

    ◇ $\Delta\phi(t) := \phi(t) - Ha(t - \Delta t)$.
    ◇ $\Delta s(t) := s(t) - Ga(t - \Delta t)$.

Thus the data collected are not perfect.

Open-loop Model

- Assume a linear relationship between open-loop WFS measurement s and turbulence-induced phase profile ϕ:

$$\boxed{s = W\phi + \epsilon}. \tag{1}$$

  ◇ ϵ = measurement noise with mean zero.

  ◇ In the H-WFS model, W represents a quadrature of the integral operator evaluated at designated positions $\vec{x}_j$, $j = 1, \ldots n$.

- Want to estimate ϕ using $\tilde{\phi}$ from the model

$$\tilde{\phi} = E_{open}s$$

  so that the variance

$$\mathcal{E}[\|\phi - \tilde{\phi}\|^2]$$

  is minimized.

  ◇ The wave front reconstruction matrix E_{open} is given by

$$E_{open} = \mathcal{E}[\phi s^T](\mathcal{E}[ss^T])^{-1}.$$

  ◇ For unbiased estimation, need to enforce the condition that $E_{open}W = I$.

Closed-loop Model

- For the H-WFS model, it is reasonable to assume the relationship

$$WH = G. \tag{2}$$

- Then

$$
\begin{aligned}
s &= W\phi + \epsilon \\
&= W(Ha + \Delta\phi) + \epsilon \\
&= WHa + (W\Delta\phi + \epsilon).
\end{aligned}
$$

It follows that

$$\boxed{\Delta s = W\Delta\phi + \epsilon}. \tag{3}$$

  ◇ The closed-loop relationship (3) is identical to the open-loop relationship (1).

- Can estimate the residual phase error $\Delta\phi(t)$ using $\Delta\tilde{\phi}(t)$ from the model

$$\Delta\tilde{\phi} = E_{closed}\Delta s$$

  ◇ E_{closed} = wavefront reconstruction matrix.

  ◇ For unbiased estimation, it requires that $E_{closed}W = I$. Hence

$$E_{closed}G = e_{closed}(WH) = H.$$

Actuator Control

- An Ideal Control:

  ◇ $\Delta\phi$ = residual error after DM correction by current command a_c.

  ◇ New command a_+ should reduce the residual error, i.e., want to

  $$\min_a \|Ha - \phi\|.$$

  ◇ Define $\Delta a := a_+ - a_c$, then want to

  $$\min_{\Delta a} \|H\Delta a - \Delta\phi\|.$$

  ◇ But $\Delta\phi$ is not observable directly. It has to be estimated from Δs.

- Estimating Δa directly from Δs:

  $$\boxed{\Delta a = M\Delta s} \tag{4}$$

An Inverse Problem

$$G$$

$$A \longrightarrow S$$

$$M$$

$$H$$

$$W$$

$$E$$

$$\Phi$$
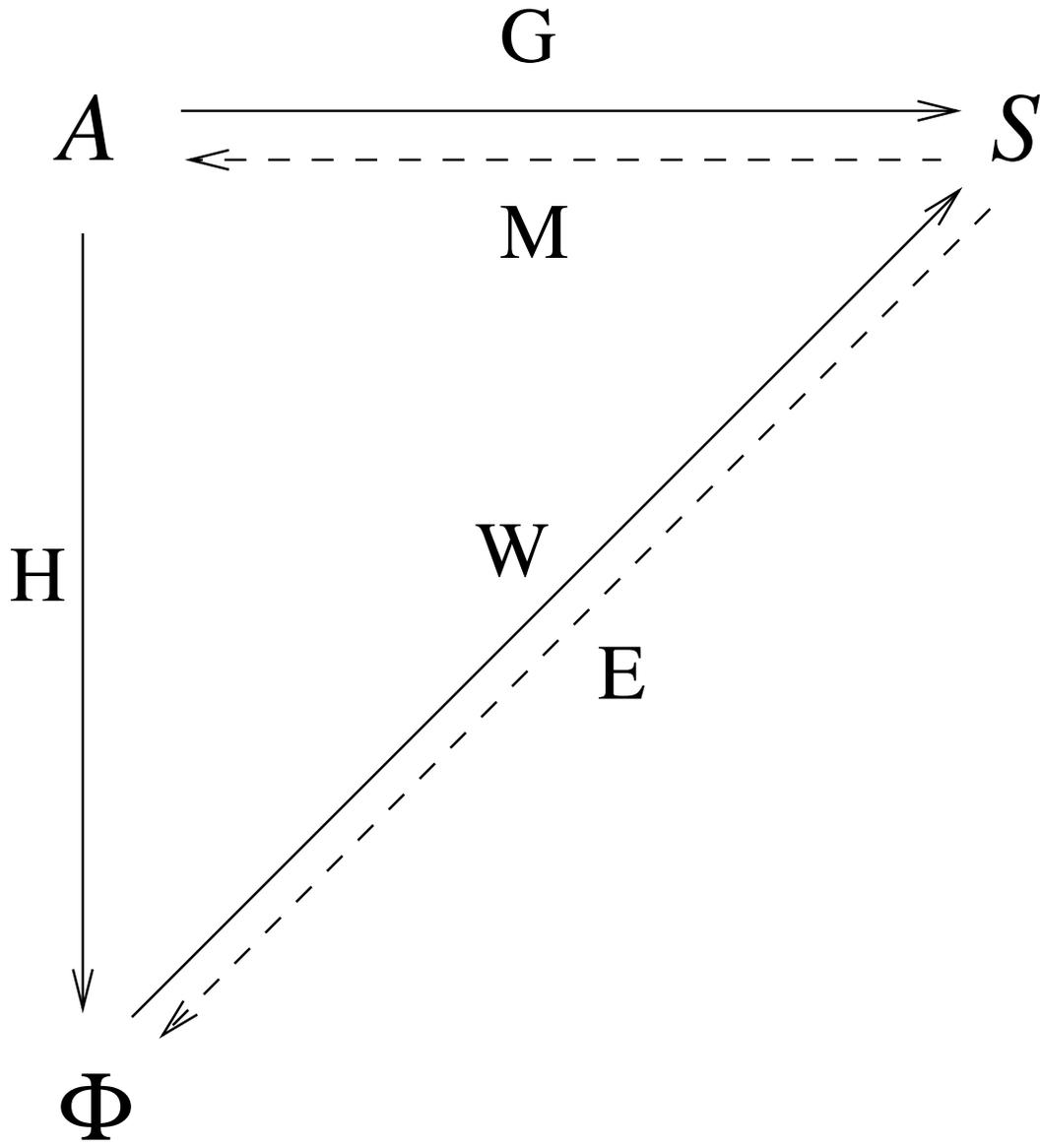
Actuator Control with Temporary Latency

- Due to finite bandwidth of the control loop, Δs is not immediately available.

- Time line for the scenario of a 2-cycle delay,
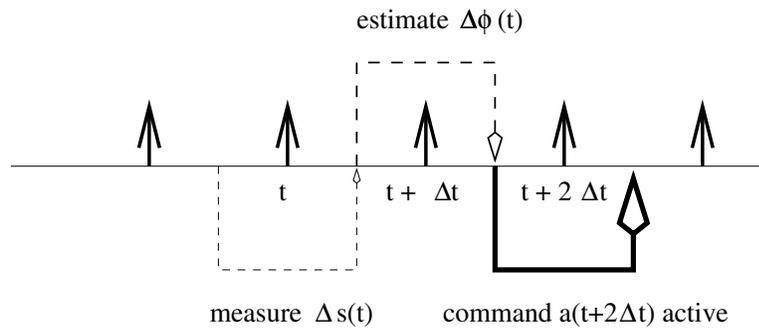
- ARMA control scheme:

$$a(t + 2\Delta t) \quad := \quad \sum_{k=0}^{p} c_k a(t + (1 - k)\Delta t)$$
$$+ \sum_{j=0}^{q} b_j M_j \Delta s(t - j\Delta t).$$

$$a^{(r+2)} = \sum_{k=0}^{p} c_k a^{(r+1-k)} + \sum_{j=0}^{q} b_j M_j \Delta s^{(r-j)}, r = 0, 1, \ldots$$

Expected Effect on the AO System

- Suppose

  ⋄ $\exp[s(t)]$ is independent of time t throughout the cycle of computation.

  ⋄ Matrix $\sum_{j=0}^{q} b_j M_j$ is of full column rank.

- Then

  ⋄ The WFS feedback measurement $\Delta s^{(n)}$ is eventually nullified by the actuators, i.e.,

  $$\exp[s] = G \lim_{n \to \infty} \exp[a^{(n)}].$$

  ⋄ The expected residual phase error is inversely related to the expected WFS measurement noise ϵ via

  $$0 = W \lim_{n \to \infty} \exp[\Delta \phi^{(n)}] + \exp[\epsilon].$$

- Compare with the ideal control:

  ⋄ Even if $\exp[\epsilon] = 0$, not necessarily $\exp[\|\lim_{n \to \infty} \Delta \phi_n\|^2]$ will be small because W has non-trivial null space.

Almost Sure Convergence

- Each control $a^{(r+j)}$ is a random variable $\implies$ The control scheme is a stochastic process.

- Each control $A^{(r+j)}$ is also a realization of the corresponding random variable $\implies$ The control scheme is a deterministic iteration.

- Convergence of deterministic iteration on independent random samples $\implies$ Almost sure convergence of stochastic process.

- Need fast convergence:

  $\diamond$ Stationary statistic is not realistic.

  $\diamond$ Atmospheric turbulence changes rapidly.

  $\diamond$ Can only assume stationary statistic for a short period of time.

- Define

$$\mathbf{a}_{r+2} \; := \; [a^{(r+2)}, a^{(r+1)}, \ldots a^{(r-q+1)}]^T, \quad r = 0, 1, \ldots$$

$$\mathbf{b} \; := \; [\sum_{j=0}^{q} b_j M_j G s', 0, \ldots, 0]^T.$$

- The ARMA scheme becomes

$$\mathbf{a}_{r+2} = A\mathbf{a}_{r+1} + \mathbf{b}$$

  where A is the $m(q+2) \times m(q+2)$ matrix

$$A := \begin{bmatrix} c_0 I_m & c_1 I_m - b_0 M_1 G & \ldots & & c_{q+1} I_m - b_q M_q G \\ I_m & 0 & & & 0 \\ 0 & I_m & & & \\ \vdots & \vdots & \ddots & & \vdots \\ 0 & 0 & \ldots & I_m & 0 \end{bmatrix}$$

- Almost convergence $\Longleftrightarrow$ Spectral radius $\rho(A)$ of A is less than one.

- Asymptotic convergence factor is precisely $\rho(A)$.

Numerical Simulation

- Consider the 2-cycle delay scheme

$$a(t + 2\Delta t) = a(t + \Delta t) + 0.6 H^\dagger W^\dagger \Delta s(t).$$

- Test data:

surface positions n	$=$	5
number of actuators m	$=$	4
number of subapertures ℓ	$=$	3
size of random samples z	$=$	2500
H	$=$	$rand(n, m)$
W	$=$	$rand(\ell, n)$
G	$=$	WH
L_ϕ	$=$	$rand(n, n)$
L_ϵ	$=$	$diag(rand(\ell, 1))$
μ_ϕ	$=$	$zeros(n, 1)$
μ_ϵ	$=$	$zeros(\ell, 1)$

- Random samples:

$$
\begin{aligned}
\phi &= \mu_\phi * ones(1, z) + L_\phi * randn(n, z), \\
\epsilon &= \mu_\epsilon * ones(1, z) + L_\epsilon * randn(\ell, z).
\end{aligned}
$$