We present here a brief review of some concepts that are assumed as background for the text. Good references include Gantmacher (1977), Brogan (1974), and Strang (1980).

A.1 BASIC DEFINITIONS AND FACTS

The **determinant** of an \( n \times n \) matrix is symbolized as \(|A|\). If \( A \) and \( B \) are both square, then

\[
|A| = |A^T|, \quad (A.1-1)
\]

\[
|AB| = |A| \cdot |B|, \quad (A.1-2)
\]

where the superscript \( T \) represents transpose. If \( A \in \mathbb{C}^{m \times n} \) and \( B \in \mathbb{C}^{n \times m} \) (where \( n \) can equal \( m \)), then

\[
\text{trace}(AB) = \text{trace}(BA) \quad (A.1-3)
\]

\[
|I_m + AB| = |I_n + BA|. \quad (A.1-4)
\]

\((C \text{ represents the complex numbers.})\)

For any matrices \( A \) and \( B \),

\[
(AB)^T = B^T A^T \quad (A.1-5)
\]

and if \( A \) and \( B \) are nonsingular, then

\[
(AB)^{-1} = B^{-1} A^{-1}. \quad (A.1-6)
\]
The Kronecker product of two matrices $A = [a_{ij}] \in C^{m \times n}$ and $B = [b_{ij}] \in C^{p \times q}$ is

$$A \otimes B = [a_{ij} B] \in C^{mp \times nq}.$$  \hfill (A.1-7)

(It is sometimes defined as $A \otimes B = [Ab_{ij}]$.)

If $A = [a_1 a_2 \cdots a_n]$, where $a_i$ are the columns of $A$, the stacking operator is defined by

$$s(A) = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix}.$$  \hfill (A.1-8)

It converts $A \in C^{m \times n}$ into a vector $s(A) \in C^{mn}$. An identity that is often useful is

$$s(ABD) = (D^T \otimes A) s(B).$$  \hfill (A.1-9)

If $A \in C^{m \times m}$ and $B \in C^{p \times p}$, then

$$|A \otimes B| = |A|^p \cdot |B|^m.$$  \hfill (A.1-10)

See Brewer (1978) for other results.

If $\lambda_i$ is an eigenvalue of $A$ with eigenvector $v_i$, then $1/\lambda_i$ is an eigenvalue of $A^{-1}$ with the same eigenvector, for

$$A v_i = \lambda_i v_i$$  \hfill (A.1-11)

implies that

$$\lambda_i^{-1} v_i = A^{-1} v_i.$$  \hfill (A.1-12)

If $\lambda_i$ is an eigenvalue of $A$ with eigenvector $\omega_i$, and $\mu_j$ is an eigenvalue of $B$ with eigenvector $w_j$, then $\lambda_i \mu_j$ is an eigenvalue of $A \otimes B$ with eigenvector $v_i \otimes w_j$ (Brewer 1978).

**A.2 Partitioned Matrices**

If

$$D = \begin{bmatrix} A_{11} & 0 & 0 \\ 0 & A_{22} & 0 \\ 0 & 0 & A_{33} \end{bmatrix},$$  \hfill (A.2-1)

where $A_{ij}$ are matrices, then we write $D = \text{diag}(A_{11}, A_{22}, A_{33})$ and say that $D$ is block diagonal. If the $A_{ii}$ are square, then

$$|D| = |A_{11}| \cdot |A_{22}| \cdot |A_{33}|,$$  \hfill (A.2-2)

and if $|D| \neq 0$, then

$$D^{-1} = \text{diag}(A_{11}^{-1}, A_{22}^{-1}, A_{33}^{-1}).$$  \hfill (A.2-3)
If
\[
D = \begin{bmatrix}
A_{11} & A_{12} & A_{13} \\
0 & A_{22} & A_{23} \\
0 & 0 & A_{33}
\end{bmatrix},
\]
where \(A_{ij}\) are matrices, then \(D\) is upper block triangular and (A.2-2) still holds. Lower block triangular matrices have the form of the transpose of (A.2-4).

If
\[
A = \begin{bmatrix}
A_{11} & A_{12} \\
A_{21} & A_{22}
\end{bmatrix},
\]
we define the Schur complement of \(A_{22}\) as
\[
D_{22} = A_{22} - A_{21}A_{11}^{-1}A_{12}
\]
and the Schur complement of \(A_{11}\) as
\[
D_{11} = A_{11} - A_{12}A_{22}^{-1}A_{21}.
\]

The inverse of \(A\) can be written
\[
A^{-1} = \begin{bmatrix}
A_{11}^{-1} + A_{11}^{-1}A_{12}D_{22}^{-1}A_{21} & -A_{11}^{-1}A_{12}D_{22}^{-1} \\
-D_{22}^{-1}A_{21}A_{11}^{-1} & D_{22}^{-1}
\end{bmatrix},
\]
\[
A^{-1} = \begin{bmatrix}
D_{11}^{-1} & -D_{11}^{-1}A_{12}A_{22}^{-1} \\
-A_{22}^{-1}A_{21}D_{11}^{-1} + A_{22}^{-1}A_{21}D_{11}^{-1}A_{12}A_{22}^{-1} & D_{22}^{-1}
\end{bmatrix},
\]
or
\[
A^{-1} = \begin{bmatrix}
D_{11}^{-1} & -A_{11}^{-1}A_{12}D_{22}^{-1} \\
-A_{22}^{-1}A_{21}D_{11}^{-1} & D_{22}^{-1}
\end{bmatrix},
\]
depending, of course, on whether \(|A_{11}| \neq 0, |A_{22}| \neq 0,\) or both. These can be verified by checking that \(AA^{-1} = A^{-1}A = I.\) By comparing these various forms, we obtain the well-known matrix inversion lemma
\[
(A_{11}^{-1} + A_{12}A_{22}A_{21})^{-1} = A_{11}^{-1} - A_{11}^{-1}A_{12}(A_{21}A_{11}^{-1}A_{12} + A_{22}^{-1})^{-1}A_{21}A_{11}^{-1}.
\]

The Schur complement arises naturally in the solution of linear simultaneous equations, for if
\[
\begin{bmatrix}
A_{11} & A_{12} \\
A_{21} & A_{22}
\end{bmatrix} \begin{bmatrix}
X \\
Y
\end{bmatrix} = \begin{bmatrix}
0 \\
Z
\end{bmatrix},
\]
then from the first equation
\[
X = -A_{11}^{-1}A_{12}Y,
\]
and using this in the second equation yields

\[(A_{22} - A_{21}A_{11}^{-1}A_{12})Y = Z. \quad (A.2-13)\]

If \(A\) is given by (A.2-5), then

\[|A| = |A_{11}| \cdot |A_{22} - A_{21}A_{11}^{-1}A_{12}| = |A_{22}| \cdot |A_{11} - A_{12}A_{22}^{-1}A_{21}|. \quad (A.2-14)\]

Therefore, the determinant of \(A\) is the product of the determinant of \(A_{11}\) (or \(A_{22}\)) and the determinant of the Schur complement of \(A_{22}\) (or \(A_{11}\)).

### A.3 QUADRATIC FORMS AND DEFINITENESS

If \(x \in \mathbb{R}^n\) is a vector, then the square of the Euclidean norm is

\[\|x\|^2 = x^T x. \quad (A.3-1)\]

If \(S\) is any nonsingular transformation, the vector \(Sx\) has a norm squared of \((Sx)^T Sx = x^T S^T Sx\). Letting \(P = S^T S\), we write

\[\|x\|^2_P = x^T P x \quad (A.3-2)\]
as the norm squared of \(Sx\). We call \(\|x\|_P\) the norm of \(x\) with respect to \(P\). We call

\[x^T Q x \quad (A.3-3)\]
a quadratic form. We shall assume \(Q\) is real.

Every real square matrix \(Q\) can be decomposed into a symmetric part \(Q_s\) (i.e., \(Q_s^T = Q_s\)) and an antisymmetric part \(Q_a\) (i.e., \(Q_a^T = -Q_a\)):

\[Q = Q_s + Q_a, \quad (A.3-4)\]

where

\[Q_s = (Q + Q^T)/2, \quad (A.3-5)\]

\[Q_a = (Q - Q^T)/2. \quad (A.3-6)\]

If the quadratic form \(x^T Ax\) has \(A\) antisymmetric, then it must be equal to zero since \(x^T Ax\) is a scalar, so that \(x^T Ax = (x^T Ax)^T = x^T A^T x = -x^T Ax\). For a general real square \(Q\), then

\[x^T Q x = x^T (Q_s + Q_a) x = x^T Q_s x. \quad (A.3-7)\]

We can therefore assume without loss of generality that \(Q\) in (A.3-3) is symmetric. Let us do so.

We say \(Q\) is:

*Positive definite* \((Q > 0)\) if \(x^T Q x > 0\) for all nonzero \(x\).
Positive semi-definite \((Q \geq 0)\) if \(x^T Q x \geq 0\) for all nonzero \(x\).

Negative semi-definite \((Q \leq 0)\) if \(x^T Q x \leq 0\) for all nonzero \(x\).

Negative definite \((Q < 0)\) if \(x^T Q x < 0\) for all nonzero \(x\).

Indefinite if \(x^T Q x > 0\) for some \(x\), \(x^T Q x < 0\) for other \(x\).

We can test for definiteness independently of the vectors \(x\). If \(\lambda_i\) are the eigenvalues of \(Q\), then

\[
\begin{align*}
Q > 0 & \text{ if all } \lambda_i > 0, \\
Q \geq 0 & \text{ if all } \lambda_i \geq 0, \\
Q \leq 0 & \text{ if all } \lambda_i \leq 0, \\
Q < 0 & \text{ if all } \lambda_i < 0.
\end{align*}
\]

Another test is provided as follows. Let \(Q = [q_{ij}] \in \mathbb{R}^{n \times n}\). The leading minors or \(Q\) are

\[
\begin{align*}
m_1 &= q_{11}, \\
m_2 &= \begin{vmatrix} q_{11} & q_{12} \\ q_{21} & q_{22} \end{vmatrix}, \\
m_3 &= \begin{vmatrix} q_{11} & q_{12} & q_{13} \\ q_{21} & q_{22} & q_{23} \\ q_{31} & q_{32} & q_{33} \end{vmatrix}, \ldots, \\
m_n &= |Q|.
\end{align*}
\]

In terms of the minors, we have

\[
\begin{align*}
Q > 0 & \text{ if } m_i > 0, \text{ all } i, \\
Q \geq 0 & \text{ if all principal minor not only leading minors) are nonnegative.} \\
Q \leq 0 & \text{ if } -Q \geq 0, \\
Q < 0 & \text{ if } \begin{cases} m_i < 0, & \text{all odd } i \\ m_i > 0, & \text{all even } i \end{cases}
\end{align*}
\]

Any positive semidefinite matrix \(Q\) can be factored into square roots either as

\[
Q = \sqrt{Q} \sqrt{Q}^T
\]

or as

\[
Q = \sqrt{Q}^T \sqrt{Q}.
\]

The (“left” and “right”) square roots in (A.3-11) and (A.3-12) are not in general the same. Indeed, \(Q\) may have several roots since each of these factorizations is not even unique. If \(Q > 0\), then all square roots are nonsingular.
If \( P > 0 \), then (A.3-2) is a norm. If \( P \geq 0 \), it is called a **seminorm** since \( x^T P x \) may be zero even if \( x \) is not.

### A.4 MATRIX CALCULUS

Let \( x \in \mathbb{C}^n = [x_1 \, x_2 \cdots x_n]^T \) be a vector, \( s \in \mathbb{C} \) be a scalar, and \( f(x) \in \mathbb{C}^m \) be an \( m \)-vector function of \( x \). The differential in \( x \) is

\[
\frac{dx}{ds} = \begin{bmatrix}
\frac{dx_1}{ds} \\
\frac{dx_2}{ds} \\
\vdots \\
\frac{dx_n}{ds}
\end{bmatrix},
\]

and the derivative of \( x \) with respect to \( s \) (which could be time) is

\[
\frac{dx}{ds} = \begin{bmatrix}
\frac{dx_1}{ds} \\
\frac{dx_2}{ds} \\
\vdots \\
\frac{dx_n}{ds}
\end{bmatrix}.
\]

If \( s \) is a function of \( x \). Then the **gradient** of \( s \) with respect to \( x \) is the column vector

\[
s_x = \frac{\partial s}{\partial x} = \begin{bmatrix}
\frac{\partial s}{\partial x_1} \\
\frac{\partial s}{\partial x_2} \\
\vdots \\
\frac{\partial s}{\partial x_n}
\end{bmatrix}.
\]

(The gradient is defined as a row vector in some references.) Then the total differential in \( s \) is

\[
ds = \left( \frac{\partial s}{\partial x} \right)^T dx = \sum_{i=1}^{n} \frac{\partial s}{\partial x_i} dx_i.
\]

If \( s \) is a function of two vectors \( x \) and \( y \), then

\[
ds = \left( \frac{\partial s}{\partial x} \right)^T dx + \left( \frac{\partial s}{\partial y} \right)^T dy.
\]

The **Hessian** of \( s \) with respect to \( x \) is the second derivative

\[
s_{xx} = \frac{\partial^2 s}{\partial x^2} = \begin{bmatrix}
\frac{\partial^2 s}{\partial x_i \partial x_j}
\end{bmatrix},
\]

which is a symmetric \( n \times n \) matrix. In terms of the gradient and the Hessian, the **Taylor series expansion** of \( s(x) \) about \( x_0 \) is

\[
s(x) = s(x_0) + \left( \frac{\partial s}{\partial x} \right)^T (x - x_0) + \frac{1}{2}(x - x_0)^T \frac{\partial^2 s}{\partial x^2} (x - x_0) + O(3),
\]

where \( O(3) \) represents terms of order 3, and \( s_x \) and \( s_{xx} \) are evaluated at \( x_0 \).
The *Jacobian* of $f$ with respect to $x$ is the $m \times n$ matrix

$$f_x = \frac{\partial f}{\partial x} = \begin{bmatrix} \frac{\partial f}{\partial x_1} & \frac{\partial f}{\partial x_2} & \cdots & \frac{\partial f}{\partial x_n} \end{bmatrix},$$  

(A.4-8)

so that the total differential of $f$ is

$$df = \frac{\partial f}{\partial x} dx = \sum_{i=1}^{n} \frac{\partial f}{\partial x_i} dx_i.$$  

(A.4-9)

We shall use the shorthand notation

$$\frac{\partial f^T}{\partial x} \Delta \left( \frac{\partial f}{\partial x} \right)^T \in \mathbb{C}^{n \times m}. \quad \text{(A.4-10)}$$

If $y$ is a vector and $A, B, D, Q$ are matrices, all with dimensions so that the following expressions make sense, then we have the following results:

$$\frac{d}{dt} (A^{-1}) = -A^{-1} \dot{A} A^{-1}. \quad \text{(A.4-11)}$$

Some useful gradients are

$$\frac{\partial}{\partial x} (y^T x) = \frac{\partial}{\partial x} (x^T y) = y, \quad \text{(A.4-12)}$$

$$\frac{\partial}{\partial x} (y^T Ax) = \frac{\partial}{\partial x} (x^T A^T y) = A^T y, \quad \text{(A.4-13)}$$

$$\frac{\partial}{\partial x} (y^T f(x)) = \frac{\partial}{\partial x} (f^T(x) y) = f_x^T y, \quad \text{(A.4-14)}$$

$$\frac{\partial}{\partial x} (x^T Ax) = Ax + A^T x, \quad \text{(A.4-15)}$$

and if $Q$ is symmetric, then

$$\frac{\partial}{\partial x} (x^T Q x) = 2Qx, \quad \text{(A.4-16)}$$

$$\frac{\partial}{\partial x} (x - y)^T Q (x - y) = 2Q(x - y). \quad \text{(A.4-17)}$$

The chain rule for two vector functions becomes

$$\frac{\partial}{\partial x} (f^T y) = f_x^T y + y_x^T f. \quad \text{(A.4-18)}$$

Some useful Hessians are

$$\frac{\partial^2 (x^T Ax)}{\partial x^2} = A + A^T, \quad \text{(A.4-19)}$$
and if $Q$ is symmetric
\[ \frac{\partial^2 x^T Q x}{\partial x^2} = 2Q, \tag{A.4-20} \]
\[ \frac{\partial^2}{\partial x^2} (x - y)^T Q (x - y) = 2Q. \tag{A.4-21} \]

Some useful Jacobians are
\[ \frac{\partial}{\partial x} (Ax) = A \tag{A.4-22} \]
(contrast this with (A.4-12)), and the chain rule
\[ \frac{\partial}{\partial x} (sf) = \frac{\partial}{\partial x} (fs) = sf_x + fs_x^T \tag{A.4-23} \]
(contrast this with (A.4-18)).
Some useful derivatives involving the trace and determinant are
\[ \frac{\partial}{\partial A} \text{trace}(A) = I, \tag{A.4-24} \]
\[ \frac{\partial}{\partial A} \text{trace}(BAD) = B^T D^T, \tag{A.4-25} \]
\[ \frac{\partial}{\partial A} \text{trace}(ABA^T) = 2AB, \text{ if } B = B^T \tag{A.4-26} \]
\[ \frac{\partial}{\partial A} |BAD| = |BAD| A^{-T}, \tag{A.4-27} \]
where $A^{-T} \triangleq (A^{-1})^T$.

A.5 THE GENERALIZED EIGENVALUE PROBLEM

Consider the generalized eigenvalue problem
\[ Gz = \mu Fz, \tag{A.5-1} \]
where
\[ \text{det}(\mu F - G) \equiv 0. \tag{A.5-2} \]

Then the finite generalized eigenvalues are the roots of $\text{det}(\mu F - G)$. Let $\mu_i$ be the roots of $\text{det}(\mu F - G)$ and define
\[ \eta_i = \text{dimker}(\mu_i F - G). \tag{A.5-3} \]

Then the rank 1 finite generalized eigenvectors are defined by
\[ (\mu_i F - G)z^1_{ij} = 0, \quad j \in \hat{\eta}_i \tag{A.5-4} \]
(where $\hat{\eta}_i = \{1, 2, \ldots, \eta_i\}$) and the rank $k$ finite eigenvectors for $k > 1$ and each $i$ and $j$ by

$$
(\mu_i F - G)z^{k+1}_{ij} = -F^k z^k_{ij}, \quad k \geq 1. \tag{A.5-5}
$$

If $F$ is nonsingular, the above equation can be used to solve recursively for the $z^k_{ij}$ beginning with the highest rank eigenvector in each chain. In that case this construction provides the eigenstructure of $F^{-1} G$. In the case where $F$ in singular, the above equation cannot generally be used to recursively generate the $z^k_{ij}$. Furthermore, there exist eigenvalues at infinity and corresponding eigenvectors that can be constructed as follows. Define $\eta = \dim \ker(F)$. Then the rank 1 infinite eigenvectors are defined by

$$
Fz^1_{\infty j} = 0, \quad j = \hat{\eta} \tag{A.5-6}
$$

and the rank $k$ infinite eigenvectors for $k > 1$ and each $j$ by

$$
F z^{k+1}_{\infty j} = G z^k_{\infty j}, \quad k \geq 1. \tag{A.5-7}
$$

By arranging the eigenvectors as the columns of two nonsingular matrices according to

$$
Z = [z^k_{ij} | z^k_{\infty j}], \quad W = [F z^k_{ij} | G z^k_{\infty j}] \tag{A.5-8}
$$

with $i, j, k$ incrementing in odometer order, then

$$
W^{-1} F V = \begin{bmatrix} I & 0 \\ 0 & N \end{bmatrix}, \quad W^{-1} G V = \begin{bmatrix} M & 0 \\ 0 & I \end{bmatrix}, \tag{A.5-9}
$$

where $M$ is a Jordan form matrix containing the finite generalized eigenvalues of $(G, F)$ and $N$ is a nilpotent Jordan matrix representing the infinite generalized eigenvalues. The above canonical form is also known as the Weierstrass form.
REFERENCES


REFERENCES


REFERENCES


INDEX

Abnormal function-of-final-state-fixed regulator problem, 185, 204
Ackermann’s formula, 91, 100, 161
Action-dependent heuristic dynamic programming, 501
Actor–critic implementation (of DT optimal adaptive control), 500
Actor–critic structures, 463, 475
Adaptive control, optimal: for discrete-time systems, 490–491
using policy iteration algorithm, 494–495
using value iteration algorithm, 495–496
Adaptive controllers, 461–462
direct vs. indirect, 463–464
Adaptive dynamic programming, 464
Adjoint system, 313
continuous, 114
discrete, 23, 34, 194, 205
Admissible controls, 439, 491
Admissible cost, 265
Affine state-variable feedback, 179, 194
Aircraft:
longitudinal autopilot, example, 166–167, 293–295
routing, example, 261–262
Arithmetic mean, 18
Asymptotic properties of the LQR, 307
Asynchronous value iteration, 478–479
Augmented state description, 343
Bandwidth, 356, 366
Bang-bang control, 234
Bang-off-bang control, 246–248
Bellman equation, 439–440, 446, 455, 462
and dynamic programming, 468–469
policy evaluation/improvement by, 474
Bellman Ford algorithm, 479
Bellman’s principle of optimality, 260–261
for continuous systems, 277–278
for discrete systems, 263–264
and dynamic programming, 260–261, 467–468
Bilinear system:
continuous optimal control, 169
discrete optimal control, 102
dynamic programming, 283–284
perturbation control, 212
Bilinear tangent control law, 126
Bode magnitude plot, 357, 359
Bode multivariable plot, 361–363
Bode singular value plot, 364–365
Boundary conditions, 20, 23
Bounded $L_2$-gain problem, 450–452
Brachistochrone problem, 220–224
Cargo loading, 285
Chain rule for differentiation, 525
Chang-Letov equation, 99–100, 165
Closed-loop, poles, 297
Closed-loop control, 41–53,
143–146, 190, 195, 307
Closed-loop Markov chain, 472–473
Closed-loop system, 90, 97, 144,
154, 179, 194, 300
adjoint, 179, 185, 194, 205
characteristic polynomial, 97,
164, 291
locus of poles, 78, 93, 167
optimal steady-state poles, 91,
100, 108, 161, 165
for polynomial regulator, 291
Closest point of approach, 16
Command-generator tracker (CGT), 332–338
Complimentary sensitivity, see Cosensitivity
Computer simulation, 21
bang-bang control, 243–245
bang-off-bang control, 253
Computer simulation (continued)  
Digital control, 54  
Harmonic oscillator, 106, 152  
Linear quadratic regulator, 146, 149–150  
Linear quadratic tracker, 180  
Newton’s system, 77, 243–245, 253  
Preliminary analysis for, 49, 61  
Scalar optimal control, 30–31, 149–150  
Scalar system, 49, 149–150  
Scalar tracker, 181  
Conjugate gradient method, 15  
Conservation of energy, 117  
Constant output feedback, 316  
Constraints:  
on control, 232, 262, 263–264  
on state, 262, 263–264  
Constraint equation, 4–5  
Continuous-time systems:  
Integrals reinforcement learning for optimal adaptive control of, 503–505  
online implementation, 507–508  
using policy iteration, 506  
using value iteration, 506  
synchronous optimal adaptive control for, 513–514  
Control delay, 288  
Controllability, see Reachability  
Control-weighting matrix, 299  
Convergence, conditions for, 304–305  
Cooperative control systems, 481  
Coriolis force, 224  
Cosensitivity, 360  
and sensitivity, 357–361  
Costate, 23  
equation, 24, 32–35, 115, 135  
Coupled nonlinear matrix equation, 302, 346–347  
Critical point, 2  
Cubic equation for optimal solution, 16, 124  
Curse of dimensionality, 274  
Curvature matrix:  
constrained, 9, 210  
continuous, 145–146, 189  
discrete, 46, 210  
unconstrained, 2  
Curve, length between two points, 117  
Cycloid, 224  
Damping ratio, 158  
Deadbeat control, 104  
Dead-zone function, 248  
Decentralized control, 343–344  
Linear quadratic regulator, 345–347  
Definiteness of matrices, 521–523  
Delay operator, 198, 288  
Descriptor systems, 102  
Design parameters, 305  
tuning the, 305  
Detectability, 70, 156  
Deterministic policies, 465  
Deviation system, 317–319  
Deyst filter, 401  
Harmonic oscillator, 106  
Newton’s system, example, 59–63  
RC circuit, example, 55–58  
Diophantine equation, 290  
Discretization:  
of continuous performance index, 271–274  
of continuous system, 53, 271–274  
of transfer function, 292  
Disturbance(s):  
discrete system with, 182–183  
and performance robustness, 356  
Dual optimization problems, 17, 18  
Dynamic(s):  
augment the, 368  
compensator, 314  
optimization, 299–300  
Dynamic programming, 462, 467–468  
Eigenstructure assignment design of steady-state regulator, 90–92, 160–161  
Eigenvalues:  
of inverse of a matrix, 519  
of Kronecker product, 519  
Eigenvectors:  
of Hamiltonian matrix, 34, 81, 90, 108, 159  
of Kronecker product, 519  
of optimal closed-loop system, 90–91  
Ergodic Markov chains, 466  
Estimation error, 396  
Euclidean norm, 298  
Euler’s approximation, 273  
Euler’s equation, 117, 131  
via HJB equation, 281  
Explicit model-following design, 338–343  
Feedback:  
output, 291  
state, see State-variable feedback  
suboptimal, 65–68  
Fictitious follower, 349  
Fictitious output, 70, 98, 156, 164  
Field of extremals, 269, 284  
Filters, washout, 313  
Final state:  
fixed, 24, 25–28, 38–40, 46, 119–120, 138–141, 170, 201  
free, 24, 28–30, 41–53, 143–146, 170, 201  
on moving point, 214, 228  
on surface, 215  
Fixed-final-state control, 38–40, 46, 138–141  
State feedback formulation, 149, 171, 183–185, 194–195  
Free-final-state control, 44, 120–121  
Frequency domain design of linear quadratic regulator, 164–167  
Frequency-domain techniques, 355  
Functional equation of dynamic programming, 264  
Gain(s):  
optimal, 375  
scheduling, 311–313  
Game theory, 344, 438–439. See also Zero-sum games  
Generalized state-space systems, discrete optimal control, 102  
Geometric mean, 18  
Gradient, 1  
Based solution, 321  
Minimization algorithm, 321  
Numerical methods, 15  
Vector, 523  
Gramian, see Observability gramian; Reachability gramian  
$H_{\infty}$ control, application of zero-sum games to, 450–453  
$H_{\infty}$ design, 357, 430–435  
Hamiltonian, 348  
Hamiltonian function, 6, 22, 32, 113, 135, 278  
Hamiltonian matrix:  
continuous, 136, 159  
discrete, 34, 80, 90, 101
INDEX

537
eigenvectors, 34, 81, 90, 108, 159, 175
Hamiltonian system:
  continuous, 131, 136, 158–159, 172, 178
discrete, 34, 80, 90, 192
Hamilton-Jacobi-Bellman
  equation, 277–279, 440–441, 443, 449, 468–469
Hamilton’s equations of motion, 117
Hamilton’s principle, 116–117
Harmonic oscillator:
  digital control of, 106
eigenstructure design, 92–95, 161–163
  linear quadratic regulator, 150–153
minimum-fuel control, 259
root-locus design, 99
steady-state regulator, 99
zero input cost, 172
Helicopter longitudinal autopilot, 293–295
Hessian matrix, 1, 523
Hewer’s algorithm, 482–483
Indirect adaptive controllers, 462, 463
Infinite horizon optimal control
  problem, 75, 93, 157, 180, 196
Integral reinforcement learning
  for optimal adaptive
  control of continuous-time
  systems, 503–505
  online implementation, 507–508
  using policy iteration, 506
  using value iteration, 506
Interpolation for discrete dynamic
  programming, 274–276
Inverse of partitioned matrix, 520
Iterative learning control, 488
Jacobian, 5, 524
Kalman filter, 391–404
Kalman gain:
  continuous dynamic, 144
discrete dynamic, 43, 105, 193, 203
  static, 17
  steady-state, 69, 74, 90, 161, 174
Kernel matrix, 36, 46, 137
Kronecker product, 103, 172, 519
Lagrange multiplier, 6–7, 22, 113, 200, 301, 399
Lagrange’s equations of motion, 116–117
Lagrangian for a dynamical
  system, 116
Leading minors of matrix, 522
Learning:
  Monte Carlo, 488
  reinforcement, 462–464, 503–505
temporal difference, 489–490
Leibniz’s rule, 111
Linearized plant model, 355
Linear minimum-energy problem, 254–257
Linear minimum-fuel problem, 246–248
Linear minimum-time problem, 213–214, 228–230
Linear quadratic
  Gaussian/loop-transfer
  recovery (LQG/LTR), 357
Linear quadratic regulator,
  443–444
  continuous state-costate
  formulation, 135
  continuous state feedback
  formulation, 144–145
derivation by dynamic
  programming, 270–271
derivation via HJB equation, 281–283
discrete state-costate
  formulation, 32
static, example, 11
linear quadratic regulator, 443–444
continuous state-costate
  formulation, 135
continuous state feedback
  formulation, 144–145
derivation by dynamic
  programming, 270–271
derivation via HJB equation, 281–283
discrete state-costate
  formulation, 32
  discrete state feedback
  formulation, 96
eigenstructure design, 108, 160–161
frequency domain design, 96–100, 165–166
with function of final state
  fixed, 185, 204–205
perturbation, 188, 206–209
for polynomial systems, 290
root-locus design, 99, 108, 165
steady-state, 90, 96–97, 156
suboptimal, 65–66, 154
with weighting of state-input
  inner product, 52–53, 104, 153–154
Linear quadratic tracker:
  continuous affine feedback
  formulation, 179
continuous state-costate
  formulation, 178–179
derivation via HJB equation, 286
discrete affine feedback
  formulation, 194
discrete state-costate
  formulation, 192
formulated as regulator, 183, 198–199
for polynomial systems, 288
suboptimal, 180, 195–196
time-invariant, 196
Linear quadratic zero-sum games, 452–453
Linear tangent control law, 217
Loop gain, 97, 164, 356
singular value, 357
Loop transfer recovery (LTR), 408–430
Low frequency specifications, 367
LQ algorithm, 304–305
LQ tracker with output feedback, 322
Lyapunov equation, 184, 300, 347, 350
algebraic, 37, 103, 136
closed-loop, 66, 74, 154
continuous observability, 136, 175
continuous reachability, 140, 175
discrete observability, 36, 46
discrete reachability, 36
online solution, 496
scalar, 48
solution, 36, 40, 46, 136, 137, 140
and value iteration, 483
as vector equation, 103, 172
Markov chain, closed-loop, 472–473
Markov decision processes, 464–473
Matrix design equation, 297
Matrix inversion lemma, 360, 520
Maximum, 2
Microprocessors, 53
Minimum, see Necessary
  conditions for minimum;
  Sufficient conditions for
  minimum
Minimum-energy problem, 21, 25, 38–39, 139
constrained, 254–257
Minimum-fuel problem, 261–262
linear, 246–248
normal, 248
Minimum-time problem, 213–214, 228–230
  with control weighting, 257–258
  linear, 234
  normal, 238
Model-following control, 289, 296
Model-following regulator, 340–343
Modelling errors, 355
Model reduction, 373–378
Monte Carlo learning, 488
Multi-Input Multi-Output (MIMO) systems, 297, 356
Multiplayer games, linear quadratic, 459–460
Multiplayer non-zero-sum games, 453–458
Multiplicative uncertainties, 372
Multivariable Bode plot, 361–363
Nash equilibrium, 454, 456
Nash game, 347–348
Natural frequency, 158
Necessary conditions for minimum, 185
  continuous linear quadratic regulator, 144–145, 185
  continuous linear quadratic tracker, 179
discrete linear quadratic regulator, 34, 205
discrete linear quadratic tracker, 194
general continuous systems, 115
general discrete systems, 23 static, 2, 6, 8
Negative definite, 522
Neighboring optimal solutions, 14, 187, 208
Neural networks, 497–498
Neuro-dynamic programming, 493
Newton’s system, examples:
  bang-bang control, 239–245
  bang-off-bang control, 239–245
  constrained minimum-energy control, 248–253
digital control, 59–63
  limiting control, 77–80
  open-loop control, 141–143
optimal control, problem, 171
optimal control via HJB equation, 286
optimal steady-state poles, 157, 173
steady-state control, 78, 157, 173
suboptimal control, 78, 174
  tracker, 196–197, 211
zero-input cost, 172
Nominal trajectory, 209
Nonlinear matrix equations, coupled, 302
Nonlinear non-zero-sum games, 453–458
Nonlinear systems:
  discretization of, 271–274
  function-of-final-state-fixed regulator, 199–201
optimal control, 24, 102, 115, 131–132, 439–441
  optimal control by approximation, 168–169
  optimal control using dynamic programming,
    264–274, 276
  perturbation control, 186–187, 206–209
  tracking problem, 177–178, 190–191
Non-zero-sum games, 453
  cooperative/competitive aspects of, 459
Norm, 521
Normal function-of-final-state-fixed regulator problem, 185, 204
Normal minimum-fuel problem, 248
Normal minimum-time problem, 238–239
Normal system, 238
Numerical solution methods:
  gradient, 15
  steepest descent, 15
Observability, 37, 70, 156, 305–306
canonical, 32, 341, 342
time-varying plant, 96, 163–164, 175
Observability gramian:
  continuous, 137
discrete, 37
time-varying plant, 96, 163–164
Observer(s):
  design, 384–387
  filter, ARE, 387
  the Kalman filter, 383–408
  output-input, 386
Observer, state, 291
Open-loop control:
  continuous, 136–137, 143
discrete, 40
  with function of final state fixed, 172
scalar system, example, 27, 40–41, 135
Operator gain, 366–367
Optimal adaptive control
  (continuous-time systems):
    hybrid controller, 507–508
    integral reinforcement learning for, 503–505
Optimal adaptive control
  (discrete-time systems), 490–491
  actor–critic implementation, 500
Q learning, 501–503
Optimal control problem:
  constrained minimum-energy, 254–257
  continuous, 112
  continuous linear quadratic, 135–136
discrete, 19
discrete linear quadratic, 32
  infinite horizon, 75, 93, 157, 180
  linear minimum-fuel, 246–248
  linear minimum-time, 234, 257–258
  nonlinear systems, 19, 112, 439–441
  solution via HJB equation, 279–280
Optimal feedback gain, 305, 321
Optimal gains, 311–313
Optimal output feedback solution algorithm, 304
Optimal policy, 467
Optimal quality function, 484–485
Optimal value, 467
Optimization:
  constrained, 301
dual problem, 17
  by scalar manipulations, 4
  by scalar manipulations, 4
Orbit injection, minimum-time, example, 224–226
Output feedback, 298
  in decentralized control, 343–344
design, 302, 351–352
dynamic, 291
gain, 302
  in game theory, 343–344
LQR with, 302–303
  problem, 305
step-response shaping, 313
  theory, 356
Output injection, 70
Output stabilization, 340
INDEX

Parameters, design, 305
PBH rank test, 72
Performance index, 319–320, 344–345
for continuous dynamic systems, 112, 131, 135–157
cubic, 104
for discrete dynamic systems, 20, 32
discretization of, 271–274
infinite horizon, 75, 93, 157
linear, 104
minimum fuel, 21
minimum-time, 20, 213–214
minimum-time with control weighting, 370
zero steady-state, 370
Predictive formulation of polynomial system, 290
Preliminary analysis for computer simulation, 49, 61
Proportional navigation, 125
Q function, 484–485
defined, 484
policy iteration using, 487
value iteration using, 487
Q learning, for optimal adaptive control, 494–495
using Q function, 487
Riccati equation, 434–435
algebraic, 69, 97, 155, 174, 175
analytic solution, 80–84, 107–108, 155, 158–160
continuous, 144, 283
destabilizing solution, 175
discrete, 43, 194, 202
information formulation, 105
Joseph formulation, 44, 66, 103, 144, 154, 172, 271
limiting behavior, 70–71, 96, 153, 155–156
online solution, 483–484, 496, 508–509
solution from Hamiltonian system solutions, 172
square-root formulations, 105
stabilizing solution, 175
as vector equation, 173
Robust design, 313, 356–357, 380–383
Rollout algorithms, 479
Root locus, 78, 94, 99, 108, 165
Runge-Kutta integrator, 40
Saddle point, 2
Satellite, 113
Saturated control, 236
Saturation function, 257
Scalar system, examples: constrained minimum-energy control, 59
digital control, 55–58
dynamic programming, 264–270, 274–276
linear quadratic regulator, 47
minimum-fuel control, 59
minimum-time control, 258–259
open-loop control, 40–41, 141
optimal control, 25, 148–151
optimal control via HJB equation, 279–280
steady-state control, 75–77, 174
steady-state tracker, 181–182
tracker, example, 180–182
uncontrolled, 138
Schur complement, 6, 520
Seminorm, 523
Sensitivity:
and cosensitivity, 357–361
function, 356
Sensitivity matrix for initial costate, 132
Separation of variables, 148, 280
Separation principle, 404–408
Sequential decision problems, 465–467
Servo compensator, 334–335
Shortest distance:
- from point to line, 16, 226–228
- between two points, 16, 117

Shortest path problems, 479

Signum function, 237

Singular control interval, 238, 248

Singular point, 3

Singular value:
- maximum, 362
- minimum, 363

Solution, gradient-based, 321

Square root of matrix, 522

Stability augmentation systems (SAS), 297

Stability robustness, 355, 356, 371

Stabilizability, 70, 156

Stabilization of multi-input plant, 74, 157

Stackelberg games, 348–351

Stacking operator, 103, 172, 519

Stage cost, 465

affine, 179, 194
- for bang-bang control, 247–248
- graphical, 219, 223, 261, 268
- implicit, 219, 223
- nonlinear, 219, 223, 258

Stationarity condition, 7, 12, 23, 33, 115, 136, 281

Stationary point, 2

Steady-state control, 75–77, 90, 99, 155, 165–166, 180

Steady-state cost, 37, 137, 149

Steepest descent algorithm, 15

Stochastic strategies or policies, 465

Suboptimal control, 66, 78, 180, 195–196

Suboptimal cost, 7

Suboptimal feedback gain, 65–66, 154

Sufficient conditions for minimum:
- continuous systems, 187–190
- discrete systems, 208
- static, 2, 9

Sweep method:
- continuous, 143, 179, 184
- discrete, 42, 192, 202

Switching:
- curve, 242
- function, 237
- time, 240–241

Symmetric part:
- of matrix, 521
- of polynomial, 108, 176

Symplectic matrix, 81

Synchronous optimal adaptive (continuous-time systems), 513–514

Target set of final states, 215, 227

Taylor series, 1, 8, 186, 206, 278, 523

Temperature control, example, 118–121

Temporal difference error, 482, 489

Temporal difference learning, 489–490

policy iteration using, 492

value iteration using, 492–493

Thrust angle programming, 126

in gravitational field, 257

Time-varying systems:
- limiting control of, 95–96, 163–164
- observability, 95–96, 163–164
- reachability, 95–96, 163–164

Total differential, 524

Tracker problem, 316–317

Tracking:
- with disturbance rejection, 337
- reference-input, 313–314
- a unit step, 329–330

Tracking error, 183, 198, 319–320

Transversality condition, 214–215, 239

Two-degrees-of-freedom regulator, 291

Two-player zero-sum games, 444–449

Two-point boundary-value problem:
- continuous, 114
- discrete, 23
- unit solution method, 170

Unit solution method, 170–171

Unmodelled dynamics, 355

Utility, 466

Value (of a policy), 466

optimal, 467

Value function (of a policy), 466

Value function approximation, 493

Value iteration:
- algorithm, 477–479
- asynchronous, 478–479
- implementation models, 488
- integral reinforcement learning for optimal adaptive control using, 506
- optimal adaptive control using, 495–496
- temporal difference learning using, 492–493
- using Q function, 487

Variation in function, 111

Washout filters, 313

White noise, 395

Zermelo’s problem, 216–220, 258

Zero-input cost:
- continuous, 136–137
- discrete, 35–37
- scalar system, example, 137

Zero-order hold, 53

Zero-sum games:
- application to $H_{\infty}$ control, 450–453

Bellman equation, 446

defined, 445

linear quadratic, 452–453

two-player, 444–449