1

Getting Started with Model Predictive Control

1.1 Introduction

The main purpose of this chapter is to provide a compact and accessible overview of the essential elements of model predictive control (MPC). We introduce deterministic and stochastic models, regulation, state estimation, dynamic programming (DP), tracking, disturbances, and some important performance properties such as closed-loop stability and zero offset to disturbances. The reader with background in MPC and linear systems theory may wish to skim this chapter briefly and proceed to Chapter 2. Other introductory texts covering the basics of MPC include Maciejowski (2002); Camacho and Bordons (2004); Rossiter (2004); Goodwin, Serón, and De Doná (2005); Kwon (2005); Wang (2009).

1.2 Models and Modeling

Model predictive control has its roots in optimal control. The basic concept of MPC is to use a dynamic model to forecast system behavior, and optimize the forecast to produce the best decision—the control move at the current time. Models are therefore central to every form of MPC. Because the optimal control move depends on the initial state of the dynamic system, a second basic concept in MPC is to use the past record of measurements to determine the most likely initial state of the system. The state estimation problem is to examine the record of past data, and reconcile these measurements with the model to determine the most likely value of the state at the current time. Both the regulation problem, in which a model forecast is used to produce the optimal control action, and the estimation problem, in which the past record of measurements is used to produce an optimal state estimate, involve dynamic models and optimization.

We first discuss the dynamic models used in this text. We start with the familiar differential equation models

$$\begin{aligned}\frac{dx}{dt} &= f(x,u,t)\\ y &= h(x,u,t)\\ x(t_0) &= x_0\end{aligned}$$

in which $x \in \mathbb{R}^n$ is the state, $u \in \mathbb{R}^m$ is the input, $y \in \mathbb{R}^p$ is the output, and $t \in \mathbb{R}$ is time. We use $\mathbb{R}^n$ to denote the set of real-valued n-vectors. The initial condition specifies the value of the state x at time $t = t_0$, and we seek a solution to the differential equation for time greater than t_0, $t \in \mathbb{R}_{\geq t_0}$. Often we define the initial time to be zero, with a corresponding initial condition, in which case $t \in \mathbb{R}_{\geq 0}$.

1.2.1 Linear Dynamic Models

Time-varying model. The most general *linear* state space model is the time-varying model

$$\begin{aligned}\frac{dx}{dt} &= A(t)x + B(t)u\\ y &= C(t)x + D(t)u\\ x(0) &= x_0\end{aligned}$$

in which $A(t) \in \mathbb{R}^{n\times n}$ is the state transition matrix, $B(t) \in \mathbb{R}^{n\times m}$ is the input matrix, $C(t) \in \mathbb{R}^{p\times n}$ is the output matrix, and $D(t) \in \mathbb{R}^{p\times m}$ allows a direct coupling between u and y. In many applications $D = 0$.

Time-invariant model. If A, B, C, and D are time invariant, the linear model reduces to

$$\begin{aligned}\frac{dx}{dt} &= Ax + Bu\\ y &= Cx + Du\\ x(0) &= x_0\end{aligned} \tag{1.1}$$

One of the main motivations for using linear models to approximate physical systems is the ease of solution and analysis of linear models. Equation (1.1) can be solved to yield

$$x(t) = e^{At}x_0 + \int_0^t e^{A(t-\tau)}Bu(\tau)d\tau \tag{1.2}$$

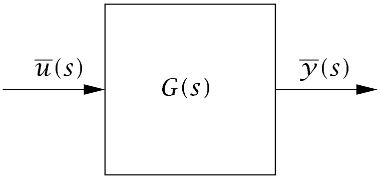

Figure 1.1: System with input $\overline{u}$, output $\overline{y}$, and transfer function matrix G connecting them; the model is $\overline{y} = G\overline{u}$.

in which $e^{At} \in \mathbb{R}^{n\times n}$ is the matrix exponential.[1] Notice the solution is a convolution integral of the entire $u(t)$ behavior weighted by the matrix exponential of At. We will see later that the eigenvalues of A determine whether the past $u(t)$ has more effect or less effect on the current $x(t)$ as time increases.

1.2.2 Input-Output Models

If we know little about the internal structure of a system, it may be convenient to take another approach in which we suppress the state variable, and focus attention only on the manipulatable inputs and measurable outputs. As shown in Figure 1.1, we consider the system to be the connection between u and y. In this viewpoint, we usually perform system identification experiments in which we manipulate u and measure y, and develop simple linear models for G. To take advantage of the usual block diagram manipulation of simple series and feedback connections, it is convenient to consider the Laplace transform of the signals rather than the time functions

$$\overline{y}(s) := \int_0^\infty e^{-st} y(t)dt$$

in which $s \in \mathbb{C}$ is the complex-valued Laplace transform variable, in contrast to t, which is the real-valued time variable. The symbol := means "equal by definition" or "is defined by." The transfer function matrix is then identified from the data, and the block diagram represents the

[1] We can define the exponential of matrix X in terms of its Taylor series

$$e^X := \frac{1}{0!}I + \frac{1}{1!}X + \frac{1}{2!}X^2 + \frac{1}{3!}X^3 + \cdots$$

This series converges for all X.

following mathematical relationship between input and output

$$\overline{y}(s) = G(s)\overline{u}(s)$$

$G(s) \in \mathbb{C}^{p\times m}$ is the transfer function matrix. Notice the state does not appear in this input-output description. If we are obtaining $G(s)$ instead from a state space model, then $G(s) = C(sI - A)^{-1}B + D$, and we assume $x(0) = 0$ as the system initial condition.

1.2.3 Distributed Models

Distributed models arise whenever we consider systems that are not spatially uniform. Consider, for example, a multicomponent, chemical mixture undergoing convection and chemical reaction. The microscopic mass balance for species A is

$$\frac{\partial c_A}{\partial t} + \nabla \cdot (c_A v_A) - R_A = 0$$

in which c_A is the molar concentration of species A, v_A is the velocity of species A, and R_A is the production rate of species A due to chemical reaction, in which

$$\nabla := \delta_x \frac{\partial}{\partial x} + \delta_y \frac{\partial}{\partial y} + \delta_z \frac{\partial}{\partial z}$$

and the $\delta_{x,y,z}$ are the respective unit vectors in the (x, y, z) spatial coordinates.

We also should note that the distribution does not have to be "spatial." Consider a particle size distribution $f(r,t)$ in which $f(r,t)dr$ represents the number of particles of size r to $r + dr$ in a particle reactor at time t. The reactor volume is considered well mixed and spatially homogeneous. If the particles nucleate at zero size with nucleation rate $B(t)$ and grow with growth rate, $G(t)$, the evolution of the particle size distribution is given by

$$\begin{aligned}
\frac{\partial f}{\partial t} &= -G\frac{\partial f}{\partial r} \\
f(r,t) &= B/G \qquad r = 0 \qquad t \geq 0 \\
f(r,t) &= f_0(r) \qquad r \geq 0 \qquad t = 0
\end{aligned}$$

Again we have partial differential equation descriptions even though the particle reactor is well mixed and spatially uniform.

1.2.4 Discrete Time Models

Discrete time models are often convenient if the system of interest is sampled at discrete times. If the sampling rate is chosen appropriately, the behavior between the samples can be safely ignored and the model describes exclusively the behavior at the sample times. The finite dimensional, linear, time-invariant, discrete time model is

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) \\ y(k) &= Cx(k) + Du(k) \\ x(0) &= x_0 \end{aligned} \tag{1.3}$$

in which $k \in \mathbb{I}_{\geq 0}$ is a nonnegative integer denoting the sample number, which is connected to time by $t = k\Delta$ in which Δ is the sample time. We use $\mathbb{I}$ to denote the set of integers and $\mathbb{I}_{\geq 0}$ to denote the set of nonnegative integers. The linear discrete time model is a linear difference equation.

It is sometimes convenient to write the time index with a subscript

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k \\ y_k &= Cx_k + Du_k \\ x_0 &\ \text{given} \end{aligned}$$

but we avoid this notation in this text. To reduce the notational complexity we usually express (1.3) as

$$\begin{aligned} x^+ &= Ax + Bu \\ y &= Cx + Du \\ x(0) &= x_0 \end{aligned}$$

in which the superscript $^+$ means the state at the next sample time. The linear discrete time model is convenient for presenting the ideas and concepts of MPC in the simplest possible mathematical setting. Because the model is linear, analytical solutions are readily derived. The solution to (1.3) is

$$x(k) = A^k x_0 + \sum_{j=0}^{k-1} A^{k-j-1} Bu(j) \tag{1.4}$$

Notice that a convolution sum corresponds to the convolution integral of (1.2) and powers of A correspond to the matrix exponential. Because (1.4) involves only multiplication and addition, it is convenient to program for computation.

The discrete time analog of the continuous time input-output model is obtained by defining the Z-transform of the signals

$$\overline{y}(z) := \sum_{k=0}^{\infty} z^k y(k)$$

The discrete transfer function matrix $G(z)$ then represents the discrete input-output model

$$\overline{y}(z) = G(z)\overline{u}(z)$$

and $G(z) \in \mathbb{C}^{p\times m}$ is the transfer function matrix. Notice the state does not appear in this input-output description. We make only passing reference to transfer function models in this text.

1.2.5 Constraints

The manipulated inputs (valve positions, voltages, torques, etc.) to most physical systems are bounded. We include these constraints by linear inequalities

$$Eu(k) \leq e \qquad k \in \mathbb{I}_{\geq 0}$$

in which

$$E = \begin{bmatrix} I \\ -I \end{bmatrix} \qquad e = \begin{bmatrix} \overline{u} \\ -\underline{u} \end{bmatrix}$$

are chosen to describe simple bounds such as

$$\underline{u} \leq u(k) \leq \overline{u} \qquad k \in \mathbb{I}_{\geq 0}$$

We sometimes wish to impose constraints on states or outputs for reasons of safety, operability, product quality, etc. These can be stated as

$$Fx(k) \leq f \qquad k \in \mathbb{I}_{\geq 0}$$

Practitioners find it convenient in some applications to limit the rate of change of the input, $u(k) - u(k-1)$. To maintain the state space form of the model, we may augment the state as

$$\widetilde{x}(k) = \begin{bmatrix} x(k) \\ u(k-1) \end{bmatrix}$$

and the augmented system model becomes

$$\begin{aligned} \widetilde{x}^+ &= \widetilde{A}\widetilde{x} + \widetilde{B}u \\ y &= \widetilde{C}\widetilde{x} \end{aligned}$$

in which

$$\widetilde{A} = \begin{bmatrix} A & 0 \\ 0 & 0 \end{bmatrix} \qquad \widetilde{B} = \begin{bmatrix} B \\ I \end{bmatrix} \qquad \widetilde{C} = \begin{bmatrix} C & 0 \end{bmatrix}$$

A rate of change constraint such as

$$\underline{\Delta} \leq u(k) - u(k-1) \leq \overline{\Delta} \qquad k \in \mathbb{I}_{\geq 0}$$

is then stated as

$$F\widetilde{x}(k) + Eu(k) \leq e \qquad F = \begin{bmatrix} 0 & -I \\ 0 & I \end{bmatrix} \qquad E = \begin{bmatrix} I \\ -I \end{bmatrix} \qquad e = \begin{bmatrix} \overline{\Delta} \\ -\underline{\Delta} \end{bmatrix}$$

To simplify analysis, it pays to maintain linear constraints when using linear dynamic models. So if we want to consider fairly general constraints for a linear system, we choose the form

$$Fx(k) + Eu(k) \leq e \qquad k \in \mathbb{I}_{\geq 0}$$

which subsumes all the forms listed previously.

When we consider nonlinear systems, analysis of the controller is not significantly simplified by maintaining linear inequalities, and we generalize the constraints to set membership

$$x(k) \in \mathbb{X} \qquad u(k) \in \mathbb{U} \qquad k \in \mathbb{I}_{\geq 0}$$

or, more generally

$$(x(k), u(k)) \in \mathbb{Z} \qquad k \in \mathbb{I}_{\geq 0}$$

We should bear in mind one general distinction between input constraints, and output or state constraints. The input constraints often represent *physical limits*. In these cases, if the controller does not respect the input constraints, the physical system enforces them. In contrast, the output or state constraints are usually *desirables*. They may not be achievable depending on the disturbances affecting the system. It is often the function of an MPC controller to determine in real time that the output or state constraints are not achievable, and relax them in some satisfactory manner. As we discuss in Chapter 2, these considerations lead implementers of MPC often to set up the optimization problem using hard constraints for the input constraints and some form of soft constraints for the output or state constraints.

Soft state or output constraints. A simple formulation for soft state or output constraints is presented next. Consider a set of hard input and state constraints such as those described previously

$$Eu(k) \leq e \qquad Fx(k) \leq f \qquad k \in \mathbb{I}_{\geq 0}$$

To soften state constraints one introduces slack variables, $\varepsilon(k)$, which are considered decision variables, like the manipulated inputs. One then relaxes the state constraints via

$$Fx(k) \leq f + \varepsilon(k) \qquad k \in \mathbb{I}_{\geq 0}$$

and adds the new "input" constraint

$$\varepsilon(k) \geq 0 \qquad k \in \mathbb{I}_{\geq 0}$$

Consider the augmented input to be $\widetilde{u}(k) = (u(k), \varepsilon(k))$, the soft state constraint formulation is then a set of mixed input-state constraints

$$\widetilde{F}x(k) + \widetilde{E}\widetilde{u}(k) \leq \widetilde{e} \qquad k \geq 0$$

with

$$\widetilde{F} = \begin{bmatrix} 0 \\ 0 \\ F \end{bmatrix} \qquad \widetilde{E} = \begin{bmatrix} E & 0 \\ 0 & -I \\ 0 & -I \end{bmatrix} \qquad \widetilde{u} = \begin{bmatrix} u \\ \varepsilon \end{bmatrix} \qquad \widetilde{e} = \begin{bmatrix} e \\ 0 \\ f \end{bmatrix}$$

As we discuss subsequently, one then formulates a stage-cost penalty that weights how much one cares about the state x, the input u and the violation of the hard state constraint, which is given by ε. The hard state constraint has been replaced by a mixed state-input constraint. The benefit of this reformulation is that the state constraint cannot cause an infeasiblity in the control problem because it can be relaxed by choosing ε; large values of ε may be undesirable as measured by the stage-cost function, but they are not infeasible.

Discrete actuators and integrality constraints. In many industrial applications, a subset of the actuators or decision variables may be integer valued or discrete. A common case arises when the process has banks of similar units such as furnaces, heaters, chillers, compressors, etc., operating in parallel. In this kind of process, part of the control problem is to decide how many and which of these discrete units should be on or off during process operation to meet the setpoint or reject a disturbance. Discrete decisions also arise in many scheduling problems. In chemical production scheduling, for example, the discrete decisions can be whether or not to produce a certain chemical in a certain

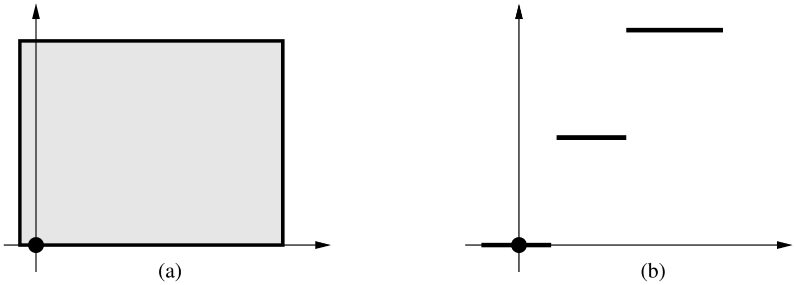

Figure 1.2: Typical input constraint sets $\mathbb{U}$ for (a) continuous actuators and (b) mixed continuous/discrete actuators. The origin (circle) represents the steady-state operating point.

reactor during the production schedule. Since these decisions are often made repeatedly as new measurement information becomes available, these (re)scheduling problems are also feedback control problems.

To define discrete-valued actuators, one may add constraints like

$$u_i(k) \in \{0, 1\} \qquad i \in I_D, \quad k \in \mathbb{I}_{\geq 0}$$

in which the set $I_D \subset \{1, 2, \ldots, m\}$ represents the indices of the actuators that are discrete, which are binary (on/off) decisions in the case illustrated above. Alternatively, one may use the general set membership constraint $u(k) \in \mathbb{U}$, and employ the set $\mathbb{U}$ to define the discrete actuators as shown in Figure 1.2. In the remainder of this introductory chapter we focus exclusively on continuous actuators, but return to discrete actuators in later chapters.

1.2.6 Deterministic and Stochastic

If one examines measurements coming from any complex, physical process, fluctuations in the data as depicted in Figure 1.3 are invariably present. For applications at small length scales, the fluctuations may be caused by the random behavior of small numbers of molecules. This type of application is becoming increasingly prevalent as scientists and engineers study applications in nanotechnology. This type of system also arises in life science applications when modeling the interactions of a few virus particles or protein molecules with living cells. In these applications there is no deterministic simulation model; the only system model available is stochastic.

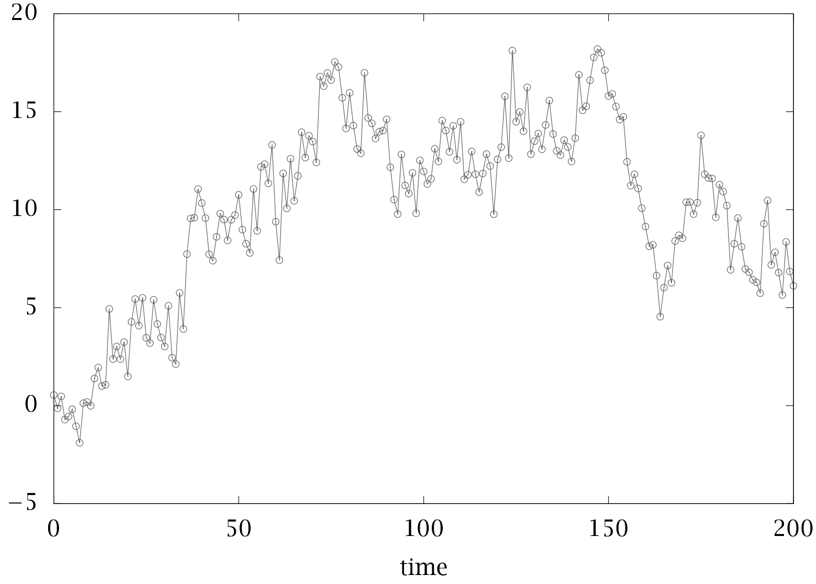

Figure 1.3: Output of a stochastic system versus time.

Linear time-invariant models. In mainstream, classical process control problems, we are usually concerned with modeling, monitoring and controlling macroscopic systems, i.e., we are not considering systems composed of small numbers of molecules. So one may naturally ask (many do) what is the motivation for stochastic models in this arena? The motivation for stochastic models is to account for the unmodeled effects of the environment (disturbances) on the system under study. If we examine the measurement from any process control system of interest, no matter how "macroscopic," we are confronted with the physical reality that the measurement still looks a lot like Figure 1.3. If it is important to model the observed measurement fluctuations, we turn to stochastic models.

Some of the observed fluctuation in the data is assignable to the measurement device. This source of fluctuation is known as measurement "noise." Some of the observed fluctuation in the data is assignable to unmodeled disturbances from the environment affecting the state of the system. The simplest stochastic model for representing these two possible sources of disturbances is a linear model with added random variables

$$\begin{aligned} x^+ &= Ax + Bu + Gw \\ y &= Cx + Du + v \end{aligned}$$

with initial condition $x(0) = x_0$. The variable $w \in \mathbb{R}^g$ is the random variable acting on the state transition, $v \in \mathbb{R}^p$ is a random variable acting on the measured output, and x_0 is a random variable specifying the initial state. The random variable v is used to model the measurement noise and w models the process disturbance. The matrix $G \in \mathbb{R}^{n\times g}$ allows further refinement of the modeling between the source of the disturbance and its effect on the state. Often G is chosen to be the identity matrix with $g = n$.

1.3 Introductory MPC Regulator

1.3.1 Linear Quadratic Problem

We start by designing a controller to take the state of a deterministic, linear system to the origin. If the setpoint is not the origin, or we wish to track a time-varying setpoint trajectory, we will subsequently make modifications of the zero setpoint problem to account for that. The system model is

$$\begin{aligned} x^+ &= Ax + Bu \\ y &= Cx \end{aligned} \tag{1.5}$$

In this first problem, we assume that the state is measured, or $C = I$. We will handle the output measurement problem with state estimation in the next section. Using the model we can predict how the state evolves given any set of inputs we are considering. Consider N time steps into the future and collect the input sequence into $\mathbf{u}$

$$\mathbf{u} = (u(0), u(1), \ldots, u(N-1))$$

Constraints on the $\mathbf{u}$ sequence (i.e., valve saturations, etc.) are covered extensively in Chapter 2. The constraints are the main feature that distinguishes MPC from the standard linear quadratic (LQ) control.

We first define an objective function $V(\cdot)$ to measure the deviation of the trajectory of $x(k), u(k)$ from zero by summing the weighted squares

$$V(x(0), \mathbf{u}) = \frac{1}{2}\sum_{k=0}^{N-1}\left[x(k)'Qx(k) + u(k)'Ru(k)\right] + \frac{1}{2}x(N)'P_f x(N)$$

subject to

$$x^+ = Ax + Bu$$

The objective function depends on the input sequence and state sequence. The initial state is available from the measurement. The remainder of the state trajectory, $x(k)$, $k = 1, \ldots, N$, is determined by the model and the input sequence $\mathbf{u}$. So we show the objective function's explicit dependence on the input sequence and initial state. The tuning parameters in the controller are the matrices Q and R. We allow the final state penalty to have a different weighting matrix, P_f, for generality. Large values of Q in comparison to R reflect the designer's intent to drive the state to the origin quickly at the expense of large control action. Penalizing the control action through large values of R relative to Q is the way to reduce the control action and slow down the rate at which the state approaches the origin. Choosing appropriate values of Q and R (i.e., tuning) is not always obvious, and this difficulty is one of the challenges faced by industrial practitioners of LQ control. Notice that MPC inherits this tuning challenge.

We then formulate the following optimal LQ control problem

$$\min_{\mathbf{u}} V(x(0), \mathbf{u}) \tag{1.6}$$

The Q, P_f, and R matrices often are chosen to be diagonal, but we do not assume that here. We assume, however, that Q, P_f, and R are *real and symmetric*; Q and P_f are *positive semidefinite*; and R is *positive definite*. These assumptions guarantee that the solution to the optimal control problem exists and is unique.

1.3.2 Optimizing Multistage Functions

We next provide a brief introduction to methods for solving multistage optimization problems like (1.6). Consider the set of variables w, x, y, and z, and the following function to be optimized

$$f(w, \underbracket{x) + g(x}, \overbracket{y) + h(y}, z)$$

Notice that the objective function has a special structure in which each stage's cost function in the sum depends only on adjacent variable pairs. For the first version of this problem, we consider w to be a fixed parameter, and we would like to solve the problem

$$\min_{x,y,z} f(w, x) + g(x, y) + h(y, z) \qquad w \text{ fixed}$$

One option is to optimize simultaneously over all three decision variables. Because of the objective function's special structure, however, we can obtain the solution by optimizing a sequence of three single-variable problems defined as follows

$$\min_x \left[f(w,x) + \min_y \left[g(x,y) + \min_z h(y,z) \right] \right]$$

We solve the inner problem over z first, and denote the optimal value and solution as follows

$$\underline{h}^0(y) = \min_z h(y,z) \qquad \underline{z}^0(y) = \arg\min_z h(y,z)$$

Notice that the optimal z and value function for this problem are both expressed as a function of the y variable. We then move to the next optimization problem and solve for the y variable

$$\min_y g(x,y) + \underline{h}^0(y)$$

and denote the solution and value function as

$$\underline{g}^0(x) = \min_y g(x,y) + \underline{h}^0(y) \qquad \underline{y}^0(x) = \arg\min_y g(x,y) + \underline{h}^0(y)$$

The optimal solution for y is a function of x, the remaining variable to be optimized. The third and final optimization is

$$\min_x f(w,x) + \underline{g}^0(x)$$

with solution and value function

$$\underline{f}^0(w) = \min_x f(w,x) + \underline{g}^0(x) \qquad \underline{x}^0(w) = \arg\min_x f(w,x) + \underline{g}^0(x)$$

We summarize the recursion with the following annotated equation

$$\underbrace{\min_x \left[f(w,x) + \overbrace{\min_y \left[g(x,y) + \underbrace{\min_z h(y,z)}_{\underline{h}^0(y),\, \underline{z}^0(y)} \right]}^{\underline{g}^0(x),\, \underline{y}^0(x)} \right]}_{\underline{f}^0(w),\, \underline{x}^0(w)}$$

If we are mainly interested in the first variable x, then the function $\underline{x}^0(w)$ is of primary interest and we have obtained this function quite efficiently. This nested solution approach is an example of a class of techniques known as dynamic programming (DP). DP was developed by Bellman (Bellman, 1957; Bellman and Dreyfus, 1962) as an efficient means for solving these kinds of multistage optimization problems. Bertsekas (1987) provides an overview of DP.

The version of the method we just used is called *backward* DP because we find the variables in reverse order: first z, then y, and finally x. Notice we find the optimal solutions as *functions* of the variables to be optimized at the next stage. If we wish to find the other variables y and z as a function of the known parameter w, then we nest the optimal solutions found by the backward DP recursion

$$\underset{\sim}{y}^0(w) = \underline{y}^0(\underline{x}^0(w)) \qquad \underset{\sim}{z}^0(w) = \underline{z}^0(\underset{\sim}{y}^0(w)) = \underline{z}^0(\underline{y}^0(\underline{x}^0(w)))$$

As we see shortly, backward DP is the method of choice for the regulator problem.

In the state estimation problem to be considered later in this chapter, w becomes a variable to be optimized, and z plays the role of a parameter. We wish to solve the problem

$$\min_{w,x,y} f(w,x) + g(x,y) + h(y,z) \qquad z \text{ fixed}$$

We can still break the problem into three smaller nested problems, but the order is reversed

$$\underbrace{\min_y \left[h(y,z) + \overbrace{\min_x \left[g(x,y) + \underbrace{\min_w f(w,x)}_{\overline{f}^0(x),\, \overline{w}^0(x)} \right]}^{\overline{g}^0(y),\, \overline{x}^0(y)} \right]}_{\overline{h}^0(z),\, \overline{y}^0(z)} \tag{1.7}$$

This form is called *forward* DP because we find the variables in the order given: first w, then x, and finally y. The optimal value functions and optimal solutions at each of the three stages are shown in (1.7). This version is preferable if we are primarily interested in finding the final variable y as a function of the parameter z. As before, if we need the other optimized variables x and w as a function of the parameter z, we must insert the optimal functions found by the forward DP recursion

$$\tilde{x}^0(z) = \overline{x}^0(\overline{y}^0(z)) \qquad \tilde{w}^0(z) = \overline{w}^0(\tilde{x}^0(z)) = \overline{w}^0(\overline{x}^0(\overline{y}^0(z)))$$

For the reader interested in trying some exercises to reinforce the concepts of DP, Exercise 1.15 considers finding the function $\tilde{w}^0(z)$ with

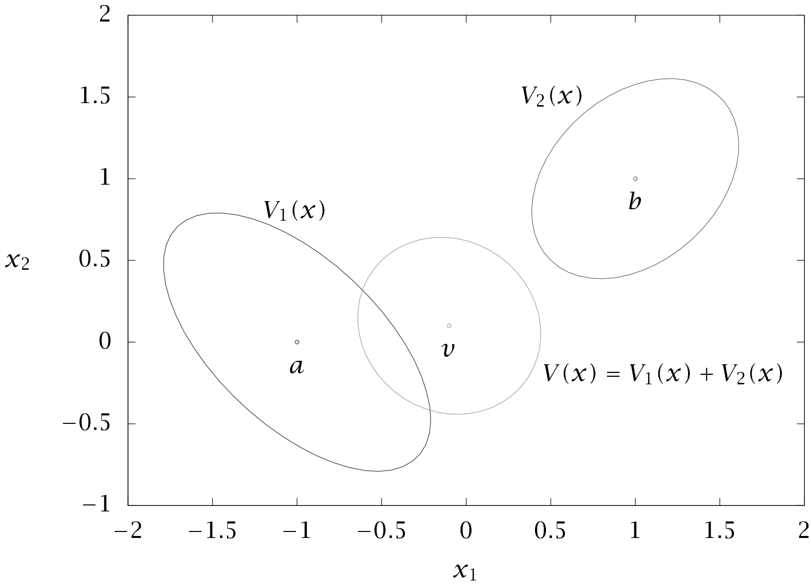

Figure 1.4: Level sets of two quadratic functions $V_1(x) = (1/4)$, $V_2(x) = (1/4)$, and their sum; $V(x) = V_1(x)+V_2(x) = 2$.

backward DP instead of forward DP as we just did here. Exercise C.1 discusses showing that the nested optimizations indeed give the same answer as simultaneous optimization over all decision variables.

Finally, if we optimize over all four variables, including the one considered as a fixed parameter in the two versions of DP we used, then we have two equivalent ways to express the value of the complete optimization

$$\min_{w,x,y,z} f(w,x) + g(x,y) + h(y,z) = \min_{w} \underline{f}^0(w) = \min_{z} \overline{h}^0(z)$$

The result in the next example proves useful in combining quadratic functions to solve the LQ problem.

Example 1.1: Sum of quadratic functions

Consider the two quadratic functions given by

$$V_1(x) = (1/2)(x-a)'A(x-a) \qquad V_2(x) = (1/2)(x-b)'B(x-b)$$

in which $A, B > 0$ are positive definite matrices and a and b are n-vectors locating the minimum of each function. Figure 1.4 displays the ellipses defined by the level sets $V_1(x) = 1/4$ and $V_2(x) = 1/4$ for the following data

$$A = \begin{bmatrix} 1.25 & 0.75 \\ 0.75 & 1.25 \end{bmatrix} \qquad a = \begin{bmatrix} -1 \\ 0 \end{bmatrix} \qquad B = \begin{bmatrix} 1.5 & -0.5 \\ -0.5 & 1.5 \end{bmatrix} \qquad b = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

(a) Show that the sum $V(x) = V_1(x) + V_2(x)$ is also quadratic

$$V(x) = (1/2)((x - v)'H(x - v) + d)$$

in which

$$\begin{gathered} H = A + B \qquad v = H^{-1}(Aa + Bb) \\ d = -(Aa + Bb)'H^{-1}(Aa + Bb) + a'Aa + b'Bb \end{gathered}$$

and verify the three ellipses given in Figure 1.4.

(b) Consider a generalization useful in the discussion of the upcoming regulation and state estimation problems. Let

$$V_1(x) = (1/2)(x-a)'A(x-a) \qquad V_2(x) = (1/2)(Cx-b)'B(Cx-b)$$

Derive the formulas for H, v, d for this case.

(c) Use the matrix inversion lemma (see Exercise 1.12) and show that $V(x)$ of part (b) can be expressed also in an inverse form, which is useful in state estimation problems

$$\begin{aligned} V(x) &= (1/2)(x - v)'\widetilde{H}^{-1}(x - v) + \text{constant} \\ \widetilde{H} &= A^{-1} - A^{-1}C'(CA^{-1}C' + B^{-1})^{-1}CA^{-1} \\ v &= a + A^{-1}C'(CA^{-1}C' + B^{-1})^{-1}(b - Ca) \end{aligned}$$

Solution

(a) The sum of two quadratics is also quadratic, so we parameterize the sum as

$$V(x) = (1/2)\left((x - v)'H(x - v) + d\right)$$

and solve for v, H, and d. Comparing the expansion of the quadratics of the right- and left-hand sides gives

$$x'Hx - 2x'Hv + v'Hv + d = x'(A+B)x - 2x'(Aa+Bb) + a'Aa + b'Bb$$

Equating terms at each order gives

$$\begin{aligned} H &= A + B \\ v &= H^{-1}(Aa + Bb) \\ d &= -v'Hv + a'Aa + b'Bb \\ &= -(Aa + Bb)'H^{-1}(Aa + Bb) + a'Aa + b'Bb \end{aligned}$$

Notice that H is positive definite since A and B are positive definite. Substituting the values of a, A, b, and B gives

$$H = \begin{bmatrix} 2.75 & 0.25 \\ 0.25 & 2.75 \end{bmatrix} \qquad v = \begin{bmatrix} -0.1 \\ 0.1 \end{bmatrix} \qquad d = 3.2$$

The level set $V(x) = 2$ is also plotted in Figure 1.4.

(b) Expanding and comparing terms as before, we obtain

$$\begin{aligned} H &= A + C'BC \\ v &= H^{-1}(Aa + C'Bb) \\ d &= -(Aa + C'Bb)'H^{-1}(Aa + C'Bb) + a'Aa + b'Bb \end{aligned} \tag{1.8}$$

Notice that H is positive definite since A is positive definite and $C'BC$ is positive semidefinite for any C.

(c) Define $\overline{x} = x - a$ and $\overline{b} = b - Ca$, and express the problem as

$$\begin{aligned} V(x) &= (1/2)\overline{x}'A\overline{x} + (1/2)(C(\overline{x} + a) - b)'B(C(\overline{x} + a) - b) \\ &= (1/2)\overline{x}'A\overline{x} + (1/2)(C\overline{x} - \overline{b})'B(C\overline{x} - \overline{b}) \end{aligned}$$

Apply the solution of part (b) to obtain

$$V(x) = (1/2)((\overline{x} - \overline{v})'H(\overline{x} - \overline{v}) + d)$$
$$H = A + C'BC \qquad \overline{v} = H^{-1}C'B\overline{b}$$

and we do not need to evaluate the constant d. From the matrix inversion lemma, use (1.54) on H and (1.55) on $\overline{v}$ to obtain

$$\begin{aligned} \widetilde{H} &= A^{-1} - A^{-1}C'(CA^{-1}C' + B^{-1})^{-1}CA^{-1} \\ \overline{v} &= A^{-1}C'(CA^{-1}C' + B^{-1})^{-1}\overline{b} \end{aligned}$$

The function $V(x)$ is then given by

$$\begin{aligned} V(x) &= (1/2)((\overline{x}-\overline{v})'\widetilde{H}^{-1}(\overline{x}-\overline{v})+d) \\ V(x) &= (1/2)((x-(a+\overline{v}))'\widetilde{H}^{-1}(x-(a+\overline{v}))+d) \\ V(x) &= (1/2)((x-v)'\widetilde{H}^{-1}(x-v)+d) \end{aligned}$$

in which

$$v = a + A^{-1}C'(CA^{-1}C' + B^{-1})^{-1}\,(b - Ca) \qquad \square$$

1.3.3 Dynamic Programming Solution

After this brief introduction to DP, we apply it to solve the LQ control problem. We first rewrite (1.6) in the following form to see the structure clearly

$$V(x(0), \mathbf{u}) = \sum_{k=0}^{N-1} \ell(x(k), u(k)) + \ell_N(x(N)) \qquad \text{s.t. } x^+ = Ax + Bu$$

in which the *stage cost* $\ell(x,u) = (1/2)(x'Qx + u'Ru), k = 0, \ldots, N-1$ and the terminal stage cost $\ell_N(x) = (1/2)x'P_f x$. Since $x(0)$ is known, we choose *backward* DP as the convenient method to solve this problem. We first rearrange the overall objective function so we can optimize over input $u(N-1)$ and state $x(N)$

$$\begin{aligned} \min_{u(0),x(1),\ldots u(N-2),x(N-1)} & \ell(x(0),u(0)) + \ell(x(1),u(1)) + \cdots + \\ & \min_{u(N-1),x(N)} \ell(x(N-1),u(N-1)) + \ell_N(x(N)) \end{aligned}$$

subject to

$$x(k+1) = Ax(k) + Bu(k) \qquad k = 0, \ldots N-1$$

The problem to be solved at the last stage is

$$\min_{u(N-1),x(N)} \ell(x(N-1),u(N-1)) + \ell_N(x(N)) \tag{1.9}$$

subject to

$$x(N) = Ax(N-1) + Bu(N-1)$$

in which $x(N-1)$ appears in this stage as a parameter. We denote the optimal cost by $V_{N-1}^0(x(N-1))$ and the optimal decision variables by $u_{N-1}^0(x(N-1))$ and $x_N^0(x(N-1))$. The optimal cost and decisions at the last stage are parameterized by the state at the previous stage as we expect in backward DP. We next solve this optimization. First we substitute the state equation for $x(N)$ and combine the two quadratic terms using (1.8)

$$\begin{aligned}
&\ell(x(N-1), u(N-1)) + \ell_N(x(N)) \\
&\quad = (1/2)\left(|x(N-1)|_Q^2 + |u(N-1)|_R^2 + |Ax(N-1) + Bu(N-1)|_{P_f}^2\right) \\
&\quad = (1/2)\left(|x(N-1)|_Q^2 + |(u(N-1) - v)|_H^2 + d\right)
\end{aligned}$$

in which

$$\begin{aligned}
H &= R + B'P_fB \\
v &= -(B'P_fB + R)^{-1}B'P_fA\, x(N-1) \\
d &= x(N-1)'\left(A'P_fA - A'P_fB(B'P_fB + R)^{-1}B'P_fA\right)x(N-1)
\end{aligned}$$

Given this form of the cost function, we see by inspection that the optimal input for $u(N-1)$ is v, so the optimal control law at stage $N-1$ is a *linear* function of the state $x(N-1)$. Then using the model equation, the optimal final state is also a linear function of state $x(N-1)$. The optimal cost is $(1/2)(|x(N-1)|_Q^2 + d)$, which makes the optimal cost a quadratic function of $x(N-1)$. Summarizing, for all x

$$\begin{aligned}
u_{N-1}^0(x) &= K(N-1)\, x \\
x_N^0(x) &= (A + BK(N-1))\, x \\
V_{N-1}^0(x) &= (1/2)x'\, \Pi(N-1)\, x
\end{aligned}$$

with the definitions

$$\begin{aligned}
K(N-1) &:= -(B'P_fB + R)^{-1}B'P_fA \\
\Pi(N-1) &:= Q + A'P_fA - A'P_fB(B'P_fB + R)^{-1}B'P_fA
\end{aligned}$$

The function $V_{N-1}^0(x)$ defines the optimal *cost to go* from state x for the last stage under the optimal control law $u_{N-1}^0(x)$. Having this function allows us to move to the next stage of the DP recursion. For the next stage we solve the optimization

$$\min_{u(N-2), x(N-1)} \ell(x(N-2), u(N-2)) + V_{N-1}^0(x(N-1))$$

subject to

$$x(N-1) = Ax(N-2) + Bu(N-2)$$

Notice that this problem is identical in structure to the stage we just solved, (1.9), and we can write out the solution by simply renaming variables

$$\begin{aligned}
u^0_{N-2}(x) &= K(N-2)\, x \\
x^0_{N-1}(x) &= (A + BK(N-2))\, x \\
V^0_{N-2}(x) &= (1/2)x'\, \Pi(N-2)\, x \\
K(N-2) &:= -(B'\Pi(N-1)B + R)^{-1}B'\Pi(N-1)A \\
\Pi(N-2) &:= Q + A'\Pi(N-1)A - \\
&\qquad A'\Pi(N-1)B(B'\Pi(N-1)B + R)^{-1}B'\Pi(N-1)A
\end{aligned}$$

The recursion from $\Pi(N-1)$ to $\Pi(N-2)$ is known as a backward Riccati iteration. To summarize, the backward Riccati iteration is defined as follows

$$\Pi(k-1) = Q + A'\Pi(k)A - A'\Pi(k)B\left(B'\Pi(k)B + R\right)^{-1}B'\Pi(k)A \qquad k = N, N-1, \ldots, 1 \tag{1.10}$$

with terminal condition

$$\Pi(N) = P_f \tag{1.11}$$

The terminal condition replaces the typical initial condition because the iteration is running backward. The optimal control policy at each stage is

$$u^0_k(x) = K(k)x \qquad k = N-1, N-2, \ldots, 0 \tag{1.12}$$

The optimal gain at time k is computed from the Riccati matrix at time $k+1$

$$K(k) = -\left(B'\Pi(k+1)B + R\right)^{-1}B'\Pi(k+1)A \qquad k = N-1, N-2, \ldots, 0 \tag{1.13}$$

and the optimal cost to go from time k to time N is

$$V^0_k(x) = (1/2)x'\Pi(k)x \qquad k = N, N-1, \ldots, 0 \tag{1.14}$$

1.3.4 The Infinite Horizon LQ Problem

Let us motivate the infinite horizon problem by showing a weakness of the finite horizon problem. Kalman (1960b, p.113) pointed out in his classic 1960 paper that optimality does not ensure stability.

> In the engineering literature it is often assumed (tacitly and incorrectly) that a system with optimal control law (6.8) is necessarily stable.

Assume that we use as our control law the first feedback gain of the finite horizon problem, $K(0)$

$$u(k) = K(0)x(k)$$

Then the stability of the closed-loop system is determined by the eigenvalues of $A+BK(0)$. We now construct an example that shows choosing $Q > 0$, $R > 0$, and $N \geq 1$ does not ensure stability. In fact, we can find reasonable values of these parameters such that the controller destabilizes a stable system.[2] Let

$$A = \begin{bmatrix} 4/3 & -2/3 \\ 1 & 0 \end{bmatrix} \qquad B = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \qquad C = [-2/3 \ \ 1]$$

This system is chosen so that $G(z)$ has a *zero* at $z = 3/2$, i.e., an unstable zero. We now construct an LQ controller that inverts this zero and hence produces an unstable system. We would like to choose $Q = C'C$ so that y itself is penalized, but that Q is only semidefinite. We add a small positive definite piece to $C'C$ so that Q is positive definite, and choose a *small* positive R penalty (to encourage the controller to misbehave), and $N = 5$

$$Q = C'C + 0.001I = \begin{bmatrix} 4/9 + .001 & -2/3 \\ -2/3 & 1.001 \end{bmatrix} \qquad R = 0.001$$

We now iterate the Riccati equation four times starting from $\Pi = P_f = Q$ and compute $K(0)$ for $N = 5$; then we compute the eigenvalues of $A + BK(0)$ and achieve[3]

$$\text{eig}(A + BK_5(0)) = \{1.307, 0.001\}$$

[2] In Chapter 2, we present several controller design methods that prevent this kind of instability.

[3] Please check this answer with Octave or MATLAB.

Using this controller the closed-loop system evolution is $x(k) = (A + BK_5(0))^k x_0$. Since an eigenvalue of $A + BK_5(0)$ is greater than unity, $x(k) \to \infty$ as $k \to \infty$. In other words the closed-loop system is unstable.

If we continue to iterate the Riccati equation, which corresponds to increasing the horizon in the controller, we obtain for $N = 7$

$$\text{eig}(A + BK_7(0)) = \{0.989, 0.001\}$$

and the controller is stabilizing. If we continue iterating the Riccati equation, we converge to the following steady-state closed-loop eigenvalues

$$\text{eig}(A + BK_\infty(0)) = \{0.664, 0.001\}$$

This controller corresponds to an infinite horizon control law. Notice that it is stabilizing and has a reasonable stability margin. Nominal stability is a guaranteed property of infinite horizon controllers as we prove in the next section.

With this motivation, we are led to consider directly the infinite horizon case

$$V(x(0), \mathbf{u}) = \frac{1}{2} \sum_{k=0}^{\infty} x(k)'Qx(k) + u(k)'Ru(k) \tag{1.15}$$

in which $x(k)$ is the solution at time k of $x^+ = Ax + Bu$ if the initial state is $x(0)$ and the input sequence is $\mathbf{u}$. If we are interested in a continuous process (i.e., no final time), then the natural cost function is an infinite horizon cost. If we were truly interested in a batch process (i.e., the process does stop at $k = N$), then stability is not a relevant property, and we naturally would use the finite horizon LQ controller and the *time-varying* controller, $u(k) = K(k)x(k), k = 0, 1, \ldots, N$.

In considering the infinite horizon problem, we first restrict attention to systems for which there exist input sequences that give bounded cost. Consider the case $A = I$ and $B = 0$, for example. Regardless of the choice of input sequence, (1.15) is unbounded for $x(0) \neq 0$. It seems clear that we are not going to stabilize an unstable system ($A = I$) without any input ($B = 0$). This is an example of an *uncontrollable* system. In order to state the sharpest results on stabilization, we require the concepts of controllability, stabilizability, observability, and detectability. We shall define these concepts subsequently.

1.3.5 Controllability

A system is *controllable* if, for any pair of states x, z in the state space, z can be reached in finite time from x (or x controlled to z) (Sontag, 1998, p.83). A *linear discrete time* system $x^+ = Ax + Bu$ is therefore controllable if there exists a finite time N and a sequence of inputs

$$(u(0), u(1), \dots u(N-1))$$

that can transfer the system from any x to any z in which

$$z = A^N x + \begin{bmatrix} B & AB & \cdots & A^{N-1}B \end{bmatrix} \begin{bmatrix} u(N-1) \\ u(N-2) \\ \vdots \\ u(0) \end{bmatrix}$$

We can simplify this condition by noting that the matrix powers A^k for $k \geq n$ are expressible as linear combinations of the powers 0 to $n-1$. This result is a consequence of the Cayley-Hamilton theorem (Horn and Johnson, 1985, pp. 86–87). Therefore the range of the matrix $\begin{bmatrix} B & AB & \cdots & A^{N-1}B \end{bmatrix}$ for $N \geq n$ is the same as $\begin{bmatrix} B & AB & \cdots & A^{n-1}B \end{bmatrix}$. In other words, for an unconstrained linear system, if we cannot reach z in n moves, we cannot reach z in any number of moves. The question of *controllability* of a linear time-invariant system is therefore a question of *existence* of solutions to linear equations for an arbitrary right-hand side

$$\begin{bmatrix} B & AB & \cdots & A^{n-1}B \end{bmatrix} \begin{bmatrix} u(n-1) \\ u(n-2) \\ \vdots \\ u(0) \end{bmatrix} = z - A^n x$$

The matrix appearing in this equation is known as the *controllability matrix C*

$$C = \begin{bmatrix} B & AB & \cdots & A^{n-1}B \end{bmatrix} \tag{1.16}$$

From the fundamental theorem of linear algebra, we know a solution exists for all right-hand sides if and only if the *rows* of the $n \times nm$ controllability matrix are linearly independent.[4] Therefore, the system (A, B) is controllable if and only if

$$\text{rank}(C) = n$$

[4] See Section A.4 of Appendix A or (Strang, 1980, pp.87–88) for a review of this result.

The following result for checking controllability also proves useful (Hautus, 1972).

Lemma 1.2 (Hautus lemma for controllability). *A system is controllable if and only if*

$$\operatorname{rank}\begin{bmatrix} \lambda I - A & B \end{bmatrix} = n \qquad \textit{for all } \lambda \in \mathbb{C} \tag{1.17}$$

in which $\mathbb{C}$ is the set of complex numbers.

Notice that the first n columns of the matrix in (1.17) are linearly independent if λ is not an eigenvalue of A, so (1.17) is equivalent to checking the rank at just the eigenvalues of A

$$\operatorname{rank}\begin{bmatrix} \lambda I - A & B \end{bmatrix} = n \qquad \text{for all } \lambda \in \operatorname{eig}(A)$$

1.3.6 Convergence of the Linear Quadratic Regulator

We now show that the infinite horizon regulator asymptotically stabilizes the origin for the closed-loop system. Define the infinite horizon objective function

$$V(x, \mathbf{u}) = \frac{1}{2}\sum_{k=0}^{\infty} x(k)'Qx(k) + u(k)'Ru(k)$$

subject to

$$\begin{aligned} x^+ &= Ax + Bu \\ x(0) &= x \end{aligned}$$

with $Q, R > 0$. If (A, B) is controllable, the solution to the optimization problem

$$\min_{\mathbf{u}} V(x, \mathbf{u})$$

exists and is unique for all x. We denote the optimal solution by $\mathbf{u}^0(x)$, and the first input in the optimal sequence by $u^0(x)$. The feedback control law $\kappa_\infty(\cdot)$ for this infinite horizon case is then defined as $u = \kappa_\infty(x)$ in which $\kappa_\infty(x) = u^0(x) = \mathbf{u}^0(0; x)$. As stated in the following lemma, this infinite horizon linear quadratic regulator (LQR) is stabilizing.

Lemma 1.3 (LQR convergence). *For (A, B) controllable, the infinite horizon LQR with $Q, R > 0$ gives a convergent closed-loop system*

$$x^+ = Ax + B\kappa_\infty(x)$$

Proof. The cost of the infinite horizon objective is bounded above for all $x(0)$ because (A, B) is controllable. Controllability implies that there exists a sequence of n inputs $(u(0), u(1), \ldots, u(n-1))$ that transfers the state from any $x(0)$ to $x(n) = 0$. A zero control sequence after $k = n$ for $(u(n+1), u(n+2), \ldots)$ generates zero cost for all terms in V after $k = n$, and the objective function for this infinite control sequence is therefore finite. The cost function is strictly convex in $\mathbf{u}$ because $R > 0$ so the solution to the optimization is unique.

If we consider the sequence of costs to go along the closed-loop trajectory, we have

$$V_{k+1} = V_k - (1/2)\left(x(k)'Qx(k) + u(k)'Ru(k)\right)$$

in which $V_k = V^0(x(k))$ is the cost at time k for state value $x(k)$ and $u(k) = u^0(x(k))$ is the optimal control for state $x(k)$. The cost along the closed-loop trajectory is nonincreasing and bounded below (by zero). Therefore, the sequence (V_k) converges and

$$x(k)'Qx(k) \to 0 \qquad u(k)'Ru(k) \to 0 \qquad \text{as } k \to \infty$$

Since $Q, R > 0$, we have

$$x(k) \to 0 \qquad u(k) \to 0 \qquad \text{as } k \to \infty$$

and closed-loop convergence is established. ■

In fact we know more. From the previous sections, we know the optimal solution is found by iterating the Riccati equation, and the optimal infinite horizon control law and optimal cost are given by

$$u^0(x) = Kx \qquad V^0(x) = (1/2)x'\Pi x$$

in which

$$\begin{aligned} K &= -(B'\Pi B + R)^{-1}B'\Pi A \\ \Pi &= Q + A'\Pi A - A'\Pi B(B'\Pi B + R)^{-1}B'\Pi A \end{aligned} \tag{1.18}$$

Proving Lemma 1.3 has shown also that for (A, B) controllable and Q, $R > 0$, a positive definite solution to the discrete algebraic Riccati equation (DARE), (1.18), exists and the eigenvalues of $(A+BK)$ are asymptotically stable for the K corresponding to this solution (Bertsekas, 1987, pp.58–64).

This basic approach to establishing regulator stability will be generalized in Chapter 2 to handle constrained and nonlinear systems, so it is helpful for the new student to first become familiar with these ideas in the unconstrained, linear setting. For linear systems, asymptotic convergence is equivalent to asymptotic stability, and we delay the discussion of stability until Chapter 2. In Chapter 2 the optimal cost is shown to be a Lyapunov function for the closed-loop system. We also can strengthen the stability for linear systems from asymptotic stability to exponential stability based on the form of the Lyapunov function.

The LQR convergence result in Lemma 1.3 is the simplest to establish, but we can enlarge the class of systems and penalties for which closed-loop stability is guaranteed. The system restriction can be weakened from controllability to *stabilizability*, which is discussed in Exercises 1.19 and 1.20. The restriction on the allowable state penalty Q can be weakened from $Q > 0$ to $Q \geq 0$ and (A, Q) *detectable*, which is also discussed in Exercise 1.20. The restriction $R > 0$ is retained to ensure uniqueness of the control law. In applications, if one cares little about the cost of the control, then R is chosen to be small, but positive definite.

1.4 Introductory State Estimation

The next topic is state estimation. In most applications, the variables that are conveniently or economically measurable (y) are a small subset of the variables required to model the system (x). Moreover, the measurement is corrupted with sensor noise and the state evolution is corrupted with process noise. Determining a good state estimate for use in the regulator in the face of a noisy and incomplete output measurement is a challenging task. That is the challenge of state estimation.

To fully appreciate the fundamentals of state estimation, we must address the fluctuations in the data. Probability theory has proven itself as the most successful and versatile approach to modeling these fluctuations. In this section we introduce the probability fundamentals necessary to develop an optimal state estimator in the simplest possible setting: a linear discrete time model subject to normally distributed process and measurement noise. This optimal state estimator is known as the Kalman filter (Kalman, 1960a). In Chapter 4 we revisit the state estimation problem in a much wider setting, and consider nonlinear models and constraints on the system that preclude an analytical solution such as the Kalman filter. The probability theory presented here is also preparation for understanding that chapter.

1.4.1 Linear Systems and Normal Distributions

This section summarizes the probability and random variable results required for deriving a linear optimal estimator such as the Kalman filter. We assume that the reader is familiar with the concepts of a random variable, probability density and distribution, the multivariate normal distribution, mean and variance, statistical independence, and conditional probability. Readers unfamiliar with these terms should study the material in Appendix A before reading this and the next sections.

In the following discussion let x, y, and z be vectors of random variables. We use the notation

$$x \sim N(m, P)$$
$$p_x(x) = n(x, m, P)$$

to denote random variable x is normally distributed with mean m and covariance (or simply variance) P, in which

$$n(x, m, P) = \frac{1}{(2\pi)^{n/2}(\det P)^{1/2}} \exp\left[-\frac{1}{2}(x-m)'P^{-1}(x-m)\right] \tag{1.19}$$

and $\det P$ denotes the determinant of matrix P. Note that if $x \in \mathbb{R}^n$, then $m \in \mathbb{R}^n$ and $P \in \mathbb{R}^{n\times n}$ is a positive definite matrix. We require three main results. The simplest version can be stated as follows.

Joint independent normals. If x and y are normally distributed and (statistically) independent[5]

$$x \sim N(m_x, P_x) \qquad y \sim N(m_y, P_y)$$

then their joint density is given by

$$p_{x,y}(x, y) = n(x, m_x, P_x)\; n(y, m_y, P_y)$$

$$\begin{bmatrix} x \\ y \end{bmatrix} \sim N\left(\begin{bmatrix} m_x \\ m_y \end{bmatrix}, \begin{bmatrix} P_x & 0 \\ 0 & P_y \end{bmatrix}\right) \tag{1.20}$$

Note that, depending on convenience, we use both (x, y) and the vector $\left[\begin{smallmatrix} x \\ y \end{smallmatrix}\right]$ to denote the pair of random variables.

Linear transformation of a normal. If x is normally distributed with mean m and variance P, and y is a linear transformation of x, $y = Ax$, then y is distributed with mean Am and variance APA'

$$x \sim N(m, P) \qquad y = Ax \qquad y \sim N(Am, APA') \tag{1.21}$$

[5] We may emphasize that two vectors of random variables are independent using *statistically independent* to distinguish this concept from linear independence of vectors.

Conditional of a joint normal. If x and y are jointly normally distributed as

$$\begin{bmatrix} x \\ y \end{bmatrix} \sim N\left(\begin{bmatrix} m_x \\ m_y \end{bmatrix} \begin{bmatrix} P_x & P_{xy} \\ P_{yx} & P_y \end{bmatrix} \right)$$

then the conditional density of x given y is also normal

$$p_{x|y}(x|y) = n(x, m, P) \tag{1.22}$$

in which the mean is

$$m = m_x + P_{xy} P_y^{-1} (y - m_y)$$

and the covariance is

$$P = P_x - P_{xy} P_y^{-1} P_{yx}$$

Note that the conditional mean m is itself a random variable because it depends on the random variable y.

To derive the optimal estimator, we actually require these three main results conditioned on additional random variables. The analogous results are the following.

Joint independent normals. If $p_{x|z}(x|z)$ is normal, and y is statistically independent of x and z and normally distributed

$$p_{x|z}(x|z) = n(x, m_x, P_x)$$
$$y \sim N(m_y, P_y) \qquad y \text{ independent of } x \text{ and } z$$

then the conditional joint density of (x, y) given z is

$$p_{x,y|z}(x, y|z) = n(x, m_x, P_x)\; n(y, m_y, P_y)$$
$$p_{x,y|z}\left(\begin{bmatrix} x \\ y \end{bmatrix} \middle| z \right) = n\left(\begin{bmatrix} x \\ y \end{bmatrix}, \begin{bmatrix} m_x \\ m_y \end{bmatrix}, \begin{bmatrix} P_x & 0 \\ 0 & P_y \end{bmatrix} \right) \tag{1.23}$$

Linear transformation of a normal.

$$p_{x|z}(x|z) = n(x, m, P) \qquad y = Ax$$
$$p_{y|z}(y|z) = n(y, Am, APA') \tag{1.24}$$

Conditional of a joint normal. If x and y are jointly normally distributed as

$$p_{x,y|z}\left(\begin{bmatrix} x \\ y \end{bmatrix} \middle| z\right) = n\left(\begin{bmatrix} x \\ y \end{bmatrix}, \begin{bmatrix} m_x \\ m_y \end{bmatrix}, \begin{bmatrix} P_x & P_{xy} \\ P_{yx} & P_y \end{bmatrix}\right)$$

then the conditional density of x given y, z is also normal

$$p_{x|y,z}(x|y,z) = n(x, m, P) \tag{1.25}$$

in which

$$\begin{aligned} m &= m_x + P_{xy}P_y^{-1}(y - m_y) \\ P &= P_x - P_{xy}P_y^{-1}P_{yx} \end{aligned}$$

1.4.2 Linear Optimal State Estimation

We start by assuming the initial state $x(0)$ is normally distributed with some mean and covariance

$$x(0) \sim N(\overline{x}(0), Q(0))$$

In applications, we often do not know $\overline{x}(0)$ or $Q(0)$. In such cases we often set $\overline{x}(0) = 0$ and choose a large value for $Q(0)$ to indicate our lack of prior knowledge. The choice of a large variance prior forces the upcoming $y(k)$ measurements to determine the state estimate $\hat{x}(k)$.

Combining the measurement. We obtain noisy measurement $y(0)$ satisfying

$$y(0) = Cx(0) + v(0)$$

in which $v(0) \sim N(0, R)$ is the measurement noise. If the measurement process is quite noisy, then R is large. If the measurements are highly accurate, then R is small. We choose a zero mean for v because all of the deterministic effects with nonzero mean are considered part of the model, and the measurement noise reflects what is left after all these other effects have been considered. Given the measurement $y(0)$, we want to obtain the conditional density $p_{x(0)|y(0)}(x(0)|y(0))$. This conditional density describes the change in our knowledge about $x(0)$ after we obtain measurement $y(0)$. This step is the essence of state estimation. To derive this conditional density, first consider the pair of variables $(x(0), y(0))$ given as

$$\begin{bmatrix} x(0) \\ y(0) \end{bmatrix} = \begin{bmatrix} I & 0 \\ C & I \end{bmatrix} \begin{bmatrix} x(0) \\ v(0) \end{bmatrix}$$

We assume that the noise $v(0)$ is statistically independent of $x(0)$, and use the independent joint normal result (1.20) to express the joint density of $(x(0), v(0))$

$$\begin{bmatrix} x(0) \\ v(0) \end{bmatrix} \sim N\left(\begin{bmatrix} \overline{x}(0) \\ 0 \end{bmatrix}, \begin{bmatrix} Q(0) & 0 \\ 0 & R \end{bmatrix} \right)$$

From the previous equation, the pair $(x(0), y(0))$ is a linear transformation of the pair $(x(0), v(0))$. Therefore, using the linear transformation of normal result (1.21), and the density of $(x(0), v(0))$ gives the density of $(x(0), y(0))$

$$\begin{bmatrix} x(0) \\ y(0) \end{bmatrix} \sim N\left(\begin{bmatrix} \overline{x}(0) \\ C\overline{x}(0) \end{bmatrix}, \begin{bmatrix} Q(0) & Q(0)C' \\ CQ(0) & CQ(0)C' + R \end{bmatrix} \right)$$

Given this joint density, we then use the conditional of a joint normal result (1.22) to obtain

$$p_{x(0)|y(0)}\left(x(0)|y(0)\right) = n\left(x(0), m, P\right)$$

in which

$$\begin{aligned} m &= \overline{x}(0) + L(0)\left(y(0) - C\overline{x}(0)\right) \\ L(0) &= Q(0)C'(CQ(0)C' + R)^{-1} \\ P &= Q(0) - Q(0)C'(CQ(0)C' + R)^{-1}CQ(0) \end{aligned}$$

We see that the conditional density $p_{x(0)|y(0)}$ is normal. The *optimal* state estimate is the value of $x(0)$ that maximizes this conditional density. For a normal, that is the mean, and we choose $\hat{x}(0) = m$. We also denote the variance in this conditional after measurement $y(0)$ by $P(0) = P$ with P given in the previous equation. The change in variance after measurement ($Q(0)$ to $P(0)$) quantifies the information increase by obtaining measurement $y(0)$. The variance after measurement, $P(0)$, is always less than or equal to $Q(0)$, which implies that we can only gain information by measurement; but the information gain may be small if the measurement device is poor and the measurement noise variance R is large.

Forecasting the state evolution. Next we consider the state evolution from $k = 0$ to $k = 1$, which satisfies

$$x(1) = \begin{bmatrix} A & I \end{bmatrix} \begin{bmatrix} x(0) \\ w(0) \end{bmatrix}$$

in which $w(0) \sim N(0, Q)$ is the process noise. If the state is subjected to large disturbances, then Q is large, and if the disturbances are small, Q is small. Again we choose zero mean for w because the nonzero-mean disturbances should have been accounted for in the system model. We next calculate the conditional density $p_{x(1)|y(0)}$. Now we require the conditional version of the joint density $(x(0), w(0))$. We assume that the process noise $w(0)$ is statistically independent of both $x(0)$ and $v(0)$, hence it is also independent of $y(0)$, which is a linear combination of $x(0)$ and $v(0)$. Therefore we use (1.23) to obtain

$$\begin{bmatrix} x(0) \\ w(0) \end{bmatrix} \sim N\left(\begin{bmatrix} \hat{x}(0) \\ 0 \end{bmatrix}, \begin{bmatrix} P(0) & 0 \\ 0 & Q \end{bmatrix} \right)$$

We then use the conditional version of the linear transformation of a normal (1.24) to obtain

$$p_{x(1)|y(0)}(x(1)|y(0)) = n(x(1), \hat{x}^-(1), P^-(1))$$

in which the mean and variance are

$$\hat{x}^-(1) = A\hat{x}(0) \qquad P^-(1) = AP(0)A' + Q$$

We see that forecasting forward one time step may increase or decrease the conditional variance of the state. If the eigenvalues of A are less than unity, for example, the term $AP(0)A'$ *may* be smaller than $P(0)$, but the process noise Q adds a positive contribution. If the system is unstable, $AP(0)A'$ *may* be larger than $P(0)$, and then the conditional variance definitely increases upon forecasting. See also Exercise 1.27 for further discussion of this point.

Given that $p_{x(1)|y(0)}$ is also a normal, we are situated to add measurement $y(1)$ and continue the process of adding measurements followed by forecasting forward one time step until we have processed all the available data. Because this process is recursive, the storage requirements are small. We need to store only the current state estimate and variance, and can discard the measurements as they are processed. The required online calculation is minor. These features make the optimal linear estimator an ideal candidate for rapid online application. We next summarize the state estimation recursion.

Summary. Denote the measurement trajectory by

$$\mathbf{y}(k) := (y(0), y(1), \ldots y(k))$$

At time k the conditional density with data $\mathbf{y}(k-1)$ is normal

$$p_{x(k)|\mathbf{y}(k-1)}(x(k)|\mathbf{y}(k-1)) = n(x(k), \hat{x}^-(k), P^-(k))$$

and we denote the mean and variance with a superscript minus to indicate these are the statistics *before* measurement $y(k)$. At $k=0$, the recursion starts with $\hat{x}^-(0) = \overline{x}(0)$ and $P^-(0) = Q(0)$ as discussed previously. We obtain measurement $y(k)$ which satisfies

$$\begin{bmatrix} x(k) \\ y(k) \end{bmatrix} = \begin{bmatrix} I & 0 \\ C & I \end{bmatrix} \begin{bmatrix} x(k) \\ v(k) \end{bmatrix}$$

The density of $(x(k), v(k))$ follows from (1.23) since measurement noise $v(k)$ is independent of $x(k)$ and $\mathbf{y}(k-1)$

$$\begin{bmatrix} x(k) \\ v(k) \end{bmatrix} \sim N\left(\begin{bmatrix} \hat{x}^-(k) \\ 0 \end{bmatrix}, \begin{bmatrix} P^-(k) & 0 \\ 0 & R \end{bmatrix}\right)$$

Equation (1.24) then gives the joint density

$$\begin{bmatrix} x(k) \\ y(k) \end{bmatrix} \sim N\left(\begin{bmatrix} \hat{x}^-(k) \\ C\hat{x}^-(k) \end{bmatrix}, \begin{bmatrix} P^-(k) & P^-(k)C' \\ CP^-(k) & CP^-(k)C' + R \end{bmatrix}\right)$$

We note $(\mathbf{y}(k-1), y(k)) = \mathbf{y}(k)$, and using the conditional density result (1.25) gives

$$p_{x(k)|\mathbf{y}(k)}\left(x(k)|\mathbf{y}(k)\right) = n\left(x(k), \hat{x}(k), P(k)\right)$$

in which

$$\begin{aligned}
\hat{x}(k) &= \hat{x}^-(k) + L(k)\left(y(k) - C\hat{x}^-(k)\right) \\
L(k) &= P^-(k)C'(CP^-(k)C' + R)^{-1} \\
P(k) &= P^-(k) - P^-(k)C'(CP^-(k)C' + R)^{-1}CP^-(k)
\end{aligned}$$

We forecast from k to $k+1$ using the model

$$x(k+1) = \begin{bmatrix} A & I \end{bmatrix} \begin{bmatrix} x(k) \\ w(k) \end{bmatrix}$$

Because $w(k)$ is independent of $x(k)$ and $\mathbf{y}(k)$, the joint density of $(x(k), w(k))$ follows from a second use of (1.23)

$$\begin{bmatrix} x(k) \\ w(k) \end{bmatrix} \sim N\left(\begin{bmatrix} \hat{x}(k) \\ 0 \end{bmatrix}, \begin{bmatrix} P(k) & 0 \\ 0 & Q \end{bmatrix}\right)$$

and a second use of the linear transformation result (1.24) gives

$$p_{x(k+1)|\mathbf{y}(k)}(x(k+1)|\mathbf{y}(k)) = n(x(k+1), \hat{x}^-(k+1), P^-(k+1))$$

in which

$$\begin{aligned} \hat{x}^-(k+1) &= A\hat{x}(k) \\ P^-(k+1) &= AP(k)A' + Q \end{aligned}$$

and the recursion is complete.

1.4.3 Least Squares Estimation

We next consider the state estimation problem as a deterministic optimization problem rather than an exercise in maximizing conditional density. This viewpoint proves valuable in Chapter 4 when we wish to add constraints to the state estimator. Consider a time horizon with measurements $y(k), k = 0, 1, \ldots, T$. We consider the prior information to be our best initial guess of the initial state $x(0)$, denoted $\overline{x}(0)$, and weighting matrices $P^-(0)$, Q, and R for the initial state, process disturbance, and measurement disturbance. A reasonably flexible choice for objective function is

$$\begin{aligned} V_T(\mathbf{x}(T)) = \frac{1}{2}\Bigg(&|x(0) - \overline{x}(0)|^2_{(P^-(0))^{-1}} + \\ &\sum_{k=0}^{T-1} |x(k+1) - Ax(k)|^2_{Q^{-1}} + \sum_{k=0}^{T} |y(k) - Cx(k)|^2_{R^{-1}} \Bigg) \end{aligned} \tag{1.26}$$

in which $\mathbf{x}(T) := (x(0), x(1), \ldots, x(T))$. We claim and then show that the following (deterministic) least squares optimization problem produces the same result as the conditional density function maximization of the Kalman filter

$$\min_{\mathbf{x}(T)} V_T(\mathbf{x}(T)) \tag{1.27}$$

Game plan. Using forward DP, we can decompose and solve recursively the least squares state estimation problem. To see clearly how the procedure works, first we write out the terms in the state estimation least squares problem (1.27)

$$\min_{x(0),\ldots,x(T)} \frac{1}{2}\Big(|x(0)-\overline{x}(0)|^2_{(P^-(0))^{-1}} + |y(0)-Cx(0)|^2_{R^{-1}} + |x(1)-Ax(0)|^2_{Q^{-1}} + |y(1)-Cx(1)|^2_{R^{-1}} + |x(2)-Ax(1)|^2_{Q^{-1}} + \cdots + |x(T)-Ax(T-1)|^2_{Q^{-1}} + |y(T)-Cx(T)|^2_{R^{-1}}\Big) \tag{1.28}$$

We decompose this T-stage optimization problem with forward DP. First we combine the prior and the measurement $y(0)$ into the quadratic function $V_0(x(0))$ as shown in the following equation

$$\min_{x(T),\ldots,x(1)} \overbrace{\min_{x(0)} \frac{1}{2}\Big(\underbrace{|x(0)-\overline{x}(0)|^2_{(P^-(0))^{-1}} + |y(0)-Cx(0)|^2_{R^{-1}}}_{\text{combine } V_0(x(0))} + |x(1)-Ax(0)|^2_{Q^{-1}}}^{\text{arrival cost } V_1^-(x(1))} + |y(1)-Cx(1)|^2_{R^{-1}} + |x(2)-Ax(1)|^2_{Q^{-1}} + \cdots + |x(T)-Ax(T-1)|^2_{Q^{-1}} + |y(T)-Cx(T)|^2_{R^{-1}}\Big)$$

Then we optimize over the first state, $x(0)$. This produces the arrival cost for the first stage, $V_1^-(x(1))$, which we will show is also quadratic

$$V_1^-(x(1)) = \frac{1}{2}\,|x(1)-\hat{x}^-(1)|^2_{(P^-(1))^{-1}}$$

Next we combine the arrival cost of the first stage with the next measurement $y(1)$ to obtain $V_1(x(1))$

$$\min_{x(T),\ldots,x(2)} \overbrace{\min_{x(1)} \frac{1}{2}\Big(\underbrace{|x(1)-\hat{x}^-(1)|^2_{(P^-(1))^{-1}} + |y(1)-Cx(1)|^2_{R^{-1}}}_{\text{combine } V_1(x(1))} + |x(2)-Ax(1)|^2_{Q^{-1}}}^{\text{arrival cost } V_2^-(x(2))} + |y(2)-Cx(2)|^2_{R^{-1}} + |x(3)-Ax(2)|^2_{Q^{-1}} + \cdots + |x(T)-Ax(T-1)|^2_{Q^{-1}} + |y(T)-Cx(T)|^2_{R^{-1}}\Big) \tag{1.29}$$

We optimize over the second state, $x(1)$, which defines arrival cost for the first two stages, $V_2^-(x(2))$. We continue in this fashion until we have optimized finally over $x(T)$ and have solved (1.28). Now that we have in mind an overall game plan for solving the problem, we look at each step in detail and develop the recursion formulas of forward DP.

Combine prior and measurement. Combining the prior and measurement defines V_0

$$V_0(x(0)) = \frac{1}{2}\Big(\underbrace{|x(0) - \overline{x}(0)|^2_{(P^-(0))^{-1}}}_{\text{prior}} + \underbrace{|y(0) - Cx(0)|^2_{R^{-1}}}_{\text{measurement}}\Big) \tag{1.30}$$

which can be expressed also as

$$V_0(x(0)) = \frac{1}{2}\Big(|x(0) - \overline{x}(0)|^2_{(P^-(0))^{-1}} + \\ |(y(0) - C\overline{x}(0)) - C(x(0) - \overline{x}(0))|^2_{R^{-1}} \Big)$$

Using the third form in Example 1.1 we can combine these two terms into a single quadratic function

$$V_0(x(0)) = (1/2)\ (x(0) - \overline{x}(0) - v)'\widetilde{H}^{-1}(x(0) - \overline{x}(0) - v) + \text{constant}$$

in which

$$v = P^-(0)C'(CP^-(0)C' + R)^{-1}\,(y(0) - C\overline{x}(0))$$
$$\widetilde{H} = P^-(0) - P^-(0)C'(CP^-(0)C' + R)^{-1}CP^-(0)$$

and we set the constant term to zero because it does not depend on $x(1)$. If we define

$$P(0) = P^-(0) - P^-(0)C'(CP^-(0)C' + R)^{-1}CP^-(0)$$
$$L(0) = P^-(0)C'(CP^-(0)C' + R)^{-1}$$

and define the state estimate $\hat{x}(0)$ as follows

$$\hat{x}(0) = \overline{x}(0) + v$$
$$\hat{x}(0) = \overline{x}(0) + L(0)\,(y(0) - C\overline{x}(0))$$

and we have derived the following compact expression for the function V_0

$$V_0(x(0)) = (1/2)\ |x(0) - \hat{x}(0)|^2_{P(0)^{-1}}$$

State evolution and arrival cost. Now we add the next term in (1.28) to the function $V_0(\cdot)$ and denote the sum as $V(\cdot)$

$$V(x(0), x(1)) = V_0(x(0)) + (1/2)\ |x(1) - Ax(0)|^2_{Q^{-1}}$$
$$V(x(0), x(1)) = \frac{1}{2}\big(|x(0) - \hat{x}(0)|^2_{P(0)^{-1}} + |x(1) - Ax(0)|^2_{Q^{-1}} \big)$$

Again using the third form in Example 1.1, we can add the two quadratics to obtain

$$V(x(0), x(1)) = (1/2)\,|x(0) - v|^2_{\tilde{H}^{-1}} + d$$

in which

$$\begin{aligned} v &= \hat{x}(0) + P(0)A'\left(AP(0)A' + Q\right)^{-1}(x(1) - A\hat{x}(0)) \\ d &= (1/2)\left(|v - \hat{x}(0)|^2_{P(0)^{-1}} + |x(1) - Av|^2_{Q^{-1}}\right) \end{aligned}$$

This form is convenient for optimization over the first decision variable $x(0)$; by inspection the solution is $x(0) = v$ and the cost is d. We define the arrival cost to be the result of this optimization

$$V_1^-(x(1)) = \min_{x(0)} V(x(0), x(1))$$

Substituting v into the expression for d and simplifying gives

$$V_1^-(x(1)) = (1/2)\,|x(1) - A\hat{x}(0)|^2_{(P^-(1))^{-1}}$$

in which

$$P^-(1) = AP(0)A' + Q$$

We define $\hat{x}^-(1) = A\hat{x}(0)$ and express the arrival cost compactly as

$$V_1^-(x(1)) = (1/2)\,\left|x(1) - \hat{x}^-(1)\right|^2_{(P^-(1))^{-1}}$$

Combine arrival cost and measurement. We now combine the arrival cost and measurement for the next stage of the optimization to obtain

$$\begin{aligned} V_1(x(1)) &= \underbrace{V_1^-(x(1))}_{\text{prior}} + \underbrace{(1/2)\,\left|(y(1) - Cx(1))\right|^2_{R^{-1}}}_{\text{measurement}} \\ V_1(x(1)) &= \frac{1}{2}\left(\left|x(1) - \hat{x}^-(1)\right|^2_{(P^-(1))^{-1}} + \left|y(1) - Cx(1)\right|^2_{R^{-1}}\right) \end{aligned}$$

We can see that this equation is exactly the form as (1.30) of the previous step, and, by simply changing the variable names, we have that

$$\begin{aligned} P(1) &= P^-(1) - P^-(1)C'(CP^-(1)C' + R)^{-1}CP^-(1) \\ L(1) &= P^-(1)C'(CP^-(1)C' + R)^{-1} \\ \hat{x}(1) &= \hat{x}^-(1) + L(1)(y(1) - C\hat{x}^-(1)) \end{aligned}$$

and the cost function V_1 is defined as

$$V_1(x(1)) = (1/2)(x(1) - \hat{x}(1))' P(1)^{-1}(x(1) - \hat{x}(1))$$

in which

$$\begin{aligned} \hat{x}^-(1) &= A\hat{x}(0) \\ P^-(1) &= AP(0)A' + Q \end{aligned}$$

Recursion and termination. The recursion can be summarized by two steps. Adding the measurement at time k produces

$$\begin{aligned} P(k) &= P^-(k) - P^-(k)C'(CP^-(k)C' + R)^{-1}CP^-(k) \\ L(k) &= P^-(k)C'(CP^-(k)C' + R)^{-1} \\ \hat{x}(k) &= \hat{x}^-(k) + L(k)(y(k) - C\hat{x}^-(k)) \end{aligned}$$

Propagating the model to time $k + 1$ produces

$$\begin{aligned} \hat{x}^-(k+1) &= A\hat{x}(k) \\ P^-(k+1) &= AP(k)A' + Q \end{aligned}$$

and the recursion starts with the prior information $\hat{x}^-(0) = \overline{x}(0)$ and $P^-(0)$. The arrival cost, V_k^-, and arrival cost plus measurement, V_k, for each stage are given by

$$\begin{aligned} V_k^-(x(k)) &= (1/2)\left|x(k) - \hat{x}^-(k)\right|^2_{(P^-(k))^{-1}} \\ V_k(x(k)) &= (1/2)\left|x(k) - \hat{x}(k)\right|^2_{(P(k))^{-1}} \end{aligned}$$

The process terminates with the final measurement $y(T)$, at which point we have recursively solved the original problem (1.28).

We see by inspection that the recursion formulas given by forward DP of (1.28) are the same as those found by calculating the conditional density function in Section 1.4.2. Moreover, the conditional densities before and after measurement are closely related to the least squares value functions as shown below

$$\begin{aligned} p(x(k)|\mathbf{y}(k-1)) &= \frac{1}{(2\pi)^{n/2}(\det P^-(k))^{1/2}} \exp(-V_k^-(x(k))) \qquad (1.31) \\ p(x(k)|\mathbf{y}(k)) &= \frac{1}{(2\pi)^{n/2}(\det P(k))^{1/2}} \exp(-V_k(x(k))) \end{aligned}$$

The discovery (and rediscovery) of the close connection between recursive least squares and optimal statistical estimation has not always been greeted happily by researchers:

> The recursive least squares approach was actually inspired by probabilistic results that automatically produce an equation of evolution for the estimate (the conditional mean). In fact, much of the recent least squares work did nothing more than rederive the probabilistic results (perhaps in an attempt to understand them). As a result, much of the least squares work contributes very little to estimation theory.
> —Jazwinski (1970, pp.152–153)

In contrast with this view, we find both approaches valuable in the subsequent development. The probabilistic approach, which views the state estimator as maximizing conditional density of the state given measurement, offers the most insight. It provides a rigorous basis for comparing different estimators based on the variance of their estimate error. It also specifies what information is required to define an optimal estimator, with variances Q and R of primary importance. In the probabilistic framework, these parameters should be found from modeling and data. The main deficiency in the least squares viewpoint is that the objective function, although reasonable, is ad hoc and not justified. The choice of weighting matrices Q and R is arbitrary. Practitioners generally choose these parameters based on a tradeoff between the competing goals of speed of estimator response and insensitivity to measurement noise. But a careful statement of this tradeoff often just leads back to the probabilistic viewpoint in which the process disturbance and measurement disturbance are modeled as normal distributions. If we restrict attention to unconstrained linear systems, the probabilistic viewpoint is clearly superior.

Approaching state estimation with the perspective of least squares pays off, however, when the models are significantly more complex. It is generally intractable to find and maximize the conditional density of the state given measurements for complex, nonlinear and constrained models. Although the state estimation problem can be stated in the language of probability, it cannot be solved with current methods. But reasonable objective functions can be chosen for even complex, nonlinear and constrained models. Moreover, knowing which least squares problems correspond to which statistically optimal estimation problems for the simple linear case, provides the engineer with valuable insight in choosing useful objective functions for nonlinear estimation. We explore these more complex and realistic estimation problems in Chapter 4. The perspective of least squares also leads to succinct arguments for establishing estimator stability, which we take up shortly.

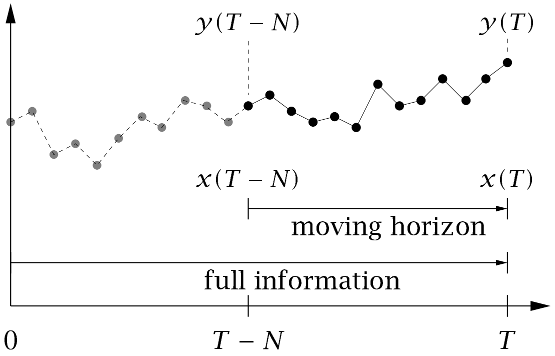

Figure 1.5: Schematic of the moving horizon estimation problem.

First we consider situations in which it is advantageous to use moving horizon estimation.

1.4.4 Moving Horizon Estimation

When using nonlinear models or considering constraints on the estimates, we cannot calculate the conditional density recursively in closed form as we did in Kalman filtering. Similarly, we cannot solve recursively the least squares problem. If we use least squares we must optimize all the states in the trajectory $\mathbf{x}(T)$ simultaneously to obtain the state estimates. This optimization problem becomes computationally intractable as T increases. Moving horizon estimation (MHE) removes this difficulty by considering only the most recent N measurements and finds only the most recent N values of the state trajectory as sketched in Figure 1.5. The states to be estimated are $\mathbf{x}_N(T) = \big(x(T-N), \ldots, x(T)\big)$ given measurements $\mathbf{y}_N(T) = \big(y(T-N), \ldots, y(T)\big)$. The data have been broken into two sections with $(\mathbf{y}(T-N-1), \mathbf{y}_N(T)) = \mathbf{y}(T)$. We assume here that $T \geq N-1$ to ignore the initial period in which the estimation window fills with measurements and assume that the window is always full.

The simplest form of MHE is the following least squares problem

$$\min_{\mathbf{x}_N(T)} \hat{V}_T(\mathbf{x}_N(T)) \tag{1.32}$$

in which the objective function is

$$\hat{V}_T(\mathbf{x}_N(T)) = \frac{1}{2}\Bigg(\sum_{k=T-N}^{T-1} |x(k+1) - Ax(k)|^2_{Q^{-1}} + \sum_{k=T-N}^{T} |y(k) - Cx(k)|^2_{R^{-1}} \Bigg) \quad (1.33)$$

We use the circumflex (hat) to indicate this is the MHE cost function considering data sequence from $T-N$ to T rather than the full information or least squares cost considering the data from 0 to T.

MHE in terms of least squares. Notice that from our previous DP recursion in (1.29), we can write the full least squares problem as

$$V_T(\mathbf{x}_N(T)) = V^-_{T-N}(x(T-N)) + \frac{1}{2}\Bigg(\sum_{k=T-N}^{T-1} |x(k+1) - Ax(k)|^2_{Q^{-1}} + \sum_{k=T-N}^{T} |y(k) - Cx(k)|^2_{R^{-1}} \Bigg)$$

in which $V^-_{T-N}(\cdot)$ is the arrival cost at time $T-N$. Comparing these two objective functions, it is clear that the simplest form of MHE is equivalent to setting up a full least squares problem, but then setting the arrival cost function $V^-_{T-N}(\cdot)$ to zero.

MHE in terms of conditional density. Because we have established the close connection between least squares and conditional density in (1.31), we can write the full least squares problem also as an equivalent conditional density maximization

$$\max_{x(T)} p_{x(T)|\mathbf{y}_N(T)}(x(T)|\mathbf{y}_N(T))$$

with prior density

$$p_{x(T-N)|\mathbf{y}(T-N-1)}(x|\mathbf{y}(T-N-1)) = c\exp(-V^-_{T-N}(x)) \quad (1.34)$$

in which the constant c can be found from (1.19) if desired, but its value does not change the solution to the optimization. We can see from (1.34) that setting $V^-_{T-N}(\cdot)$ to zero in the simplest form of MHE is equivalent to giving infinite variance to the conditional density of $x(T-N)|\mathbf{y}(T-N-1)$. This means we are using no information about the state $x(T-N)$ and completely discounting the previous measurements $\mathbf{y}(T-N-1)$.

To provide a more flexible MHE problem, we therefore introduce a penalty on the first state to account for the neglected data $\mathbf{y}(T-N-1)$

$$\hat{V}_T(\mathbf{x}_N(T)) = \Gamma_{T-N}(x(T-N)) + \\ \frac{1}{2}\left(\sum_{k=T-N}^{T-1} |x(k+1) - Ax(k)|^2_{Q^{-1}} + \sum_{k=T-N}^{T} |y(k) - Cx(k)|^2_{R^{-1}}\right)$$

For the linear Gaussian case, we can account for the neglected data exactly with no approximation by setting Γ equal to the arrival cost, or, equivalently, the negative logarithm of the conditional density of the state given the prior measurements. Indeed, there is no need to use MHE for the linear Gaussian problem at all because we can solve the full problem recursively. When addressing nonlinear and constrained problems in Chapter 4, however, we must approximate the conditional density of the state given the prior measurements in MHE to obtain a computationally tractable and high-quality estimator.

1.4.5 Observability

We next explore the convergence properties of the state estimators. For this we require the concept of system observability. The basic idea of observability is that any two distinct states can be *distinguished* by applying some input and observing the two system outputs over some finite time interval (Sontag, 1998, p.262–263). We discuss this general definition in more detail when treating nonlinear systems in Chapter 4, but observability for linear systems is much simpler. First of all, the applied input is irrelevant and we can set it to zero. Therefore consider the linear time-invariant system (A, C) with zero input

$$\begin{aligned} x(k+1) &= Ax(k) \\ y(k) &= Cx(k) \end{aligned}$$

The system is observable if there exists a finite N, such that for every $x(0)$, N measurements $(y(0), y(1), \ldots, y(N-1))$ distinguish uniquely the initial state $x(0)$. Similarly to the case of controllability, if we cannot determine the initial state using n measurements, we cannot determine it using $N > n$ measurements. Therefore we can develop a convenient test for observability as follows. For n measurements, the system model gives

$$\begin{bmatrix} y(0) \\ y(1) \\ \vdots \\ y(n-1) \end{bmatrix} = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix} x(0) \tag{1.35}$$

The question of *observability* is therefore a question of *uniqueness* of solutions to these linear equations. The matrix appearing in this equation is known as the *observability matrix* $\mathcal{O}$

$$\mathcal{O} = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix} \tag{1.36}$$

From the fundamental theorem of linear algebra, we know the solution to (1.35) is unique if and only if the *columns* of the $np \times n$ observability matrix are linearly independent.[6] Therefore, we have that the system (A, C) is observable if and only if

$$\text{rank}(\mathcal{O}) = n$$

The following result for checking observability also proves useful (Hautus, 1972).

Lemma 1.4 (Hautus lemma for observability). *A system is observable if and only if*

$$\text{rank} \begin{bmatrix} \lambda I - A \\ C \end{bmatrix} = n \qquad \textit{for all } \lambda \in \mathbb{C} \tag{1.37}$$

in which $\mathbb{C}$ is the set of complex numbers.

Notice that the first n rows of the matrix in (1.37) are linearly independent if $\lambda \notin \text{eig}(A)$, so (1.37) is equivalent to checking the rank at just the eigenvalues of A

$$\text{rank} \begin{bmatrix} \lambda I - A \\ C \end{bmatrix} = n \qquad \text{for all } \lambda \in \text{eig}(A)$$

[6] See Section A.4 of Appendix A or (Strang, 1980, pp.87–88) for a review of this result.

1.4.6 Convergence of the State Estimator

Next we consider the question of convergence of the estimates of several of the estimators we have considered. The simplest convergence question to ask is the following. Given an initial estimate error, and zero state and measurement noises, does the state estimate converge to the state as time increases and more measurements become available? If the answer to this question is yes, we say the estimates converge; sometimes we say the estimator converges. As with the regulator, optimality of an estimator does not ensure its stability. Consider the case $A = I, C = 0$. The optimal estimate is $\hat{x}(k) = \overline{x}(0)$, which does not converge to the true state unless we have luckily chosen $\overline{x}(0) = x(0)$.[7] Obviously the lack of stability is caused by our choosing an unobservable (undetectable) system.

We treat first the Kalman filtering or full least squares problem. Recall that this estimator optimizes over the entire state trajectory $\mathbf{x}(T) := (x(0), \ldots, x(T))$ based on all measurements $\mathbf{y}(T) := (y(0), \ldots, y(T))$. In order to establish convergence, the following result on the optimal estimator cost function proves useful.

Lemma 1.5 (Convergence of estimator cost). *Given noise-free measurements $\mathbf{y}(T) = (Cx(0), CAx(0), \ldots, CA^T x(0))$, the optimal estimator cost $V_T^0(\mathbf{y}(T))$ converges as $T \to \infty$.*

Proof. Denote the optimal state sequence at time T given measurement $\mathbf{y}(T)$ by

$$(\hat{x}(0|T),\ \hat{x}(1|T),\ \ldots,\ \hat{x}(T|T))$$

We wish to compare the optimal costs at time T and $T-1$. Therefore, consider using the first $T-1$ elements of the solution at time T as decision variables in the state estimation problem at time $T-1$. The cost for those decision variables at time $T-1$ is given by

$$V_T^0 - \frac{1}{2}\left(|\hat{x}(T|T) - A\hat{x}(T-1|T)|^2_{Q^{-1}} + |y(T) - C\hat{x}(T|T)|^2_{R^{-1}} \right)$$

In other words, we have the full cost at time T and we deduct the cost of the last stage, which is not present at $T-1$. Now this choice of decision variables is not necessarily optimal at time $T-1$, so we have the inequality

$$V_{T-1}^0 \leq V_T^0 - \frac{1}{2}\left(|\hat{x}(T|T) - A\hat{x}(T-1|T)|^2_{Q^{-1}} + |y(T) - C\hat{x}(T|T)|^2_{R^{-1}} \right)$$

[7] If we could count on that kind of luck, we would have no need for state estimation.

Because the quadratic terms are nonnegative, the sequence of optimal estimator costs is nondecreasing with increasing T. We can establish that the optimal cost is bounded above as follows: at any time T we can choose the decision variables to be $(x(0), Ax(0), \dots, A^T x(0))$, which achieves cost $|x(0) - \overline{x}(0)|^2_{(P^-(0))^{-1}}$ independent of T. The optimal cost sequence is nondecreasing and bounded above and, therefore, converges. ■

The optimal estimator cost converges regardless of system observability. But if we want the optimal estimate to converge to the state, we have to restrict the system further. The following lemma provides an example of what is required.

Lemma 1.6 (Estimator convergence). *For (A, C) observable, $Q, R > 0$, and noise-free measurements $\mathbf{y}(T) = (Cx(0), CAx(0), \dots, CA^T x(0))$, the optimal linear state estimate converges to the state*

$$\hat{x}(T) \to x(T) \quad \textit{as } T \to \infty$$

Proof. To compress the notation somewhat, let $\hat{w}_T(j) = \hat{x}(T + j + 1|T + n - 1) - A\hat{x}(T + j|T + n - 1)$. Using the optimal solution at time $T + n - 1$ as decision variables at time $T - 1$ allows us to write the following inequality

$$V^0_{T-1} \leq V^0_{T+n-1} - \frac{1}{2}\left(\sum_{j=-1}^{n-2} |\hat{w}_T(j)|^2_{Q^{-1}} + \sum_{j=0}^{n-1} |y(T + j) - C\hat{x}(T + j|T + n - 1)|^2_{R^{-1}} \right)$$

Because the sequence of optimal costs converges with increasing T, and $Q^{-1}, R^{-1} > 0$, we have established that for increasing T

$$\begin{aligned} \hat{w}_T(j) &\to 0 \quad j = -1, \dots, n - 2 \\ y(T + j) - C\hat{x}(T + j|T + n - 1) &\to 0 \quad j = 0, \dots, n - 1 \end{aligned} \tag{1.38}$$

From the system model we have the following relationship between the last n stages in the optimization problem at time $T + n - 1$ with data

$\mathbf{y}(T+n-1)$

$$\begin{bmatrix} \hat{x}(T|T+n-1) \\ \hat{x}(T+1|T+n-1) \\ \vdots \\ \hat{x}(T+n-1|T+n-1) \end{bmatrix} = \begin{bmatrix} I \\ A \\ \vdots \\ A^{n-1} \end{bmatrix} \hat{x}(T|T+n-1) + \begin{bmatrix} 0 & & & \\ I & 0 & & \\ \vdots & \vdots & \ddots & \\ A^{n-2} & A^{n-3} & \cdots & I \end{bmatrix} \begin{bmatrix} \hat{w}_T(0) \\ \hat{w}_T(1) \\ \vdots \\ \hat{w}_T(n-2) \end{bmatrix} \tag{1.39}$$

We note the measurements satisfy

$$\begin{bmatrix} y(T) \\ y(T+1) \\ \vdots \\ y(T+n-1) \end{bmatrix} = \mathcal{O}x(T)$$

Multiplying (1.39) by C and subtracting gives

$$\begin{bmatrix} y(T) - C\hat{x}(T|T+n-1) \\ y(T+1) - C\hat{x}(T+1|T+n-1) \\ \vdots \\ y(T+n-1) - C\hat{x}(T+n-1|T+n-1) \end{bmatrix} = \mathcal{O}(x(T) - \hat{x}(T|T+n-1)) - \begin{bmatrix} 0 & & & \\ C & 0 & & \\ \vdots & \vdots & \ddots & \\ CA^{n-2} & CA^{n-3} & \cdots & C \end{bmatrix} \begin{bmatrix} \hat{w}_T(0) \\ \hat{w}_T(1) \\ \vdots \\ \hat{w}_T(n-2) \end{bmatrix}$$

Applying (1.38) to this equation, we conclude $\mathcal{O}(x(T) - \hat{x}(T|T+n-1)) \to 0$ with increasing T. Because the observability matrix has independent columns, we conclude $x(T) - \hat{x}(T|T+n-1) \to 0$ as $T \to \infty$. Thus we conclude that the *smoothed* estimate $\hat{x}(T|T+n-1)$ converges to the state $x(T)$. Because the $\hat{w}_T(j)$ terms go to zero with increasing T, the last line of (1.39) gives $\hat{x}(T+n-1|T+n-1) \to A^{n-1}\hat{x}(T|T+n-1)$ as $T \to \infty$. From the system model $A^{n-1}x(T) = x(T+n-1)$ and, therefore, after replacing $T+n-1$ by T, we have

$$\hat{x}(T|T) \to x(T) \quad \text{as } T \to \infty$$

and asymptotic convergence of the estimator is established. ■

This convergence result also covers MHE with prior weighting set to the exact arrival cost because that is equivalent to Kalman filtering and full least squares. The simplest form of MHE, which discounts prior data completely, is also a convergent estimator, however, as discussed in Exercise 1.28.

The estimator convergence result in Lemma 1.6 is the simplest to establish, but, as in the case of the LQ regulator, we can enlarge the class of systems and weighting matrices (variances) for which estimator convergence is guaranteed. The system restriction can be weakened from observability to *detectability*, which is discussed in Exercises 1.31 and 1.32. The restriction on the process disturbance weight (variance) Q can be weakened from $Q > 0$ to $Q \geq 0$ and (A, Q) *stabilizable*, which is discussed in Exercise 1.33. The restriction $R > 0$ remains to ensure uniqueness of the estimator.

1.5 Tracking, Disturbances, and Zero Offset

In the last section of this chapter we show briefly how to use the MPC regulator and MHE estimator to handle different kinds of control problems, including setpoint tracking and rejecting nonzero disturbances.

1.5.1 Tracking

It is a standard objective in applications to use a feedback controller to move the measured outputs of a system to a specified and constant setpoint. This problem is known as setpoint tracking. In Chapter 5 we consider the case in which the system is nonlinear and constrained, but for simplicity here we consider the linear unconstrained system in which y_{sp} is an arbitrary constant. In the regulation problem of Section 1.3 we assumed that the goal was to take the state of the system to the origin. Such a regulator can be used to treat the setpoint tracking problem with a coordinate transformation. Denote the desired output setpoint as y_{sp}. Denote a steady state of the system model as (x_s, u_s). From (1.5), the steady state satisfies

$$\begin{bmatrix} I - A & -B \end{bmatrix} \begin{bmatrix} x_s \\ u_s \end{bmatrix} = 0$$

For *unconstrained* systems, we also impose the requirement that the steady state satisfies $Cx_s = y_{\mathrm{sp}}$ for the tracking problem, giving the set of equations

$$\begin{bmatrix} I-A & -B \\ C & 0 \end{bmatrix} \begin{bmatrix} x_s \\ u_s \end{bmatrix} = \begin{bmatrix} 0 \\ y_{\text{sp}} \end{bmatrix} \tag{1.40}$$

If this set of equations has a solution, we can then define deviation variables

$$\begin{aligned} \widetilde{x}(k) &= x(k) - x_s \\ \widetilde{u}(k) &= u(k) - u_s \end{aligned}$$

that satisfy the dynamic model

$$\begin{aligned} \widetilde{x}(k+1) &= x(k+1) - x_s \\ &= Ax(k) + Bu(k) - (Ax_s + Bu_s) \\ \widetilde{x}(k+1) &= A\widetilde{x}(k) + B\widetilde{u}(k) \end{aligned}$$

so that the deviation variables satisfy the same model equation as the original variables. The zero regulation problem applied to the system in deviation variables finds $\widetilde{u}(k)$ that takes $\widetilde{x}(k)$ to zero, or, equivalently, which takes $x(k)$ to x_s, so that at steady state, $Cx(k) = Cx_s = y_{\text{sp}}$, which is the goal of the setpoint tracking problem. After solving the regulation problem in deviation variables, the input applied to the system is $u(k) = \widetilde{u}(k) + u_s$.

We next discuss when we can solve (1.40). We also note that for *constrained* systems, we must impose the constraints on the steady state (x_s, u_s). The matrix in (1.40) is a $(n+p) \times (n+m)$ matrix. For (1.40) to have a solution for all y_{sp}, it is sufficient that the rows of the matrix are linearly independent. That requires $p \leq m$: we require at least as many inputs as outputs with setpoints. But it is not uncommon in applications to have many more measured outputs than manipulated inputs. To handle these more general situations, we choose a matrix H and denote a new variable $r = Hy$ as a selection of linear combinations of the measured outputs. The variable $r \in \mathbb{R}^{n_c}$ is known as the *controlled variable*. For cases in which $p > m$, we choose some set of outputs $n_c \leq m$, as controlled variables, and assign setpoints to r, denoted r_{sp}.

We also wish to treat systems with more inputs than outputs, $m > p$. For these cases, the solution to (1.40) may exist for some choice of H and r_{sp}, but cannot be unique. If we wish to obtain a unique steady state, then we also must provide desired values for the steady inputs, u_{sp}. To handle constrained systems, we simply impose the constraints on (x_s, u_s).

Steady-state target problem. Our candidate optimization problem is therefore

$$\min_{x_s,u_s} \frac{1}{2}\left(\left|u_s - u_{\text{sp}}\right|^2_{R_s} + \left|Cx_s - y_{\text{sp}}\right|^2_{Q_s}\right) \tag{1.41a}$$

subject to

$$\begin{bmatrix} I-A & -B \\ HC & 0 \end{bmatrix}\begin{bmatrix} x_s \\ u_s \end{bmatrix} = \begin{bmatrix} 0 \\ r_{\text{sp}} \end{bmatrix} \tag{1.41b}$$

$$Eu_s \leq e \tag{1.41c}$$

$$FCx_s \leq f \tag{1.41d}$$

We make the following assumptions.

Assumption 1.7 (Target feasibility and uniqueness)**.**

(a) The target problem is feasible for the controlled variable setpoints of interest r_{sp}.

(b) The steady-state input penalty R_s is positive definite.

Assumption 1.7 (a) ensures that the solution (x_s, u_s) exists, and Assumption 1.7 (b) ensures that the solution is unique. If one chooses $n_c = 0$, then no controlled variables are required to be at setpoint, and the problem is feasible for any $(u_{\text{sp}}, y_{\text{sp}})$ because $(x_s, u_s) = (0, 0)$ is a feasible point. Exercises 1.56 and 1.57 explore the connection between feasibility of the equality constraints and the number of controlled variables relative to the number of inputs and outputs. One restriction is that the number of controlled variables chosen to be offset free must be less than or equal to the number of manipulated variables and the number of measurements, $n_c \leq m$ and $n_c \leq p$.

Dynamic regulation problem. Given the steady-state solution, we define the following multistage objective function

$$V(\tilde{x}(0), \tilde{\mathbf{u}}) = \frac{1}{2}\sum_{k=0}^{N-1} \left|\tilde{x}(k)\right|^2_Q + \left|\tilde{u}(k)\right|^2_R \qquad \text{s.t. } \tilde{x}^+ = A\tilde{x} + B\tilde{u}$$

in which $\tilde{x}(0) = \hat{x}(k) - x_s$, i.e., the initial condition for the regulation problem comes from the state estimate shifted by the steady-state x_s. The regulator solves the following dynamic, zero-state regulation problem

$$\min_{\tilde{\mathbf{u}}} V(\tilde{x}(0), \tilde{\mathbf{u}})$$

subject to

$$E\tilde{u} \leq e - Eu_s$$
$$FC\tilde{x} \leq f - FCx_s$$

in which the constraints also are shifted by the steady state (x_s, u_s). The optimal cost and solution are $V^0(\tilde{x}(0))$ and $\tilde{\mathbf{u}}^0(\tilde{x}(0))$. The moving horizon control law uses the first move of this optimal sequence, $\tilde{u}^0(\tilde{x}(0)) = \tilde{\mathbf{u}}^0(0; \tilde{x}(0))$, so the controller output is $u(k) = \tilde{u}^0(\tilde{x}(0)) + u_s$.

1.5.2 Disturbances and Zero Offset

Another common objective in applications is to use a feedback controller to compensate for an unmeasured disturbance to the system with the input so the disturbance's effect on the controlled variable is mitigated. This problem is known as disturbance rejection. We may wish to design a feedback controller that compensates for nonzero disturbances such that the selected controlled variables asymptotically approach their setpoints without offset. This property is known as zero offset. In this section we show a simple method for constructing an MPC controller to achieve zero offset.

In Chapter 5, we address the full problem. Here we must be content to limit our objective. We will ensure that *if the system is stabilized in the presence of the disturbance*, then there is zero offset. But we will not attempt to construct the controller that ensures stabilization over an interesting class of disturbances. That topic is treated in Chapter 5.

This more limited objective is similar to what one achieves when using the integral mode in proportional-integral-derivative (PID) control of an unconstrained system: either there is zero steady offset, or the system trajectory is unbounded. In a constrained system, the statement is amended to: either there is zero steady offset, or the system trajectory is unbounded, or the system constraints are active at steady state. In both constrained and unconstrained systems, the zero-offset property *precludes* one undesirable possibility: the system settles at an unconstrained steady state, and the steady state displays offset in the controlled variables.

A simple method to compensate for an unmeasured disturbance is to (i) model the disturbance, (ii) use the measurements and model to estimate the disturbance, and (iii) find the inputs that minimize the effect of the disturbance on the controlled variables. The choice of disturbance model is motivated by the zero-offset goal. To achieve offset-free performance we augment the system state with an *integrating* disturbance d driven by a white noise w_d

$$d^+ = d + w_d \tag{1.42}$$

This choice is motivated by the works of Davison and Smith (1971, 1974); Qiu and Davison (1993) and the Internal Model Principle of Francis and Wonham (1976). To remove offset, one designs a control system that can remove asymptotically constant, nonzero disturbances (Davison and Smith, 1971), (Kwakernaak and Sivan, 1972, p.278). To accomplish this end, the original system is augmented with a replicate of the constant, nonzero disturbance model, (1.42). Thus the states of the original system are moved onto the manifold that cancels the effect of the disturbance on the controlled variables. The augmented system model used for the state estimator is given by

$$\begin{bmatrix} x \\ d \end{bmatrix}^+ = \begin{bmatrix} A & B_d \\ 0 & I \end{bmatrix} \begin{bmatrix} x \\ d \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u + w \tag{1.43a}$$

$$y = \begin{bmatrix} C & C_d \end{bmatrix} \begin{bmatrix} x \\ d \end{bmatrix} + v \tag{1.43b}$$

and we are free to choose how the integrating disturbance affects the states and measured outputs through the choice of B_d and C_d. The only restriction is that the augmented system is detectable. That restriction can be easily checked using the following result.

Lemma 1.8 (Detectability of the augmented system)**.** *The augmented system* (1.43) *is detectable if and only if the unaugmented system* (A, C) *is detectable, and the following condition holds*

$$\operatorname{rank} \begin{bmatrix} I - A & -B_d \\ C & C_d \end{bmatrix} = n + n_d \tag{1.44}$$

Corollary 1.9 (Dimension of the disturbance)**.** *The maximal dimension of the disturbance d in* (1.43) *such that the augmented system is detectable is equal to the number of measurements, that is*

$$n_d \leq p$$

A pair of matrices (B_d, C_d) such that (1.44) is satisfied always exists. In fact, since (A, C) is detectable, the submatrix $\begin{bmatrix} I-A \\ C \end{bmatrix} \in \mathbb{R}^{(p+n)\times n}$ has rank n. Thus, we can choose any $n_d \leq p$ columns in $\mathbb{R}^{p+n}$ independent of $\left[\begin{smallmatrix} I-A \\ C \end{smallmatrix}\right]$ for $\left[\begin{smallmatrix} -B_d \\ C_d \end{smallmatrix}\right]$.

The state and the additional integrating disturbance are estimated from the plant measurement using a Kalman filter designed for the augmented system. The variances of the stochastic disturbances w and v may be treated as adjustable parameters or found from input-output measurements (Odelson, Rajamani, and Rawlings, 2006). The estimator provides $\hat{x}(k)$ and $\hat{d}(k)$ at each time k. The best forecast of the steady-state disturbance using (1.42) is simply

$$\hat{d}_s = \hat{d}(k)$$

The steady-state target problem is therefore modified to account for the nonzero disturbance $\hat{d}_s$

$$\min_{x_s, u_s} \frac{1}{2}\left(\left| u_s - u_{\text{sp}} \right|^2_{R_s} + \left| Cx_s + C_d\hat{d}_s - y_{\text{sp}} \right|^2_{Q_s} \right) \tag{1.45a}$$

subject to

$$\begin{bmatrix} I-A & -B \\ HC & 0 \end{bmatrix} \begin{bmatrix} x_s \\ u_s \end{bmatrix} = \begin{bmatrix} B_d\hat{d}_s \\ r_{\text{sp}} - HC_d\hat{d}_s \end{bmatrix} \tag{1.45b}$$

$$Eu_s \leq e \tag{1.45c}$$

$$FCx_s \leq f - FC_d\hat{d}_s \tag{1.45d}$$

Comparing (1.41) to (1.45), we see the disturbance model affects the steady-state target determination in four ways.

1. The output target is modified in (1.45a) to account for the effect of the disturbance on the measured output ($y_{\text{sp}} \to y_{\text{sp}} - C_d\hat{d}_s$).

2. The output constraint in (1.45d) is similarly modified ($f \to f - FC_d\hat{d}_s$).

3. The system steady-state relation in (1.45b) is modified to account for the effect of the disturbance on the state evolution ($0 \to B_d\hat{d}_s$).

4. The controlled variable target in (1.45b) is modified to account for the effect of the disturbance on the controlled variable ($r_{\text{sp}} \to r_{\text{sp}} - HC_d\hat{d}_s$).

Given the steady-state target, the same dynamic regulation problem as presented in the tracking section, Section 1.5, is used for the regulator.

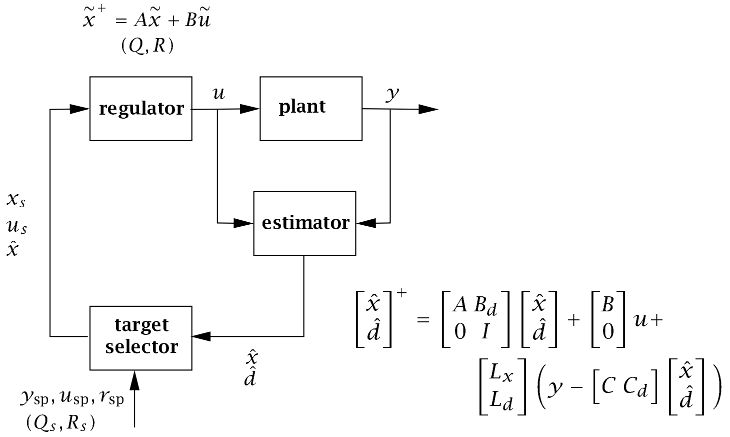
$\tilde{x}^+ = A\tilde{x} + B\tilde{u}$
(Q, R)
regulator
u
plant
y
estimator
x_s
u_s
$\hat{x}$
target selector
$\hat{x}$
$\hat{d}$
$y_{\text{sp}}, u_{\text{sp}}, r_{\text{sp}}$
(Q_s, R_s)
$\begin{bmatrix} \hat{x} \\ \hat{d} \end{bmatrix}^+ = \begin{bmatrix} A & B_d \\ 0 & I \end{bmatrix} \begin{bmatrix} \hat{x} \\ \hat{d} \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u +$
$\begin{bmatrix} L_x \\ L_d \end{bmatrix} \left(y - \begin{bmatrix} C & C_d \end{bmatrix} \begin{bmatrix} \hat{x} \\ \hat{d} \end{bmatrix} \right)$

Figure 1.6: MPC controller consisting of: receding horizon regulator, state estimator, and target selector.

In other words, the regulator is based on the deterministic system (A, B) in which the current state is $\hat{x}(k) - x_s$ and the goal is to take the system to the origin.

The following lemma summarizes the offset-free control property of the combined control system.

Lemma 1.10 (Offset-free control). *Consider a system controlled by the MPC algorithm as shown in Figure 1.6. The target problem* (1.45) *is assumed feasible. Augment the system model with a number of integrating disturbances equal to the number of measurements* $(n_d = p)$*; choose any* $B_d \in \mathbb{R}^{n \times p}$, $C_d \in \mathbb{R}^{p \times p}$ *such that*

$$\operatorname{rank} \begin{bmatrix} I - A & -B_d \\ C & C_d \end{bmatrix} = n + p$$

If the plant output $y(k)$ *goes to steady state* y_s*, the closed-loop system is stable, and constraints are not active at steady state, then there is zero offset in the controlled variables, that is*

$$Hy_s = r_{\text{sp}}$$

The proof of this lemma is given in Pannocchia and Rawlings (2003). It may seem surprising that the number of integrating disturbances must be equal to the number of *measurements* used for feedback rather than the number of *controlled variables* to guarantee offset-free control. To gain insight into the reason, consider the disturbance part (bottom half) of the Kalman filter equations shown in Figure 1.6

$$\hat{d}^+ = \hat{d} + L_d\left(y - \begin{bmatrix} C & C_d \end{bmatrix} \begin{bmatrix} \hat{x} \\ \hat{d} \end{bmatrix}\right)$$

Because of the integrator, the disturbance estimate cannot converge until

$$L_d\left(y - \begin{bmatrix} C & C_d \end{bmatrix} \begin{bmatrix} \hat{x} \\ \hat{d} \end{bmatrix}\right) = 0$$

But notice this condition merely restricts the output prediction error to lie in the nullspace of the matrix L_d, which is an $n_d \times p$ matrix. If we choose $n_d = n_c < p$, then the number of columns of L_d is greater than the number of rows and L_d has a nonzero nullspace.[8] In general, we require the output prediction error to be *zero* to achieve zero offset independently of the regulator tuning. For L_d to have only the zero vector in its nullspace, we require $n_d \geq p$. Since we also know $n_d \leq p$ from Corollary 1.9, we conclude $n_d = p$.

Notice also that Lemma 1.10 does not require that the plant output be generated by the model. The theorem applies regardless of what generates the plant output. *If the plant is identical to the system plus disturbance model assumed in the estimator*, then the conclusion can be strengthened. In the nominal case without measurement or process noise ($w = 0$, $v = 0$), *for a set of plant initial states*, the closed-loop system *converges to a steady state* and the feasible steady-state target is achieved leading to zero offset in the controlled variables. Characterizing the set of initial states in the region of convergence, and stabilizing the system when the plant and the model differ, are treated in Chapters 3 and 5. We conclude the chapter with a nonlinear example that demonstrates the use of Lemma 1.10.

Example 1.11: More measured outputs than inputs and zero offset

We consider a well-stirred chemical reactor depicted in Figure 1.7, as in Pannocchia and Rawlings (2003). An irreversible, first-order reaction A$\longrightarrow$ B occurs in the liquid phase and the reactor temperature is regulated with external cooling. Mass and energy balances lead to the

[8] This is another consequence of the fundamental theorem of linear algebra. The result is depicted in Figure A.1.

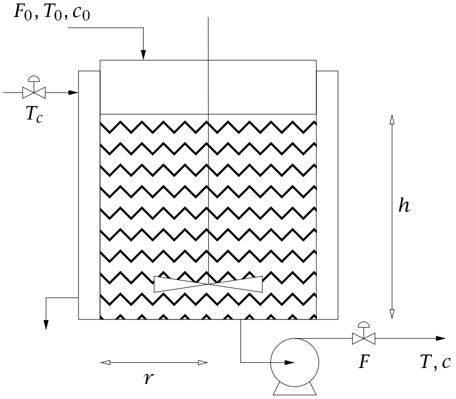

Figure 1.7: Schematic of the well-stirred reactor.

following nonlinear state space model

$$\begin{aligned}\frac{dc}{dt} &= \frac{F_0(c_0 - c)}{\pi r^2 h} - k_0 \exp\left(-\frac{E}{RT}\right) c\\ \frac{dT}{dt} &= \frac{F_0(T_0 - T)}{\pi r^2 h} + \frac{-\Delta H}{\rho C_p} k_0 \exp\left(-\frac{E}{RT}\right) c + \frac{2U}{r\rho C_p}(T_c - T)\\ \frac{dh}{dt} &= \frac{F_0 - F}{\pi r^2}\end{aligned}$$

The controlled variables are h, the level of the tank, and c, the molar concentration of species A. The additional state variable is T, the reactor temperature; while the manipulated variables are T_c, the coolant liquid temperature, and F, the outlet flowrate. Moreover, it is assumed that the inlet flowrate acts as an unmeasured disturbance. The model parameters in nominal conditions are reported in Table 1.1. The open-loop stable steady-state operating conditions are the following

$$c^s = 0.878\,\mathrm{kmol/m^3} \qquad T^s = 324.5\,\mathrm{K} \qquad h^s = 0.659\,\mathrm{m}$$
$$T_c^s = 300\,\mathrm{K} \qquad F^s = 0.1\,\mathrm{m^3/min}$$

Using a sampling time of 1 min, a linearized discrete state space model is obtained and, assuming that all the states are measured, the state space variables are

$$x = \begin{bmatrix} c - c^s \\ T - T^s \\ h - h^s \end{bmatrix} \qquad u = \begin{bmatrix} T_c - T_c^s \\ F - F^s \end{bmatrix} \qquad y = \begin{bmatrix} c - c^s \\ T - T^s \\ h - h^s \end{bmatrix} \qquad p = F_0 - F_0^s$$

Parameter	Nominal value	Units
F_0	0.1	$\mathrm{m^3/min}$
T_0	350	K
c_0	1	$\mathrm{kmol/m^3}$
r	0.219	m
k_0	7.2×10^{10}	$\mathrm{min^{-1}}$
E/R	8750	K
U	54.94	$\mathrm{kJ/min{\cdot}m^2{\cdot}K}$
ρ	1000	$\mathrm{kg/m^3}$
C_p	0.239	$\mathrm{kJ/kg{\cdot}K}$
ΔH	-5×10^4	kJ/kmol

Table 1.1: Parameters of the well-stirred reactor.

The corresponding linear model is

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) + B_p p \\ y(k) &= Cx(k) \end{aligned}$$

in which

$$A = \begin{bmatrix} 0.2681 & -0.00338 & -0.00728 \\ 9.703 & 0.3279 & -25.44 \\ 0 & 0 & 1 \end{bmatrix} \qquad C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$B = \begin{bmatrix} -0.00537 & 0.1655 \\ 1.297 & 97.91 \\ 0 & -6.637 \end{bmatrix} \qquad B_p = \begin{bmatrix} -0.1175 \\ 69.74 \\ 6.637 \end{bmatrix}$$

(a) Since we have two inputs, T_c and F, we try to remove offset in two controlled variables, c and h. Model the disturbance with *two* integrating output disturbances on the two controlled variables. Assume that the covariances of the state noises are zero except for the two integrating states. Assume that the covariances of the three measurements' noises are also zero.

Notice that although there are only two controlled variables, this choice of *two* integrating disturbances does not follow the prescription of Lemma 1.10 for zero offset.

Simulate the response of the controlled system after a 10% increase in the inlet flowrate F_0 at time $t = 10\,\mathrm{min}$. Use the nonlinear differential equations for the plant model. Do you have steady offset in any of the outputs? Which ones?

(b) Follow the prescription of Lemma 1.10 and choose a disturbance model with *three* integrating modes. Can you choose three integrating output disturbances for this plant? If so, prove it. If not, state why not.

(c) Again choose a disturbance model with three integrating modes; choose two integrating output disturbances on the two controlled variables. Choose one integrating input disturbance on the outlet flowrate F. Is the augmented system detectable?

Simulate again the response of the controlled system after a 10% increase in the inlet flowrate F_0 at time $t = 10\,\text{min}$. Again use the nonlinear differential equations for the plant model. Do you have steady offset in any of the outputs? Which ones?

Compare and contrast the closed-loop performance for the design with two integrating disturbances and the design with three integrating disturbances. Which control system do you recommend and why?

Solution

(a) Integrating disturbances are added to the two controlled variables (first and third outputs) by choosing

$$C_d = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \qquad B_d = 0$$

The results with two integrating disturbances are shown in Figures 1.8 and 1.9. Notice that despite adding integrating disturbances to the two controlled variables, c and h, both of these controlled variables as well as the third output, T, all display nonzero offset at steady state.

(b) A third integrating disturbance is added to the second output giving

$$C_d = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \qquad B_d = 0$$

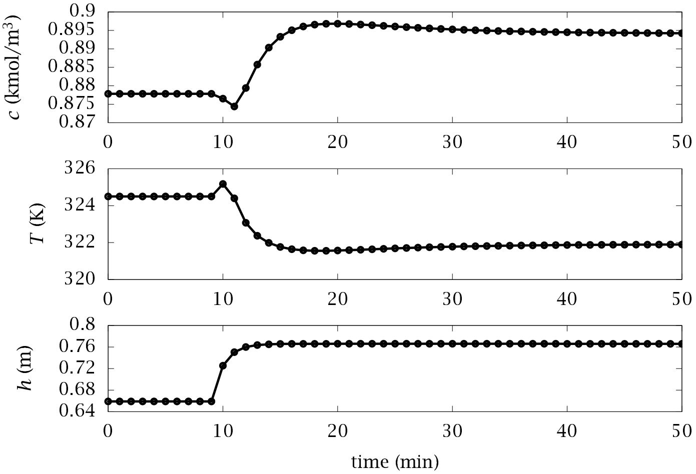

Figure 1.8: Three measured outputs versus time after a step change in inlet flowrate at 10 minutes; $n_d = 2$.

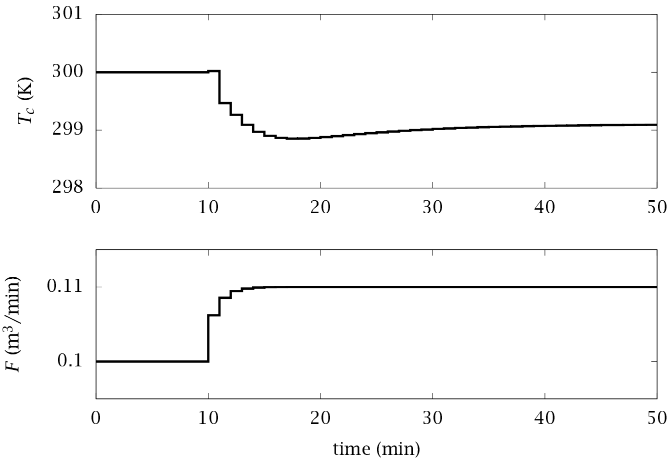

Figure 1.9: Two manipulated inputs versus time after a step change in inlet flowrate at 10 minutes; $n_d = 2$.

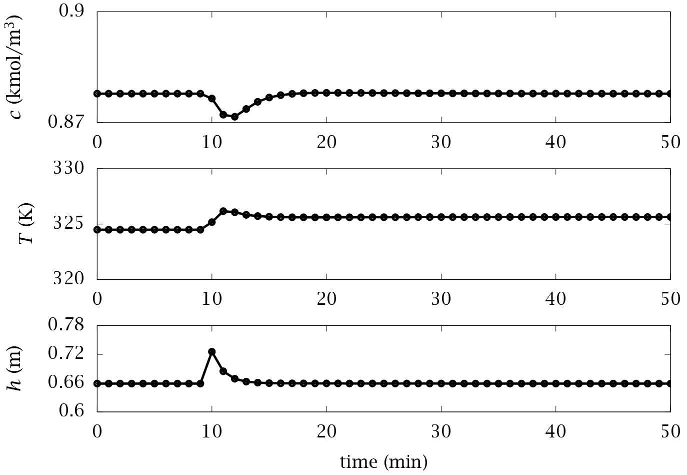

Figure 1.10: Three measured outputs versus time after a step change in inlet flowrate at 10 minutes; $n_d = 3$.

The augmented system is not detectable with this disturbance model. The rank of $\begin{bmatrix} I-A & -B_d \\ C & C_d \end{bmatrix}$ is only 5 instead of 6. The problem here is that the system level is itself an integrator, and we cannot distinguish h from the integrating disturbance added to h.

(c) Next we try three integrating disturbances: two added to the two controlled variables, and one added to the second manipulated variable

$$C_d = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \qquad B_d = \begin{bmatrix} 0 & 0 & 0.1655 \\ 0 & 0 & 97.91 \\ 0 & 0 & -6.637 \end{bmatrix}$$

The augmented system is detectable for this disturbance model.

The results for this choice of three integrating disturbances are shown in Figures 1.10 and 1.11. Notice that we have zero offset in the two controlled variables, c and h, and have successfully forced the steady-state effect of the inlet flowrate disturbance entirely into the second output, T.

Notice also that the dynamic behavior of all three outputs is superior to that achieved with the model using two integrating disturbances. The true disturbance, which is a step at the inlet flowrate,

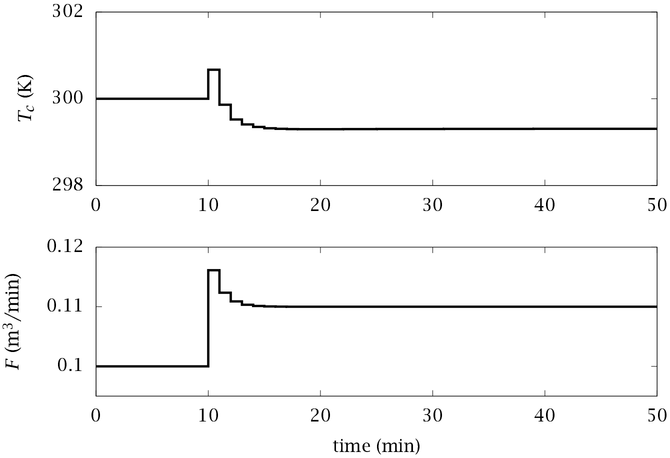

Figure 1.11: Two manipulated inputs versus time after a step change in inlet flowrate at 10 minutes; $n_d = 3$.

is better represented by including the integrator in the outlet flowrate. With a more accurate disturbance model, better overall control is achieved. The controller uses smaller manipulated variable action and also achieves better output variable behavior. An added bonus is that steady offset is removed in the maximum possible number of outputs. □

Further notation

G	transfer function matrix
m	mean of normally distributed random variable
T	reactor temperature
$\tilde{u}$	input deviation variable
x, y, z	spatial coordinates for a distributed system
$\tilde{x}$	state deviation variable

1.6 Exercises

Exercise 1.1: State space form for chemical reaction model

Consider the following chemical reaction kinetics for a two-step series reaction

$$\mathrm{A} \xrightarrow{k_1} \mathrm{B} \qquad \mathrm{B} \xrightarrow{k_2} \mathrm{C}$$

We wish to follow the reaction in a constant volume, well-mixed, batch reactor. As taught in the undergraduate chemical engineering curriculum, we proceed by writing material balances for the three species giving

$$\frac{dc_A}{dt} = -r_1 \qquad \frac{dc_B}{dt} = r_1 - r_2 \qquad \frac{dc_C}{dt} = r_2$$

in which c_j is the concentration of species j, and r_1 and r_2 are the rates (mol/(time·vol)) at which the two reactions occur. We then assume some rate law for the reaction kinetics, such as

$$r_1 = k_1 c_A \qquad r_2 = k_2 c_B$$

We substitute the rate laws into the material balances and specify the starting concentrations to produce three differential equations for the three species concentrations.

(a) Write the linear state space model for the deterministic series chemical reaction model. Assume we can measure the component A concentration. What are x, y, A, B, C, and D for this model?

(b) Simulate this model with initial conditions and parameters given by

$$c_{A0} = 1 \quad c_{B0} = c_{C0} = 0 \qquad k_1 = 2 \quad k_2 = 1$$

Exercise 1.2: Distributed systems and time delay

We assume familiarity with the transfer function of a time delay from an undergraduate systems course

$$\overline{y}(s) = e^{-\theta s}\overline{u}(s)$$

Let's see the connection between the delay and the distributed systems, which give rise to it. A simple physical example of a time delay is the delay caused by transport in a flowing system. Consider plug flow in a tube depicted in Figure 1.12.

(a) Write down the equation of change for moles of component j for an arbitrary volume element and show that

$$\frac{\partial c_j}{\partial t} = -\nabla \cdot (c_j v_j) + R_j$$

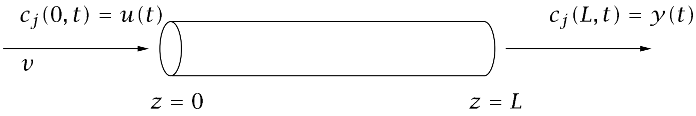

Figure 1.12: Plug-flow reactor.

in which c_j is the molar concentration of component j, v_j is the velocity of component j, and R_j is the production rate of component j due to chemical reaction.[9]

Plug flow means the fluid velocity of all components is purely in the z direction, and is independent of r and θ and, we assume here, z

$$v_j = v\delta_z$$

(b) Assuming plug flow and neglecting chemical reaction in the tube, show that the equation of change reduces to

$$\frac{\partial c_j}{\partial t} = -v\frac{\partial c_j}{\partial z} \tag{1.46}$$

This equation is known as a hyperbolic, first-order partial differential equation. Assume the boundary and initial conditions are

$$c_j(z,t) = u(t) \qquad 0 = z \qquad t \geq 0 \tag{1.47}$$

$$c_j(z,t) = c_{j0}(z) \qquad 0 \leq z \leq L \quad t = 0 \tag{1.48}$$

In other words, we are using the feed concentration as the manipulated variable, $u(t)$, and the tube starts out with some initial concentration profile of component j, $c_{j0}(z)$.

(c) Show that the solution to (1.46) with these boundary conditions is

$$c_j(z,t) = \begin{cases} u(t - z/v) & vt > z \\ c_{j0}(z - vt) & vt < z \end{cases} \tag{1.49}$$

(d) If the reactor starts out empty of component j, show that the transfer function between the outlet concentration, $y = c_j(L,t)$, and the inlet concentration, $c_j(0, t) = u(t)$, is a time delay. What is the value of θ?

Exercise 1.3: Pendulum in state space

Consider the pendulum suspended at the end of a rigid link depicted in Figure 1.13. Let r and θ denote the polar coordinates of the center of the pendulum, and let $p = r\delta_r$ be the position vector of the pendulum, in which δ_r and δ_θ are the unit vectors in polar coordinates. We wish to determine a state space description of the system. We are able to apply a torque T to the pendulum as our manipulated variable. The pendulum has mass m, the only other external force acting on the pendulum is gravity, and we neglect friction. The link provides force $-t\delta_r$ necessary to maintain the pendulum at distance $r = R$ from the axis of rotation, and we measure this force t.

(a) Provide expressions for the four partial derivatives for changes in the unit vectors with r and θ

$$\frac{\partial \delta_r}{\partial r} \qquad \frac{\partial \delta_r}{\partial \theta} \qquad \frac{\partial \delta_\theta}{\partial r} \qquad \frac{\partial \delta_\theta}{\partial \theta}$$

(b) Use the chain rule to find the velocity of the pendulum in terms of the time derivatives of r and θ. Do not simplify yet by assuming r is constant. We want the general result.

[9] You will need the Gauss divergence theorem and 3D Leibniz formula to go from a mass balance on a volume element to the equation of continuity.

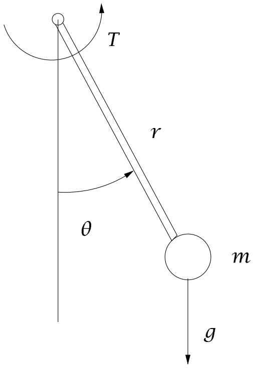

Figure 1.13: Pendulum with applied torque.

(c) Differentiate again to show that the acceleration of the pendulum is

$$\ddot{p} = (\ddot{r} - r\dot{\theta}^2)\delta_r + (r\ddot{\theta} + 2\dot{r}\dot{\theta})\delta_\theta$$

(d) Use a momentum balance on the pendulum mass (you may assume it is a point mass) to determine both the force exerted by the link

$$t = mR\dot{\theta}^2 + mg\cos\theta$$

and an equation for the acceleration of the pendulum due to gravity and the applied torque

$$mR\ddot{\theta} - T/R + mg\sin\theta = 0$$

(e) Define a state vector and give a state space description of your system. What is the physical significance of your state. Assume you measure the force exerted by the link.

One answer is

$$\begin{aligned}\frac{dx_1}{dt} &= x_2\\ \frac{dx_2}{dt} &= -(g/R)\sin x_1 + u\\ y &= mRx_2^2 + mg\cos x_1\end{aligned}$$

in which $u = T/(mR^2)$

Exercise 1.4: Time to Laplace domain

Take the Laplace transform of the following set of differential equations and find the transfer function, $G(s)$, connecting $\overline{u}(s)$ and $\overline{y}(s)$, $\overline{y} = G\overline{u}$

$$\begin{aligned}\frac{dx}{dt} &= Ax + Bu\\ y &= Cx + Du\end{aligned} \tag{1.50}$$

For $x \in \mathbb{R}^n$, $y \in \mathbb{R}^p$, and $u \in \mathbb{R}^m$, what is the dimension of the G matrix? What happens to the initial condition, $x(0) = x_0$?

Exercise 1.5: Converting between continuous and discrete time models

Given a prescribed $u(t)$, derive and check the solution to (1.50). Given a prescribed $u(k)$ sequence, what is the solution to the discrete time model

$$x(k+1) = \tilde{A}x(k) + \tilde{B}u(k)$$
$$y(k) = \tilde{C}x(k) + \tilde{D}u(k)$$

(a) Compute $\tilde{A}, \tilde{B}, \tilde{C}$, and $\tilde{D}$ so that the two solutions agree at the sample times for a zero-order hold input, i.e., $y(k) = y(t_k)$ for $u(t) = u(k)$, $t \in (t_k, t_{k+1})$ in which $t_k = k\Delta$ for sample time Δ.

(b) Is your result valid for A singular? If not, how can you find $\tilde{A}, \tilde{B}, \tilde{C}$, and $\tilde{D}$ for this case?

Exercise 1.6: Continuous to discrete time conversion for nonlinear models

Consider the autonomous nonlinear differential equation model

$$\frac{dx}{dt} = f(x,u)$$
$$x(0) = x_0 \tag{1.51}$$

Given a zero-order hold on the input, let $s(t,u,x_0), 0 \leq t \leq \Delta$, be the solution to (1.51) given initial condition x_0 at time $t = 0$, and constant input u is applied for t in the interval $0 \leq t \leq \Delta$. Consider also the nonlinear discrete time model

$$x(k+1) = F(x(k), u(k))$$

(a) What is the relationship between F and s so that the solution of the discrete time model agrees at the sample times with the continuous time model with a zero-order hold?

(b) Assume f is linear and apply this result to check the result of Exercise 1.5.

Exercise 1.7: Commuting functions of a matrix

Although matrix multiplication does not commute in general

$$AB \neq BA$$

multiplication of functions of the same matrix do commute. You may have used the following fact in Exercise 1.5

$$A^{-1}\exp(At) = \exp(At)A^{-1} \tag{1.52}$$

(a) Prove that (1.52) is true assuming A has distinct eigenvalues and can therefore be represented as

$$A = Q\Lambda Q^{-1} \qquad \Lambda = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n \end{bmatrix}$$

in which Λ is a diagonal matrix containing the eigenvalues of A, and Q is the matrix of eigenvectors such that

$$Aq_i = \lambda_i q_i, \qquad i = 1, \ldots, n$$

in which q_i is the ith column of matrix Q.

(b) Prove the more general relationship

$$f(A)g(A) = g(A)f(A) \tag{1.53}$$

in which f and g are any functions definable by Taylor series.

(c) Prove that (1.53) is true without assuming the eigenvalues are distinct.

Hint: use the Taylor series defining the functions and apply the Cayley-Hamilton theorem (Horn and Johnson, 1985, pp. 86–87).

Exercise 1.8: Finite difference formula and approximating the exponential

Instead of computing the exact conversion of a continuous time to a discrete time system as in Exercise 1.5, assume instead one simply approximates the time derivative with a first-order finite difference formula

$$\frac{dx}{dt} \approx \frac{x(t_{k+1}) - x(t_k)}{\Delta}$$

with step size equal to the sample time, Δ. For this approximation of the continuous time system, compute $\tilde{A}$ and $\tilde{B}$ so that the discrete time system agrees with the approximate continuous time system at the sample times. Comparing these answers to the exact solution, what approximation of $e^{A\Delta}$ results from the finite difference approximation? When is this a good approximation of $e^{A\Delta}$?

Exercise 1.9: Mapping eigenvalues of continuous time systems to discrete time systems

Consider the continuous time differential equation and discrete time difference equation

$$\frac{dx}{dt} = Ax$$
$$x^+ = \tilde{A}x$$

and the transformation

$$\tilde{A} = e^{A\Delta}$$

Consider the scalar A case.

(a) What A represents an integrator in continuous time? What is the corresponding $\tilde{A}$ value for the integrator in discrete time?

(b) What A give purely oscillatory solutions? What are the corresponding $\tilde{A}$?

(c) For what A is the solution of the ODE stable? Unstable? What are the corresponding $\tilde{A}$?

(d) Sketch and label these A and $\tilde{A}$ regions in two complex-plane diagrams.

Exercise 1.10: State space realization

Define a state vector and realize the following models as state space models **by hand**. One should do a few by hand to understand what the Octave or MATLAB calls are doing. Answer the following questions. What is the connection between the poles of G and the state space description? For what kinds of $G(s)$ does one obtain a nonzero D matrix? What is the order and gain of these systems? Is there a connection between order and the numbers of inputs and outputs?

(a) $G(s) = \dfrac{1}{2s+1}$

(b) $G(s) = \dfrac{1}{(2s+1)(3s+1)}$

(c) $G(s) = \dfrac{2s+1}{3s+1}$

(d) $y(k+1) = y(k) + 2u(k)$

(e) $y(k+1) = a_1 y(k) + a_2 y(k-1) + b_1 u(k) + b_2 u(k-1)$

Exercise 1.11: Minimal realization

Find minimal realizations of the state space models you found by hand in Exercise 1.10. Use Octave or MATLAB for computing minimal realizations. Were any of your hand realizations nonminimal?

Exercise 1.12: Partitioned matrix inversion lemma

Let matrix Z be partitioned into

$$Z = \begin{bmatrix} B & C \\ D & E \end{bmatrix}$$

and assume Z^{-1}, B^{-1} and E^{-1} exist.

(a) Perform row elimination and show that

$$Z^{-1} = \begin{bmatrix} B^{-1} + B^{-1}C(E - DB^{-1}C)^{-1}DB^{-1} & -B^{-1}C(E - DB^{-1}C)^{-1} \\ -(E - DB^{-1}C)^{-1}DB^{-1} & (E - DB^{-1}C)^{-1} \end{bmatrix}$$

Note that this result is still valid if E is singular.

(b) Perform column elimination and show that

$$Z^{-1} = \begin{bmatrix} (B - CE^{-1}D)^{-1} & -(B - CE^{-1}D)^{-1}CE^{-1} \\ -E^{-1}D(B - CE^{-1}D)^{-1} & E^{-1} + E^{-1}D(B - CE^{-1}D)^{-1}CE^{-1} \end{bmatrix}$$

Note that this result is still valid if B is singular.

(c) A host of other useful control-related inversion formulas follow from these results. Equate the (1,1) or (2,2) entries of Z^{-1} and derive the identity

$$(A + BCD)^{-1} = A^{-1} - A^{-1}B(DA^{-1}B + C^{-1})^{-1}DA^{-1} \tag{1.54}$$

A useful special case of this result is

$$(I + X^{-1})^{-1} = I - (I + X)^{-1}$$

(d) Equate the (1,2) or (2,1) entries of Z^{-1} and derive the identity

$$(A + BCD)^{-1}BC = A^{-1}B(DA^{-1}B + C^{-1})^{-1} \tag{1.55}$$

Equations (1.54) and (1.55) prove especially useful in rearranging formulas in least squares estimation.

Exercise 1.13: Perturbation to an asymptotically stable linear system

Given the system

$$x^+ = Ax + Bu$$

If A is an asymptotically stable matrix, prove that if $u(k) \to 0$, then $x(k) \to 0$.

Exercise 1.14: Exponential stability of a perturbed linear system

Given the system

$$x^+ = Ax + Bu$$

If A is an asymptotically stable matrix, prove that if $u(k)$ decreases exponentially to zero, then $x(k)$ decreases exponentially to zero.

Exercise 1.15: Are we going forward or backward today?

In the chapter we derived the solution to

$$\min_{w,x,y} f(w,x) + g(x,y) + h(y,z)$$

in which z is a fixed parameter using forward dynamic programming (DP)

$$\begin{aligned}
&\overline{y}^0(z) \\
&\tilde{x}^0(z) = \overline{x}^0(\overline{y}^0(z)) \\
&\tilde{w}^0(z) = \overline{w}^0(\overline{x}^0(\overline{y}^0(z)))
\end{aligned}$$

(a) Solve for optimal w as a function of z using backward DP.

(b) Is forward or backward DP more efficient if you want optimal w as a function of z?

Exercise 1.16: Method of Lagrange multipliers

Consider the objective function $V(x) = (1/2)x'Hx + h'x$ and optimization problem

$$\min_x V(x) \tag{1.56}$$

subject to

$$Dx = d$$

in which $H > 0$, $x \in \mathbb{R}^n$, $d \in \mathbb{R}^m$, $m < n$, i.e., fewer constraints than decisions. Rather than partially solving for x using the constraint and eliminating it, we make use of the method of Lagrange multipliers for treating the equality constraints (Fletcher, 1987; Nocedal and Wright, 2006).

In the method of Lagrange multipliers, we augment the objective function with the constraints to form the Lagrangian function, L

$$L(x, \lambda) = (1/2)x'Hx + h'x - \lambda'(Dx - d)$$

in which $\lambda \in \mathbb{R}^m$ is the vector of Lagrange multipliers. The necessary and sufficient conditions for a global minimizer are that the partial derivatives of L with respect to x and λ vanish (Nocedal and Wright, 2006, p. 451), (Fletcher, 1987, p.198,236)

(a) Show that the necessary and sufficient conditions are equivalent to the matrix equation

$$\begin{bmatrix} H & -D' \\ -D & 0 \end{bmatrix} \begin{bmatrix} x \\ \lambda \end{bmatrix} = - \begin{bmatrix} h \\ d \end{bmatrix} \tag{1.57}$$

The solution to (1.57) then provides the solution to the original problem (1.56).

(b) We note one other important feature of the Lagrange multipliers, their relationship to the optimal cost of the purely quadratic case. For $h = 0$, the cost is given by

$$V^0 = (1/2)(x^0)'Hx^0$$

Show that this can also be expressed in terms of λ^0 by the following

$$V^0 = (1/2)d'\lambda^0$$

Exercise 1.17: Minimizing a constrained, quadratic function

Consider optimizing the positive definite quadratic function subject to a linear constraint

$$\min_x (1/2)x'Hx \qquad \text{s.t. } Ax = b$$

Using the method of Lagrange multipliers presented in Exercise 1.16, show that the optimal solution, multiplier, and cost are given by

$$\begin{aligned} x^0 &= H^{-1}A'(AH^{-1}A')^{-1}b \\ \lambda^0 &= (AH^{-1}A')^{-1}b \\ V^0 &= (1/2)b'(AH^{-1}A')^{-1}b \end{aligned}$$

Exercise 1.18: Minimizing a partitioned quadratic function

Consider the partitioned constrained minimization

$$\min_{x_1,x_2} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}' \begin{bmatrix} H_1 & \\ & H_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

subject to

$$\begin{bmatrix} D & I \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = d$$

The solution to this optimization is required in two different forms, depending on whether one is solving an estimation or regulation problem. Show that the solution can be expressed in the following two forms if both H_1 and H_2 are full rank.

- Regulator form

$$\begin{aligned} V^0(d) &= d'(H_2 - H_2D(D'H_2D + H_1)^{-1}D'H_2)d \\ x_1^0(d) &= \widetilde{K}d \qquad \widetilde{K} = (D'H_2D + H_1)^{-1}D'H_2 \\ x_2^0(d) &= (I - D\widetilde{K})d \end{aligned}$$

- Estimator form

$$\begin{aligned} V^0(d) &= d'(DH_1^{-1}D' + H_2^{-1})^{-1}d \\ x_1^0(d) &= \widetilde{L}d \qquad \widetilde{L} = H_1^{-1}D'(DH_1^{-1}D' + H_2^{-1})^{-1} \\ x_2^0(d) &= (I - D\widetilde{L})d \end{aligned}$$

Exercise 1.19: Stabilizability and controllability canonical forms

Consider the partitioned system

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}^+ = \begin{bmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} B_1 \\ 0 \end{bmatrix} u$$

with (A_{11}, B_1) controllable. This form is known as controllability canonical form.

(a) Show that the system is *not* controllable by checking the rank of the controllability matrix.

(b) Show that the modes x_1 can be controlled from any $x_1(0)$ to any $x_1(n)$ with a sequence of inputs $u(0), \ldots, u(n-1)$, but the modes x_2 *cannot* be controlled from any $x_2(0)$ to any $x_2(n)$. The states x_2 are termed the uncontrollable modes.

(c) If A_{22} is stable the system is termed *stabilizable*. Although not all modes can be controlled, the uncontrollable modes are stable and decay to steady state.

The following lemma gives an equivalent condition for stabilizability.

Lemma 1.12 (Hautus lemma for stabilizability)**.** *A system is stabilizable if and only if*

$$\operatorname{rank}\begin{bmatrix} \lambda I - A & B \end{bmatrix} = n \qquad \textit{for all } |\lambda| \geq 1$$

Prove this lemma using Lemma 1.2 as the condition for controllability.

Exercise 1.20: Regulator stability, stabilizable systems, and semidefinite state penalty

(a) Show that the infinite horizon LQR is stabilizing for (A, B) *stabilizable* with R, $Q > 0$.

(b) Show that the infinite horizon LQR is stabilizing for (A, B) stabilizable and $R > 0$, $Q \geq 0$, and (A, Q) detectable. Discuss what happens to the controller's stabilizing property if Q is not positive semidefinite or (A, Q) is not detectable.

Exercise 1.21: Time-varying linear quadratic problem

Consider the time-varying version of the LQ problem solved in the chapter. The system model is

$$x(k+1) = A(k)x(k) + B(k)u(k)$$

The objective function also contains time-varying penalties

$$\min_{\mathbf{u}} V(x(0), \mathbf{u}) = \frac{1}{2}\left(\sum_{k=0}^{N-1} \left(x(k)'Q(k)x(k) + u(k)'R(k)u(k)\right) + x(N)'Q(N)x(N)\right)$$

subject to the model. Notice the penalty on the final state is now simply $Q(N)$ instead of P_f.

Apply the DP argument to this problem and determine the optimal input sequence and cost. Can this problem also be solved in closed form like the time-invariant case?

Exercise 1.22: Steady-state Riccati equation

Generate a random A and B for a system model for whatever $n(\geq 3)$ and $m(\geq 3)$ you wish. Choose a positive semidefinite Q and positive definite R of the appropriate sizes.

(a) Iterate the DARE by hand with Octave or MATLAB until Π stops changing. Save this result. Now call the MATLAB or Octave function to solve the steady-state DARE. Do the solutions agree? Where in the complex plane are the eigenvalues of $A + BK$? Increase the size of Q relative to R. Where do the eigenvalues move?

(b) Repeat for a singular A matrix. What happens to the two solution techniques?

(c) Repeat for an unstable A matrix.

Exercise 1.23: Positive definite Riccati iteration

If $\Pi(k), Q, R > 0$ in (1.10), show that $\Pi(k-1) > 0$.

Hint: apply (1.54) to the term $(B'\Pi(k)B + R)^{-1}$.

Exercise 1.24: Existence and uniqueness of the solution to constrained least squares

Consider the least squares problem subject to linear constraint

$$\min_x (1/2)x'Qx \qquad \text{subject to} \quad Ax = b$$

in which $x \in \mathbb{R}^n$, $b \in \mathbb{R}^p$, $Q \in \mathbb{R}^{n\times n}$, $Q \geq 0$, $A \in \mathbb{R}^{p\times n}$. Show that this problem has a solution for every b and the solution is unique if and only if

$$\operatorname{rank}(A) = p \qquad \operatorname{rank}\begin{bmatrix} Q \\ A \end{bmatrix} = n$$

Exercise 1.25: Rate-of-change penalty

Consider the generalized LQR problem with the cross term between $x(k)$ and $u(k)$

$$V(x(0),\mathbf{u}) = \frac{1}{2}\sum_{k=0}^{N-1}\left(x(k)'Qx(k) + u(k)'Ru(k) + 2x(k)'Mu(k)\right) + (1/2)x(N)'P_f x(N)$$

(a) Solve this problem with backward DP and write out the Riccati iteration and feedback gain.

(b) Control engineers often wish to tune a regulator by penalizing the rate of change of the input rather than the absolute size of the input. Consider the additional positive definite penalty matrix S and the modified objective function

$$V(x(0),\mathbf{u}) = \frac{1}{2}\sum_{k=0}^{N-1}\left(x(k)'Qx(k) + u(k)'Ru(k) + \Delta u(k)'S\Delta u(k)\right) + (1/2)x(N)'P_f x(N)$$

in which $\Delta u(k) = u(k) - u(k-1)$. Show that you can augment the state to include $u(k-1)$ via

$$\tilde{x}(k) = \begin{bmatrix} x(k) \\ u(k-1) \end{bmatrix}$$

and reduce this new problem to the standard LQR with the cross term. What are $\widetilde{A}$, $\widetilde{B}$, $\widetilde{Q}$, $\widetilde{R}$, and $\widetilde{M}$ for the augmented problem (Rao and Rawlings, 1999)?

Exercise 1.26: Existence, uniqueness and stability with the cross term

Consider the linear quadratic problem with system

$$x^+ = Ax + Bu \tag{1.58}$$

and infinite horizon cost function

$$V(x(0), \mathbf{u}) = (1/2) \sum_{k=0}^{\infty} x(k)'Qx(k) + u(k)'Ru(k)$$

The existence, uniqueness and stability conditions for this problem are: (A, B) stabilizable, $Q \geq 0$, (A, Q) detectable, and $R > 0$. Consider the modified objective function with the cross term

$$V = (1/2) \sum_{k=0}^{\infty} x(k)'Qx(k) + u(k)'Ru(k) + 2x(k)'Mu(k) \tag{1.59}$$

(a) Consider reparameterizing the input as

$$v(k) = u(k) + Tx(k) \tag{1.60}$$

Choose T such that the cost function in x and v does not have a cross term, and express the existence, uniqueness and stability conditions for the transformed system. Goodwin and Sin (1984, p.251) discuss this procedure in the state estimation problem with nonzero covariance between state and output measurement noises.

(b) Translate and simplify these to obtain the existence, uniqueness and stability conditions for the original system with cross term.

Exercise 1.27: Forecasting and variance increase or decrease

Given positive definite initial state variance $P(0)$ and process disturbance variance Q, the variance after forecasting one sample time was shown to be

$$P^-(1) = AP(0)A' + Q$$

(a) If A is stable, is it true that $AP(0)A' < P(0)$? If so, prove it. If not, provide a counterexample.

(b) If A is unstable, is it true that $AP(0)A' > P(0)$? If so, prove it. If not, provide a counterexample.

(c) If the magnitudes of *all* the eigenvalues of A are unstable, is it true that $AP(0)A' > P(0)$? If so, prove it. If not, provide a counterexample.

Exercise 1.28: Convergence of MHE with zero prior weighting

Show that the simplest form of MHE defined in (1.32) and (1.33) is also a convergent estimator for an observable system. What restrictions on the horizon length N do you require for this result to hold?

Hint: you can solve the MHE optimization problem by inspection when there is no prior weighting of the data.

Exercise 1.29: Symmetry in regulation and estimation

In this exercise we display the symmetry of the backward DP recursion for regulation, and the forward DP recursion for estimation. In the regulation problem we solve at stage k

$$\min_{x,u} \ell(z,u) + V_k^0(x) \qquad \text{s.t. } x = Az + Bu$$

In backward DP, x is the state at the current stage and z is the state at the previous stage. The stage cost and cost to go are given by

$$\ell(z,u) = (1/2)(z'Qz + u'Ru) \qquad V_k^0(x) = (1/2)x'\Pi(k)x$$

and the optimal cost is $V_{k-1}^0(z)$ since z is the state at the previous stage.

In estimation we solve at stage k

$$\min_{x,w} \ell(z,w) + V_k^0(x) \qquad \text{s.t. } z = Ax + w$$

In forward DP, x is the state at the current stage, z is the state at the next stage. The stage cost and arrival cost are given by

$$\ell(z,w) = (1/2)(\,|y(k+1) - Cz|_{R^{-1}}^2 + w'Q^{-1}w) \qquad V_k^0(x) = (1/2)\,|x - \hat{x}(k)|_{P(k)^{-1}}^2$$

and we wish to find $V_{k+1}^0(z)$ in the estimation problem.

(a) In the estimation problem, take the z term outside the optimization and solve

$$\min_{x,w} \frac{1}{2}\left(w'Q^{-1}w + (x - \hat{x}(k))'P(k)^{-1}(x - \hat{x}(k))\right) \qquad \text{s.t. } z = Ax + w$$

using the inverse form in Exercise 1.18, and show that the optimal cost is given by

$$\begin{aligned} V^0(z) &= (1/2)(z - A\hat{x}(k))'(P^-(k+1))^{-1}(z - A\hat{x}(k)) \\ P^-(k+1) &= AP(k)A' + Q \end{aligned}$$

Add the z term to this cost using the third part of Example 1.1 and show that

$$\begin{aligned} V_{k+1}^0(z) &= (1/2)(z - \hat{x}(k+1))'P^{-1}(k+1)(z - \hat{x}(k+1)) \\ P(k+1) &= P^-(k+1) - P^-(k+1)C'(CP^-(k+1)C' + R)^{-1}CP^-(k+1) \\ \hat{x}(k+1) &= A\hat{x}(k) + L(k+1)(y(k+1) - CA\hat{x}(k)) \\ L(k+1) &= P^-(k+1)C'(CP^-(k+1)C' + R)^{-1} \end{aligned}$$

(b) In the regulator problem, take the z term outside the optimization and solve the remaining two-term problem using the regulator form of Exercise 1.18. Then add the z term and show that

$$\begin{aligned}
V_{k-1}^0(z) &= (1/2)z'\Pi(k-1)z \\
\Pi(k-1) &= Q + A'\Pi(k)A - A'\Pi(k)B(B'\Pi(k)B+R)^{-1}B'\Pi(k)A \\
u^0(z) &= K(k-1)z \\
x^0(z) &= (A+BK(k-1))z \\
K(k-1) &= -(B'\Pi(k)B+R)^{-1}B'\Pi(k)A
\end{aligned}$$

This symmetry can be developed further if we pose an output tracking problem rather than zero state regulation problem in the regulator.

Exercise 1.30: Symmetry in the Riccati iteration

Show that the covariance before measurement $P^-(k+1)$ in estimation satisfies an identical iteration to the cost to go $\Pi(k-1)$ in regulation under the change of variables $P^- \longrightarrow \Pi$, $A \longrightarrow A'$, $C \longrightarrow B'$.

Exercise 1.31: Detectability and observability canonical forms

Consider the partitioned system

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}^+ = \begin{bmatrix} A_{11} & 0 \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$$y = \begin{bmatrix} C_1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

with (A_{11}, C_1) observable. This form is known as observability canonical form.

(a) Show that the system is *not* observable by checking the rank of the observability matrix.

(b) Show that the modes x_1 can be uniquely determined from a sequence of measurements, but the modes x_2 *cannot* be uniquely determined from the measurements. The states x_2 are termed the unobservable modes.

(c) If A_{22} is stable the system is termed *detectable*. Although not all modes can be observed, the unobservable modes are stable and decay to steady state.

The following lemma gives an equivalent condition for detectability.

Lemma 1.13 (Hautus lemma for detectability)**.** *A system is detectable if and only if*

$$\operatorname{rank} \begin{bmatrix} \lambda I - A \\ C \end{bmatrix} = n \qquad \textit{for all } |\lambda| \geq 1$$

Prove this lemma using Lemma 1.4 as the condition for observability.

Exercise 1.32: Estimator stability and detectable systems

Show that the least squares estimator given in (1.27) is stable for (A, C) *detectable* with $Q > 0$.

Exercise 1.33: Estimator stability and semidefinite state noise penalty

We wish to show that the least squares estimator is stable for (A, C) detectable and $Q \geq 0$, (A, Q) stabilizable.

(a) Because Q^{-1} is not defined in this problem, the objective function defined in (1.26) requires modification. Show that the objective function with semidefinite $Q \geq 0$ can be converted into the following form

$$V(x(0), \mathbf{w}(T)) = \frac{1}{2}\Big(|x(0) - \overline{x}(0)|^2_{(P^-(0))^{-1}} + \sum_{k=0}^{T-1} |w(k)|^2_{\widetilde{Q}^{-1}} + \sum_{k=0}^{T} |y(k) - Cx(k)|^2_{R^{-1}} \Big)$$

in which

$$x^+ = Ax + Gw \qquad \widetilde{Q} > 0$$

Find expressions for $\widetilde{Q}$ and G in terms of the original semidefinite Q. How are the dimension of $\widetilde{Q}$ and G related to the rank of Q?

(b) What is the probabilistic interpretation of the state estimation problem with semidefinite Q?

(c) Show that (A, Q) stabilizable implies (A, G) stabilizable in the converted form.

(d) Show that this estimator is stable for (A, C) detectable and (A, G) stabilizable with $\widetilde{Q}, R > 0$.

(e) Discuss what happens to the estimator's stability if Q is not positive semidefinite or (A, Q) is not stabilizable.

Exercise 1.34: Calculating mean and variance from data

We are sampling a real-valued scalar random variable $x(k) \in \mathbb{R}$ at time k. Assume the random variable comes from a distribution with mean $\overline{x}$ and variance P, and the samples at different times are statistically independent.

A colleague has suggested the following formulas for estimating the mean and variance from N samples

$$\hat{x}_N = \frac{1}{N} \sum_{j=1}^{N} x(j) \qquad \hat{P}_N = \frac{1}{N} \sum_{j=1}^{N} (x(j) - \hat{x}_N)^2$$

(a) Prove that the estimate of the mean is unbiased for all N, i.e., show that for all N

$$\mathcal{E}(\hat{x}_N) = \overline{x}$$

(b) Prove that the estimate of the variance is not unbiased for any N, i.e., show that for all N

$$\mathcal{E}(\hat{P}_N) \neq P$$

(c) Using the result above, provide an alternative formula for the variance estimate that is unbiased for all N. How large does N have to be before these two estimates of P are within 1%?

Exercise 1.35: Expected sum of squares

Given that a random variable x has mean m and covariance P, show that the expected sum of squares is given by the formula (Selby, 1973, p.138)

$$\mathcal{E}(x'Qx) = m'Qm + \operatorname{tr}(QP)$$

The trace of a square matrix A, written $\operatorname{tr}(A)$, is defined to be the sum of the diagonal elements

$$\operatorname{tr}(A) := \sum_i A_{ii}$$

Exercise 1.36: Normal distribution

Given a normal distribution with scalar parameters m and σ

$$p_\xi(x) = \sqrt{\frac{1}{2\pi\sigma^2}} \exp\left[-\frac{1}{2}\left(\frac{x-m}{\sigma}\right)^2\right] \tag{1.61}$$

By direct calculation, show that

(a)

$$\mathcal{E}(\xi) = m$$
$$\operatorname{var}(\xi) = \sigma^2$$

(b) Show that the mean and the maximum likelihood are equal for the normal distribution. Draw a sketch of this result. The maximum likelihood estimate, $\hat{x}$, is defined as

$$\hat{x} := \arg\max_x p_\xi(x)$$

in which arg returns the solution to the optimization problem.

Exercise 1.37: Conditional densities are positive definite

We show in Example A.44 that if ξ and η are jointly normally distributed as

$$\begin{aligned} \begin{bmatrix} \xi \\ \eta \end{bmatrix} &\sim N(m, P) \\ &\sim N\left(\begin{bmatrix} m_x \\ m_y \end{bmatrix}, \begin{bmatrix} P_x & P_{xy} \\ P_{yx} & P_y \end{bmatrix}\right) \end{aligned}$$

then the conditional density of ξ given η is also normal

$$(\xi|\eta) \sim N(m_{x|y}, P_{x|y})$$

in which the conditional mean is

$$m_{x|y} = m_x + P_{xy}P_y^{-1}(y - m_y)$$

and the conditional covariance is

$$P_{x|y} = P_x - P_{xy}P_y^{-1}P_{yx}$$

Given that the joint density is well defined, prove the marginal densities and the conditional densities also are well defined, i.e., given $P > 0$, prove $P_x > 0$, $P_y > 0$, $P_{x|y} > 0$, $P_{y|x} > 0$.

Exercise 1.38: Expectation and covariance under linear transformations

Consider the random variable $x \in \mathbb{R}^n$ with density p_x and mean and covariance

$$\mathcal{E}(x) = m_x \qquad \text{cov}(x) = P_x$$

Consider the random variable $y \in \mathbb{R}^p$ defined by the linear transformation

$$y = Cx$$

(a) Show that the mean and covariance for y are given by

$$\mathcal{E}(y) = Cm_x \qquad \text{cov}(y) = CP_xC'$$

Does this result hold for all C? If yes, prove it; if no, provide a counterexample.

(b) Apply this result to solve Exercise A.35.

Exercise 1.39: Normal distributions under linear transformations

Given the normally distributed random variable, $\xi \in \mathbb{R}^n$, consider the random variable, $\eta \in \mathbb{R}^n$, obtained by the linear transformation

$$\eta = A\xi$$

in which A is a nonsingular matrix. Using the result on transforming probability densities, show that if $\xi \sim N(m, P)$, then $\eta \sim N(Am, APA')$. This result basically says that linear transformations of normal random variables are normal.

Exercise 1.40: More on normals and linear transformations

Consider a normally distributed random variable $x \in \mathbb{R}^n$, $x \sim N(m_x, P_x)$. You showed in Exercise 1.39 for $C \in \mathbb{R}^{n\times n}$ invertible, that the random variable y defined by the linear transformation $y = Cx$ is also normal and is distributed as

$$y \sim N(Cm_x, CP_xC')$$

Does this result hold for all C? If yes, prove it; if no, provide a counterexample.

Exercise 1.41: Signal processing in the good old days—recursive least squares

Imagine we are sent back in time to 1960 and the only computers available have extremely small memories. Say we have a large amount of data coming from a process and we want to compute the least squares estimate of model parameters from these data. Our immediate challenge is that we cannot load all of these data into memory to make the standard least squares calculation.

Alternatively, go 150 years further back in time and consider the situation from Gauss's perspective,

> It occasionally happens that after we have completed all parts of an extended calculation on a sequence of observations, we learn of a new observation that we would like to include. In many cases we will not want to have to redo the entire elimination but instead to find the modifications due to the new observation in the most reliable values of the unknowns and in their weights.
> C.F. Gauss, 1823
> G.W. Stewart Translation, 1995, p. 191.

Given the linear model

$$y_i = X_i'\theta$$

in which scalar y_i is the measurement at sample i, X_i' is the independent model variable (row vector, $1 \times p$) at sample i, and θ is the parameter vector ($p \times 1$) to be estimated from these data. Given the weighted least squares objective and n measurements, we wish to compute the usual estimate

$$\hat{\theta} = (X'X)^{-1}X'y \tag{1.62}$$

in which

$$y = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} \qquad X = \begin{bmatrix} X_1' \\ \vdots \\ X_n' \end{bmatrix}$$

We do not wish to store the large matrices $X(n \times p)$ and $y(n \times 1)$ required for this calculation. Because we are planning to process the data one at a time, we first modify our usual least squares problem to deal with small n. For example, we wish to estimate the parameters when $n < p$ and the inverse in (1.62) does not exist. In such cases, we may choose to regularize the problem by modifying the objective function as follows

$$\Phi(\theta) = (\theta - \overline{\theta})'P_0^{-1}(\theta - \overline{\theta}) + \sum_{i=1}^{n}(y_i - X_i'\theta)^2$$

in which $\overline{\theta}$ and P_0 are chosen by the user. In Bayesian estimation, we call $\overline{\theta}$ and P_0 the prior information, and often assume that the prior density of θ (without measurements) is normal

$$\theta \sim N(\overline{\theta}, P_0)$$

The solution to this modified least squares estimation problem is

$$\hat{\theta} = \overline{\theta} + (X'X + P_0^{-1})^{-1}X'(y - X\overline{\theta}) \tag{1.63}$$

Devise a means to *recursively* estimate θ so that:

1. We never store more than one measurement at a time in memory.
2. After processing all the measurements, we obtain the same least squares estimate given in (1.63).

Exercise 1.42: Least squares parameter estimation and Bayesian estimation

Consider a model linear in the parameters

$$y = X\theta + e \tag{1.64}$$

in which $y \in \mathbb{R}^p$ is a vector of measurements, $\theta \in \mathbb{R}^m$ is a vector of parameters, $X \in \mathbb{R}^{p \times m}$ is a matrix of known constants, and $e \in \mathbb{R}^p$ is a random variable modeling the measurement error. The standard parameter estimation problem is to find the best estimate of θ given the measurements y corrupted with measurement error e, which we assume is distributed as

$$e \sim N(0, R)$$

(a) Consider the case in which the errors in the measurements are independently and identically distributed with variance σ^2, $R = \sigma^2 I$. For this case, the classic least squares problem and solution are

$$\min_{\theta} \left| y - X\theta \right|^2 \qquad \hat{\theta} = \left(X'X\right)^{-1} X'y$$

Consider the measurements to be sampled from (1.64) with true parameter value θ_0. Show that using the least squares formula, the parameter estimate is distributed as

$$\hat{\theta} \sim N(\theta_0, P_{\hat{\theta}}) \qquad P_{\hat{\theta}} = \sigma^2 \left(X'X\right)^{-1}$$

(b) Now consider again the model of (1.64) and a Bayesian estimation problem. Assume a prior distribution for the random variable θ

$$\theta \sim N(\overline{\theta}, \overline{P})$$

Compute the conditional density of θ given measurement y, show that this density is normal, and find its mean and covariance

$$p_{\theta|y}(\theta|y) = n(\theta, m, P)$$

Show that Bayesian estimation and least squares estimation give the same result in the limit of an infinite variance prior. In other words, if the covariance of the prior is large compared to the covariance of the measurement error, show that

$$m \approx (X'X)^{-1}X'y \qquad P \approx P_{\hat{\theta}}$$

(c) What (weighted) least squares minimization problem is solved for the general measurement error covariance

$$e \sim N(0, R)$$

Derive the least squares estimate formula for this case.

(d) Again consider the measurements to be sampled from (1.64) with true parameter value θ_0. Show that the weighted least squares formula gives parameter estimates that are distributed as

$$\hat{\theta} \sim N(\theta_0, P_{\hat{\theta}})$$

and find $P_{\hat{\theta}}$ for this case.

(e) Show again that Bayesian estimation and least squares estimation give the same result in the limit of an infinite variance prior.

Exercise 1.43: Least squares and minimum variance estimation

Consider again the model linear in the parameters and the least squares estimator from Exercise 1.42

$$y = X\theta + e \qquad e \sim N(0, R)$$
$$\hat{\theta} = \left(X'R^{-1}X\right)^{-1} X'R^{-1}y$$

Show that the covariance of the least squares estimator is the smallest covariance of all linear unbiased estimators.

Exercise 1.44: Two stages are not better than one

We often can decompose an estimation problem into stages. Consider the following case in which we wish to estimate x from measurements of z, but we have the model between x and an intermediate variable, y, and the model between y and z

$$y = Ax + e_1 \qquad \text{cov}(e_1) = Q_1$$
$$z = By + e_2 \qquad \text{cov}(e_2) = Q_2$$

(a) Write down the optimal least squares problem to solve for $\hat{y}$ given the z measurements and the second model. Given $\hat{y}$, write down the optimal least squares problem for $\hat{x}$ in terms of $\hat{y}$. Combine these two results together and write the resulting estimate of $\hat{x}$ given measurements of z. Call this the two-stage estimate of x.

(b) Combine the two models together into a single model and show that the relationship between z and x is

$$z = BAx + e_3 \qquad \text{cov}(e_3) = Q_3$$

Express Q_3 in terms of Q_1, Q_2 and the models A, B. What is the optimal least squares estimate of $\hat{x}$ given measurements of z and the one-stage model? Call this the one-stage estimate of x.

(c) Are the one-stage and two-stage estimates of x the same? If yes, prove it. If no, provide a counterexample. Do you have to make any assumptions about the models A, B?

Exercise 1.45: Time-varying Kalman filter

Derive formulas for the conditional densities of $x(k)|\mathbf{y}(k-1)$ and $x(k)|\mathbf{y}(k)$ for the time-varying linear system

$$\begin{aligned} x(k+1) &= A(k)x(k) + G(k)w(k) \\ y(k) &= C(k)x(k) + v(k) \end{aligned}$$

in which the initial state, state noise and measurement noise are independently distributed as

$$x(0) \sim N(\overline{x}_0, Q_0) \qquad w(k) \sim N(0, Q) \qquad v(k) \sim N(0, R)$$

Exercise 1.46: More on conditional densities

In deriving the discrete time Kalman filter, we have $p_{x|\mathbf{y}}(x(k)|\mathbf{y}(k))$ and we wish to calculate recursively $p_{x|\mathbf{y}}(x(k+1)|\mathbf{y}(k+1))$ after we collect the output measurement at time $k+1$. It is straightforward to calculate $p_{x,y|\mathbf{y}}(x(k+1), y(k+1)|\mathbf{y}(k))$ from our established results on normal densities and knowledge of $p_{x|\mathbf{y}}(x(k)|\mathbf{y}(k))$, but we still need to establish a formula for pushing the $y(k+1)$ to the other side of the conditional density bar. Consider the following statement as a possible lemma to aid in this operation.

$$p_{a|b,c}(a|b,c) = \frac{p_{a,b|c}(a,b|c)}{p_{b|c}(b|c)}$$

If this statement is true, prove it. If it is false, give a counterexample.

Exercise 1.47: Other useful conditional densities

Using the definitions of marginal and conditional density, establish the following useful conditional density relations

1. $p_{A|B}(a|b) = \int p_{A|B,C}(a|b,c)p_{C|B}(c|b)dc$
2. $p_{A|B,C}(a|b,c) = p_{C|A,B}(c|a,b)\dfrac{p_{A|B}(a|b)}{p_{C|B}(c|b)}$

Exercise 1.48: Optimal filtering and deterministic least squares

Given the data sequence $(y(0), \ldots, y(k))$ and the system model

$$\begin{aligned} x^+ &= Ax + w \\ y &= Cx + v \end{aligned}$$

(a) Write down a least squares problem whose solution would provide a good state estimate for $x(k)$ in this situation. What probabilistic interpretation can you assign to the estimate calculated from this least squares problem?

(b) Now consider the nonlinear model

$$\begin{aligned} x^+ &= f(x) + w \\ y &= g(x) + v \end{aligned}$$

What is the corresponding nonlinear least squares problem for estimating $x(k)$ in this situation? What probabilistic interpretation, if any, can you assign to this estimate in the nonlinear model context?

(c) What is the motivation for changing from these least squares estimators to the moving horizon estimators we discussed in the chapter?

Exercise 1.49: A nonlinear transformation and conditional density

Consider the following relationship between the random variable y, and x and v

$$y = f(x) + v$$

The author of a famous textbook wants us to believe that

$$p_{y|x}(y|x) = p_v(y - f(x))$$

Derive this result and state what additional assumptions on the random variables x and v are required for this result to be correct.

Exercise 1.50: Some smoothing

One of the problems with asking you to derive the Kalman filter is that the derivation is in so many textbooks that it is difficult to tell if you are thinking independently. So here's a variation on the theme that should help you evaluate your level of understanding of these ideas. Let's calculate a smoothed rather than filtered estimate and covariance. Here's the problem.

We have the usual setup with a prior on $x(0)$

$$x(0) \sim N(\overline{x}(0), Q_0)$$

and we receive data from the following system

$$\begin{aligned} x(k+1) &= Ax(k) + w(k) \\ y(k) &= Cx(k) + v(k) \end{aligned}$$

in which the random variables $w(k)$ and $v(k)$ are independent, identically distributed normals, $w(k) \sim N(0, Q)$, $v(k) \sim N(0, R)$.

(a) Calculate the standard density for the filtering problem, $p_{x(0)|y(0)}(x(0)|y(0))$.

(b) Now calculate the density for the smoothing problem

$$p_{x(0)|y(0),y(1)}(x(0)|y(0),y(1))$$

that is, *not* the usual $p_{x(1)|y(0),y(1)}(x(1)|y(0),y(1))$.

Exercise 1.51: Alive on arrival

The following two optimization problems are helpful in understanding the arrival cost decomposition in state estimation.

(a) Let $V(x,y,z)$ be a positive, strictly convex function consisting of the sum of two functions, one of which depends on both x and y, and the other of which depends on y and z

$$V(x,y,z) = g(x,y) + h(y,z) \qquad V : \mathbb{R}^m \times \mathbb{R}^n \times \mathbb{R}^p \to \mathbb{R}_{\geq 0}$$

Consider the optimization problem

$$P1 : \min_{x,y,z} V(x,y,z)$$

The arrival cost decomposes this three-variable optimization problem into two, smaller dimensional optimization problems. Define the "arrival cost" $\widetilde{g}$ for this problem as the solution to the following single-variable optimization problem

$$\widetilde{g}(y) = \min_x g(x,y)$$

and define optimization problem $P2$ as follows

$$P2 : \min_{y,z} \widetilde{g}(y) + h(y,z)$$

Let (x',y',z') denote the solution to $P1$ and (x^0,y^0,z^0) denote the solution to $P2$, in which

$$x^0 = \arg\min_x g(x,y^0)$$

Prove that the two solutions are equal

$$(x',y',z') = (x^0,y^0,z^0)$$

(b) Repeat the previous part for the following optimization problems

$$V(x,y,z) = g(x) + h(y,z)$$

Here the y variables do not appear in g but restrict the x variables through a linear constraint. The two optimization problems are

$$P1 : \min_{x,y,z} V(x,y,z) \qquad \text{subject to } Ex = y$$

$$P2 : \min_{y,z} \widetilde{g}(y) + h(y,z)$$

in which

$$\widetilde{g}(y) = \min_x g(x) \qquad \text{subject to } Ex = y$$

Exercise 1.52: On-time arrival

Consider the deterministic, full information state estimation optimization problem

$$\min_{x(0),\mathbf{w},\mathbf{v}} \frac{1}{2}\left(|x(0) - \overline{x}(0)|^2_{(P^-(0))^{-1}} + \sum_{i=0}^{T-1} |w(i)|^2_{Q^{-1}} + |v(i)|^2_{R^{-1}} \right) \tag{1.65}$$

subject to

$$\begin{aligned} x^+ &= Ax + w \\ y &= Cx + v \end{aligned} \tag{1.66}$$

in which the sequence of measurements $\mathbf{y}(T)$ are known values. Notice we assume the noise-shaping matrix, G, is an identity matrix here. See Exercise 1.53 for the general case. Using the result of the first part of Exercise 1.51, show that this problem is equivalent to the following problem

$$\min_{x(T-N),\mathbf{w},\mathbf{v}} V^-_{T-N}(x(T-N)) + \frac{1}{2} \sum_{i=T-N}^{T-1} |w(i)|^2_{Q^{-1}} + |v(i)|^2_{R^{-1}}$$

subject to (1.66). The arrival cost is defined as

$$V^-_N(a) := \min_{x(0),\mathbf{w},\mathbf{v}} \frac{1}{2}\left(|x(0) - \overline{x}(0)|^2_{(P^-(0))^{-1}} + \sum_{i=0}^{N-1} |w(i)|^2_{Q^{-1}} + |v(i)|^2_{R^{-1}} \right)$$

subject to (1.66) and $x(N) = a$. Notice that any value of N, $0 \leq N \leq T$, can be used to split the cost function using the arrival cost.

Exercise 1.53: Arrival cost with noise-shaping matrix *G*

Consider the deterministic, full information state estimation optimization problem

$$\min_{x(0),\mathbf{w},\mathbf{v}} \frac{1}{2}\left(|x(0) - \overline{x}(0)|^2_{(P^-(0))^{-1}} + \sum_{i=0}^{T-1} |w(i)|^2_{Q^{-1}} + |v(i)|^2_{R^{-1}} \right)$$

subject to

$$\begin{aligned} x^+ &= Ax + Gw \\ y &= Cx + v \end{aligned} \tag{1.67}$$

in which the sequence of measurements $\mathbf{y}$ are known values. Using the result of the second part of Exercise 1.51, show that this problem also is equivalent to the following problem

$$\min_{x(T-N),\mathbf{w},\mathbf{v}} V^-_{T-N}(x(T-N)) + \frac{1}{2}\left(\sum_{i=T-N}^{T-1} |w(i)|^2_{Q^{-1}} + |v(i)|^2_{R^{-1}} \right)$$

subject to (1.67). The arrival cost is defined for all $k \geq 0$ and $a \in \mathbb{R}^n$ by

$$V^-_k(a) := \min_{x(0),\mathbf{w},\mathbf{v}} \frac{1}{2}\left(|x(0) - \overline{x}(0)|^2_{(P^-(0))^{-1}} + \sum_{i=0}^{k-1} |w(i)|^2_{Q^{-1}} + |v(i)|^2_{R^{-1}} \right)$$

subject to $x(k) = a$ and the model (1.67). Notice that any value of N, $0 \leq N \leq T$, can be used to split the cost function using the arrival cost.

Exercise 1.54: Where is the steady state?

Consider the two-input, two-output system

$$A = \begin{bmatrix} 0.5 & 0 & 0 & 0 \\ 0 & 0.6 & 0 & 0 \\ 0 & 0 & 0.5 & 0 \\ 0 & 0 & 0 & 0.6 \end{bmatrix} \qquad B = \begin{bmatrix} 0.5 & 0 \\ 0 & 0.4 \\ 0.25 & 0 \\ 0 & 0.6 \end{bmatrix} \qquad C = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

(a) The output setpoint is $y_{\mathrm{sp}} = \begin{bmatrix} 1 & -1 \end{bmatrix}'$ and the input setpoint is $u_{\mathrm{sp}} = \begin{bmatrix} 0 & 0 \end{bmatrix}'$. Calculate the target triple (x_s, u_s, y_s). Is the output setpoint feasible, i.e., does $y_s = y_{\mathrm{sp}}$?

(b) Assume only input one u_1 is available for control. Is the output setpoint feasible? What is the target in this case using $Q_s = I$?

(c) Assume both inputs are available for control but only the first output has a setpoint, $y_{1t} = 1$. What is the solution to the target problem for $R_s = I$?

Exercise 1.55: Detectability of integrating disturbance models

(a) Prove Lemma 1.8; the augmented system is detectable if and only if the system (A, C) is detectable and

$$\mathrm{rank} \begin{bmatrix} I - A & -B_d \\ C & C_d \end{bmatrix} = n + n_d$$

(b) Prove Corollary 1.9; the augmented system is detectable only if $n_d \leq p$.

Exercise 1.56: Unconstrained tracking problem

(a) For an *unconstrained* system, show that the following condition is *sufficient* for feasibility of the target problem for any r_{sp}.

$$\mathrm{rank} \begin{bmatrix} I - A & -B \\ HC & 0 \end{bmatrix} = n + n_c \tag{1.68}$$

(b) Show that (1.68) implies that the number of controlled variables without offset is less than or equal to the number of manipulated variables and the number of measurements, $n_c \leq m$ and $n_c \leq p$.

(c) Show that (1.68) implies the rows of H are independent.

(d) Does (1.68) imply that the rows of C are independent? If so, prove it; if not, provide a counterexample.

(e) By choosing H, how can one satisfy (1.68) if one has installed redundant sensors so several rows of C are identical?

Exercise 1.57: Unconstrained tracking problem for stabilizable systems

If we restrict attention to stabilizable systems, the sufficient condition of Exercise 1.56 becomes a necessary and sufficient condition. Prove the following lemma.

Lemma 1.14 (Stabilizable systems and feasible targets). *Consider an unconstrained, stabilizable system (A, B). The target is feasible for any r_{sp} if and only if*

$$\operatorname{rank}\begin{bmatrix} I-A & -B \\ HC & 0 \end{bmatrix} = n + n_c$$

Exercise 1.58: Existence and uniqueness of the unconstrained target

Assume a system having p controlled variables $z = Hx$, with setpoints r_{sp}, and m manipulated variables u, with setpoints u_{sp}. Consider the steady-state target problem

$$\min_{x,u}(1/2)(u - u_{\text{sp}})'R(u - u_{\text{sp}}) \qquad R > 0$$

subject to

$$\begin{bmatrix} I-A & -B \\ H & 0 \end{bmatrix}\begin{bmatrix} x \\ u \end{bmatrix} = \begin{bmatrix} 0 \\ r_{\text{sp}} \end{bmatrix}$$

Show that the steady-state solution (x, u) exists for any $(r_{\text{sp}}, u_{\text{sp}})$ and is unique if

$$\operatorname{rank}\begin{bmatrix} I-A & -B \\ H & 0 \end{bmatrix} = n + p \qquad \operatorname{rank}\begin{bmatrix} I-A \\ H \end{bmatrix} = n$$

Exercise 1.59: Choose a sample time

Consider the unstable continuous time system

$$\frac{dx}{dt} = Ax + Bu \qquad y = Cx$$

in which

$$A = \begin{bmatrix} -0.281 & 0.935 & 0.035 & 0.008 \\ 0.047 & -0.116 & 0.053 & 0.383 \\ 0.679 & 0.519 & 0.030 & 0.067 \\ 0.679 & 0.831 & 0.671 & -0.083 \end{bmatrix} \qquad B = \begin{bmatrix} 0.687 \\ 0.589 \\ 0.930 \\ 0.846 \end{bmatrix} \qquad C = I$$

Consider regulator tuning parameters and constraints

$$Q = \operatorname{diag}(1,2,1,2) \qquad R = 1 \qquad N = 10 \qquad |x| \leq \begin{bmatrix} 1 \\ 2 \\ 1 \\ 3 \end{bmatrix}$$

(a) Compute the eigenvalues of A. Choose a sample time of $\Delta = 0.04$ and simulate the MPC regulator response given $x(0) = \begin{bmatrix} -0.9 & -1.8 & 0.7 & 2 \end{bmatrix}'$ until $t = 20$. Use an ODE solver to simulate the continuous time plant response. Plot all states and the input versus time.

Now add an input disturbance to the regulator so the control applied to the plant is u_d instead of u in which

$$u_d(k) = (1 + 0.1w_1)u(k) + 0.1w_2$$

and w_1 and w_2 are zero-mean, normally distributed random variables with unit variance. Simulate the regulator's performance given this disturbance. Plot all states and $u_d(k)$ versus time.

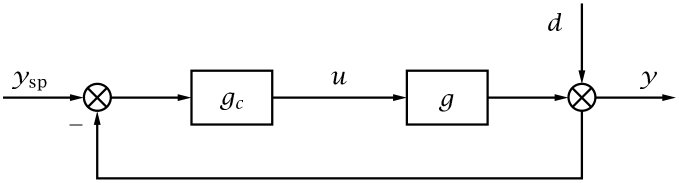

Figure 1.14: Feedback control system with output disturbance d, and setpoint y_{sp}.

(b) Repeat the simulations with and without disturbance for $\Delta = 0.4$ and $\Delta = 2$.

(c) Compare the simulations for the different sample times. What happens if the sample time is too large? Choose an appropriate sample time for this system and justify your choice.

Exercise 1.60: Disturbance models and offset

Consider the following two-input, three-output plant discussed in Example 1.11

$$\begin{aligned} x^+ &= Ax + Bu + B_p p \\ y &= Cx \end{aligned}$$

in which

$$A = \begin{bmatrix} 0.2681 & -0.00338 & -0.00728 \\ 9.703 & 0.3279 & -25.44 \\ 0 & 0 & 1 \end{bmatrix} \qquad C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$B = \begin{bmatrix} -0.00537 & 0.1655 \\ 1.297 & 97.91 \\ 0 & -6.637 \end{bmatrix} \qquad B_p = \begin{bmatrix} -0.1175 \\ 69.74 \\ 6.637 \end{bmatrix}$$

The input disturbance p results from a reactor inlet flowrate disturbance.

(a) Since there are two inputs, choose two outputs in which to remove steady-state offset. Build an output disturbance model with two integrators. Is your augmented model detectable?

(b) Implement your controller using $p = 0.01$ as a step disturbance at $k = 0$. Do you remove offset in your chosen outputs? Do you remove offset in any outputs?

(c) Can you find any two-integrator disturbance model that removes offset in two outputs? If so, which disturbance model do you use? If not, why not?

Exercise 1.61: MPC, PID, and time delay

Consider the following first-order system with time delay shown in Figure 1.14

$$g(s) = \frac{k}{\tau s + 1} e^{-\theta s}, \qquad k = 1, \tau = 1, \theta = 5$$

Consider a unit step change in setpoint y_{sp}, at $t = 0$.

(a) Choose a reasonable sample time, Δ, and disturbance model, and simulate an offset-free discrete time MPC controller for this setpoint change. List all of your chosen parameters.

(b) Choose PID tuning parameters to achieve "good performance" for this system. List your PID tuning parameters. Compare the performances of the two controllers.

Exercise 1.62: CSTR heat-transfer coefficient

Your mission is to design the controller for the nonlinear CSTR model given in Example 1.11. We wish to use a linear controller and estimator with three integrating disturbances to remove offset in two controlled variables: *temperature* and *level*; use the nonlinear CSTR model as the plant.

(a) You are particularly concerned about disturbances to the heat-transfer rate (parameter U) for this reactor. If changes to U are the primary disturbance, what disturbance model do you recommend and what covariances do you recommend for the three disturbances so that the disturbance state accounting for heat transfer is used primarily to explain the output error in the state estimator? First do a simulation with no measurement noise to test your estimator design. In the simulation let the reactor's heat-transfer coefficient decrease (and increase) by 20% at 10 minutes to test your control system design. Comment on the performance of the control system.

(b) Now let's add some measurement noise to all three sensors. So we all work on the same problem, choose the variance of the measurement error R_v to be

$$R_v = 10^{-3}\,\mathrm{diag}\,(c_s^2, T_s^2, h_s^2)$$

in which (c_s, T_s, h_s) are the nominal steady states of the three measurements. Is the performance from the previous part assuming no measurement noise acceptable? How do you adjust your estimator from the previous part to obtain good performance? Rerun the simulation with measurement noise and your adjusted state estimator. Comment on the change in the performance of your new design that accounts for the measurement noise.

(c) Recall that the offset lemma 1.10 is an either-or proposition, i.e., *either* the controller removes steady offset in the controlled variables *or* the system is closed-loop unstable. From closed-loop simulation, approximate the range of plant U values for which the controller is stabilizing (with zero measurement noise). From a stabilization perspective, which disturbance is worse, an increase or decrease in the plant's heat-transfer coefficient?

Exercise 1.63: System identification of the nonlinear CSTR

In many practical applications, it may not be convenient to express system dynamics from first principles. Hence, identifying a suitable model from data is a critical step in the design of an MPC controller. Your final mission is to obtain a 2-input, 3-output process model for the nonlinear CSTR given in Example 1.11 using the System Identification Toolbox in MATLAB. Relevant functions are provided.

(a) Begin first by creating a dataset for identification. Generate a pseudo-random, binary signal (PRBS) for the inputs using `idinput`. Ensure you have generated uncorrelated signals for each input. Think about the amplitude of the PRBS to use when collecting data from a nonlinear process keeping in mind that large perturbations may lead to undesirable phenomena such as reactor ignition. Inject these generated input sequences into the nonlinear plant of Example 1.11 and simulate the system by solving the nonlinear ODEs. Add measurement noise to the simulation so that you have a realistic dataset for the ID and plot the input-output data.

(b) Use the data to identify a third-order linear state-space model by calling `iddata` and `ssest`. Compare the step tests of your identified model with those from the linear model used in Example 1.11. Which is more accurate compared to the true plant simulation?

(c) Using the code for Example 1.11 as a starting point, replace the linear model in the MPC controller with your identified model and recalculate Figures 1.10 and 1.11 from the example. Is your control system robust enough to obtain good closed-loop control of the nonlinear plant using your linear model identified from data in the MPC controller? Do you maintain zero offset in the controlled variables?

Bibliography

R. E. Bellman. *Dynamic Programming.* Princeton University Press, Princeton, New Jersey, 1957.

R. E. Bellman and S. E. Dreyfus. *Applied Dynamic Programming.* Princeton University Press, Princeton, New Jersey, 1962.

D. P. Bertsekas. *Dynamic Programming.* Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1987.

E. F. Camacho and C. Bordons. *Model Predictive Control.* Springer-Verlag, London, second edition, 2004.

E. J. Davison and H. W. Smith. Pole assignment in linear time-invariant multivariable systems with constant disturbances. *Automatica*, 7:489–498, 1971.

E. J. Davison and H. W. Smith. A note on the design of industrial regulators: Integral feedback and feedforward controllers. *Automatica*, 10:329–332, 1974.

R. Fletcher. *Practical Methods of Optimization.* John Wiley & Sons, New York, 1987.

B. A. Francis and W. M. Wonham. The internal model principle of control theory. *Automatica*, 12:457–465, 1976.

G. C. Goodwin and K. S. Sin. *Adaptive Filtering Prediction and Control.* Prentice-Hall, Englewood Cliffs, New Jersey, 1984.

G. C. Goodwin, M. M. Serón, and J. A. De Doná. *Constrained control and estimation: an optimization approach.* Springer, New York, 2005.

M. L. J. Hautus. Controllability and stabilizability of sampled systems. *IEEE Trans. Auto. Cont.*, 17(4):528–531, August 1972.

R. A. Horn and C. R. Johnson. *Matrix Analysis.* Cambridge University Press, 1985.

A. H. Jazwinski. *Stochastic Processes and Filtering Theory.* Academic Press, New York, 1970.

R. E. Kalman. A new approach to linear filtering and prediction problems. *Trans. ASME, J. Basic Engineering*, pages 35–45, March 1960a.

R. E. Kalman. Contributions to the theory of optimal control. *Bull. Soc. Math. Mex.*, 5:102–119, 1960b.

H. Kwakernaak and R. Sivan. *Linear Optimal Control Systems*. John Wiley and Sons, New York, 1972.

W. H. Kwon. *Receding horizon control: model predictive control for state models*. Springer-Verlag, London, 2005.

J. M. Maciejowski. *Predictive Control with Contraints*. Prentice-Hall, Harlow, UK, 2002.

J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer, New York, second edition, 2006.

B. J. Odelson, M. R. Rajamani, and J. B. Rawlings. A new autocovariance least-squares method for estimating noise covariances. *Automatica*, 42(2):303–308, February 2006.

G. Pannocchia and J. B. Rawlings. Disturbance models for offset-free MPC control. *AIChE J.*, 49(2):426–437, 2003.

L. Qiu and E. J. Davison. Performance limitations of non-minimum phase systems in the servomechanism problem. *Automatica*, 29(2):337–349, 1993.

C. V. Rao and J. B. Rawlings. Steady states and constraints in model predictive control. *AIChE J.*, 45(6):1266–1278, 1999.

J. A. Rossiter. *Model-based predictive control: a practical approach*. CRC Press LLC, Boca Raton, FL, 2004.

S. M. Selby. *CRC Standard Mathematical Tables*. CRC Press, twenty-first edition, 1973.

E. D. Sontag. *Mathematical Control Theory*. Springer-Verlag, New York, second edition, 1998.

G. Strang. *Linear Algebra and its Applications*. Academic Press, New York, second edition, 1980.

L. Wang. *Model Predictive Control System Design and Implementation Using Matlab*. Springer, New York, 2009.