Review of Matrix Algebra

Basic Properties of Matrices

* Equality: $A = B$, $a_{ij} = b_{ij}$ for all $i, j$.
* Matrix Addition & Subtraction
* Matrix Multiplication
* Null Matrix & Unit Matrix.

The Laws of Matrix Algebra

1. $A + B = B + A$; $A + (-B) = -B + A = A - B$
2. $A + (B + C) = (A + B) + C$; $\alpha(A + B) = \alpha A + \alpha B$
3. $\alpha A = A \alpha$; $A(BC) = (AB)C$
4. $(A + C) = AB + AC$; $(B + C)A = BA + BC$

$AB \neq BA$ - Multiplication not commutative.

Matrix Transpose, Conjugate & Associate Matrix

A symmetric matrix is one for which $A = A^T$.
If $A = -A^T$ then $A$ is skew-symmetric.

Important property: $(AB)^T = B^TA^T$; $(ABC)^T = C^TB^TA^T$

The conjugate of $A$ is $\bar{A}$, where each element is replaced by its complex conjugate.
For $A = A^T$, $A$ is called Hermitian.

For real matrices, symmetric and Hermitian matrices are the same thing.

The associated matrix of $A$ is the conjugate transpose of $A$.

Properties of Determinants

If $A$ and $B$ are both $n \times n$, then $|AB| = |A||B|$. $|A| = |A^T|.$

Matrix Inversion

If $B = A^{-1}$ then $A \cdot B = I$.
For inverse to exist, $A$ must be nonsingular.

$$(ABC \ldots W)^{-1} = W^{-1} \cdots C^{-1} B^{-1} A^{-1}$$

If $A^{-1} = A$, $A$ is involutory.
$A^{-1} = A^T$, $A$ is orthogonal.
$A^{-1} = A^T$, $A$ is unitary.

Matrix Differentiation and Integration

$A(t) = \begin{bmatrix} a_1(t) \\ a_2(t) \end{bmatrix}$

$$\frac{dA}{dt} = \dot{A} = \begin{bmatrix} \dot{a}_1(t) \\ \dot{a}_2(t) \end{bmatrix}$$

$$\int A(x) \, dx = \begin{bmatrix} \int a_1(x) \, dx \\ \int a_2(x) \, dx \end{bmatrix}$$
Linear Vector Spaces

Vector addition, subtraction & multiplication by a scalar:

\[ \mathbf{v}_1 + \mathbf{v}_2 \]
\[ \mathbf{v}_1 - \mathbf{v}_2 \]
\[ \alpha \mathbf{v} \]

Vector products:

Scalar Inner product:
\[ \langle \mathbf{v}, \mathbf{w} \rangle = \mathbf{v} \cdot \mathbf{w} = \mathbf{w} \cdot \mathbf{v} \]

Orthogonal vectors if \( \langle \mathbf{v}, \mathbf{w} \rangle = 0 \).
\[ ||\mathbf{v}|| = \langle \mathbf{v}, \mathbf{v} \rangle^{1/2} \] - magnitude.

Matrix Outer product:
\[ \mathbf{v} = [v_1, v_2, v_3]^T \] & \[ \mathbf{w} = [w_1, w_2, w_3]^T \]

\[ \mathbf{v} \times \mathbf{w} = \mathbf{v} \mathbf{w}^T = \begin{bmatrix}
  v_1 w_1 & v_1 w_2 & v_1 w_3 \\
  v_2 w_1 & v_2 w_2 & v_2 w_3 \\
  v_3 w_1 & v_3 w_2 & v_3 w_3
\end{bmatrix} \]

So, \( \mathbf{v} \mathbf{w}^T \neq \mathbf{w} \mathbf{v}^T \)

Vector Cross product:
\[ \mathbf{v} \times \mathbf{w} \] yields another vector perpendicular to the plane of \( \mathbf{v} \) & \( \mathbf{w} \).
\[ ||\mathbf{v} \times \mathbf{w}|| = \mathbf{v} \mathbf{w} \sin \theta. \]
Definition: Let a finite number of vectors \( \{ \mathbf{x}_i \} = \{ \mathbf{x}_1, \ldots, \mathbf{x}_n \} \) belong to a vector space. If there exists a set of \( n \) scalars, \( a_i \), at least one of which is not zero, which satisfies
\[ a_1 \mathbf{x}_1 + \cdots + a_n \mathbf{x}_n = \mathbf{0}, \]
then the vector \( \{ \mathbf{x}_i \} \) are linearly dependent.

Definition: If \( a_1 \mathbf{x}_1 + \cdots + a_n \mathbf{x}_n = \mathbf{0} \) implies that \( a_i = 0, i = 1, \ldots, n \), then \( \{ \mathbf{x}_i \} \) is a set of linearly independent vectors.

Tests for linear dependence:

\[ \mathbf{x}_1 = \begin{bmatrix} 5 \\ 2 \\ 3 \end{bmatrix}; \quad \mathbf{x}_2 = \begin{bmatrix} -1 \\ 7 \\ 4 \end{bmatrix}; \quad \mathbf{x}_3 = \begin{bmatrix} 14 \\ 50 \\ 36 \end{bmatrix} \]

\[ A = \begin{bmatrix} 5 & -1 & 14 \\ 2 & 7 & 60 \\ 3 & 4 & 36 \end{bmatrix} \]

\[ |A| = 5 \cdot 7 \cdot 36 - 2 \cdot 14 \cdot 36 + \cdots = 0 \Rightarrow \text{linearly dependent vectors.} \]

Not applicable if we don't have \( n \) vectors of \( n \) dimensions each.
Matrix Diagonalization

The intent here is to transform a state-space representation of the form

\[ \dot{x} = Ax + Bu \]

into

\[ \dot{\tilde{z}} = \Lambda \tilde{z} + Bu \]

where \( \Lambda \) is diagonal. So, we need to diagonalize matrix \( A \).

Assume the following transformation:

\[ x = T\tilde{z} \]

where \( T \) is a constant transformation matrix. Then,

\[ \dot{x} = \dot{T}\tilde{z} \]

Substituting these equations into the original state-space results into:

\[ \dot{T}\tilde{z} = AT\tilde{z} + Bu \implies \dot{\tilde{z}} = T^{-1}AT\tilde{z} + T^{-1}Bu \]

The corresponding output equation is

\[ y = CT\tilde{z} + Du \]

So,

\[ \dot{\tilde{z}} = \tilde{A}\tilde{z} + \tilde{B}u \]

\[ y = \tilde{C}\tilde{z} + \tilde{D}u \]

where \( \tilde{A} = T^{-1}AT, \tilde{B} = T^{-1}B, \tilde{C} = CT, \tilde{D} = D. \)
We can show that the original and the transformed system have the same eigenvalues.

For the transformed system
\[ |\lambda I - T^*A T| = 0 \Rightarrow \]
\[ |A^* T^* - T^*^* A| = 0 \Rightarrow \]
\[ |T^* (A^* - A) T| = 0 \Rightarrow \]
\[ |T|^{-1} \cdot |A| - |A| \cdot |T| = 0 \Rightarrow \]
\[ |T|^{-1} \cdot |T| \cdot |A - A| = 0 \]
\[ |T| = 1 \]
\[ \Rightarrow |\lambda I - A| = 0 \]

So, \[ |\lambda I - A| = 0 \]

When \( T \) is selected so that \( T^* A T \) is diagonal, then \( T \)
is called the modal matrix. This transformation is also called similarity transformation.

Special Operations in Vector Spaces

**Inner Product**

Any scalar valued function \( \langle x, y \rangle \) can be defined as the inner product, \( \langle x, y \rangle \), if

1. \( \langle x, y \rangle = \langle y, x \rangle \) (complex conjugate)
2. \( \langle x, \alpha y + \beta z \rangle = \alpha \langle x, y \rangle + \beta \langle x, z \rangle \) (linear, homogeneous)
3. \( \langle x, x \rangle \geq 0 \) for all \( x \), \( \langle x, x \rangle = 0 \Leftrightarrow x = 0 \).

**Vector Norm**

2-norm:
\[ ||x|| = \sqrt{\langle x, x \rangle} \]

Many other norms can be defined; the only requirements are that \( ||x|| \) be a non-negative real scalar satisfying...
(1) \(|x| = 0 \iff x = 0\)

(2) \(\|x\| = |x| \cdot \|x\|\) for any \(x \in \mathbb{R}\)

(3) \(\|x + y\| \leq \|x\| + \|y\|\) (triangle inequality)

Cauchy–Schwarz inequality

\[ |\langle x, y \rangle| \leq \|x\| \cdot \|y\|\]

Equality holds only if \(x\) and \(y\) are linearly dependent.

Unit Vectors

\[ x = \frac{x}{\|x\|}, \quad \|x\| = 1. \]

Metric or Distance Measure

Distance between two points \(x\) and \(y\)

\[ f(x, y) = \|x - y\| = \sqrt{\langle x - y, x - y \rangle} \]

Generalized Angles in \(n\)-dimensional space

\[ x \cdot y = \langle x, y \rangle = \|x\| \cdot \|y\| \cdot \cos \theta \]

\[ \cos \theta = \frac{1}{\|x\| \cdot \|y\|} \cdot \langle x, y \rangle = \langle x, y \rangle \]

Orthogonal vectors

Any two vectors \(x\) and \(y\) that belong to a linear vector space \(V\) are said to be orthogonal iff

\[ \langle x, y \rangle = 0 \]
Simultaneous Linear Equations

Consider

\[\begin{align*}
q_1 x_1 + \cdots + q_n x_n &= y_1 \\
q_1 x_1 + \cdots + q_n x_n &= y_2 \\
&\vdots \\
q_1 x_1 + \cdots + q_n x_n &= y_m
\end{align*}\]

where \( A \) is known as \( \vec{a} \) and the vector \( \vec{y} \).

Let's form the augmented matrix

\[W = [A \mid \vec{y}]\]

\( W \) indicates whether or not the system has a solution.

1. If \( r_w \neq r_A \), no solution exists (equations are inconsistent).
2. If \( r_w = r_A \), at least one solution exists.
   a. If \( r_\vec{w} = r_A = n \) \( \Rightarrow \) unique solution for \( \vec{x} \).
   b. If \( r_\vec{w} = r_A < n \) \( \Rightarrow \) infinite solutions.

So the possibilities are: No solutions, unique solution, or infinite solutions.

Solution by Partitioning

Assume that \( r_A = r_w \), so there is at least one solution.

By definition, the matrix \( A \) (\( m \times n \)) will contain a non-singular \( r \times r \) matrix.
The original equation $Ax = y$ can be rearranged into

$$
\begin{bmatrix}
A_1 & A_2 \\
-1 & -1 \\
A_3 & A_4
\end{bmatrix}
\begin{bmatrix}
x_1 \\
x_2 \\
x_3
\end{bmatrix}
= 
\begin{bmatrix}
y_1 \\
y_2 \\
y_3
\end{bmatrix}
$$

where $A_4$ is $r_4 \times r_4$ & nonsingular. In general,

$$A_1 x_1 + A_2 x_2 = y_1 \quad \text{or} \quad x_1 = A_1^{-1} [y_1 - A_2 x_2]$$

The $g$ degeneracy of $k$ is $g_A = n - r_4$. The values of the $g_A$ components of $x_2$ are completely arbitrary.

**Example**

$$
\begin{bmatrix}
1 & 0 & 1 \\
0 & 1 & 1 \\
1 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
x_1 \\
x_2 \\
x_3
\end{bmatrix}
= 
\begin{bmatrix}
3 \\
-1 \\
3
\end{bmatrix}
$$

Here $r_4 = 2$ & $r_w = 2$ also. $\Rightarrow$ infinite set of solutions.

Let, $A_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, $x_1 = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$, $y_1 = \begin{bmatrix} 3 \\ -1 \end{bmatrix}$

$A_2 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$, $x_2 = x_3$, $y_2 = 3$

Then,

$$x_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}^{-1} \left\{ \begin{bmatrix} 3 \\ -1 \end{bmatrix} - \begin{bmatrix} 1 \\ 1 \end{bmatrix} x_3 \right\} = \begin{bmatrix} 3 - x_3 \\ -1 - x_3 \end{bmatrix}$$

$\therefore$

$$x = \begin{bmatrix} 3 - x_3, -1 - x_3, x_3 \end{bmatrix}^T \text{ with } x_3 \text{ arbitrary.}$$

**Gram-Schmidt method of expansion.**
Homogeneous Linear Equations

Assume: $Ax = 0$; this has at least one solution, $x = 0$. This is because $A$ and $W$ have the same rank. $x = 0$ is called the trivial solution. For a non-trivial solution to exist we must have $r_A < n$. If this is the case then we have infinite non-trivial solutions with $n - r_A$ free parameters.

The non-homogeneous equation $Ax = y$ can always be written as an equivalent set of homogeneous equations:

$$
\begin{bmatrix}
A \\
y
\end{bmatrix}
\begin{bmatrix}
x \\
-1
\end{bmatrix} = 0 	ext{ or } W
\begin{bmatrix}
x \\
-1
\end{bmatrix} = 0.
$$

Underdetermined Case

When there are more equations than unknowns, the matrix $A$ has $m < n$.

When the matrix $A$ has $m < n$, there is no possibility of a unique solution $x$. Then we can find families of solutions, and further we can single-out some particular solutions, such as the minimum norm solution.

Overdetermined Case

When there are more equations than unknowns, the matrix $A$ has $m > n$. If the equations are inconsistent, no solution exists.
Usually, because of measurement errors and modeling errors associated with $x$, we would like to obtain approximate values of $x$.

There are approaches for this: (1) Ignore some equations, weight only those that remain, (2) least-squares, weight all equations equally, (3) weighted-least squares, weight some more than others.

1. Ignore some equations

Assume that $A_1$ is an $m \times n$ non-singular matrix formed by deleting rows from $A$ & $y$, is the $n \times 1$ vector obtained by deleting the corresponding elements. Then,

$$x_1 = A_1^{-1}y_1$$

satisfies one of the original $m$ equations. The remaining $m-n$ are ignored.

2. Least-Squares Approximate Solution

In this case no one $x$ can satisfy the equations. So,

$$e = y - Ax$$

The least-squares approach minimizes some form of $e$.

Let's choose $e$ to minimize

$$\|e\|^2 = e^Te = (y - Ax)^T(y - Ax)$$
The solution to the set of equations is
\[ x = A^+ y \]
where \( A^+ = (A^T A)^{-1} A^T \) (pseudo-inverse).

This is an approximate solution, where the approximation error is given by
\[ \| y - A x \|^2 = y^T [ I - R (A^T A)^{-1} A^T ] y \]

If \( A^{-1} \) exists then \( A^+ = A^{-1} \) and \( \| e \|^2 = 0 \).

(3) **Weighted Least-Squares**

We could try to minimize a weighted sum-squared of the error, for example,
\[ e^T R^{-1} e = (y - A x)^T R^{-1} (y - A x) \]

where \( R \) is the weighting matrix, usually diagonal.

Then,
\[ x = (A^T R^{-1} A)^{-1} A^T R^{-1} y \]

These results can be derived by setting \( \frac{\partial}{\partial x_i} \| e \|^2 = 0 \).
Recursive Weighted Least Squares

Up to this point we dealt with 'batch least squares' that is all equations are tested simultaneously. A recursive method of using each equation (or data) as becomes available is also very common, especially in real-time applications.

Assume a set of \( m \) equations

\[
Y_k = Ax + e
\]

and compute the weighted least-squares \( x_k \) as

\[
x_k = (A^T R^{-1} A)^{-1} A^T R^{-1} Y_k
\]

Now, assume an additional set of relations

\[
Y_{k+1} = H_{k+1} x + e_{k+1}
\]

It is desired to obtain a new estimate for \( x_{k+1} \) which combining both data sets of minimizes

\[
J = e^T e_{k+1}
\]

\[
= e^T e_{k+1} \left[ \begin{array}{c}
R^{-1} \quad 0 \\
0 \quad R_{k+1}^{-1}
\end{array} \right] \left[ \begin{array}{c}
e_{k+1} \\
\end{array} \right]
\]

\[
= e_{k+1}^T R_{k+1}^{-1} e_{k+1}
\]

(weighting matrices)

It can be shown that

\[
x_{k+1} = x_k + K_k \left[ Y_{k+1} - H_{k+1} x_k \right]
\]
where \( K_k = P_k H_k^T \left[ H_k P_k H_k + R_k \right]^{-1} \)

\( P_k = (A^T R^{-1} A)^{-1} \)

If we obtain additional measurements, we can use the above relation recursively. But now we need to define a new matrix \( P_{k+1} \) as

\[ P_{k+1} = \left[ P_k^{-1} + H_k R_k^{-1} H_k^T \right]^{-1} \]

for use in

\[ K_{k+1} = P_{k+1} H_{k+2}^T \left[ H_{k+2} P_{k+1} H_{k+2}^T + R_{k+2} \right]^{-1} \]
Eigenvectors & Eigenvalues

Definitions: Let \( A \) be a linear transformation. The non-zero elements \( x_i \neq 0 \) in the particular solutions \( x_i \) that satisfy

\[
A(x_i) = \lambda_i x_i
\]

are called eigenvectors & eigenvalues of \( A \), respectively.

If \( A \) is an \( m \times n \) matrix, then

\[
(\lambda - \lambda_i) x_i = 0
\]

& the determination of the eigenvectors \( x_i \) find the non-trivial solution of this set of homogeneous equations.

To find the eigenvalues we must require that

\[
\text{det}(A - \lambda I) = 0.
\]

When the determinant is expanded it yields an \( n \)th degree polynomial in \( \lambda_i \):

\[
|\lambda - \lambda I| = (\lambda - \lambda_1)(\lambda - \lambda_2) \cdots (\lambda - \lambda_n) = \Delta(\lambda)
\]

The roots of the equation are the eigenvalues \( \lambda_i \). -

There are no restrictions, so an \( m \times n \) matrix has \( n \) eigenvalues.

\[
\Delta(\lambda) = (-1)^n(\lambda - \lambda_1)(\lambda - \lambda_2) \cdots (\lambda - \lambda_n) = 0
\]

In general, some of the roots might be repeated. If there are \( p < n \) distinct roots, \( \Delta(\lambda) \) becomes
\[\Delta(A) = (-1)^n (a_{11} - a_{12})^{m_1} (a_{21} - a_{22})^{m_2} \cdots (a_{n1} - a_{n2})^{m_n}\]
\[m_1 + m_2 + \cdots + m_n = n\]

**Properties:**

1. If the coeff. of the \(\Delta(A)\) are real, then if \(\lambda_i\) is a complex eigenvalue, so is \(\overline{\lambda_i}\).

2. \(\text{Tr}(A) = \lambda_1 + \lambda_2 + \cdots + \lambda_n = (-1)^{n-1} \Delta_{n-1}\)

3. \(|A| = \lambda_1 \lambda_2 \cdots \lambda_n = c_0\)

**Finding Eigenvalues**

Case I: All eigenvalues are distinct.

Case II: Some eigenvalues are multiple roots of the characteristic eq.

Not to be covered here!

**Uses of Eigenvalues & Eigenvectors**

1. Existence of solution for sets of linear equations.

Existence of solutions for homogeneous equations & of a unique solution for non-homogeneous equations was seen to depend on whether or not the coefficient matrix \(A\) has a zero determinant. This can be translated to whether or not the zero is an eigenvalue of the problem at hand.
(2) Stability of Linear Differential & Difference Equations.

It can be shown that the system poles are equal to the eigenvalues of the system matrix $A$. The stability of a system can therefore be determined by examining the system eigenvalues.
Functions of Square Matrices & Cayley-Hamilton

Theorem

In this development we only consider square matrices. All the usual rules of exponents apply to square matrices. That is,

\[ A^n A^m = \underbrace{(A \cdots A)}_{m \text{ factors}} \underbrace{(A \cdots A)}_{n \text{ factors}} = A^{n+m} \]

\[ (A^n)^m = \underbrace{(A \cdots A)}_{m \text{ factors}} \underbrace{(A \cdots A)}_{n \text{ factors}} = A^{mn} \]

If \( A \) is non-singular then \( (A^{-1})^n = A^{-n} \), &

\[ A^{-n} A^n = A^n (A^{-1})^n = (AA^n)^n = I^n = I \text{ or } A^n A^{-n} = A^{-n} A^n = A^n = I. \]

Matrix polynomial:

\[ P(A) = c_m A^m + c_{m-1} A^{m-1} + \cdots + c_1 A + c_0 I. \]

As for scalar polynomials, also for matrix polynomials we can write:

\[ P(A) = c(A - \lambda_1) (A - \lambda_2) \cdots (A - \lambda_k) \]

Characteristic Polynomial & Cayley-Hamilton

If the characteristic poly of a matrix \( A \) is written as

\[ |A - \lambda I| = (\lambda^n + c_{n-1} \lambda^{n-1} + \cdots + c_1 \lambda + c_0) = \Delta(A) \]

then the corresponding matrix polynomial is

\[ \Delta(A) = (-1)^n A^n + c_{n-1} A^{n-1} + \cdots + c_1 A + c_0 I. \]
**Cayley-Hamilton Theorem:** Every matrix satisfies its own characteristic equation; that is, $\Delta(A) = [0]$.

**Uses of the Cayley-Hamilton Theorem**

**Matrix Inversion**

$A; \text{n x n}$

$\Delta(A) = (-1)^n + c_{n-1} A^{n-1} + \cdots + c_1 A + c_0 = 0$

Recall that $c_0 = \lambda_1 \lambda_2 \cdots \lambda_n = |A|$ so it's zero iff $A$ is singular.

We know that

$\Delta(A) = (-1)^n A^n + c_{n-1} A^{n-1} + \cdots + c_1 A + c_0 I = 0$

Assume $A^{-1}$ exists,

$(-1)^n A^{-1} + c_{n-1} A^{-2} + \cdots + c_1 I + c_0 A^{-1} = 0$

$\Rightarrow$

$A^{-1} = -\frac{1}{c_0} \left[ (-1)^n A^{n-1} + c_{n-1} A^{n-2} + \cdots + c_1 I \right]$

Many other uses of this theorem.
Solution of Unforced State Equations

Continuous-Time Case

Set the control to zero,
\[ \dot{x}(t) = A \dot{x}(t) \quad ; \quad x(0) = x_0 \]

Traditional solutions deal with the *d.e.* of the equation. Another method is w/ model decoupling. Assume that $A$ can be diagonalized, i.e., write the equation as,

\[ \dot{x}(t) = M A M^{-1} \dot{x}(t). \]

Then we can write the solution $x(t)$ as:

\[ x(t) = M \text{Diag}[e^{\lambda_1 t}, e^{\lambda_2 t}, \ldots, e^{\lambda_n t}] M^{-1} x(0). \]

Or

\[ x(t) = M e^{A^t M^{-1}} x(0). \]

where

\[ e^{A^t} = M e^{A^t M^{-1}} \]

The same procedure can be used for the discrete-time case where, $x(k+1) = Ax(k)$. 
Analysis of Continuous-Time Linear Systems

Start from

\[ \dot{x}(t) = a(t) x(t) + b(t) u(t) \] if set \( u(t) = 0 \).

Then,

\[ \frac{dx}{dt} = a(t) x(t) \quad \Rightarrow \quad \frac{dx}{x} = a(t) dt \]

\[ \Rightarrow \int_{x(t_0)}^{x(t)} \frac{dx}{x} = \int_{t_0}^{t} a(t) dt \]

or

\[ \ln x(t) - \ln x(t_0) = \int_{t_0}^{t} a(t) dt \]

\[ \Rightarrow \frac{x(t)}{x(t_0)} = e^{\int_{t_0}^{t} a(t) dt} \quad \Rightarrow \quad x(t) = x(t_0) e^{\int_{t_0}^{t} a(t) dt} \]

When, \( a(t) = a = \text{constant} \),

\[ x(t) = x(t_0) e^{(t-t_0)a} \]

When, we include the non-homogeneous term,

\[ x(t) = e^{(t-t_0)a} x(t_0) + \int_{t_0}^{t} e^{(t-t) a} b(t) u(t) dt \]

The same results can be obtained if we generalize to the multi-dimensional state case.
For a
\[ \dot{x}(t) = A \cdot x(t); \quad x(t_0) \text{ given} \text{ & } A \text{ constant} \]
\[ x(t) = e^{(t-t_0)A} x(t_0) \]

For the nonhomogeneous case:
\[ \dot{x}(t) = A \cdot x(t) + B \cdot u(t) \quad \text{w/ } x(t_0) \text{ given.} \]
\[ x(t) = e^{(t-t_0)A} x(t_0) + \int_{t_0}^{t} e^{(t-s)A} B(s) u(s) \, ds \]

The Transition Matrix

If the matrix \( A \) is constant then,
\[ \Phi(t, \tau) = e^{(t-\tau)A} \]

Provided it is such that
\[ \frac{d}{dt} \Phi(t, \tau) = A(t) \Phi(t, \tau) \]

Note time dependency.

Obtaining the state transition matrix:
\[ \Phi(t, \tau) = e^{(t-t_0)A} \]
\[ \{ \Phi(t, \tau) \text{ is found by replacing } t \text{ by } t-\tau. \} \]
Discrete-time models of continuous-time systems

\[ x(t) = A x(t) + B u(t) \]

The solution of \( x(t) \) for any interval \([t_n, t_{n+1}]\) can be obtained by using \( x(t_n) \) as an I.C.:

\[ x(t_{n+1}) = \phi(t_{n+1}, t_n) x(t_n) + \int_{t_n}^{t_{n+1}} \phi(t, t_n) B(t) u(t) dt \]

where \( \phi(t, t_n) = e^{AT} \), \( T = t_{n+1} - t_n \)

S\text{\textsuperscript{s}}: commonly we use:

\[ x(k+1) = A_1 x(k) + B_1 u(k) \]

where \( A_1 = e^{AT} \) and \( B_1 = \int e^{A(t_{n+1} - t)} dt B \)

Depends on the sampling rate.
5. Starting from the discrete-time state-space we proceed as follows:

\[ x(k+1) = Ax(k) \quad \text{w/} \quad x(0) \text{ known.} \]

\[
\begin{align*}
\hat{x}(1) &= A \hat{x}(0) \\
\hat{x}(2) &= A \hat{x}(1) = A^2 \hat{x}(0) \\
&\vdots \\
\hat{x}(k) &= A^k \hat{x}(0)
\end{align*}
\]

For the non-homogeneous problem we assume we know \( x(0) \) and \( u(0), y(0), y(1), \ldots \). Then,

\[
\begin{align*}
\hat{x}(1) &= A \hat{x}(0) + B(0) y(0) \\
\hat{x}(2) &= A^2 \hat{x}(0) + B(1) y(1) = A^2 \hat{x}(0) + A B(0) u(0) + B(1) y(1) \\
&\vdots \\
\hat{x}(k) &= A^k \hat{x}(0) + \sum_{j=0}^{k-1} A^{k-1-j} B(j) u(j)
\end{align*}
\]

In general,

\[ \hat{x}(k) = A^k \hat{x}(0) + \sum_{j=0}^{k-1} A^{k-1-j} B(j) u(j) \]

Shift the indices:

\[ \hat{x}(k) = A^k \hat{x}(0) + \sum_{j=1}^{k} A^{k-j} B(j-1) u(j-1) \]

The output equation is expressed as

\[ y(k) = C \hat{x}(k) \]
**Stability**

In the state-space, an equilibrium point for a continuous-time system is a point at which \( \dot{x} = 0 \) in the absence of all external inputs or disturbances. For discrete-time systems, we say \( x(k+1) = x(k) \).

- Asymptotic stability
- BIBO stability

**Linear System Stability**

This is studied in terms of the eigenvalues of \( A \) of the state-space form; eigenvalues of the form \( \lambda = \beta \pm j\omega \):

\[
\dot{x} = Ax
\]

**Unstable**

- If \( \beta > 0 \) for any simple roots
- \( \beta > 0 \) for repeated roots

**Stable** (or **stable**)

- \( \beta < 0 \) for all simple roots
- \( \beta < 0 \) for all repeated roots

**Unstable**

- \( |\lambda| > 1 \) for simple roots
- \( |\lambda| > 1 \) for repeated roots

**Stable**

- \( |\lambda| < 1 \) simple root
- \( |\lambda| < 1 \) repeated roots
| Asymptotic Stability | bi:co for all roots | i:le: for all roots | 50 |
Towards the MIMO Poles, Zeros, Modes, Stabilizability and Detectability

For SISO systems:

\[
\dot{x}(t) = A x(t) + B u(t) \\
y(t) = C x(t) + D u(t)
\]

and \( x(s) = L(x(t)) \), then,

\[
y(s) = \left[ C (sI-A)^{-1} B + D \right] u(s)
\]

For MIMO systems:

\[
\dot{x}(t) = A x(t) + B u(t) \\
y(t) = C x(t) + D u(t)
\]

\[
y(s) = \left[ C (sI-A)^{-1} B + D \right] u(s)
\]

Concept: Define "poles" and "zeros" in the time-domain. Poles and zeros of \((A, B, C, D)\).

The purpose of this section is to present the key ideas behind the definition of the transmission zeros for a multivariable linear time-invariant system starting from its state-space description.
Before we start the discussion of multivariable zeros, it is helpful to define the generalized eigenvalue problem. As it will be seen later, the solution to this problem provides an algorithm for calculating the transmission zeros.

The Ordinary Eigenvalue Problem

Let $A$ be an $mxn$ matrix. Then the eigenvalues $\lambda_i$, right eigenvectors $x_i$, and the left eigenvectors $y_i$ of $A$ are defined by the ordinary eigenvalue problem,

$$(s_i \cdot I - A) \cdot x_i = 0; \quad i = 1, \ldots, n$$

$$y_i^T (s_i \cdot I - A) = 0; \quad i = 1, \ldots, n$$

Clearly, the eigenvalues $s_i$ are the $m$ roots of the polynomial,

$$\det(s_i \cdot I - A) = 0$$
**Uniformed System:**

\[ x(t) = A \cdot x(t), \quad x(0) = \xi \]

then,

\[ x(t) = e^{-At} \xi \]

\[ e^{At} = \sum_{k=0}^{\infty} \frac{1}{k!} (At)^k \]

\[ \phi(\dot{x}(t)) = s \cdot \phi(s) - \xi \]

\[ s \cdot \phi(s) - \xi = A \cdot \phi(s) \implies \phi(s) = (sI - A)^{-1} \xi \]

\[ \therefore \phi(e^{At}) = (sI - A)^{-1} = \Phi(s) \]

---

**FACT:**

\[ e^{At} = \sum_{i=1}^{n} e^{A_i t} \cdot \vec{x}_i \cdot \vec{y}_i^H \]

\[ A = \sum_{i=1}^{n} \Lambda_i \cdot \vec{x}_i \cdot \vec{y}_i^H \]

So,

\[ \phi(e^{At}) = (sI - A)^{-1} = \sum_{i=1}^{n} \phi(e^{A_i t}) \cdot \vec{x}_i \cdot \vec{y}_i^H = \]

\[ = \sum_{i=1}^{n} \frac{1}{s - \Lambda_i} \cdot \vec{x}_i \cdot \vec{y}_i^H \]
Observability, Detectability

Consider the following system:

\[
\dot{x}(t) = Ax(t)
\]

\[
y(t) = C \cdot x(t)
\]

\[
C = \begin{bmatrix}
C_1 \\
C_2 \\
\vdots \\
C_m
\end{bmatrix}
\]

\[
y_k(t) = C_k \cdot x(t).
\]

\[
x(t) = e^{At} \cdot \xi = \sum_{i=1}^{\infty} e^{\lambda_i t} \cdot \xi_i \cdot (\phi_i \cdot e^{\lambda_i t})
\]

\[
\therefore \quad \dot{y}_k(t) = \sum_{i=1}^{\infty} \lambda_i e^{\lambda_i t} \cdot (C_k \cdot \xi_i) \cdot (\phi_i \cdot e^{\lambda_i t})
\]

\[(C_k \cdot \xi_i) \text{ is the degree that the } i^{th} \text{ mode shows up in the } k^{th} \text{ output (sensor).}\]

If \((C_k \cdot \xi_i) = 0\) then the \(i^{th}\) mode is unobservable in the \(k^{th}\) output.

If \((C_k \cdot \xi_i) = 0\) then, \(k = 1, \ldots, m\)

or,

\[C \cdot \phi_i = 0\text{ then, the } i^{th} \text{ mode is unobservable.}\]

and therefore the pair \([A, C]\) is unobservable.

or, \(\text{rank } [C, A \cdot C', \ldots, A^{m-1} \cdot C'] < m\)

If all of the unobservable modes are stable,
then \([A, C]\) is detectable.

Now, rewrite \(x(s)\) and \(y(s)\).

\[
x(s) = \sum_{i=1}^{n} \left( \frac{1}{s-\lambda_i} \right) A_i \left( \frac{y_i^H}{y_i^H \cdot B} \right) + \left( \frac{1}{s-\lambda_i} \right) B_i + (s)
\]

\(0\) if the \(i\)th mode is not controllable.

\[
y(s) = \sum_{i=1}^{n} \left( \frac{1}{s-\lambda_i} \right) (y_i \cdot F) \left( C \cdot X_i \right) + \sum_{i=1}^{n} \left( \frac{1}{s-\lambda_i} \right) (C \cdot X_i \cdot y_i^H \cdot B) + (s)
\]

\(0\) if the \(i\)th mode is not observable.

**MIMO Residue Expansion**

\[
y(s) = (G(s) - H(s)) \cdot F = 0
\]

\[
G(s) = \left\{ \sum_{i=1}^{n} \left( \frac{1}{s-\lambda_i} \right) C X_i \cdot y_i^H \cdot B \right\} + D
\]

\(R_i = C X_i \cdot y_i^H \cdot B = i\)th residue matrix at the pole \( s = \lambda_i \)

If the \(i\)th mode is either uncontrollable or unobservable or both then \( R_i = 0 \).
Poles of \((A, B, C, D)\):

Poles are the eigenvalues of \(A\)

\[
x(t) = e^{At} = \left[ \sum_{i=1}^{n} e^{\lambda_i t} X_i \right] Y^H
\]

\((Y^H)\) is the degree that the initial condition excites the \(i\)-th mode \(e^{\lambda_i t} X_i\)

Stable mode: \(\lambda_i < 0\)
Unstable mode: \(\lambda_i > 0\)

Stabilizability, Controllability

Let's consider the forced system,

\[
\dot{x} = Ax + Bu, \quad x(0) = 0, \quad x \in \mathbb{R}^n, \quad u \in \mathbb{R}^m
\]

\[
y = Cx + Du
\]

\[
\dot{x} = \sum_{j=1}^{m} b_j u_j(t)
\]

\[
x(t) = \int_0^t e^{A(t-\tau)} B u(\tau) d\tau
\]

\[
B_{j}(t) \sum_{j=1}^{m} b_j u_j(t)
\]

\[
e^{A(t-t)} = \sum_{i=1}^{n} e^{\lambda_i(t-t)} X_i Y^H
\]

Therefore,
\[ x(t) = \sum_{i=1}^{n} \sum_{j=1}^{m} x_i(y_j^H b_j^*) \int_{0}^{t} e^{A_i(t-\tau)} u_j(\tau) d\tau \]

The scalar \((y_j^H b_j^*)\) can be interpreted as the degree that the \(j\)th control influences the \(i\)th mode.

If \((y_j^H b_j^*)\) is zero then the \(i\)th mode is uncontrollable from the \(j\)th input.

If any one of the modes is uncontrollable from all inputs,

\[ (y_j^H b_j^*) = \ldots \quad (y_j^H b_n^*) = 0 \quad \text{then} \]

\((A, B)\) is uncontrollable

\[ \text{rank } [B, AB, \ldots, A^{n-1}B] < m. \]

**Definition:** If all uncontrollable modes are stable then \((A, B)\) is stabilizable.

\[ x(s) = \sum_{i=1}^{n} \sum_{j=1}^{m} x_i(y_j^H b_j) \frac{1}{s - \lambda_i} u_j(s). \]

\[ x(s) = \sum_{i=1}^{n} \left( \frac{1}{s - \lambda_i} \right) x_i \frac{y_i^H B}{S} \mu(s) \quad \text{and} \]

\[ y(s) = \sum_{i=1}^{n} \frac{1}{s - \lambda_i} (C x_i)(y_i^H B) \mu(s) + D \mu(s). \]

**MIMO Residue Expansion:**
Canonical State-Space Representations

We know that there are infinite representations of a transfer function in state-space form. Some specific representations are more convenient to deal with than others. In particular, there are 4 representations, called canonical forms, that are widely used in control systems synthesis.

There are:
(1) Controller canonical form
(2) Observer
(3) Observer Canonical Form
(4) Observer

\[ G(s) = \frac{b_0 s^n + b_1 s^{n-1} + \ldots + b_n}{s^n + a_1 s^{n-1} + \ldots + a_n s + a_0} \]

(1) Controller Canonical Form

\[ x(t) = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ -a_0 & -a_1 & \cdots & -a_{n-1} & 1 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix} u(t) \]

\[ y(t) = \begin{bmatrix} b_0 & b_1 & \cdots & b_{n-1} \end{bmatrix} x(t) \]
(2) Observability Can. Form

\[
\dot{x}(t) = \begin{bmatrix}
\text{Same as in (1)}
\end{bmatrix} x(t) + \begin{bmatrix}
b_{-1} \\
\vdots \\
b_1 \\
b_0
\end{bmatrix} u(t)
\]

\[y(t) = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix} x(t)\]

(3) Observer Can. Form

\[
\dot{x}(t) = \begin{bmatrix}
-a_1 & 1 & 0 & \vdots \\
\vdots & \ddots & \ddots & \vdots \\
0 & \cdots & -a_1 & 1 \\
-b_0 & \cdots & 0 & -a_1
\end{bmatrix} x(t) + \begin{bmatrix}
b_{-1} \\
\vdots \\
b_1 \\
b_0
\end{bmatrix} u(t)
\]

\[y(t) = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix} x(t)\]

(4) Controllability Can. Form

\[
\dot{x}(t) = \begin{bmatrix}
\text{Same as (3)}
\end{bmatrix} x(t) + \begin{bmatrix}
0 \\
\vdots \\
0
\end{bmatrix}
\]

\[y(t) = \begin{bmatrix} b_0 & \cdots & b_{m-1} \end{bmatrix} x(t)\]

These 4 are also known as companion forms.
**Transformation to Companion Forms**

Assume a physical state space:

\[ x_p(t) = A_p x_p(t) + b_p u(t) \]

We wish to transform it to the phase variable:

\[ \dot{x}(t) = A_c x(t) + b_c u(t) \]

where the companion form \( A_c, b_c \) is given by:

\[
A_c = \begin{bmatrix}
1 & 1 & \cdots & 0 \\
0 & \ddots & & \vdots \\
& \ddots & \ddots & \ddots \\
-\alpha_0 & -\alpha_1 & \cdots & -\alpha_{n-2} & -\alpha_{n-1}
\end{bmatrix}
\]

\[
b_c = \begin{bmatrix}
0 \\
0 \\
\vdots \\
1
\end{bmatrix}
\]

This is the controller canonical form.

Assume \( x_p(t) = T x(t) \), then we can compute the relation:

\[
A_c = T A_p T^{-1}
\]

\[
b_c = T^{-1} b_p
\]

The characeristic polynomial is invariant to similarity transformations:

\[
Q(\lambda) = |\lambda I - A_p| = \lambda^n - a_{n-1} \lambda^{n-1} + \cdots + a_1 \lambda + a_0
\]
We can form the controllability matrix for the place-variable form as:

\[ M_{ec} = [b_e A^0 b_e \ldots A^{-1} b_e] \]

It is only in terms of the coefficient of the char equation.

\[ M_{ec} = [T^{-1} b_p T^{-1} A_p b_p \ldots T^{-1} A^{-1} A_p b_p] \]

\[ = T^{-1} M_{ep} \]

where \( M_{ep} = [b_p A_p b_p \ldots A^{-1} A_p b_p] \)

So,

\[ T = M_{ep} M_{ec}^{-1} \]
ME651

Canonical State-Space Representations

These notes should help alleviate some of the confusion I have created today in class.

Let me start with the following transfer function:

\[ G(s) = \frac{b_m s^m + b_{m-1} s^{m-1} + \ldots + b_1 s + b_0}{a_n s^n + a_{n-1} s^{n-1} + \ldots + a_1 s + a_0} \]

with \( a_n \neq 0 \), and assume that \( m = n \). We know that \( m \leq n \), so if \( m < n \) some of the \( b_i \)'s will be zero if you can substitute the zeros at the end.

This is the most general form of the TF we can start with. Now, with \( m = n \), \( G(s) \) can be rewritten as:

\[ G(s) = \frac{b_n + \frac{c_n s^{n-1} + c_{n-2} s^{n-2} + \ldots + c_1 s + c_0}{a_n}}{a_n s^n + a_{n-1} s^{n-1} + \ldots + a_1 s + a_0} \]

where \( c_i = b_i - \frac{b_n}{a_n} a_i \) for \( i = 0, \ldots, n-1 \)

Further,

\[ G(s) = \frac{b_n}{a_n} + \frac{\frac{c_n}{a_n} s^{n-1} + \frac{c_{n-2}}{a_n} s^{n-2} + \ldots + \frac{c_1}{a_n} s + \frac{c_0}{a_n}}{s^n + \frac{a_{n-1}}{a_n} s^{n-1} + \ldots + \frac{a_1}{a_n} s + \frac{a_0}{a_n}} \]
Now define,

\[ d = \frac{b_n}{a_n} \quad \text{and} \quad C_i = \frac{c_i}{a_n} \quad i = 0, \ldots, n-1 \]

\[ \tilde{a}_i = \frac{a_i}{a_n} \quad i = 0, \ldots, n-1 \]

Then \( G(s) \) becomes

\[
G(s) = d + \frac{c_{n-1} s^{n-1} + \ldots + c_1 s + c_0}{s^n + \tilde{a}_{n-1} s^{n-1} + \ldots + \tilde{a}_1 s + \tilde{a}_0}
\]

Now we can construct the controller canonical form as follows:

Now the equations for the controller canonical form can be written as:
\[
\begin{bmatrix}
X(t) & \dot{X}(t) + B u(t)
\end{bmatrix}
\]

\[
y(t) = \begin{bmatrix}
\tilde{c}_0 & \tilde{c}_1 & \cdots & \tilde{c}_{n-1}
\end{bmatrix}
\begin{bmatrix}
x(t)
\end{bmatrix}
+ \frac{d}{D} u(t)
\]

The observability canonical form is

\[
\begin{bmatrix}
\dot{x}_2(t) & \text{Same as Controller Canonical Form}
\end{bmatrix}
\begin{bmatrix}
x_2(t) + \begin{bmatrix}
\tilde{c}_{n-1}
\end{bmatrix}
\end{bmatrix}
\]

\[
y(t) = \begin{bmatrix}
1 & 0 & \cdots & 0
\end{bmatrix}
\begin{bmatrix}
x_2(t)
\end{bmatrix}
+ d u(t)
\]

The observer canonical form is

\[
\begin{bmatrix}
\dot{x}_3(t) & \text{\begin{bmatrix}
\tilde{c}_{n-1}
\end{bmatrix}}
\end{bmatrix}
\begin{bmatrix}
x_3(t) + \begin{bmatrix}
\tilde{c}_0
\end{bmatrix}
\end{bmatrix}
\]

\[
y(t) = \begin{bmatrix}
1 & 0 & \cdots & 0
\end{bmatrix}
\begin{bmatrix}
x_3(t)
\end{bmatrix}
+ d u(t)
\]
The controllability canonical form is

\[
\dot{x}_r(t) = \begin{bmatrix}
    \text{Same as the Observer canonical form}
\end{bmatrix} \begin{bmatrix} x_r(t) \\ \end{bmatrix} + \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} k(t)
\]

\[
y(t) = \begin{bmatrix} \tilde{c}_0 & \tilde{c}_1 & \cdots & \tilde{c}_n \end{bmatrix} \begin{bmatrix} x_r(t) \\ \end{bmatrix} + d u(t)
\]

**Example** Find the four canonical forms for

\[
G(s) = \frac{9s^2 + 35s + 1}{4s^2 + 25s + 6}
\]

Here \( n = 2 \), and \( b_2 = 2, b_1 = 3, b_0 = 1; a_2 = \frac{1}{2}, a_1 = 2, a_0 = 6 \).

Also \( d = \frac{b_2}{a_2} = \frac{2}{4} = \frac{1}{2} \).

Then, \( c_0 = b_0 - d \cdot a_0 = 1 - \frac{1}{2} \cdot 6 = -2 \)

\( c_1 = b_1 - d \cdot a_1 = 3 - \frac{1}{2} \cdot 2 = \frac{5}{2} \)

\( \tilde{c}_0 = c_0 / a_2 = -\frac{1}{2} \); \( \tilde{c}_0 = \frac{a_0}{a_2} = \frac{6}{4} = \frac{3}{2} \)

\( \tilde{c}_1 = c_1 / a_2 = \frac{5}{4} = \frac{5}{2} \); \( q_1 = \frac{a_1}{a_2} = \frac{2}{4} = \frac{1}{2} \)

So, we can write \( G(s) \) as,

\[
G(s) = \frac{1}{2} + \frac{\frac{1}{2} s - \frac{1}{2}}{s^2 + \frac{1}{2} s + \frac{3}{2}}
\]
Controller canonical form:

\[
\begin{bmatrix}
\dot{x}_1 \\
\dot{x}_2
\end{bmatrix} =
\begin{bmatrix}
0 & 1 \\
-\frac{3}{2} & -\frac{1}{2}
\end{bmatrix}
\begin{bmatrix}
x_1 \\
x_2
\end{bmatrix}
+ \begin{bmatrix}
0 \\
1
\end{bmatrix} u
\]

\[
y = \begin{bmatrix}
-\frac{1}{2} & \frac{1}{2}
\end{bmatrix}
\begin{bmatrix}
x_1 \\
x_2
\end{bmatrix}
+ \frac{1}{2} u
\]

Observability canonical form:

\[
\begin{bmatrix}
\dot{x}_1^* \\
\dot{x}_2^*
\end{bmatrix} =
\begin{bmatrix}
0 & 1 \\
-\frac{3}{2} & -\frac{1}{2}
\end{bmatrix}
\begin{bmatrix}
x_1^* \\
x_2^*
\end{bmatrix}
+ \begin{bmatrix}
\frac{1}{2} \\
-\frac{1}{2}
\end{bmatrix} u
\]

\[
y = \begin{bmatrix}
1 & 0
\end{bmatrix}
\begin{bmatrix}
x_1^* \\
x_2^*
\end{bmatrix}
+ \frac{1}{2} u
\]

Observer canonical form:

\[
\begin{bmatrix}
\dot{x}_1 \\
\dot{x}_2
\end{bmatrix} =
\begin{bmatrix}
-\frac{1}{2} & 1 \\
-\frac{3}{2} & 0
\end{bmatrix}
\begin{bmatrix}
x_1^* \\
x_2^*
\end{bmatrix}
+ \begin{bmatrix}
\frac{1}{2} \\
-\frac{1}{2}
\end{bmatrix} u
\]

\[
y = \begin{bmatrix}
1 & 0
\end{bmatrix}
\begin{bmatrix}
x_1^* \\
x_2^*
\end{bmatrix}
+ \frac{1}{2} u
\]

Controllability Canonical Form:

\[
\begin{bmatrix}
\dot{x}_1^* \\
\dot{x}_2^*
\end{bmatrix} =
\begin{bmatrix}
-\frac{1}{2} & 1 \\
-\frac{3}{2} & 0
\end{bmatrix}
\begin{bmatrix}
x_1^* \\
x_2^*
\end{bmatrix}
+ \begin{bmatrix}
0 \\
1
\end{bmatrix} u
\]

\[
y = \begin{bmatrix}
-\frac{1}{2} & \frac{1}{2}
\end{bmatrix}
\begin{bmatrix}
x_1^* \\
x_2^*
\end{bmatrix}
+ \frac{1}{2} u
\]
**State Variables & Control System Design**

Assume the following state-space representation:

\[
\dot{x}(t) = Ax(t) + Bu(t)
\]
\[
y(t) = Cx(t) + Du(t)
\]

and its discrete-time equivalent,

\[
x(k+1) = Ax(k) + Bu(k)
\]
\[
y(k) = Cx(k) + Du(k)
\]

It is assumed that we can not alter \(A, B, C, D\) in order to control the system.

There are two commonly used control system structures:

1. Full-state variable feedback
2. Output feedback.

---

![Block diagram showing control system configurations.](attachment:control_system_diagram.png)

**Full-state feedback control**

or

**Output feedback control**
Even though state variable feedback is an idealization, i.e. we almost never have full-state information, it is useful to study it because

1. The state contains all needed info. about a system; so it is good to know what is the best that can be done.

2. There are some instances in which $C = I$, i.e. all state can be measured.

3. Many optimal control laws are expressed in this form.

4. There are effective means to reconstruct or estimate the state from input/output measurements.

The equations for state feedback are:

$$y(k) = Fv(k) - Kx(k) \quad \text{or} \quad u(k) = Fv(k) - Kx(k)$$

The closed-loop equations are:

$$\dot{x}(k) = [A - BK]x(k) + [BF]v(k) \quad \text{or} \quad \dot{x}(k) = [A - BK]x(k) + [BF]v(k)$$

$$y(k) = [C - DK]x(k) + [DF]v(k) \quad \text{or} \quad y(k) = [C - DK]x(k) + [DF]v(k)$$
So, now the question becomes how can we pick $K$ and $F$ such that the closed-loop has desirable properties?

1. Stability depends on $[A - BK]$
2. Controllability depends on $[A + BK, B F]$
3. Observability depends on $[A - BK, C - DK]$

For output feedback system

$u(t) = F_x - K y(t)$

$y(t) = C x + D u - C x + D F y(t)$

$\Rightarrow [I + D K] y(t) = C x + D F y(t)$

$y(t) = [I + D K]^{-1} [C x + D F y(t)]$

The state dynamics can be expressed as:

$\dot{x}(t) = \{ A - BK \} \{ I + DK^{-1} C \} x(t) + B \{ I + K D \}^{-1} F y(t)$

Effect of Feedback on System Properties

**Stability**

Feedback impacts the pole location; open-loop & closed-loop poles will not be the same. So, by appropriately selecting the control gain matrix, we can place the closed-loop poles at appropriate locations.
PROPERTIES OF STATE VARIABLE

FEEDBACK

Assume,

\[ \dot{x}(t) = A x(t) + B u(t) \]
\[ y(t) = C x(t) \]  \hspace{1cm} (1)

\( x \in \mathbb{R}^m, \ u \in \mathbb{R}^n, \ y \in \mathbb{R}^p \). The system (1) represents the open-loop dynamics. In the frequency domain (1) is described as:

\[ x(s) = (sI - A)^{-1} B u(s) \]
\[ y(s) = C (sI - A)^{-1} B u(s) \]  \hspace{1cm} (2)

It is useful to think of the physical meaning of state variables as those variables whose knowledge at each instant of time characterizes the energy stored in the system; physical state variables are inductor currents, e.g., voltages, positions, accelerations, forces, pressures, etc., since they define energy. If such physical state variables are directly available for measurement, one can adjust the controls (which absorb or introduce energy to the system) as a function of the state variables so as to make the system outputs have desirable time-response characteristics.
If we use linear state feedback, given by
\[ U(t) = -C_x(t) + H_c(t) \]  (3)

where \( C \) is a constant \( m \times n \) gain matrix. In general \( H_c(t) \) is designed for good command following & disturbance rejection.

The closed-loop system is described as follows:
\[
\begin{align*}
\dot{x}(t) &= [A - BG] x(t) + B H_c(t) \\
y(t) &= C x(t)
\end{align*}
\] (4)

in the frequency domain:
\[
\begin{align*}
x(s) &= \left(sI - A + BG\right)^{-1} B H_c(s) \\
y(s) &= C \left(sI - A + BG\right)^{-1} B H_c(s)
\end{align*}
\] (5)

Clearly the eigenvalues \( \lambda_i(A) \); \( i = 1, \ldots, n \) are the open-loop poles & the eigenvalues \( \lambda_i(A - BG) \); \( i = 1, \ldots, n \) are the closed-loop poles.
The following facts are known from linear theory:

**Theorem 1**: If \([A, B]\) is controllable then there exists at least one gain matrix \(G\) so that the closed-loop poles \(\lambda_i(A-BG)\) can be arbitrarily placed in the \(s\)-plane, of course complex closed-loop poles must be in complex conjugate pairs.

The implication of this theorem is that by appropriately selecting \(G\) we can always get a stable closed-loop system. This is true even if all of the poles of \(A\) are on the RHP.

Another property of state feedback is that it does not add or change the location of the open-loop zeros.

**Theorem 2**: The zeros of the open-loop transfer matrix \(G_{OL}(s)\) defined by:

\[
G_{OL}(s) = C(sI - A)B
\]

are identical to the zeros of the closed-loop transfer matrix \(1\) defined by:

\[
G_{CL}(s) = C(sI - A + BG)B
\]
Interpretation of the State Variable Feedback Method as Compensation

We can visualize the effects of state feedback as defining the LIM for a standard MIMO feedback configuration.

\[ \frac{M_{c}(s)}{M(s)} \xrightarrow{\text{feedback}} \sqrt{(sI-A)^{-1}}B \sqrt{x(s)} \rightarrow G \]

If we define the LIM, \( G_{\text{loop}}(s) \) by:

\[ G_{\text{loop}}(s) = G(sI-A)^{-1}B \]

Then,

\[ \frac{M_{c}(s)}{M(s)} \xrightarrow{\text{feedback}} G_{\text{loop}}(s) \]

Now we can use all the machinery we built for robustness and stability using the SV's.

It is important to notice that use of state variables feedback does not provide dynamic compensation. The importance of this remark will become obvious when we try to shape the loop dynamics augmenting the open-loop dynamics \((sI-A)B\).
with additional dynamics allows for superior command input tracking and disturbance rejection.

The main value of state variable feedback is that it stabilizes a causal system. However, the remaining measures of goodness are still not answered.

Even though we know that we can stabilize a system, we still don't know how to pick good G's. That's why we try the IQ design method.
Pole Assignment Using State Feedback

The eigenvalues of the closed-loop system are the roots of

$$\Delta(\lambda) = |\lambda I - A + BK| = 0$$

It can be proven that if the open-loop system is $[A, B]$ completely controllable, then we can place the closed-loop poles $\{z_1, \ldots, z_n\}$ to any desired location using full-state feedback.

**Problem:** What should be the closed-loop pole locations?

**What if the system is not completely controllable?**

**Ex.**

$$A = \begin{bmatrix} 0 & 2 \\ 0 & 3 \end{bmatrix}; \quad B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

Assume we want closed-loop poles at $z = -3$ and $-4$.

This means the characteristic poly. that we want is:

$$(\lambda + 3)(\lambda + 4) = \lambda^2 + 7\lambda + 12$$

$$|\lambda I - A + BK| = \begin{vmatrix} \lambda & 0 \\ 0 & \lambda \end{vmatrix} - \begin{vmatrix} 0 & 2 \\ 0 & 3 \end{vmatrix} + \begin{vmatrix} k_1 & 0 \\ k_1 & k_2 \end{vmatrix} =$$

$$= \begin{vmatrix} \lambda - 2 \\ k_1 & \lambda - 3k_2 \end{vmatrix} = \lambda(\lambda - 3 + k_2) + 2k_1 = \lambda^2 - 3\lambda + k_2\lambda + 2k_1 = \lambda^2 + (k_2 - 3)\lambda + 2k_1$$

So,

$$k_2 - 3 = 7 \Rightarrow k_2 = 10 \quad k = [6 \quad 10]$$

$$2k_1 = 12 \quad \Rightarrow k_1 = 6$$
This method works well for higher order problems, as long as the state space is in the controllable canonical form.

\[
\dot{x} = \begin{bmatrix}
  0 & 1 & 0 & \cdots & 0 \\
  0 & 0 & 1 & \cdots & 0 \\
  \vdots & \vdots & \ddots & \ddots & \vdots \\
  0 & 0 & \cdots & 0 & 1 \\
  -q_0 & -q_1 & \cdots & -q_{n-1} & 0
\end{bmatrix} x + \begin{bmatrix}
  0 \\
  0 \\
  \vdots \\
  0 \\
  1
\end{bmatrix} u(t)
\]

Because of the special structure of \( A \) & \( B \) the closed-loop determinant can be written so

\[
\begin{vmatrix}
  2 & -1 & 0 & \cdots & 0 \\
  0 & 2 & -1 & \cdots & 0 \\
  \vdots & \vdots & \ddots & \ddots & \vdots \\
  0 & \vdots & \ddots & 2 & -1 \\
  k_1 + a_0 & k_2 + a_1 & k_3 + a_2 & \cdots & a_0 + k_n + a_{n-1}
\end{vmatrix} = (\lambda + k_n + a_{n-1})^n + \cdots + (k_2 + a_1)\lambda + (k_1 + a_0)
\]

Assuming the desired closed-loop characteristic polynomial is

\[
\lambda^n + c_{n-1}\lambda^{n-1} + \cdots + c_1\lambda + c_0
\]

we can equate the two:

\[
k_0 = c_{n-1} - a_{n-1}
\]

If the state space is not in the controllable form we can transform it using similarity transformations, as long as we have a single input only.

Let \( x \) be the vector of any representation. Then there exist \( T \) such that \( \dot{x}^* = T\dot{x} \) is in the controllable canonical representation.
Then the \( \{A, B\} \) matrices will be transformed to,

\[
A' = TAT^{-1} \quad \& \\
B' = TB
\]

& the resulting gain will be \( K'x' = Kx \)

using \( K'Tx = Kx \) results into \( K = K'T \)

Example: Prior example, \( A + B \).

\[
A = \begin{bmatrix} 0 & 2 \\ 0 & 3 \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}
\]

\[
T = \begin{bmatrix} \frac{1}{2} & 0 \\ 0 & 1 \end{bmatrix}
\]

transforms this to controllable canonical form.

\[
A' = TAT^{-1} = \begin{bmatrix} 0 & 1 \\ 0 & 3 \end{bmatrix} \quad \& \quad B'T = \begin{bmatrix} 0 \\ 1 \end{bmatrix}
\]

So, if we want \( A^2 + 7A + 12 \)

\[
K_2' = (-3) = 7 \quad \Rightarrow \quad K_2' = 10
\]

\[
K_1' + \frac{Q}{Q_0} = 12 \quad \Rightarrow \quad K_1' = 12
\]

\[
[K_1, K_2] = \begin{bmatrix} \frac{1}{2} & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 12 & 10 \\ 10 & 10 \end{bmatrix} = \begin{bmatrix} 6 & 10 \\ 10 & 10 \end{bmatrix} \] same as before!

This transformation method will not work if we have multiple input systems.
Now, let's develop a method that is applicable to any general system with full-state feedback.

We know that $|\lambda I - A + BK| = 0$, which implies that there is at least one non-zero vector $\Psi_i$ such that

$$(\lambda_i I - A + BK)\Psi_i = 0 \Rightarrow (A - BK)\Psi_i = \lambda_i \Psi_i$$

polar eigenvector of $A - BK$

Also, rewrite this equation as

$$(\lambda_i I - A)\Psi_i = -BK\Psi_i \quad \text{or}$$

$$(\lambda_i I - A) \begin{bmatrix} \Psi_i \\ K\Psi_i \end{bmatrix} = 0$$

has both $\Psi_i$ and $K$ as unknowns.

Set $z_i = \begin{bmatrix} \Psi_i \\ K\Psi_i \end{bmatrix}$

Finding the $K$ consists of two steps. First, a sufficient # of solution vectors $z_i$ is found. Then, we use the definition of $\Psi_i$ to find $K$.

Example: $A = \begin{bmatrix} 0 & 2 \\ 0 & 3 \end{bmatrix}$, $B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$, $\lambda_1 = -3$, $\lambda_2 = -4$

$$(\lambda_i I - A) \begin{bmatrix} \Psi_i \\ z_i \end{bmatrix} = 0 \Rightarrow$$

$\begin{bmatrix} -3 & -2 & 0 \\ 0 & 6 & 1 \end{bmatrix} \begin{bmatrix} \Psi_i \\ s_1 \\ z_i \end{bmatrix} = 0$; Arbitrarily select $s_2 = 1$ gives

$s_1 = -\frac{3}{2}$, $s_3 = 6$
So,  \( x_1 = \begin{bmatrix} -\frac{2}{3} & 1 & 6 \end{bmatrix}^T \), this implies

\[ K \begin{bmatrix} -\frac{2}{3} \\ 1 \end{bmatrix} = 6 \]

Similarly, for \( x_2 = -4 \)

\[ x_2 = \begin{bmatrix} -\frac{1}{2} & 1 & 7 \end{bmatrix}^T \Rightarrow K \begin{bmatrix} -\frac{1}{2} \\ 1 \end{bmatrix} = 7 \]

So,

\[ K \begin{bmatrix} -\frac{2}{3} & -\frac{1}{2} \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 6 & 7 \end{bmatrix} \]

And \( K = \begin{bmatrix} 6 & 10 \end{bmatrix} \)

So we can place the poles of the closed-loop \( A - BK \) anywhere we wish as long as \( [A, B] \) is controllable. If it is only stabilizable, then we can only find \( K \) that does stabilize the closed-loop.

If instead of full state feedback we have output feedback i.e.,

\[ u = -Kx \]

then the static control cannot place all closed-loop poles at independent locations. Worse of all, we cannot even show that we can stabilize the closed-loop system with a static output feedback.
Observers - Reconstruction of a State from Available Outputs

Observer - An estimator for the state.
Full state observer - All states estimated.
Reduced state observer - Only states that are not measured or estimated. The others are measured.

Here we are considering a deterministic world, i.e., no noise!

Given $[A, B, C, D]$, the input & output $x(t), y(t)$ obtain an estimate $\hat{x}(t)$ of the state.
This is the state reconstruction problem.

If $C$ is square & nonsingular $\hat{x}(t) = C^{-1}y(t)$.

We can construct another dynamic system called the observer of inputs $\bar{x}, u$ & output an estimate of $x(t)$:

$$\dot{\hat{x}}(t) = A_c \hat{x}(t) + L \bar{x}(t) + \xi(t)$$

where $\hat{x}(t)$ is the state estimate. $A_c, L, \xi$ will be selected such as to give good estimates.
Define the estimation error as:

\[ e = \hat{x} - x \]  

Then

\[ \dot{e} = \dot{\hat{x}} - \dot{x} = A\hat{x} - A\dot{x} - By + Bu - e \]

Select \( z = Bu \) & use \( y = Cx \) then,

\[ \dot{z} = (A - LC)x - A\dot{\hat{x}} \]

Selecting \( A_c = A - LC \) reduces to

\[ \dot{e} = A_c e \]

If the eigenvalues of \( A_c \) are stable then \( e(t) \to 0 \) as \( t \to \infty \), or \( \hat{x}(t) \to x(t) \) as \( t \to \infty \).

We still don't know how to pick \( L \) but we can force the eigenvalues of \( A - LC = A_c \) to have specific values, i.e., pole placement.

The controllability requirement is replaced by the observability requirement of \([A\; C]\) (or detectability).
Reduced order Observer

Assume that we selected the state equations such that

\[ C = [I_m, 0] \]

and write the top state dynamics as:

\[
\dot{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}
\]

\[ A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}, \quad B = \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} \]

The states \( x_1(t) \) are treated as known because \( x_1(t) = y(t) \). The states \( \hat{x}_1(t) \) can be expressed as

\[
\dot{\hat{x}}_1(t) = A_{11} \hat{x}_1(t) + \{ \underbrace{A_{12} x_2(t) + B_2 u(t)}_{\text{Known}} \}
\]

The top half of the dynamics are:

\[
\dot{x}_1(t) = A_{11} x_1(t) + B_1 u(t) + A_{12} x_2(t)
\]

\[ \rightarrow \dot{x}_1(t) - \underbrace{A_{11} x_1(t) - B_1 u(t)}_{\text{Known}} = A_{12} \hat{x}_2(t) \]

So, the reduced order observer becomes:

\[
\hat{x}_1 = x_1 = y
\]

\[
\dot{\hat{x}}_2 = A_r \hat{x}_2 + L \gamma + e_r
\]

where

\[
A_r = A_{22} - L \gamma A_{12} \quad \hat{x}_r = \dot{x}_1 - A_{11} \hat{x}_1 - B_1 u \quad e_r = A_{21} x_1 + B_2 u
\]
Separation Principle

One of the major uses of observers is in feedback control, when full-state measurements are not available.

The closed-loop system is of 2n order if we need to determine \( K \) and \( L \).

The separation principle says that we can design the \( K \), or if we had all states available, the a separate observer can be designed, to provide the desired observer response.

Let's demonstrate this:

\[
\begin{align*}
x &= Ax + Bu \\
\dot{x} &= A_c \hat{x} + LCx + Bu \\
u &= Fr - K\hat{x}
\end{align*}
\]
Combine this into:

\[
\begin{bmatrix}
\dot{X} \\
\hat{X}
\end{bmatrix} =
\begin{bmatrix}
A & -BK \\
LC & A_c - BK
\end{bmatrix}
\begin{bmatrix}
X \\
\hat{X}
\end{bmatrix} +
\begin{bmatrix}
BF \\
BF
\end{bmatrix}
\]

The 2n-closed loop eigenvalues are the roots of

\[
\begin{vmatrix}
I_n \lambda - A & BK \\
-LC & I_n \lambda - A_c + BK
\end{vmatrix} = 0
\]

Row and column operations:

\[
\begin{vmatrix}
I_n \lambda - A & BK \\
-I_n \lambda + A - LC & I_n \lambda - A_c
\end{vmatrix}
\equiv
\begin{vmatrix}
I_n \lambda - A + BK & BK \\
0 & I_n \lambda - A_c
\end{vmatrix}
\]

\[
= \left| I_n \lambda - A + BK \right| \left| I_n \lambda - A_c \right| = 0
\]

Separate design:

given that \( \{A, B, C\} \) are completely controllable and observable (or at least stabilizable and detectable)
THE LINEAR QUADRATIC REGULATOR (LQR) PROBLEM

In this note we summarize the solution and properties of the Linear Quadratic Regulator problem.

The style followed here is to expose the properties of the LQR problem with respect to

1. Stability of the Closed Loop Control System
2. Solution to a Dynamic Optimal Control Problem
3. Inherent Robustness Properties in terms of Singular Values of the resultant Return Difference Matrix through LQR compensation.

1. The Open Loop System

In the time-domain the open-loop system is described in state variable form as follows:

\[
\dot{x}(t) = A x(t) + B u(t) \\
y(t) = C x(t)
\]  

(1)

with \(x(t) \in \mathbb{R}^n\), \(u(t) \in \mathbb{R}^m\), \(y(t) \in \mathbb{R}^p\).

2. Assumptions

We make the following assumptions

Assumption 1. The entire state vector \(x(t)\) is available for feedback.
Assumption 2. The system (1) is stabilizable. Recall that a system is stabilizable if all of its unstable modes are controllable. We also remark that if \([A, B]\) is controllable, then the system (1) is stabilizable; thus controllability is a stronger assumption than stabilizability.

Assumption 3. The system (1) is detectable. Recall that a system is detectable if all of its unstable modes are observable. We also remark that if \([A, C]\) is observable, then the system (1) is detectable; thus observability is a stronger assumption than detectability.

3. The Algebraic Riccati Equation (ARE)

Let \(R\) be an arbitrary symmetric positive definite \(m \times m\) matrix, i.e.

\[
R = R' > 0; \quad R^{-1} \text{ exists}
\]  

Then consider the following matrix ARE

\[
0 = -KA - A'K - C'C + KBR^{-1}B'K
\]  

Equation (3) may be frightening at first but it is nothing more than a shorthand for a bunch of complex algebraic quadratic equations in the elements \(K_{ij}\) of the \(n \times n\) matrix \(K\). Think of \(A, B, C\) and \(R\) as being the coefficients and that the matrix \(K\) is being the "unknown".

The ARE has several important properties, that we outline below.

Property 1: The solution matrices are symmetric, i.e.

\[
K = K'
\]  

This can be readily verified by taking the transpose of both sides of Eq (3).

Property 2. In general, there is a very large number of matrices \(K\) that are solutions to the ARE. However, if \([A, B]\) is controllable and \([A, C]\) is observable, then there exists one and only one solution matrix \(K\) to the ARE that is positive definite, i.e.

\[
K = K' > 0
\]
If we relax the controllability assumption to that of stabilizability, and the observability assumption to that of detectability, then there exist one and only one positive semidefinite solution matrix to the ARE, i.e.

$$K = K^t > 0$$  \hspace{1cm} (6)

**Remark:** Given $A$, $B$, $C$, and $R$ one needs a computer to find the positive (semi) definite solution matrix $K$ to the ARE which is important for designing stable control systems. Modern software* find the desired solution matrix directly; they do not find all solution matrices and check for condition (5) or (6)!

Henceforth, when we refer to the solution matrix $K$ to the ARE we shall imply the unique positive (semi) definite one.

**The LQ Control Gain Matrix**

Once we are given $A$, $B$, $C$ and $R$ then we can calculate the solution matrix $K$ of the ARE. On the basis of this calculation we define the LQ control gain matrix $G$ as follows:

$$G = R^{-1}B'K$$  \hspace{1cm} (7)

Clearly $G$ is a constant $m \times n$ matrix.

**The LQR Control Law**

The LQR control law utilizes full state variable feedback for the system (1) using the control gain $G$ calculated according to Eq. (7). Thus, the control law is

$$u(t) = -G \ x(t)$$  \hspace{1cm} (8)

The LQR Closed Loop System

Substituting Eq (8) into Eq (1) we readily deduce that the closed-loop system evolves according to the state equations

\[ \dot{x}(t) = (A-BG)x(t) \]

\[ y(t) = Cx(t) \]  \hspace{1cm} (9)

The LQR design has several very important properties which we shall summarize in the sequel. It is important to stress that the properties of LQR designs hinge upon

(a) the fact that full-state feedback is used  
(b) the specific way that the control gain matrix \( G \) is computed via the solution matrix \( K \) of the ARE.

Furthermore, it is important to stress that these properties hold

(a) for any order system, where the order of the system, \( n \), is the dimensionality of the state vector  
(b) for any number of controls, \( m \), and outputs, \( p \),  
(c) for any matrices \( A, B, C \), (modulo the stabilizability and detectability assumptions) and \( R = R' > 0 \).

Thus the open-loop system may have several unstable open-loop poles and several nonminimum phase zeros.

Property 1

Guaranteed Stability of the LQR Closed Loop System

The poles of the closed-loop system (9) are strictly in the left half of the \( s \)-plane and hence the closed-loop system is asymptotically stable, i.e.

\[ \Re \lambda_i [A-BG] < 0 \]  \hspace{1cm} (10)
Property 2: Optimality

The LQR control law (8) generates the minimum possible value of the quadratic cost functional $J$ given by

$$J = \int_0^\infty [y'(t)y(t) + u'(t)Ru(t)]dt$$

subject to the dynamic constraints imposed by the open-loop dynamics (1).

Property 3: Excellent Robustness Properties

We have seen that any state variable feedback compensation scheme generates a specific loop transfer matrix. Since we have computed the LQR control gain matrix in a specific way, we are going to get a specific loop transfer matrix for any given $A$, $B$, $C$, and $R$. We can represent the closed-loop system given by Eq. (9) as shown in Fig. 1.

$$\frac{U_c}{\omega} = \frac{U_c(s)}{\omega}$$

$$G(s) = G(sI-A)^{-1}B$$

where the loop transfer matrix, denoted by $G_{LQ}(s)$, induced by the LQR design scheme is given by

$$G_{LQ}(s) = G(sI-A)^{-1}B$$

Recall that the multivariable robustness properties of any design depend on the size of

$$\sigma_{\min}[I+G(j\omega)]$$

and

$$\sigma_{\min}[I+G(j\omega)]$$

Now we make the following additional assumption:
Assumption 4. The matrix $R = R^r > 0$ is also diagonal.

Under the above assumption we have the following guaranteed LQR robustness properties

\[
\begin{align*}
\sigma_{\min}(I + G_{LQ}(j\omega)) & \geq 1 \quad \text{for all } \omega \\
\sigma_{\min}(I + G_{LQ}^{-1}(j\omega)) & \geq \frac{1}{2} \quad \text{for all } \omega
\end{align*}
\]  

(14)  

(15)

From the inequalities (14) and (15) we can deduce the following guaranteed multivariable gain and phase margin properties of any LQR design:

\[
\begin{align*}
(1) & \quad \text{Upward Gain Margin is infinite} \\
(2) & \quad \text{Downward Gain Margin is at least } \frac{1}{2} \quad \text{(or -6db)} \\
(3) & \quad \text{Phase Margin is at least } +60\%.
\end{align*}
\]

We remind the reader that the gain margins $[\frac{1}{2}, \infty]$ can occur independently and simultaneously in all $m$ control channels. Similarly, the phase margins $[+60^\circ, -60^\circ]$ can occur simultaneously and independently in all on control channels.

These inherent robustness properties of LQR design are quite good for many applications. To appreciate what they mean let us consider a SISO system

\[
\begin{align*}
\dot{x}(t) &= A x(t) + b u(t) \\
y(t) &= c'x(t)
\end{align*}
\]  

(16)

and let $R$ be a positive scalar. Then the LQ control gain is simply an $n$-vector, $g$, given by

\[
g = \frac{1}{R} b'k
\]

(17)

the control law is

\[
u(t) = -g'x(t)
\]

(18)
Fig 2. Every SISO LQR design leads to a loop transfer function that avoids the unit circle about the critical point in the Nyquist diagram.
and the resulting LQ loop transfer function

\[ g_{LQ}(s) = g'(sI-A)^{-1}b \]

(19)

We can now visualize the robustness properties in the ordinary Nyquist diagram by plotting \( g_{LQ}(j\omega) \) for all \( \omega \). The visualization of the inherent robustness properties is shown in Fig. 2; the Nyquist locus \( g_{LQ}(j\omega) \) is guaranteed not to get inside the unit circle centered at the \((-1,0)\) point. This is why the LQR obtains the good gain and phase margins!

Property 4: LQR Rolloff Characteristics

For SISO LQR designs Fig. 2 shows that as \( \omega \to \infty \) the phase of \( g_{LQ}(j\omega) \) cannot be less than \(-90^\circ\). This implies that the high frequency behavior of \( g_{LQ}(j\omega) \) is of the form

\[ \lim_{\omega \to \infty} |g_{LQ}(j\omega)| \sim \frac{1}{\omega} \]

(20)

so that SISO LQR designs exhibit "an one-pole rolloff" behavior. This property is also true for MIMO designs in the sense that

\[ \lim_{\omega \to \infty} |g_{LQ}(j\omega)| \sim \frac{1}{\omega} \]

(21)

Strictly speaking, this violates the Bode-Horowitz condition that all physical systems must exhibit at least a 2-pole rolloff behavior. The reason that we get an one-pole rolloff characteristic in LQR designs is due to our assumption that we have measured all state variables and we can feed them back. If we have unmodeled dynamics (i.e. states that have not been included in the state space description (1)) then all the properties that we have discussed above are not necessarily true nor guaranteed.
If the unmodeled dynamics are high-frequency ones, and if the LQR design leads to a stable closed-loop system, then we would actually get at least a 2-pole rolloff. In SISO designs, in reference to Fig. 2, this means that the actual $g_{LQ}(j\omega)$ will penetrate the unit disk about the $(-1,0)$ point at high frequencies; this in turn will imply that real LQR designs will not have the infinite upward gain margin property.

The above remarks are not meant to degrade the power of LQ-based designs. At this point they are intended to serve as a warning to the reader that mathematical results, derived on the basis of certain assumptions, should be carefully scrutinized from a pragmatic control system point of view. A good control system designer should be intimately familiar with both the strengths and weaknesses of a computer-aided design procedure. Fortunately, as we shall see, we can "tune" LQ-based designs so that their shortcomings in practical designs can be tolerated. After all, demanding an upward gain margin does not make much engineering sense. Control system designers should be very happy with upward gain margins of the order tens of db's; otherwise, they must admit that they are truly sloppy modellers!!!

In closing, we remind the reader that LQR-based designs are special cases of general full state variable feedback designs; consequently they have the following general properties.

(a) LQR designs do not introduce extra dynamics in the loop transfer matrix.

(b) LQR designs do not change the location of the open-loop transmission zeros nor do they create any new ones.

* This can be done by controlling the gain crossover frequency in LQR designs; we shall elaborate at length upon this point in subsequent notes.
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
Department of Electrical Engineering and Computer Science
Cambridge MA 02139

6.232 MULTIVARIABLE CONTROL SYSTEMS
SPRING 1985

VARIANTS OF THE LINEAR QUADRATIC
REGULATOR (LQR) PROBLEM

Written by
Prof. Michael Athans
March 18 1985

REF. NO. 850318/6232

All Right Reserved
VARIANTS OF THE LINEAR QUADRATIC REGULATOR (LQR) PROBLEM

0. Summary

1. The objective of the lecture note is to summarize some useful variants of the LQR problem which often turn out to be useful.

1. The first problem deals with the optimal control of an LTI system with respect to a quadratic performance criterion which contains cross-penalty terms in the state and control vectors.

2. The second problem formulation deals with adding an exponential time weighting term in the standard quadratic criterion. As we shall see one can select a design parameter so that the LQR closed-loop system has a preassigned degree of stability.
3. The third problem is a stochastic version of the LQR problem. In this formulation we allow the state dynamics to include the effects of white process noise. As we shall see, the deterministic structure of the LQR problem remains optimal in the stochastic case provided that optimality is suitably defined.

1. LQR Problem with Cross Penalty

1.1 Problem Formulation

Consider an LTI system with state equation

\[ \dot{x}(t) = A x(t) + B u(t) \]

and the following quadratic cost functional

\[ J = \int_0^\infty \left[ x'(t) Q x(t) + 2 x'(t) S u(t) + u'(t) R u(t) \right] dt \]

The standard LQR problem was solved under the assumption \( S = 0 \).

In the problem formulation we make the
following assumptions:

\[ Q = Q' \succ 0 \]  \hspace{1cm} (1.2) \\
\[ R = R' \succ 0 \]  \hspace{1cm} (1.3) \\
[A, B] \text{ is stabilizable} \hspace{1cm} (1.4) \\
[A, Q'^{1/2}] \text{ is detectable} \hspace{1cm} (1.5)

\[
\begin{bmatrix}
Q & S \\
S' & R
\end{bmatrix} \succ 0 
\]  \hspace{1cm} (1.6)

We remark that assumptions (1.2) to (1.5) are identical to those in the standard LQR problem.

The additional assumption (1.6) simply guarantees that the integrand of the cost function (1.1) is non-negative.

1.2 Motivation

Let us examine two problems that give rise to the problem formulation above.

Problem A

Consider an LTI system with the property that
the control \( u(t) \) feeds through to the output \( y(t) \), i.e.

\[
\begin{align*}
\dot{x}(t) &= A x(t) + B u(t) \\
y(t) &= C x(t) + D u(t) \quad ; \quad D \neq 0
\end{align*}
\] (1.7)

Consider the following cost functional

\[
J = \int_0^\infty \left[ y'(t) y(t) + u'(t) R u(t) \right] dt
\] (1.8)

Substituting the output equation into the cost functional (1.8) we obtain

\[
J = \int_0^\infty \left[ x'(t) C' C x(t) + 2 x'(t) C' D u(t) \\
+ u'(t) \left[ D' D + R \right] u(t) \right] dt
\] (1.9)

which clearly has the structure of the cost functional (1.1)

\textbf{Problem B}

\( Q \) Sometimes we may wish to penalize the derivatives \( \dot{x}_i(t) \) of some (or all) state variables in the quadratic cost functional.
This will make the regulator more sluggish and the trick is used to create a more "damped" closed-loop response. So let us suppose that the state dynamics are given by

\[ \dot{x}(t) = Ax(t) + Bu(t) \]  \hspace{1cm} (1.10)

and the cost functional has the form

\[ J = \int_{0}^{\infty} \left[ x'(t)x(t) + u'(t)Ru(t) \right] dt \]  \hspace{1cm} (1.11)

Substituting eq. (1.10) into the cost functional (1.1), we obtain

\[ J = \int_{0}^{\infty} \left\{ x'(t)A'Ax(t) + 2x'(t)A'Bu(t) \\
+ u'(t) \left[ B'B + R \right] u(t) \right\} dt \]  \hspace{1cm} (1.12)

Once more we see that the structure of the cost functional (1.12) exhibits the cross terms of the cost functional (1.1).
1.3 Problem Solution

We now summarize the solution of the optimal control problem posed in Section 1.1. We remark that the solution can be derived by variational techniques very similar to those used in the standard LQR problem. The proof is left as an exercise for the reader.

The optimal control can be implemented by full-state feedback and is given by

\[ u(t) = -G_x(t) \]  \hspace{1cm} (1.13)

The control gain matrix \( G \) is given by

\[ G = R^{-1} \left[ S' + B'K \right] \]  \hspace{1cm} (1.14)

The matrix \( K = K' \geq 0 \) is computed from the solution of the following algebraic Riccati equation

\[ 0 = -KA - A'K - Q + [KB + S']R^{-1}[B'K + S''] \]  \hspace{1cm} (1.15)
1.4 Discussion

The resultant closed-loop regulator is asymptotically stable, i.e.

\[ \text{Re} \lambda_i [A-BG] < 0 \] \hspace{1cm} (1.16)

However, the robustness results derived for the standard LQ regulator are not necessarily true.

2. LQR Problem With Exponential Penalties

2.1 Problem Formulation

We now consider the following variant of the standard LQR problem. The state dynamics are given by

\[ \dot{x}(t) = A x(t) + B u(t) \] \hspace{1cm} (2.1)

and the cost functional is given by

\[ J = \int_0^\infty e^{-2\alpha t} \left\{ x'(t) Q x(t) + u'(t) R u(t) \right\} dt \] \hspace{1cm} (2.2)

with

\[ \alpha > 0 \] \hspace{1cm} (2.3)
The remaining assumptions are those for the standard LQR problem (see eqs. (1.2) to (1.5)).

Some thought on the part of the reader should suggest what is the impact of the exponential weighting in the cost functional (2.2). In order for the cost to be minimum, \( J \) has to be finite. Since \( x \geq 0 \), this implies that the quadratic integrand must decrease faster than \( \exp \{-2 \alpha t\} \), i.e.

\[
x'(t)Qx(t) + u'(t)Ru(t) \leq M e^{-2 \alpha t} \quad (2.4)
\]

where \( M \) is some positive constant. Hence, we should suspect that the closed-loop poles cannot be in arbitrary locations in the left half of the \( s \)-plane. The precise properties of the closed-loop poles will be discussed in the sequel.
2.2 Problem Solution

We now summarize the solution to the optimal control problem defined in Section 2.1.

The optimal control can be realized by LTI state variable feedback and is of the form

\[ u(t) = -Gx(t) \]  \hspace{1cm} (2.5)

The control gain matrix \( G \) is given by

\[ G = R^{-1}B'K \]  \hspace{1cm} (2.6)

The matrix \( K = K' > 0 \) is the unique solution of the following algebraic Riccati equation

\[ 0 = -K(A + \alpha I) - (A + \alpha I)'K - Q + KBR^{-1}B'K \]  \hspace{1cm} (2.7)

Note that the ARE can be easily evaluated with standard computer-aided design software.

2.3 Solution Properties

The major impact of the inclusion of the weighting \( \exp\{2\alpha t\} \) in the quadratic cost function (2.2) relates to the location of the
poles of the closed-loop regulator

\[ \dot{x}(t) = [A - BG]x(t) \]  \hspace{1cm} (2.8)

It no matter what are the numerical values of A, B, Q, and R (modulo our standing LQR assumptions) it is true that

\[ \Re \lambda \left[ A - BG \right] < -\alpha \]  \hspace{1cm} (2.9)

As illustrated in Fig. 2.1

---

**Fig 2.1** Region of closed-loop poles in the S-plane

Thus, both the closed-loop state, \( x(t) \), and closed-loop control, \( u(t) \), decay faster
It is easy to demonstrate that all the other inherent properties of LQR design (with \(R = \text{diagonal matrix}\)), such as the multivariable gain and phase margin properties and the singular value inequality properties remain valid in this case as well.

2.4 Elements of Proof

The solution of the optimal control problem can, of course, be derived with variational argument similar to those used to derive the standard LQR problem.

However, there is a more direct way that can be used to derive the solution summarized in Section 2.2. The idea is to make a change of variables so that the problem definition of Section 2.1 reduces to that of a standard LQR problem.
Consider the following variable definition
\[ \dot{x}(t) \triangleq e^{\alpha t} \dot{z}(t) \]  \hspace{1cm} (2.10)
\[ \dot{v}(t) \triangleq e^{\alpha t} \dot{u}(t) \]  \hspace{1cm} (2.11)

Clearly
\[ x(t) = e^{-\alpha t} \dot{x}(t) \quad ; \quad u(t) = e^{-\alpha t} \dot{v}(t) \]  \hspace{1cm} (2.12)
and so the cost functional (2.2) reduces to
\[ J = \int_{0}^{\infty} \left[ \dot{z}'(t)Q \dot{z}(t) + \dot{v}'(t)R \dot{v}(t) \right] dt \]  \hspace{1cm} (2.13)

which of course has the structure of the quadratic cost in the standard LQR problem.

Now let us compute the dynamics of \( \dot{z}(t) \).

From eq. (2.10) we have
\[ \dot{z}(t) = \alpha e^{\alpha t} \dot{z}(t) + e^{\alpha t} \dot{x}(t) \]
\[ = \alpha \dot{z}(t) + e^{\alpha t} \left[ AX(t) + BU(t) \right] \]
\[ = \alpha \dot{z}(t) + A \dot{z}(t) + \alpha \dot{z}(t) + BU(t) \]
\[ = [A + \alpha I] \dot{z}(t) + BU(t) \]  \hspace{1cm} (2.14)

Thus we want to solve the standard LQR problem for the system
\[ \ddot{z}(t) = (A + \alpha I) \dot{z}(t) + B \dot{v}(t) \]  \hspace{1cm} (2.15)
with respect to the cost functional (2.13). The results of Section 2.2 follow directly once we transform the solution of the standard LQR problem to the original variables via eq. (2.12).

3. The Stochastic LQR Problem

3.1 Problem Formulation

\( \text{In the stochastic LQR problem we allow the inclusion of white process noise in the state dynamics. We still assume that all state variables can be measured exactly, and can be used for feedback as necessary.} \)

\( \text{We assume that the state vector } x(t) \text{ satisfies the stochastic differential equation} \)

\[
\dot{x}(t) = A x(t) + B u(t) + L \xi(t) \tag{3.1}
\]

\( \text{where } \xi(t) \text{ is a zero-mean white noise process.} \)

\( \text{We seek the control that minimizes the following quadratic cost} \)
\[ J = \lim_{T \to \infty} \frac{1}{T} \mathbb{E} \int_0^T \left[ x'(t) Q x(t) + u'(t) R u(t) \right] dt \] (3.2)

The integrand of the cost functional is identical to that of the deterministic LQR problem and we make the same assumptions (see eqn (1.2) to (1.5)). The difference is that the integrand is now a stochastic process so that we must minimize its expected value \( \mathbb{E}\{ \cdot \} \). The normalization by the integration time \( T \) is necessary to ensure that the cost functional is finite.

### 3.2 Problem Solution

4 Surprisingly the optimal control for the stochastic LQR problem is identical to that of the corresponding deterministic LQR problem. This result hinges upon the assumption that \( \Sigma(t) \) is white. The proof is quite complicated if one insists on rigor; it requires advanced mathematical tools to obtain.
and is beyond the scope of this book. For the sake of completeness we state the solution to the stochastic LQR problem below.

4. The optimal control is

\[ u(t) = -Gx(t) \]  

(3.3)

where

\[ G = R^{-1}B'K \]  

(3.4)

and \( K = K' > 0 \) is the solution matrix of the algebraic Riccati equation

\[ 0 = -KA - A'K - Q + KBR^{-1}B'K \]  

(3.5)

3.3 Properties

4. All properties of the deterministic LQR problem are also true for the stochastic LQR problem.
LECTURE NOTES
ON
THE KALMAN FILTER

April 5, 1983

Written by
MICHAEL ATHANS

All rights reserved. No further reproduction allowed
THE KALMAN FILTER

0. SUMMARY

The purpose of this chapter is to discuss the celebrated Kalman filter (KF). The Kalman filter is of fundamental importance on its own right in the field of stochastic dynamic optimal estimation. However, our discussion of the Kalman filter will be slanted towards the way that we are going to use it in MIMO control system design; it is a dynamic MIMO system that will define the so-called Linear-Quadratic-Gaussian (LQG) model-based compensator.

In fact we want to stress at the outset that KF "purists" will be shocked by the way that we plan to adjust the KF free parameters, with total disregard about the mathematical definitions and paying absolutely no attention to stochastic optimality considerations. Thus, we shall use the KF as a means to an end; in particular, how it will help us

(a) shape loop singular values, and
(b) design dynamic MIMO compensators for feedback control.

1. PROBLEM DEFINITION

In this section we present the definition of the basic stochastic estimation problem for which the KF yields the "optimal" answer.

1.1 State Dynamics

Consider an LTI stochastic dynamic system whose state vector $\mathbf{x}(t)$ obeys the stochastic differential equation

$$\dot{\mathbf{x}}(t) = A \mathbf{x}(t) + B \mathbf{u}(t) + L \mathbf{\xi}(t) \quad (1-1)$$
In Eq. (1-1) \( u(t) \) is a deterministic (control) input, assumed known. The vector \( \xi(t) \) is a stochastic process called the process (or plant) noise, with certain key properties. In particular, we assume that \( \xi(t) \) is a continuous-time white Gaussian noise. It's mathematical characterization is as follows: it has zero-mean for all \( t \), i.e.

\[
E\{\xi(t)\} = 0
\]  

(1-2)

and its covariance matrix at times \( t \) and \( \tau \) is defined by

\[
\text{cov}[\xi(t);\xi(\tau)] \overset{\Delta}{=} E\{\xi(t)\xi(\tau)\} = \Xi(t-\tau)
\]

(1-3)

with \( \delta(t-\tau) \) being the Dirac delta function (impulse at \( t=\tau \)). The matrix \( \Xi \) is called the intensity matrix of \( \xi(t) \) and it has the following properties

\[
\Xi = \Xi^T > 0
\]

(1-4)

Remark: Continuous white noise does not exist in nature; it is the limit of a broad-band noise process. Continuous white noise has constant power at all frequencies; hence, infinite energy (this is why it does not really exist). It is completely unpredictable since from Eq. (1-3) it follows that is uncorrelated for any \( t \neq \tau \), while it has infinite variance at \( t=\tau \). Nonetheless, it is an excellent modeling tool.

Remark: The state \( x(t) \), the solution of Eq. (1-1) is a well-defined physical process; it actually is a colored Gaussian random process.

1.2 Measurement Equation

We assume that our sensors cannot directly measure all of the physical state variables, the components of \( x(t) \), of the system (1-1). Rather we assume that our sensors can only measure certain output variables in the presence of additive measurement noise.
The mathematical model of the measurements is as follows:

\[ \hat{y}(t) = C \hat{x}(t) + \theta(t) \]  \hfill (1-5)

The vector \( \hat{y}(t) \) is the actual sensor measurement. The measurement (or sensor) noise \( \theta(t) \) is assumed to be a continuous-time white Gaussian noise process, independent of \( \xi(t) \), with zero mean, i.e.

\[ E[\theta(t)] = 0 \]  \hfill (1-6)

and covariance matrix

\[ \text{cov}[\theta(t); \theta(t')] = E[\theta(t)\theta'(t)] = \Theta(t-t) \]  \hfill (1-7)

with

\[ \Theta = \Theta' > 0 \]  \hfill (1-8)

Figure 1-1 shows a visualization, in block diagram form, of Eqs. (1-1) and (1-5).

1.3 The State Estimation Problem

Let imagine that we have been observing the control \( u(t) \) and the output \( y(t) \) over the infinite past up to the present time \( t \). Let

\[ U(t) = \{u(t); \ -\infty < t \leq t\} \]  \hfill (1-9)

\[ Y(t) = \{y(t); \ -\infty < t \leq t\} \]  \hfill (1-10)

denote the past histories of the control and output, respectively.

The state estimation problem is as follows: Given \( U(t) \) and \( Y(t) \) find a vector \( \hat{x}(t) \), at time \( t \), which is an "optimal" estimate of the present state \( x(t) \) of the system (1-1).
Under the stated assumptions regarding the Gaussian nature of $\xi(t)$ and $\theta(t)$ the "optimal" state estimate is the conditional mean of the state, i.e.

$$\hat{x}(t) = E[x(t)|U(t), Y(t)]$$  \hspace{1cm} (1-11)

One can relax the Gaussian assumption and define the optimality of the state estimate $\hat{x}(t)$ in different ways.

One popular way is to demand that $\hat{x}(t)$ be generated by a linear transformation on the past "data" $U(t)$ and $Y(t)$, such that the state estimation error $\bar{x}(t)$

$$\bar{x}(t) \triangleq x(t) - \hat{x}(t)$$  \hspace{1cm} (1-12)

has zero mean, i.e.

$$E[\bar{x}(t)] = 0$$  \hspace{1cm} (1-13)

and the cost functional

$$J = E \left\{ \sum_{i=1}^{n} \bar{x}_i^2(t) \right\} = E[\bar{x}'(t)\bar{x}(t)] = \text{tr}[E[\bar{x}(t)\bar{x}'(t)]]$$  \hspace{1cm} (1-14)

is minimized.

The cost functional $J$ has the physical interpretation that it minimizes the sum of the error variances $E[\bar{x}_i^2(t)]$ for each state variable. If we let $\Sigma$ denote the covariance matrix (stationary) of the state estimation error

$$\Sigma \triangleq E[\bar{x}(t)\bar{x}'(t)]$$  \hspace{1cm} (1-15)

then the cost $J$ of Eq. (1-14) can also be written as

$$J = \text{tr}[\Sigma]$$  \hspace{1cm} (1-16)
Bottom Line: We need an algorithm that translates the signals we can observe, $u(t)$ and $y(t)$, into a state estimate $\hat{x}(t)$, such that the state estimation error $\bar{x}(t)$ is "small" in some well-defined sense. The KF is the algorithm that does just that!

2. SUMMARY OF THE KALMAN FILTER EQUATIONS

2.1 Additional Assumptions

In this section we summarize the on-line and off-line equations that define the Kalman filter. Before we do that we make two additional "mild" assumptions

$$[A,L] \text{ is stabilizable (or controllable)} \quad (2-1)$$

$$[A,C] \text{ is detectable (or observable)} \quad (2-2)$$

If $[A,L]$ is controllable, this means that the process noise $\xi(t)$ excites all modes of the system (1-1); if $[A,C]$ is observable that means that the "noiseless" output $y(t) = C \, x(t)$ contains information about all state variables.

2.2 The Kalman Filter Dynamics

The function of the KF is to generate in real-time the state estimate $\hat{x}(t)$ of the state $x(t)$. It actually is an LTI dynamic system, of identical order (n) to the plant (1-1), and is driven by

(a) the deterministic control input $u(t)$, and

(b) the measured output vector $y(t)$.

The Kalman filter dynamics are given as follows:

$$\dot{\hat{x}}(t) = A \, \hat{x}(t) + B \, u(t) + H[y(t) - C \, \hat{x}(t)] \quad (2-3)$$
A block diagram visualization of Eq. (2-3) is shown in Fig. 2-1. Note that in Eq. (2-3) all variables have been defined previously except for the KF gain matrix $H$ whose calculation is carried out off-line and will be discussed in Section 2.4.

The filter gain matrix $H$ multiplies the so-called residual or innovations vector

$$r(t) \triangleq y(t) - C \hat{x}(t)$$  \hspace{1cm} (2-4)

and updates the time-rate-of-change, $\dot{\hat{x}}(t)$, of the state estimate $\hat{x}(t)$. The residual $r(t)$ is like an "error" between the measured output $y(t)$, and the predicted output $C \hat{x}(t)$.

Remark: From an intuitive point of view the KF, defined by Eq. (2-3) and illustrated in Fig. 2-1, can be thought as a model-based observer or state-reconstructor. The reader should carefully compare the structures depicted in Figs. 1-1 and 2-1. The plant/sensor properties, reflected by the matrices $A, B$ and $C$, are duplicated in the KF*. Indeed the optimal predictor is obtained with $H=0$. The state estimate $\hat{x}(t)$ is continuously updated by the actual sensor measurements, through the formation of the residual $r(t)$ and "closing" the loop with the filter gain matrix $H$.

The KF dynamics of Eq. (2-3) can also be written in the form

$$\dot{\hat{x}}(t) = [A-H C] \hat{x}(t) + B u(t) + H y(t)$$  \hspace{1cm} (2-5)

* No signal corresponding to $L \xi(t)$ shows up in Fig. 2-1. This is because we assumed that $\xi(t)$ had zero-mean; being completely unpredictable its "best guess" is 0.
Fig 2-1. The structure of the Kalman Filter. The control, $u(t)$, and measured output, $y(t)$, are those associated with the stochastic system of Fig 1-1. The filter gain matrix $H$ is computed in a special way.
From the structure of Eq. (2-5) we can immediately see that the stability of the KF is governed by the matrix \([A-H\ C]\). At this point of our development we remark that Assumption (2-2), i.e. the detectability of \([A,C]\), guarantees the existence of at least one filter gain matrix \(H\) such that the KF is stable, i.e.

\[
\text{Re} \lambda_i [A-H\ C] < 0; \quad i=1,2,\ldots,n
\]  

(2-6)

2.3 Properties of Model-Based Observers

We have remarked that the KF gain matrix \(H\) is calculated in a very special way. However, it is extremely useful, and consistent with our Model-Based-Compensator (MBC) philosophy, to examine the structure of Eqs. (2-5) or (2-5) and Fig. 2-1 with an arbitrary filter gain matrix \(H\), except that it leads to a stable loop in the sense that Eq. (2-6) holds. Thus, for the development that follows in this subsection think of \(H\) as being a fixed matrix.

As before let \(\hat{x}(t)\) denote the state estimation error vector

\[
\hat{x}(t) = \dot{x}(t) - \dot{x}(t)
\]  

(2-7)

It follows that

\[
\ddot{x}(t) = \dot{x}(t) - \dot{x}(t)
\]  

(2-8)

Next, we substitute Eqs. (1-1), (1-5) and (2-5) into Eq. (2-8) and use Eq. (2-7) as appropriate. After some easy algebraic manipulations we obtain the following stochastic vector differential equation for the state estimation error \(\hat{x}(t)\):

\[
\ddot{x}(t) = [A-H\ C]\hat{x}(t) + \Sigma \xi(t) - H \theta(t)
\]  

(2-9)

Note that, in view of Eq. (2-6), the estimation error dynamic system is stable. Also note that the deterministic signal \(\bar{u}(t)\) does not appear in the error equation (2-9).
Under our assumptions that the system is stable and it was started at the indefinite past \( t_0 \rightarrow -\infty \), then it is easy to verify that
\[
E(\bar{x}(t)) = 0
\]  
(2-10)

This implies that any stable model-based estimator of the form shown in Fig. 2-1, with any filter gain matrix \( H \), gives us unbiased estimates.

Using next elementary facts from stochastic linear system theory one can calculate the error covariance matrix \( \Sigma \) of the state estimation error \( \bar{x}(t) \)
\[
\Sigma \triangleq \text{cov}[\bar{x}(t); \bar{x}(t)] = E(\bar{x}(t)\bar{x}'(t))
\]  
(2-11)

It is the solution of the so-called Lyapunov matrix equation (linear in \( \Sigma \))
\[
[A-H C] \Sigma + \Sigma [A-H C]' + L \Sigma L' + H \otimes H' = 0
\]  
(2-12)

with
\[
\Sigma = \Sigma' > 0
\]  
(2-13)

Thus, for any given filter gain matrix \( H \) we can calculate* the associated error covariance matrix \( \Sigma \) from Eq. (2-12). Recalling the discussion of Section 1.3, we can evaluate, for a given \( H \), the quality of the estimator by calculating
\[
J = \text{tr}[\Sigma]
\]  
(2-14)

The specific way that the KF gain is calculated is by solving a constrained static optimization problem of minimizing (2-14) with respect to the elements \( h_{ij} \) of the matrix \( H \) subject to the algebraic constraints (2-12) and (2-13).

* The HONEY-X software package can solve Lyapunov equations.
2.4 The Kalman Filter Gain and Associated Filter Algebraic Riccati Equation (FARE)

We now summarize the off-line calculations* that define fully the Kalman filter (2-3) or (2-5).

The KF gain matrix $H$ is computed by

$$H = \Sigma C' \Theta^{-1},$$

(2-15)

where $\Sigma$ is the unique, symmetric, and at least positive semidefinite solution matrix of the so-called Filter Algebraic Riccati Equation (FARE)

$$0 = A \Sigma + \Sigma A' + L \Sigma L' - \Sigma C' \Theta^{-1} C \Sigma$$

(2-16)

with

$$\Sigma = \Sigma' > 0$$

(2-17)

Remark: The KF gain is obtained by setting

$$\frac{\partial}{\partial h_{ij}} \text{tr}[\Sigma] = 0$$

(2-18)

where $\Sigma$ is given by Eq. (2-12). The result is Eq. (2-15). Substituting Eq. (2-15) into Eq. (2-12) one deduces the FARE (2-16).

2.5 Duality Between the KF and LQ Problems

The mathematical problems associated with the solution of the LQ and KF are dual. This duality was recognized by R.E. Kalman as early as 1960.

The duality can be used to deduce several properties of the KF simply by "dualizing" the results of the LQ problem. See Table 2.1. A summary of the KF properties is given by Section 3.

*These calculations can be executed by HONEY-X.
<table>
<thead>
<tr>
<th>LQ Problem</th>
<th>KF Problem</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A$</td>
<td>$-A'$</td>
</tr>
<tr>
<td>$B$</td>
<td>$C'$</td>
</tr>
<tr>
<td>$Q$</td>
<td>$L = L'$</td>
</tr>
<tr>
<td>$R$</td>
<td>$Q$</td>
</tr>
<tr>
<td>$K$</td>
<td>$\Sigma$</td>
</tr>
</tbody>
</table>

$[A,B]$ stabilizable

$[A,Q^{1/2}]$ detectable

$[A,C]$ detectable

$[A,L]$ stabilizable (with $\Sigma = I$)
3. **Kalman Filter Properties**

3.1 **Introduction**

In this section we summarize the key properties of the Kalman filter. These properties are the "dual" of those derived for the LQ controller.

3.2 **Guaranteed Stability**

Recall that the KF algorithm is

\[
\dot{\hat{x}}(t) = (A - H C)\hat{x}(t) + B u(t) + H y(t)
\]  

(3-1)

Then, under the assumptions of Section 2, the matrix \([A - H C]\) is strictly stable, i.e.

\[
\text{Re} \; \lambda \; [A - H C] < 0; \; i=1,2,\ldots,n
\]  

(3-2)

3.3 **Frequency Domain Equality**

One can readily derive a frequency domain* equality for the KF. In the development that follows let

\[
\Xi = I
\]  

(3-3)

Let us make the following definitions: Let \(G_{KF}(s)\) denote the KF loop-transfer matrix

\[
G_{KF}(s) \triangleq C(sI - A)^{-1}H
\]  

(3-4)

\[
G_{KF}^H(s) \triangleq H^*(-sI - A')^{-1}C^*
\]  

(3-5)

Let \(G_{POL}(s)\) denote the filter open-loop transfer matrix (from \(\xi(t)\) to \(y(t)\))

*See Ref. 811025/6232 for the frequency domain equality associated with the LQ problem.
\[
G_{\text{FOL}}(s) \triangleq \frac{C(sI-A)^{-1}L}{(sI-A)^{-1}C'} \quad (3-6)
\]

\[
G_{\text{FOL}}^H(s) \triangleq \frac{L'}{(sI-A')^{-1}C'} \quad (3-7)
\]

Then the following equality holds

\[
[I + G_{\text{KF}}(s)]^H[I + G_{\text{KF}}(s)]^H = \Theta + G_{\text{FOL}}(s)G_{\text{FOL}}^H(s) \quad (3-8)
\]

If

\[
\Theta = \mu I; \quad \mu > 0 ,
\]

then Eq. (3-8) reduces to

\[
[I + G_{\text{KF}}(s)][I + G_{\text{KF}}(s)]^H = I + \frac{1}{\mu} G_{\text{FOL}}(s)G_{\text{FOL}}^H(s) \quad (3-9)
\]

3.4 Guaranteed Robustness Properties

The KF enjoys the same type of robustness properties as the LQ regulator.

The following properties are valid if

\[
\Theta = \text{diagonal matrix} \quad (3-10)
\]

From the frequency domain equality (3-8) we deduce the inequality

\[
[I + G_{\text{KF}}(s)][I + G_{\text{KF}}(s)]^H \geq I \quad (3-11)
\]

From the definition of singular values we then deduce that

\[
\sigma_{\text{min}}[I + G_{\text{KF}}(s)] \geq 1 \quad (3-12)
\]

\[
\sigma_{\text{min}}[I + G_{\text{KF}}^{-1}(s)] \geq \frac{1}{2} \quad (3-13)
\]
3.5 Optimal KF Root Locus

The root-square locus techniques* for LQ regulators can be readily adapted to root-locus studies involving the closed-loop poles of the Kalman filter.

4. THE KALMAN FILTER AS A TRIVIAL CONTROL PROBLEM

(to be written).

*See Ref. 811028/6232
HOW TO ADAPT LQ REGULATOR DESIGNS TO COMMAND-FOLLOWING
AND OUTPUT-DISTURBANCE REJECTION FEEDBACK CONTROL
SYSTEMS

Prof. Michael Athans

April 18, 1984

All rights reserved
No further reproduction allowed without written permission of M. Athans
HOW TO ADAPT LQ REGULATOR DESIGNS TO COMMAND-FOLLOWING AND
OUTPUT-DISTURBANCE REJECTION FEEDBACK CONTROL
SYSTEMS

0. SUMMARY

The structure and properties of LQ regulators (LQR) are derived by the solution
of a specific optimal control problem. The problem formulation that we have discussed
up to now did not involve reference command-following or output disturbance-rejection.
The purpose of this note is to
(a) demonstrate how to adapt LQR designs to the above situations, and (b) to point
out certain important issues that arise.

1. MOTIVATION

1.1 The Standard LQR Loop

Let us consider an LTI system described by

\[ \dot{x}(t) = A \, x(t) + B \, u(t) \quad (1.1) \]

with \( x(t) \in \mathbb{R}^n \) and \( u(t) \in \mathbb{R}^m \).

We know how to construct the LQR solution to the problem, which implies that
the optimal control law is given by

\[ u(t) = -K \, x(t) \quad (1.2) \]

and requires full state variable feedback.

The resultant LQR Loop is shown in Figure 1. If we break the loop at point
\( \textcircled{1} \), i.e. at the plant input, then the resultant loop transfer function
Figure 1: The standard LQR loop.

say \( T_1(s) \), is the LQR loop transfer function matrix \( G_{LQ}(s) \)

\[
\begin{align*}
T_1(s) &= G_{LQ}(s) = (sI-A)^{-1}B \\
&= G_{LQ}(s)
\end{align*}
\]

We have already discussed that, under mild assumptions,

(a) the MIMO feedback system of Figure 1 is nominally stable,

(b) the LQ loop transfer matrix \( T_1(s) \) or \( G_{LQ}(s) \) has nice crossover properties as exhibited by the guaranteed singular value inequalities

\[
\begin{align*}
\sigma_{\min}(1+T_1(j\omega)) &> 1 \\
\sigma_{\min}(1+T_1^{-1}(j\omega)) &> \frac{1}{2}
\end{align*}
\]

which in turn imply good robustness properties when the modeling errors are reflected at the plant input, i.e. point \( 1 \) in Figure 1, and

(c) we can easily shape the singular values \( \sigma_{\min}(T_1(j\omega)) \) by cleverly selecting the design parameters \( Q \) and \( R \) of the LQR problem.

1.2 Some Important Design Issues

In spite of all the nice things we can say about the LQR loop Figure 1, it is not immediately obvious how to adapt the LQR design in the context of
→ (a) command-following, and/or
→ (b) output disturbance-rejection.

These are precisely the performance oriented requirements of MIMO servomechanisms.

It certainly possible to define an output vector \( y_p(t) \in \mathbb{R}^m \)

\[
y_p(t) = C \times(t)
\]  

(1.6)

an output disturbance \( d(t) \in \mathbb{R}^m \), and a command vector \( r(t) \in \mathbb{R}^m \), and change the system in Figure 1 to that shown in Figure 2. The difficulty with the system of Figure 2 is that the error signal

\[
e(t) = r(t) - y(t) = r(t) - (y_p(t) + d(t))
\]  

(1.7)

is "outside" the main feedback loop and one cannot accomplish good command-following and/or disturbance rejection.

Figure 2: An LQR loop with "meaningless" command-following and disturbance rejection performance.
So the basic question remains on how one adapts LQR loops so that the error signal, \( e(t) \), appears inside the MIMO loop.

1.3 Optimization Based Approaches

It is possible to address the command-following and disturbance-rejection problem using a variant of the LQ optimal control problem; however, its solution has some unpleasant properties. These properties are summarized below.

Property of Optimal LQ solution: The optimal control \( u(t) \), at the present time \( t \), depends on

(a) the present value of the state \( x(t) \) - this is all right and we cannot quarrel with it - and

(b) the future numerical values of both the reference input \( r(t), t \leq t < \infty \) and the future numerical value of the disturbance \( d(t), t < \infty \).

In the vast majority of practical applications we do not know nor can we "measure" the future command input (typically \( r(t) \) depends on a human action, i.e. an aircraft pilot) or the future disturbance (typically we cannot measure even the present disturbance \( d(t) \) never mind its future time-evolution, since the disturbance is the consequence of natural environmental phenomena). Hence, a brute-force optimal approach does not lead to realistic feedback designs.

The reason behind this sad state of affairs is that we have attempted to use the wrong theoretical methodology for the problem at hand. A fundamental property of optimal designs is that the optimal control at each instant of time, \( t \), wants to minimize the integrated effect of future errors. The present state \( x(t) \) summarizes

*The reader is strongly advised to read Lecture Note No. 811023/6232 at this point, so as to see all the details.

all past information as far the plant is concerned. However, the future errors will depend on the future values of \( r(t) \) and \( d(t) \), and to keep future errors small (minimum in fact!), the optimal control now must know in detail the future commands and disturbances.

The bottom line of the discussion up to now is that we cannot use the solution of a dynamic optimal control problem. We have tried to fit a square peg into a round hole.*

From a historical point of view, after control theorists figured out the unpleasant facts discussed above, they came up with a modification of the statement of the optimal control problem. Before we overview this change of strategy, we warn the reader that this method does not work too well either, although it does provide insight on what are key issues in optimal command-following and disturbance-rejection designs.

**Detailed Modeling of the Command-Inputs and Disturbances**

In this version of the optimal control problem, in addition to the plant dynamics, one imagines that the reference command input \( r(t) \) is the output of a MIMO LTI dynamic system, with state vector \( x_r(t) \), driven by zero-mean white noise. Similarly, we imagine that the disturbance \( d(t) \) is the output of a MIMO LTI dynamic system, with state vector \( x_d(t) \), once more driven by zero-mean white noise.

These modeling assumptions lead to a stochastic version of an LQ optimal control problem which can be solved. We are not going to discuss the details here, but discuss the nature of the resultant optimal control law. Under suitable assumptions, it can be shown that the optimal control law is of the form

\[
\dot{u}(t) = -G x(t) - G_r x_r(t) - G_d x_d(t)
\]  

(1.8)

*Remember: Theories have limitations, but stupidity does not.
where the gain matrices $G_r$, $G_l$, and $G_d$ can be calculated by solving a (big) algebraic Riccati equation.

What is important here is to understand the structure of the control law (1.8). The optimal control $u(t)$ now (at time $t$) does not depend on any past or future quantities. It is simply a linear combination of the present plant state vector $x(t)$, the present state vector $x_r(t)$ (that models the "dynamics" of the command vector $r(t)$), and the present state vector $x_d(t)$ (that models the "dynamics" of the disturbance $d(t)$). So at first glance, it may appear that we have "licked" the problem of the dependence of the present control upon future values of commands and disturbances.

But there is not rest for the wicked. The control in Eq. (1.8) depends on purely fictitious quantities, $x_r(t)$ and $x_d(t)$, that we cannot measure! If we "told" the mathematics that $x_r(t)$ and $x_d(t)$ cannot be measured the LQ optimal control problem would become the so-called LQG stochastic optimal control problem, in which Kalman filters would be used to estimate the fictitious states $x_r(t)$ and $x_d(t)$.

1.4 Where do we stand?

We have started with the nice deterministic LQR loop of Figure 1 and we observed that it is not immediately obvious how to modify it so that we get a meaningful command-following and disturbance-rejection MIMO servomechanism design (not like the one shown in Figure 2).

We have also asserted, by means of informal discussions, that appeals to solutions to optimal control problems do not give us anything to shout about. Of course, we can forget the whole problem (and this is precisely what has happened in the open-literature). But, if the problem is important (and it certainly is) we must find an engineering fix. This is the topic of the rest of this note.
2. **THE AD-HOC LQ SERVO**

2.1 *Introduction*

The author has coined the word LQ-servo to distinguish the structure of the command-following and disturbance-rejection design, to be discussed below, from that of the LQ-regulator. Also, the author does not know of a reference that addresses the LQ-servo issues using the philosophy that will be adopted in the sequel.

2.2 *Setting up the LQ servo problem*

Our plant is an LTI MIMO system with control $u(t)$ and plant output $y_p(t)$, with $u(t) \in \mathbb{R}^m$ and $y_p(t) \in \mathbb{R}^m$. As before, we let $r(t) \in \mathbb{R}^m$ be the reference command input. We also denote the output disturbance by $d(t) \in \mathbb{R}^m$. Then, the output $y(t)$ to be controlled is

$$y(t) = y_p(t) + d(t) \quad (2.1)$$

and the error vector $e(t) \in \mathbb{R}^m$ is

$$e(t) = r(t) - y(t) \quad (2.2)$$

Next we make some basic, but very important, points.

**Remark 2.1:** In order to physically construct the error signal (2.2) we must measure the output $y(t)$. So we assume that $y(t)$ is a physically measured set of variables associated with the plant.

**Remark 2.2:** Physical disturbances cannot be measured. In fact, physical disturbances (such as wind gusts, ocean waves, electric demands etc) "enter" the physical system at different physical locations, imparting to, or subtracting from, the dynamic system some form of physical energy. In our mathematical model (2.1) we model the net effect of all these disturbances at the plant output by the output disturbance vector $d(t)$. 
Implication 2.1: The logical consequence of the above two remarks is that the plant output vector \( y_p(t) \) is directly measured. After all, if there were no disturbances (\( d(t) = 0 \)), \( y_p(t) = y(t) \) and, by Remark 2.1, \( y_p(t) \) is directly measured.

Remark 2.3: In LQR based designs we assume that the entire plant state vector, \( x(t) \in \mathbb{R}^n \), is directly measured. Otherwise, we could not implement the LQR loop of Figure 1.

The reader at this point may wonder why we belabor the obvious. The reason is stated next.

Implication 2.2: Since both \( y_p(t) \) and \( x(t) \) are measured directly, \( y_p(t) \) must be a subvector of \( x(t) \)!

Now we return to the model of our plant which is consistent with the above remarks. With no loss of generality we can represent the plant state vector as follows:

\[
x(t) = \begin{bmatrix} y_p(t) \\ \cdots \\ x_r(t) \end{bmatrix}; \quad x(t) \in \mathbb{R}^n, \quad y_p(t) \in \mathbb{R}^m, \quad x_r(t) \in \mathbb{R}^{n-m}
\]  

(2.3)

The dynamic plant model is given by

\[
\dot{x}(t) = A x(t) + B u(t) \\
y_p(t) = \underbrace{[I_{mxm} : 0]}_{C_p} x(t)
\]  

(2.4)

where \( I_{mxm} \) is the \( mxm \) identity matrix.

We are now ready to adapt the LQR structure to derive the LQ-servo.
2.3 The LQ-Servo Structure

We start with our plant model in the specific format of Eqs. (2.3) and (2.4).

Suppose that we carry-out an arbitrary LQR design* which, of course, yields the control law

\[ u(t) = -G x(t) \]  \hspace{1cm} (2.5)

The decomposition of the plant state vector \( x(t) \) according to Eq. (2.3) suggests the natural decomposition of the LQ control \( m \times m \) gain matrix \( G \) as follows:

\[ G = \begin{bmatrix} G_y & G_r \\ \hline m \times m & m \times (n-m) \end{bmatrix} \]  \hspace{1cm} (2.6)

From Eqs. (2.3), (2.5), and (2.6) we can see that the LQR control is

\[ u(t) = -G_y y_p(t) - G_r x_r(t) \]  \hspace{1cm} (2.7)

We can now redraw the standard LQR loop of Figure 1 to that illustrated in Figure 3.

\[ \text{Figure 3: Another way of visualizing the LQ regulator of Figure 1.} \]

*Any reasonable full-state feedback design can also be interpreted along the same lines, not just LQR designs.
We stress that the loop shown in Figure 3 is identical, in all respects, to the standard LQR loop of Figure 1. Indeed, the loop transfer function matrix at point 1 is still $T_1(s)$ given by Eq. (1.3). All we have done in Figure 3 is

(a) stressed the decomposition of the plant state vector $x(t)$ into its subcomponents $y_p(t)$ and $x_r(t)$ according to Eq. (2.3),

(b) visualized the decomposition of the LQ gain matrix $G$ into its submatrices $G_y$ and $G_r$ according to Eq. (2.7), and

(c) put some "zero" vectors at specific points in the loop.

At this point it should become obvious to the reader why we have selected the structure shown in Figure 3; the location of the "zero" signals in Figure 3 are precisely where we would naturally expect our command vector, $r(t)$, and output disturbance vector, $d(t)$, to enter in a meaningful way.

We can now define the LQ-Servo loop to be the one shown in Figure 4. Notice

![Figure 4: Defined structure of the LQ servo loop.](image-url)
that (in contradistinction to Figure 2) the error signal \( e(t) \) appears in a natural way inside the main feedback loop.

Remark 2.4: The LQ-Servo loop in Figure 4 is not optimal with respect to anything related to \( r(t) \) and \( d(t) \).

Remark 2.5: The nominal LQ-servo design is guaranteed to be closed-loop stable. The reason is that in linear time-invariant systems the stability is not changed by the presence of the exogenous signals \( r(t) \) and \( d(t) \). The LQ-servo loop of Figure 4 is stable, because the LQR loop of Figure 3 is guaranteed stable. In fact, the nominal closed-loop poles of the LQ servo are those of the nominal LQR design, \( \lambda \), \( [A-B \ C] \).

Remark 2.6: One can get an intuitive idea of how the LQ servo works by closely comparing Figures 3 and 4. Recall that an LQ regulator wants to drive all state variables to zero. The LQ regulator of Figure 3 interprets any non-zero value of the state variables \( y_p(t) \) and \( x_r(t) \) as "bad" and adjusts the control to drive them toward zero; in particular, in Figure 3 any signal that appears in front of the gain \( G_y \) is interpreted as a non-zero \( y_p(t) \). Now look at Figure 4. The presence of the command \( r(t) \) and disturbance \( d(t) \) result in nonzero errors, \( e(t) \), the signal entering the gain \( G_y \). So the control system wants to drive \( e(t) \) to zero; it "thinks" that \( e(t) \) is \( y_p(t) \). But, we do want \( e(t) \) to go to zero, so our objective is met. Thus, we have used the solution of the optimal LQ regulator problem and we have adapted it to get a working (nonoptimal) servo. Given the discussion of Section 1.3 we have accomplished quite a bit. Anyway, in engineering "the end always justifies the means."

2.4: The LQ-Servo: Performance and Robustness

In this section we analyze the LQ-servo loop illustrated in Figure 4 so that we can understand fully its robustness and performance characteristics, where performance relates to command-following and disturbance-rejection.
Stability-Robustness Issues

If we reflect high-frequency modeling uncertainty to the plant input (loop broken at point 1 in Figure 4) then the loop transfer function of interest is $T_1(s)$ (see Figure 1 and Eq. (1.3))

$$T_1(s) = G_{LQ}(s) = G(sI-A)^{-1}B$$  \hspace{1cm} \star \star \star \hspace{1cm} (2.8)

We have seen several tricks by which we can easily shape the LQ loop transfer matrix $T_1(s)$, i.e. matching the singular values at low or high frequency, approximate cross-over-frequency design etc. However, if we only focus our attention to the shapes of the singular values of $T_1(j\omega)$:

(a) we obtain a realistic picture of the stability-robustness of the system of Figure 4 to modeling errors, and in particular high frequency modeling errors;

(b) we can obtain an erroneous idea of the command-following and output disturbance-rejection properties of the LQ-servo shown in Figure 4.

Performance-Issues

Whether or not the LQ servo does a good job of command-following and disturbance rejection in the typical low-frequency region would hinge upon the shape of the singular values of $T_2(s)$, $s=j\omega$, where $T_2(s)$ is the loop transfer matrix when we break the LQ-servo loop at point 2 in Figure 4.

Even a cursory examination of Figure 4 should convince the reader that, in general,

$$T_1(s) \neq T_2(s)$$  \hspace{1cm} \star \star \hspace{1cm} (2.9)

although both transfer matrices have the same dimensions (mxm matrices),
We can certainly calculate the loop transfer function matrix $T_2(s)$. To do so let us first define the $m \times m$ transfer matrix $Q(s)$ from $u_e(t)$ to $y_p(t)$ in Figure 4, i.e.

$$y_p(s) \triangleq Q(s)u_e(s)$$

Then,

$$T_2(s) = Q(s)G_y$$

To calculate $Q(s)$ we proceed as follows. We start with our special state-space model of our plant given by Eq. (2.4), i.e.

$$\dot{x}(t) = A \times x(t) + B \times u(t); \quad y_p(t) = C_p \times x(t) \tag{2.12}$$

From Figure 4 we have

$$u(t) = u_e(t) - u_r(t) = u_e(t) - G_x x_r(t) \tag{2.13}$$

Equations (2.12) and (2.13) yield

$$\dot{x}(t) = A \times x(t) - B \times G_x x_r(t) + B \times u_e(t) \tag{2.14}$$

If we define $D_p$ to be the following $(n-m) \times n$ matrix

$$D_p \triangleq \begin{bmatrix} 0_{(n-m) \times n} & I_{(n-m) \times (n-m)} \end{bmatrix} \tag{2.15}$$

then $x_r(t)$ can be written as a function of $x(t)$

$$x_r(t) = D_p x(t) \tag{2.16}$$

Therefore, from Eqs. (2.14) and (2.16) we obtain

$$\dot{x}(t) = [A - B \times G_x D_p] x(t) + B \times u_e(t)$$

Hence, the transfer matrix $Q(s)$, defined in Eq. (2.10), is

$$Q(s) = C_p (sI - A + B \times G_x D_p)^{-1} B \tag{2.17}$$
We remind the reader (see Eq. (2.4)) that the \( mxn \) matrix \( C_p \) has the special form

\[
C_p = \begin{bmatrix} I_{mxm} & 0_{mx(m-n)} \end{bmatrix}
\]  

(2.18)

Therefore, the command-following and disturbance rejection properties of the ad-hoc LQ-servo are characterized by the shapes of the singular values of

\[
T_2(s) = C_p (sI - A + B G_D)^{-1} B G_y
\]  

(2.19)

versus frequency \( (s=j\omega) \).

Remark 2.7: If we shape the singular values of \( T_1(j\omega) \), then we have effectively calculated the numerical values of the gains \( G_y \) and \( G_f \) in the LQ-servo of Figure 4. Hence, the shape of the singular values of \( T_2(j\omega) \) is fixed; we have no other degrees of freedom left. To put it another way, once we have designed a good \( T_1(s) \) then we are stuck with \( T_2(s) \). From a performance viewpoint we cannot tell whether \( T_2(s) \) is "good" or "bad"; we must look at the singular values \( \sigma_1[T_2(j\omega)] \) to make that judgement.

Remark 2.8: As we have seen, we can exploit the frequency domain equality for LQ regulators and devise tricks that making the shaping of \( \sigma_1[T_1(j\omega)] \) a relatively "easy" task. The author does not know of any easy ways of shaping directly the singular values \( \sigma_1[T_2(j\omega)] \) even only in the low-frequency region; future work may change this lack of knowledge.

Remark 2.9: A very limited number of numerical investigations indicate that if the plant \( C_p (sI-A)^{-1} B \) is minimum phase then (consistent with any stability-robustness constraints) if one makes \( \sigma_{\min}[T_1(j\omega)] \) large at low frequencies, then \( \sigma_{\min}[T_2(j\omega)] \)
will also be large (but not the same as the former). However, if the plant \( \frac{C(sI-A)^{-1}B}{P} \) is not minimum-phase and if its non-minimum phase zeros are in the low frequency region, then the magnitude of \( \sigma_{\min}[T_2(j\omega)] \) can be quite small in the frequency region of the non-minimum phase zeros, even though \( \sigma_{\min}[T_1(j\omega)] \) can be made large in that frequency region. The conclusion is that the inherent performance limitations associated with non-minimum phase plants persist in full state variable designs, and appear to be independent of the extra measurements (over and above those of the plant output variables) that are available for full-state feedback control.

3. CONCLUDING REMARKS

In this note we have shown how to adapt, in an ad-hoc manner, the LQR-based design methodology so that the important functions of command-following and disturbance-rejection are met. The resultant LQ-servo system is that shown in Figure 4.

Three are two important loop transfer matrices for the LQ-servo problem denoted by \( T_1(s) \) and \( T_2(s) \), obtained by breaking the loop of Figure 4 at points 1 and 2, respectively. The loop transfer matrix \( T_1(s) \) is the standard LQR loop transfer matrix (Eq. (2.8)) and it should be designed so that the typical high frequency stability-robustness specifications are met for plant uncertainties reflected at the plant input. Note that the singular values of \( T_1(j\omega) \) can be shaped in a relatively easy manner. Once \( T_1(s) \) is fixed, the loop transfer matrix \( T_2(s) \) is also fixed; see Eq. (2.19). The command-following and disturbance-rejection properties of the LQ-servo in Figure 4 are then specified by the low-frequency behavior of the singular values of \( T_2(j\omega) \). If the plant has non-minimum phase zeros, the command-following and disturbance-rejection performance of the LQ-servo may be poor.

At present, no easy tricks for shaping the singular values of \( T_2(j\omega) \) exist. The designer must shape \( T_1(j\omega) \) and see what happens to \( T_2(j\omega) \).

*Such an observation is the result of as yet unpublished research by the author.
Lecture Notes

on

THE ACCURATE MEASUREMENT KALMAN FILTER PROBLEM

Written by
Professor Michael Athans

April 24, 1985

REF. NO. 850425/6232

All Rights Reserved
THE ACCURATE MEASUREMENT KALMAN FILTER PROBLEM

0. SUMMARY

In this note we summarize the properties of the Kalman Filter (KF) problem when the intensity of the sensor noise approaches zero. In a mathematical sense this is the "dual" of the so-called "cheap-control LQR problem. The results are fundamental to the Loop Transfer Recovery (LTR) method applied at the plant input.

1. PROBLEM DEFINITION

Consider the stochastic LTI system

\[
\begin{align*}
\dot{x}(t) &= A_x x(t) + L \xi(t) \\
y(t) &= C_x x(t) + \theta(t)
\end{align*}
\]

We assume that the process noise \( \xi(t) \) is white, zero-mean, and with unit intensity, i.e.

\[
E(\xi(t)\xi'(t)) = I\delta(t-\tau)
\]

We also assume that the measurement noise \( \theta(t) \) is white, zero-mean and with intensity indexed by \( \mu \), i.e.

\[
E(\theta(t)\theta'(t)) = \mu I\delta(t-\tau)
\]

Definition 1.1: What we mean by the accurate measurement KF problem is defined by the limiting case

\[
\mu \to 0
\]

corresponding to essentially noiseless measurements.
Under the assumptions that $[A, L]$ is stabilizable and that $[A, C]$ is detectable we know that the KF is a stable system and generates the state estimates $\hat{x}(t)$ by

$$\hat{x}(t) = [A - H_{\mu} C] \hat{x}(t) + H_{\mu} y(t)$$

(1.6)

where we use the subscript $\mu$ to stress the dependence of the KF gain matrix $H_{\mu}$ upon the parameter $\mu$.

We recall that $H_{\mu}$ is computed by

$$H_{\mu} = \frac{1}{\mu} \Sigma_{\mu} C'$$

(1.7)

where the error covariance matrix $\Sigma_{\mu}$, also dependent upon $\mu$, is calculated by the solution of the FARE:

$$0 = A \frac{\Sigma_{\mu}}{\mu} + \frac{\Sigma_{\mu} A'}{\mu} + L L' - \frac{1}{\mu} \Sigma_{\mu} C' C \Sigma_{\mu}$$

(1.8)

We seek insight about the limiting behavior of both $\Sigma_{\mu}$ and $H_{\mu}$ as $\mu \to 0$.

2. THE MAIN RESULT

In this section we summarize the main result in terms of a theorem.

Theorem 2.1: Suppose that the TFM from the white noise $u(t)$ to the output $y(t)$ for the system (1.1), (1.2), i.e. the TFM

$$W(s) \triangleq C(sI - A)^{-1} L$$

(2.1)

is minimum phase. Then,

$$\lim_{\mu \to 0} \Sigma_{\mu} = 0$$

(2.2)
and
\[
\lim_{\mu \to 0} \sqrt{\mu} H_\mu = L W; \quad W'W = I \tag{2.5}
\]

Proof: This is theorem 4.14 in Kwakernaak and Sivan, Ref. [1], pp. 370-371.

Remark 2.1: It can be shown that the minimum phase of \( H(s) \), given by eq. (2.1), is both a necessary and a sufficient conditions for the limiting properties given by eqs. (2.2) and (2.3).

Remark 2.2: The implication of eq. (2.2) is that in the case of exact measurements upon a minimum phase plant, the KF yields exact state estimates, since the error covariance matrix is zero. This assumes that the KF has been operating upon the data for a sufficiently long time so that initial transient errors have died out.

Remark 2.3: For a non-minimum phase plant
\[
\lim_{\mu \to 0} \Sigma = \mu \neq 0 \tag{2.4}
\]
Hence, perfect state estimation is impossible for non-minimum phase plants.

Remark 2.4: The limiting behavior (with \( L = B \)) of the Kalman Filter gain
\[
\lim_{\mu \to 0} \sqrt{\mu} H_\mu = B W; \quad W'W = I \tag{2.5}
\]
is the precise dual of the limiting behavior of the LQ control gain
\[
\lim_{\rho \to 0} \sqrt{\rho} G_\rho = W C; \quad W'W = I \tag{2.6}
\]
for the minimum phase plant
\[
G(s) = C(sI - A)^{-1} B \tag{2.7}
\]
The relation (2.5) has been used by Doyle and Stein [2] to apply the LTR method at the plant input, while eq. (2.6) has been used by Kwakernaak [3] to apply the LTR method at the plant output (see also Kwakernaak and Sivan [1], pp. 419-427).

3. REFERENCES

