Chapter 3
Methods of linear control theory

"Guaranteed Margins for LQG Regulators.
Abstract – There are none."

- John Doyle, *IEEE Transactions on Automatic Control, 1978 [86]*

The most well-developed theory of control generally applies to a linear system or to the linearization of a nonlinear system about a fixed point or a periodic orbit. Linear control theory has many applications in fluid dynamics, such as the stabilization of unstable laminar boundary layers. Although the governing equations may be nonlinear, successful stabilizing controllers will regulate the system to a neighborhood where the linearization is increasingly valid.

In this chapter we introduce linear systems (Sec. 3.1) and explore $\mathscr{H}_2$ optimal control problems, including the linear quadratic regulator (LQR) in Sec. 3.2 and Kalman filters in Sec. 3.3. These problems are chosen because of their simplicity, ubiquitous application, well-defined quadratic cost-functions, and the existence of known optimal solutions. Next, linear quadratic Gaussian (LQG) control is introduced for sensor-based feedback in Sec. 3.4. Finally, methods of system linear system identification are provided in Sec. 3.5.

This chapter is not meant to be an exhaustive primer on linear control theory, although key concepts from optimal control are introduced as needed to build intuition. Note that none of the linear system theory below is required to implement the machine learning control strategies in the remainder of the book, but they are instead included to provide context and demonstrate known optimal solutions to linear control problems. In many situations, $\mathscr{H}_\infty$ robust control may be more desirable to balance the trade-off between robustness and performance in systems with uncertainty and unmodeled dynamics, and the MLC methods developed here may be generalized to other cost functions. For a more complete discussion of linear control theory, excellent books include [93, 251].

3.1 Linear systems

Many systems of interest are either linear, or correspond to the linearization of a nonlinear system, such as Eq. (1.1), about a fixed point or periodic orbit. The most complete theory of control applies to linear systems. Consider the following state-space system:

$$\frac{d}{dt}\mathbf{a} = \mathbf{A}\mathbf{a} + \mathbf{B}\mathbf{b} \tag{3.1a}$$

$$\mathbf{s} = \mathbf{C}\mathbf{a} + \mathbf{D}\mathbf{b}. \tag{3.1b}$$

The matrices $\mathbf{A}, \mathbf{B}, \mathbf{C}$, and $\mathbf{D}$ arise from the linearization of Eq. (1.1) about an equilibrium state $\mathbf{a}_0$; in Eq. (3.1), the state $\mathbf{a}$ is the deviation from the equilibrium $\mathbf{a}_0$. In the absence of an actuation input $\mathbf{b}$, the solution to Eq. (3.1a) is:

$$\mathbf{a}(t) = e^{\mathbf{A}t}\mathbf{a}(t_0), \tag{3.2}$$

where the matrix exponential $e^{\mathbf{A}t}$ is given by the infinite series:

$$e^{\mathbf{A}t} = \mathbf{I} + \mathbf{A}t + \frac{1}{2!}\mathbf{A}^2t^2 + \frac{1}{3!}\mathbf{A}^3t^3 + \cdots. \tag{3.3}$$

The stability of this system is determined by the eigenvalues of $\mathbf{A}$, and eigenvalues with positive real part are unstable. The corresponding eigenvectors represent unstable state directions, where perturbations will either grow without bound, or grow until unmodeled nonlinear dynamics become important.

In the case of an actuation input $\mathbf{b}$, the solution to Eq. (3.1a) becomes:

$$\mathbf{a}(t) = e^{\mathbf{A}t}\mathbf{a}(t_0) + \int_{t_0}^{t} e^{\mathbf{A}(t-\tau)}\mathbf{B}\mathbf{b}(\tau)\,d\tau. \tag{3.4}$$

The system in Eq. (3.1a) is *controllable* if it is possible to navigate the system to an arbitrary state $\mathbf{a}$ from the origin in finite time with a finite actuation signal $\mathbf{b}(t)$. Mathematically, this relies on the controllability matrix

$$\mathscr{C} = \begin{bmatrix}\mathbf{B} & \mathbf{A}\mathbf{B} & \mathbf{A}^2\mathbf{B} & \cdots & \mathbf{A}^{N_a-1}\mathbf{B}\end{bmatrix} \tag{3.5}$$

having full column rank. In practice, the *degree of controllability*, characterized by the singular value decomposition of the controllability matrix in Eq. (3.5), or equivalently, by the eigen-decomposition of the controllability Gramian, is often more useful. Note that if Eq. (3.1a) is the linearization of a nonlinear system about a fixed point, then it may be controllable with a nonlinear controller $\mathbf{b} = \mathbf{K}(\mathbf{a})$, even if Eq. (3.1a) is linearly uncontrollable. As long as all unstable state directions are in the span of $\mathscr{C}$, then the system is *stabilizable*; these unstable directions correspond to eigenvectors of $\mathbf{A}$ with eigenvalues having positive real part.

Similarly, the system in Eq. (3.1b) is *observable* if any state $\mathbf{a}$ may be estimated from a time-history of sensor measurements $\mathbf{s}$. Mathematically, this corresponds to

the observability matrix

$$\mathscr{O} = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \mathbf{CA}^2 \\ \vdots \\ \mathbf{CA}^{N_a-1} \end{bmatrix} \tag{3.6}$$

having full row rank. A system is *detectable* if all unstable states are *observable*, so that they are in the row-space of $\mathscr{O}$.

As in Eq. (1.1), the matrices in the linearization in Eq. (3.1) may depend on the specific bifurcation parameters μ. The linear theory above may also be generalized for linear parameter varying (LPV) systems, where the matrices in Eq. (3.1) depend on a time-varying parameter [246, 16]. For example, when linearizing about a periodic orbit, the matrices are parameterized by the phase ϕ of the trajectory on the orbit. In this case, gain-scheduling allows different controllers to be applied depending on the parameter values [246, 232].

3.2 Full-state feedback

If measurements of the full state $\mathbf{a}$ are available, then $\mathbf{D} = \mathbf{0}$ and $\mathbf{C} = \mathbf{I}$, where $\mathbf{I}$ is the $N_a \times N_a$ identity matrix. We may then consider full-state feedback control $\mathbf{b} = \mathbf{K}(\mathbf{a})$ based on measurements of the state, $\mathbf{s} = \mathbf{a}$. Although full-state feedback may be unrealistic, especially for high-dimensional systems, it is often possible to estimate the full state from limited sensor measurements, using a Kalman filter, as discussed in Sec. 3.3. Remarkably, it is possible to design an optimal full-state feedback controller and an optimal state-estimator separately, and the combined sensor-based feedback controller will also be optimal, as we will show in Sec. 3.4.

Linear quadratic regulator (LQR)

If the system in Eq. (3.1a) is controllable, then it is possible to design a proportional controller

$$\mathbf{b} = -\mathbf{K}_r \mathbf{a} \tag{3.7}$$

to arbitrarily place the eigenvalues of the closed-loop system

$$\frac{d}{dt}\mathbf{a} = \mathbf{A}\mathbf{a} + \mathbf{B}\mathbf{b} = (\mathbf{A} - \mathbf{B}\mathbf{K}_r)\,\mathbf{a}. \tag{3.8}$$

A natural goal in control theory is to stabilize the system so that the state $\mathbf{a}$ converges quickly to $\mathbf{0}$, but without expending too much control effort. We may construct a quadratic cost function J that balances the aggressive regulation of $\mathbf{a}$ with the cost of control:

$$J(t) = \int_0^t \left[\mathbf{a}^T(\tau)\mathbf{Q}\,\mathbf{a}(\tau) + \mathbf{b}^T(\tau)\mathbf{R}\mathbf{b}(\tau)\right] d\tau. \tag{3.9}$$

The goal is to develop a control strategy $\mathbf{b} = -\mathbf{K}_r\mathbf{a}$ to minimize $J = \lim_{t\to\infty} J(t)$. The matrices $\mathbf{Q}$ and $\mathbf{R}$ weight the cost of deviations of the state from zero and the cost of actuation, respectively. These matrices are often diagonal, and the diagonal elements may be tuned to change the relative importance of the control objectives. $\mathbf{R}$ is positive definite and $\mathbf{Q}$ is positive semi-definite. For example, if we increase the entries of $\mathbf{Q}$ by a factor of 10 and keep $\mathbf{R}$ the same, then accurate state regulation is more heavily weighted, and more aggressive control may be permitted. Typically, the ratios of elements in $\mathbf{Q}$ and $\mathbf{R}$ are increased or decreased by powers of 10.

Because of the well-defined quadratic cost function in Eq. (3.9), the optimal controller $\mathbf{K}_r$ may be solved for analytically. In particular, the controller $\mathbf{K}$ that minimizes the cost in Eq. (3.9) is given by

$$\mathbf{K}_r = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{X}, \tag{3.10}$$

where $\mathbf{X}$ is the solution to the algebraic Riccati equation:

$$\mathbf{A}^T\mathbf{X} + \mathbf{X}\mathbf{A} - \mathbf{X}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{X} + \mathbf{Q} = \mathbf{0}. \tag{3.11}$$

The resulting full-state feedback controller is called a linear quadratic regulator (LQR), since it is a linear control law that minimizes a quadratic cost function to regulate the system. This is shown schematically in Fig. 3.1. Solving for the LQR controller $\mathbf{K}_r$ in Eq. (3.10) is computationally robust, and it is a built-in routine in many computational packages. However, the computational cost of solving the Riccati equation in Eq. (3.11) scales with the cube of the state dimension, making it prohibitively expensive for large systems, except as an off-line calculation.

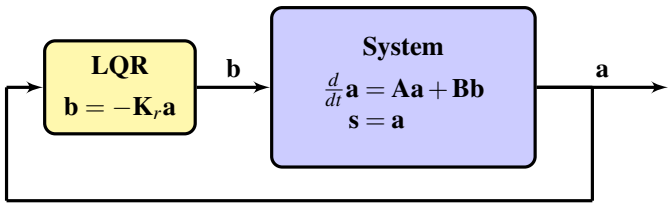

Fig. 3.1 LQR stabilization problem. The optimal control for a linear system with full-state feedback $\mathbf{s} = \mathbf{a}$ is given by proportional control $\mathbf{b} = -\mathbf{K}_r\mathbf{a}$ where $\mathbf{K}_r$ is a gain matrix obtained by solving an algebraic Riccati equation.

3.3 Sensor-based state estimation

The optimal LQR controller above relies on access to the full-state of the system. However, in many applications full-state measurements of a high-dimensional system are either technologically infeasible or prohibitively expensive to collect and process. When full measurements are available, as with the use of particle image velocimetry (PIV) [277, 127] to measure fluid velocity fields in experiments, these measurements are typically only available in controlled experimental settings, and are not practical for in-field applications such as monitoring flow over a wing in flight. The computational burden of collecting, transferring and processing full-state measurements may also limit the temporal resolution of the measurements and introduce unacceptable computational time-delays which degrade robust performance.

In practice, it is often necessary to *estimate* the full state $\mathbf{a}$ from limited noisy sensor measurements $\mathbf{s}$. This estimation process balances information from a model prediction of the state with the sensor measurements. Under a set of well-defined conditions it is possible to obtain a stable estimator that converges to an estimate of the full state $\mathbf{a}$, which can then be used in conjunction with the optimal full-state feedback LQR control law described above.

Kalman filtering

The Kalman filter [156] is perhaps the most often applied algorithm to estimate the full-state of a system from noisy sensor measurements and an uncertain model of the system. Kalman filters have been used in myriad applications, including guidance and tracking of vehicles, airplane autopilots, modeling climate and weather, seismology, and satellite navigation, to name only a few. An excellent and complete derivation of the Kalman filter may be found in [254].

In the dynamic state-estimation framework, the linear dynamics from Eq. (3.1) are generalized to include stochastic disturbances $\mathbf{w}_d$, also known as *process noise*, and sensor noise $\mathbf{w}_n$:

$$\frac{d}{dt}\mathbf{a} = \mathbf{A}\mathbf{a} + \mathbf{B}\mathbf{b} + \mathbf{w}_d \tag{3.12a}$$

$$\mathbf{s} = \mathbf{C}\mathbf{a} + \mathbf{D}\mathbf{b} + \mathbf{w}_n. \tag{3.12b}$$

Both the disturbance and noise terms are assumed to be zero-mean Gaussian white-noise processes, although generalizations exist to handle correlated and biased noise terms. We assume that the disturbance and noise covariances are known:

$$\mathbb{E}\left(\mathbf{w}_d(t)\mathbf{w}_d(\tau)^T\right) = \mathbf{V}_d\delta(t-\tau) \tag{3.13a}$$

$$\mathbb{E}\left(\mathbf{w}_n(t)\mathbf{w}_n(\tau)^T\right) = \mathbf{V}_n\delta(t-\tau) \tag{3.13b}$$

where $\mathbb{E}$ is the expectation operator, and $\delta(\cdot)$ is the Dirac delta function. The matrices $\mathbf{V}_d$ and $\mathbf{V}_n$ are diagonal matrices whose entries contain the variances of the corresponding disturbance or noise term.

A full-state estimator is a dynamical system that produces an estimate $\hat{\mathbf{a}}$ for the full-state $\mathbf{a}$ using only knowledge of the noisy sensor measurements $\mathbf{s}$, the actuation input $\mathbf{b}$, and a model of the process dynamics. If the system is observable, it is possible to construct an estimator with a filter gain $\mathbf{K}_f$ as follows:

$$\frac{d}{dt}\hat{\mathbf{a}} = \mathbf{A}\hat{\mathbf{a}} + \mathbf{B}\mathbf{b} + \mathbf{K}_f\left(\mathbf{s} - \hat{\mathbf{s}}\right) \tag{3.14a}$$

$$\hat{\mathbf{s}} = \mathbf{C}\hat{\mathbf{a}} + \mathbf{D}\mathbf{b}. \tag{3.14b}$$

The output $\hat{\mathbf{s}}$ is a prediction of the expected sensor output based on the full-state estimate $\hat{\mathbf{a}}$. Substituting the expression for $\hat{\mathbf{s}}$ from Eq. (3.14b) into Eq. (3.14a) yields a dynamical system for $\hat{\mathbf{a}}$ with $\mathbf{b}$ and $\mathbf{s}$ as inputs:

$$\frac{d}{dt}\hat{\mathbf{a}} = \left(\mathbf{A} - \mathbf{K}_f\mathbf{C}\right)\hat{\mathbf{a}} + \mathbf{K}_f\mathbf{s} + \left(\mathbf{B} - \mathbf{K}_f\mathbf{D}\right)\mathbf{b} \tag{3.15a}$$

$$= \left(\mathbf{A} - \mathbf{K}_f\mathbf{C}\right)\hat{\mathbf{a}} + \left[\mathbf{K}_f,\ \left(\mathbf{B} - \mathbf{K}_f\mathbf{D}\right)\right]\begin{bmatrix}\mathbf{s}\\ \mathbf{b}\end{bmatrix}. \tag{3.15b}$$

This is shown schematically in Fig. 3.2 for $\mathbf{D} = \mathbf{0}$.

For observable systems in Eq. (3.1), it is possible to arbitrarily place the eigenvalues of the estimator dynamics $\mathbf{A} - \mathbf{K}_f\mathbf{C}$, resulting in stable convergence of the estimate $\hat{\mathbf{a}}$ to the true state $\mathbf{a}$. To see that stable dynamics $\mathbf{A} - \mathbf{K}_f\mathbf{C}$ results in a stable estimator that converges to the full-state $\mathbf{a}$, consider the time dynamics of the estimation error $\varepsilon = \mathbf{a} - \hat{\mathbf{a}}$:

$$\begin{aligned}
\frac{d}{dt}\varepsilon &= \frac{d}{dt}\mathbf{a} - \frac{d}{dt}\hat{\mathbf{a}} \\
&= \left[\mathbf{A}\mathbf{a} + \mathbf{B}\mathbf{b} + \mathbf{w}_d\right] - \left[\left(\mathbf{A} - \mathbf{K}_f\mathbf{C}\right)\hat{\mathbf{a}} + \mathbf{K}_f\mathbf{s} + \left(\mathbf{B} - \mathbf{K}_f\mathbf{D}\right)\mathbf{b}\right] \\
&= \mathbf{A}\varepsilon + \mathbf{w}_d + \mathbf{K}_f\mathbf{C}\hat{\mathbf{a}} - \mathbf{K}_f\mathbf{s} + \mathbf{K}_f\mathbf{D}\mathbf{b} \\
&= \mathbf{A}\varepsilon + \mathbf{w}_d + \mathbf{K}_f\mathbf{C}\hat{\mathbf{a}} - \mathbf{K}_f\underbrace{\left[\mathbf{C}\mathbf{a} + \mathbf{D}\mathbf{b} + \mathbf{w}_n\right]}_{\mathbf{s}} + \mathbf{K}_f\mathbf{D}\mathbf{b} \\
&= \left(\mathbf{A} - \mathbf{K}_f\mathbf{C}\right)\varepsilon + \mathbf{w}_d - \mathbf{K}_f\mathbf{w}_n.
\end{aligned}$$

Therefore, the estimate $\hat{\mathbf{a}}$ will converge to the true state $\mathbf{a}$ as long as $\mathbf{A} - \mathbf{K}_f\mathbf{C}$ is stable. Analogous to the case of LQR, there is a balance between over-stabilization and the amplification of noise. An analogy is often made with an inexperienced automobile driver who holds the wheel too tightly and reacts to every bump and disturbance on the road.

The Kalman filter is an *optimal* full-state estimator that minimizes the following cost function:

$$J = \lim_{t\to\infty}\mathbb{E}\left(\left(\mathbf{a}(t) - \hat{\mathbf{a}}(t)\right)^T\left(\mathbf{a}(t) - \hat{\mathbf{a}}(t)\right)\right). \tag{3.16}$$

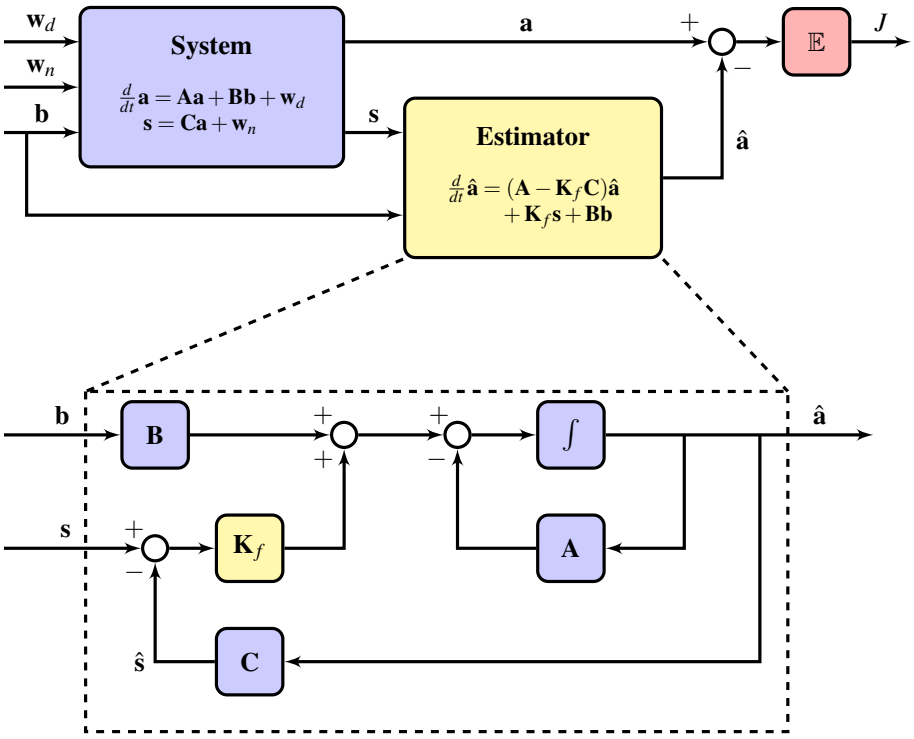

Fig. 3.2 Schematic of the Kalman filter for state estimation from noisy measurements $\mathbf{s} = \mathbf{C}\mathbf{a} + \mathbf{w}_n$ with process noise (disturbance) $\mathbf{w}_d$. Note that there is no feedthrough term $\mathbf{D}$ in this diagram.

Implicit in this cost function are the noise and disturbance covariances, which determine the optimal balance between aggressive estimation and noise attenuation. The mathematical derivation of an optimal solution is nearly identical to that of LQR, and this problem is often called linear quadratic estimation (LQE) because of the dual formulation. The optimal Kalman filter gain $\mathbf{K}_f$ is given by

$$\mathbf{K}_f = \mathbf{Y}\mathbf{C}^T\mathbf{V}_n \tag{3.17}$$

where $\mathbf{Y}$ is the solution to another algebraic Riccati equation:

$$\mathbf{Y}\mathbf{A}^T + \mathbf{A}\mathbf{Y} - \mathbf{Y}\mathbf{C}^T\mathbf{V}_n^{-1}\mathbf{C}\mathbf{Y} + \mathbf{V}_d = \mathbf{0}. \tag{3.18}$$

3.4 Sensor-based feedback

In practice, the full-state estimate from sensor-based estimation is used in conjunction with a full-state feedback control law, resulting in sensor-based feedback. The separation principle in control theory states that for linear systems it is possible to design optimal full-state feedback and full-state estimator gain matrices separately, and the resulting sensor-based feedback will remain optimal when combined. There are numerous techniques to develop sensor-based control that optimize different quantities. For instance, combining the LQR and Kalman filter solutions results in what are known as $\mathscr{H}_2$ optimal control laws, while other controllers, known as $\mathscr{H}_\infty$ controllers, may be designed to provide robustness.

In the case of model-free machine learning control, the controller dynamical system, which estimates relevant states from sensors and feeds this state estimate back into an actuation signal, must be designed and optimized as a single unit. Realistically, we may not have access to full-state data to train an estimator, even during an expensive off-line optimization. Moreover, the system under investigation may be unstable, so that a sensor-based controller must first be applied before training an estimator is even feasible. However, it will be possible to design sensor-based feedback controllers in one shot using MLC using generalized transfer function blocks, as will be explored in the following chapter.

Linear quadratic Gaussian (LQG)

The linear quadratic Gaussian (LQG) controller is the optimal sensor-based feedback control law that minimizes the cost function in Eq. (3.9) using sensors $\mathbf{s}$ from the linear model in Eq. (3.12) with sensor and process noise. Remarkably, the optimal LQG solution is obtained by combining the optimal LQR feedback gain $\mathbf{K}_r$ with the estimated state $\hat{\mathbf{a}}$ obtained by the optimal Kalman filter $\mathbf{K}_f$, as shown in Fig. 3.3. Thus, it is possible to design $\mathbf{K}_r$ and $\mathbf{K}_f$ separately by solving the respective Riccati equations in Eqs. (3.11) and (3.18) and then combine to form an optimal LQG controller; this is known as the *separation principle*.

Combining the full-state LQR control in Eq. (3.8) with the full-state Kalman filter estimate in Eq. (3.15), we obtain a dynamical system for the LQG controller where $\mathbf{s}$ is the input to the controller, $\mathbf{b}$ is the output of the controller, and internal controller state is the full-state estimate $\hat{\mathbf{a}}$:

$$\frac{d}{dt}\hat{\mathbf{a}} = \left(\mathbf{A} - \mathbf{K}_f\mathbf{C} - \mathbf{B}\mathbf{K}_r\right)\hat{\mathbf{a}} + \mathbf{K}_f\mathbf{s} \tag{3.19a}$$

$$\mathbf{b} = -\mathbf{K}_r\hat{\mathbf{a}}. \tag{3.19b}$$

The LQG cost function is the ensemble-averaged value of Eq. (3.9),

$$J(t) = \left\langle \int_0^t \left[\mathbf{a}^T(\tau)\,\mathbf{Q}\,\mathbf{a}(\tau) + \mathbf{b}^T(\tau)\mathbf{R}\mathbf{b}(\tau)\right] d\tau \right\rangle, \tag{3.20}$$

where the angular brackets denote the average over many noise realizations.

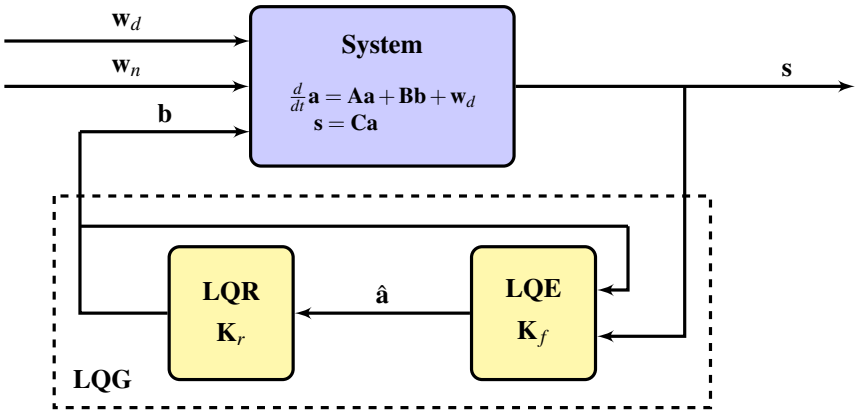

Fig. 3.3 Schematic diagram for linear quadratic Gaussian (LQG) controller. The optimal LQR and LQE gain matrices $\mathbf{K}_r$ and $\mathbf{K}_f$ are designed separately based on the solutions of two different algebraic Riccati equations. When combined, the resulting sensor-based feedback is optimal.

Applying LQR to the full-state estimate $\hat{\mathbf{a}}$ results in the following state dynamics:

$$\begin{aligned}\frac{d}{dt}\mathbf{a} &= \mathbf{A}\mathbf{a} - \mathbf{B}\mathbf{K}_r\hat{\mathbf{a}} + \mathbf{w}_d && \text{(3.21a)}\\ &= \mathbf{A}\mathbf{a} - \mathbf{B}\mathbf{K}_r\mathbf{a} + \mathbf{B}\mathbf{K}_r(\mathbf{a} - \hat{\mathbf{a}}) + \mathbf{w}_d && \text{(3.21b)}\\ &= \mathbf{A}\mathbf{a} - \mathbf{B}\mathbf{K}_r\mathbf{a} + \mathbf{B}\mathbf{K}_r\boldsymbol{\varepsilon} + \mathbf{w}_d, && \text{(3.21c)}\end{aligned}$$

where $\boldsymbol{\varepsilon} = \mathbf{a} - \hat{\mathbf{a}}$ as before. We may finally write the closed-loop system as

$$\frac{d}{dt}\begin{bmatrix}\mathbf{a}\\ \boldsymbol{\varepsilon}\end{bmatrix} = \begin{bmatrix}\mathbf{A} - \mathbf{B}\mathbf{K}_r & \mathbf{B}\mathbf{K}_r\\ \mathbf{0} & \mathbf{A} - \mathbf{K}_f\mathbf{C}\end{bmatrix}\begin{bmatrix}\mathbf{a}\\ \boldsymbol{\varepsilon}\end{bmatrix} + \begin{bmatrix}\mathbf{I} & \mathbf{0}\\ \mathbf{I} & -\mathbf{K}_f\end{bmatrix}\begin{bmatrix}\mathbf{w}_d\\ \mathbf{w}_n\end{bmatrix}. \tag{3.22}$$

It is clear that if $\mathbf{K}_r$ and $\mathbf{K}_f$ were chosen to optimally place the closed-loop eigenvalues of $\mathbf{A} - \mathbf{B}\mathbf{K}_r$ and $\mathbf{A} - \mathbf{K}_f\mathbf{C}$ in the respective LQR and Kalman filter problems, then these are still the eigenvalues of the sensor-based closed-loop LQG controller.

The LQG framework assumes an accurate system model and knowledge of the measurement and process noise magnitude; moreover, the *Gaussian* in the title refers to the assumption that these noise terms are Gaussian white noise processes. In practice, all of these assumptions are dubious for many real-world systems, and even small amounts of model uncertainty can destroy the LQG performance and cause instability [86]. The entire optimization process above is often referred to as $\mathscr{H}_2$ optimal control. The optimization problem may be modified to promote robust controllers for systems that have model uncertainty [89, 119, 88], and these controllers are often referred to as $\mathscr{H}_\infty$ robust control laws. Intuitively, robust control penalizes the *worst-case* performance of a system, so that robustness is promoted. Often, an LQG controller may be *robustified* through a process called loop transfer recovery, although this is beyond the scope of this book. An excellent treatment of robust control may be found in [93].

3.5 System Identification and Model Reduction

In many high-dimensional fluid problems, it is still possible to use linear control techniques, despite nonlinear equations of motion. For example, in fluid dynamics there are numerous success stories of linear model-based flow control, including transition delay in a spatially developing boundary layer on a flat plate and in channel flow [29, 136, 135, 63, 142, 13, 11, 12, 244, 245, 101], reducing skin-friction drag in wall turbulence [71, 72, 102, 159, 160], and stabilization of the cavity flow [56, 227, 55, 233, 229, 231, 234, 57, 144]. However, many of the linear control approaches do not scale well to large state spaces, and they may be prohibitively expensive to enact for real-time control on short timescales. It is therefore often necessary to first develop low-dimensional approximations of the full-state system for use with feedback control. There are two broad approaches to this problem: First, it is possible to start with a high-dimensional dynamical system, such as the discretized Navier-Stokes equations, and project the dynamics onto a low-dimensional subspace identified, for example, using proper orthogonal decomposition (POD) [28, 138] and Galerkin projection. This results in a reduced-order model (ROM) [220, 24]. There are many approaches to this problem, including discrete empirical interpolation methods (DEIM) [60, 210], gappy POD [100], balanced proper orthogonal decomposition (BPOD) [276, 230], and many more. The second approach is to collect data from a simulation or an experiment and try to identify a low-rank model using data-driven techniques. This approach is typically called system identification, and is often preferred for control design because of the relative ease of implementation. Examples include the dynamic mode decomposition (DMD) [237, 269] and related Koopman analysis [187, 228, 188], the eigensystem realization algorithm (ERA) [151, 181], and the observer–Kalman filter identification (OKID) [152, 213, 150].

After a linear model has been identified, either by model reduction or system identification, it may then be used for model-based control design, as described in Chapter 4. However, there are a number of issues that may arise in practice, as linear model-based control might not work for a large class of problems. First, the system being modeled may be strongly nonlinear, in which case the linear approximation might only capture a small portion of the dynamic effects. Next, the system may be stochastically driven, so that the linear model will average out the relevant fluctuations. Finally, when control is applied to the full system, the attractor dynamics may change, rendering the linearized model invalid. Exceptions include the stabilization of laminar solutions in fluid mechanics, where feedback control rejects nonlinear disturbances and keeps the system close to the fixed point where linearization is useful.

There are certainly alternative methods for system identification and model reduction that are nonlinear, involve stochasticity, and change with the attractor. However, these methods are typically advanced and they also may limit the available machinery from control theory.

3.5.1 System identification

System identification may be thought of as a form of machine learning, where an input–output map of a system is learned from training data in a representation that generalizes to data that was not in the training set. There is a vast literature on methods for system identification [150, 174], which is beyond the scope of this treatment, although many of the leading methods are based on a form of dynamic regression that fits models based on data. For this section, we consider the eigensystem realization algorithm (ERA) and observer-Kalman filter identification (OKID) methods because of their connection to balanced model reduction [192, 230, 181, 269] and their recent successful application in closed-loop flow control [13, 11, 143]. The ERA/OKID procedure is also applicable to multiple-input, multiple-output (MIMO) systems.

3.5.2 Eigensystem realization algorithm (ERA)

The eigensystem realization algorithm produces low-dimensional linear input–output models from sensor measurements of an impulse response experiment, and it is based on the "minimal realization" theory of Ho and Kalman [133]. The modern theory was developed to identify structural models for various spacecraft [151], and it has been shown by Ma *et al.* [181] that ERA models are equivalent to BPOD models[1]. ERA is based entirely on impulse response measurements and does not require prior knowledge of a model.

Given a linear system, as in Eq. (3.1), it is possible to obtain a discrete-time version:

$$\mathbf{a}_{k+1} = \mathbf{A}_d\mathbf{a}_k + \mathbf{B}_d\mathbf{b}_k \tag{3.23a}$$

$$\mathbf{s}_k = \mathbf{C}_d\mathbf{a}_k + \mathbf{D}_d\mathbf{b}_k, \tag{3.23b}$$

where subscript k denotes the time and Δt the corresponding timestep, so that $\mathbf{a}_k = \mathbf{a}(t_k) = \mathbf{a}(k\Delta t)$. The matrices in the discrete-time system are denoted with a subscript d and are related to the original continuous-time system matrices as:

$$\mathbf{A}_d = \exp(\mathbf{A}\Delta t) \tag{3.24a}$$

$$\mathbf{B}_d = \int_0^{\Delta t} \exp(\mathbf{A}\tau)\mathbf{B}d\tau \tag{3.24b}$$

$$\mathbf{C}_d = \mathbf{C} \tag{3.24c}$$

$$\mathbf{D}_d = \mathbf{D}. \tag{3.24d}$$

Now, a discrete-time delta function input in the actuation $\mathbf{b}$:

[1] BPOD and ERA models both balance the empirical Gramians and approximate balanced truncation [192] for high-dimensional systems.

$$\mathbf{b}_k^\delta \triangleq \mathbf{b}^\delta(k\Delta t) = \begin{cases} \mathbf{I},\, k=0 \\ \mathbf{0},\, k=1,2,3,\cdots \end{cases} \tag{3.25}$$

gives rise to a discrete-time impulse response in the sensors $\mathbf{s}$:

$$\mathbf{s}_k^\delta \triangleq \mathbf{s}^\delta(k\Delta t) = \begin{cases} \mathbf{D}_d,\, k=0 \\ \mathbf{C}_d\mathbf{A}_d^{k-1}\mathbf{B}_d,\, k=1,2,3,\cdots. \end{cases} \tag{3.26}$$

In an experiment or simulation, typically N_b impulse responses are performed, one for each of the N_b separate input channels. The output responses are collected for each impulsive input, and at a given time-step k, the output vector in response to the j-th impulsive input will form the j-th column of $\mathbf{s}_k^\delta$. Thus, each of the $\mathbf{s}_k^\delta$ is a $N_s \times N_b$ matrix.

A Hankel matrix $\mathbf{H}$ is formed by stacking shifted time-series of impulse-response measurements into a matrix:

$$\mathbf{H} = \begin{bmatrix} \mathbf{s}_1^\delta & \mathbf{s}_2^\delta & \cdots & \mathbf{s}_{m_c}^\delta \\ \mathbf{s}_2^\delta & \mathbf{s}_3^\delta & \cdots & \mathbf{s}_{m_c+1}^\delta \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{s}_{m_o}^\delta & \mathbf{s}_{m_o+1}^\delta & \cdots & \mathbf{s}_{m_c+m_o-1}^\delta \end{bmatrix}. \tag{3.27}$$

This matrix is closely related to the empirical discrete-time observability and controllability Gramians, $\mathbf{W}_{\mathscr{O}}^d = \mathscr{O}_d{}^*\mathscr{O}_d$ and $\mathbf{W}_{\mathscr{C}}^d = \mathscr{C}_d\mathscr{C}_d{}^*$. Substituting the expression from Eq. (3.26) into Eq. (3.27) yields:

$$\mathbf{H} = \begin{bmatrix} \mathbf{C}_d\mathbf{B}_d & \mathbf{C}_d\mathbf{A}_d\mathbf{B}_d & \cdots & \mathbf{C}_d\mathbf{A}_d^{m_c-1}\mathbf{B}_d \\ \mathbf{C}_d\mathbf{A}_d\mathbf{B}_d & \mathbf{C}_d\mathbf{A}_d^2\mathbf{B}_d & \cdots & \mathbf{C}_d\mathbf{A}_d^{m_c}\mathbf{B}_d \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{C}_d\mathbf{A}_d^{m_o-1}\mathbf{B}_d & \mathbf{C}_d\mathbf{A}_d^{m_o}\mathbf{B}_d & \cdots & \mathbf{C}_d\mathbf{A}_d^{m_c+m_o-2}\mathbf{B}_d \end{bmatrix} \tag{3.28a}$$

$$= \begin{bmatrix} \mathbf{C}_d \\ \mathbf{C}_d\mathbf{A}_d \\ \vdots \\ \mathbf{C}_d\mathbf{A}_d^{m_o-1} \end{bmatrix} \begin{bmatrix} \mathbf{B}_d & \mathbf{A}_d\mathbf{B}_d & \cdots & \mathbf{A}_d^{m_c-1}\mathbf{B}_d \end{bmatrix} \;=\; \mathscr{O}_d\mathscr{C}_d. \tag{3.28b}$$

Taking the singular value decomposition (SVD) of this Hankel matrix yields the dominant temporal patterns in this time-series:

$$\mathbf{H} = \mathbf{U}\Sigma\mathbf{V}^* = \begin{bmatrix} \mathbf{U}_r & \mathbf{U}_s \end{bmatrix} \begin{bmatrix} \Sigma_r & \mathbf{0} \\ \mathbf{0} & \Sigma_s \end{bmatrix} \begin{bmatrix} \mathbf{V}_r^* \\ \mathbf{V}_s^* \end{bmatrix} \approx \mathbf{U}_r\Sigma_r\mathbf{V}_r^*. \tag{3.29}$$

Notice that we may truncate all small singular values in Σ_s and only keep the first r singular values in Σ_r. The columns of $\mathbf{U}_r$ and $\mathbf{V}_r$ may be thought of as *eigen*-time-delay coordinates.

With sensor measurements from an impulse-response experiment, it is also possible to create a second, shifted Hankel matrix $\mathbf{H}'$:

$$\mathbf{H}' = \begin{bmatrix} \mathbf{s}_2 & \mathbf{s}_3^\delta & \cdots & \mathbf{s}_{m_c+1}^\delta \\ \mathbf{s}_3^\delta & \mathbf{s}_4^\delta & \cdots & \mathbf{s}_{m_c+2}^\delta \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{s}_{m_o+1}^\delta & \mathbf{s}_{m_o+2}^\delta & \cdots & \mathbf{s}_{m_c+m_o}^\delta \end{bmatrix} \tag{3.30a}$$

$$= \begin{bmatrix} \mathbf{C}_d\mathbf{A}_d\mathbf{B}_d & \mathbf{C}_d\mathbf{A}_d^2\mathbf{B}_d & \cdots & \mathbf{C}_d\mathbf{A}_d^{m_c}\mathbf{B}_d \\ \mathbf{C}_d\mathbf{A}_d^2\mathbf{B}_d & \mathbf{C}_d\mathbf{A}_d^3\mathbf{B}_d & \cdots & \mathbf{C}_d\mathbf{A}_d^{m_c+1}\mathbf{B}_d \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{C}_d\mathbf{A}_d^{m_o}\mathbf{B}_d & \mathbf{C}_d\mathbf{A}_d^{m_o+1}\mathbf{B}_d & \cdots & \mathbf{C}_d\mathbf{A}_d^{m_c+m_o-1}\mathbf{B}_d \end{bmatrix} = \mathscr{O}_d\mathbf{A}\mathscr{C}_d. \tag{3.30b}$$

Based on the matrices $\mathbf{H}$ and $\mathbf{H}'$, we are able to construct a reduced-order model as follows:

$$\mathbf{A}_r = \Sigma_r^{-1/2}\mathbf{U}_r^*\mathbf{H}'\mathbf{V}_r\Sigma_r^{-1/2}; \tag{3.31a}$$

$$\mathbf{B}_r = \text{ first } N_b \text{ columns of } \Sigma_r^{1/2}\mathbf{V}^*; \tag{3.31b}$$

$$\mathbf{C}_r = \text{ first } N_s \text{ columns of } \mathbf{U}\Sigma_r^{1/2}. \tag{3.31c}$$

Thus, we express the input–output dynamics in terms of a reduced system with a low-dimensional state:

$$\tilde{\mathbf{a}}_{k+1} = \mathbf{A}_r\tilde{\mathbf{a}}_k + \mathbf{B}_r\mathbf{b} \tag{3.32a}$$

$$\mathbf{s} = \mathbf{C}_r\tilde{\mathbf{a}}_k. \tag{3.32b}$$

$\mathbf{H}$ and $\mathbf{H}'$ are constructed from impulse response simulations/experiments, without the need for storing direct or adjoint snapshots, as in other balanced model reduction techniques. However, if full-state snapshots are available (for example, by collecting velocity fields in simulations or PIV experiments), it is then possible to construct direct modes. These full-state snapshots form $\mathscr{C}_d$, and modes can be constructed by:

$$\Phi_r = \mathscr{C}_d\mathbf{V}_r\Sigma_r^{-1/2}. \tag{3.33}$$

These modes may then be used to approximate the full-state of the high-dimensional system from the low-dimensional model in Eq. (3.32) by:

$$\mathbf{a} \approx \Phi_r\tilde{\mathbf{a}}. \tag{3.34}$$

ERA balances the empirical controllability and observability Gramians, $\mathscr{O}_d\mathscr{O}_d^*$ and $\mathscr{C}_d^*\mathscr{C}_d$. Unless we collect a very large amount of data, the true Gramians are only approximately balanced. Instead of collecting long tails of data, it is possible to collect data until the Hankel matrix is full rank, balance the full-rank identified model, and then truncate. This is more efficient than collecting snapshots until all transients have decayed; this idea is developed in [268] and [177].

3.5.3 Observer Kalman filter identification (OKID)

OKID was developed to compliment the ERA for lightly damped experimental systems with noise [152]. In practice, performing isolated impulse response experiments is challenging, and the effect of measurement noise can contaminate results. Moreover, if there is a large separation of timescales, then a tremendous amount of data must be collected to use ERA. This section poses the general problem of approximating the impulse response from arbitrary input/output data. Typically, one would identify reduced-order models according to the following general procedure, shown in Fig. 3.4:

1. Collect the output response to a pseudo-random input.
2. This information is passed through the OKID algorithm to obtain the de-noised linear impulse response.
3. The impulse response is passed through the ERA to obtain a reduced-order state-space system.

The output $\mathbf{s}_k$ in response to a general input signal $\mathbf{b}_k$, for zero initial condition $\mathbf{x}_0 = \mathbf{0}$, is given by:

$$\mathbf{s}_0 = \mathbf{D}_d\mathbf{b}_0 \tag{3.35a}$$
$$\mathbf{s}_1 = \mathbf{C}_d\mathbf{B}_d\mathbf{b}_0 + \mathbf{D}_d\mathbf{b}_1 \tag{3.35b}$$
$$\mathbf{s}_2 = \mathbf{C}_d\mathbf{A}_d\mathbf{B}_d\mathbf{b}_0 + \mathbf{C}_d\mathbf{B}_d\mathbf{b}_1 + \mathbf{D}_d\mathbf{b}_2 \tag{3.35c}$$
$$\cdots$$
$$\mathbf{s}_k = \mathbf{C}_d\mathbf{A}_d^{k-1}\mathbf{B}_d\mathbf{b}_0 + \mathbf{C}_d\mathbf{A}_d^{k-2}\mathbf{B}_d\mathbf{b}_1 + \cdots + \mathbf{C}_d\mathbf{B}_d\mathbf{b}_{k-1} + \mathbf{D}_d\mathbf{b}_k. \tag{3.35d}$$

Note that there is no $\mathbf{C}$ term in the expression for $\mathbf{s}_0$ since there is zero initial condition $\mathbf{x}_0 = \mathbf{0}$. This progression of measurements $\mathbf{s}_k$ may be further simplified and expressed in terms of impulse-response measurements $\mathbf{s}_k^\delta$:

$$\underbrace{\begin{bmatrix}\mathbf{s}_0 & \mathbf{s}_1 & \cdots & \mathbf{s}_N\end{bmatrix}}_{\mathscr{S}} = \underbrace{\begin{bmatrix}\mathbf{s}_0^\delta & \mathbf{s}_1^\delta & \cdots & \mathbf{s}_N^\delta\end{bmatrix}}_{\mathscr{S}^\delta} \underbrace{\begin{bmatrix}\mathbf{b}_0 & \mathbf{b}_1 & \cdots & \mathbf{b}_N \\ \mathbf{0} & \mathbf{b}_0 & \cdots & \mathbf{b}_{N-1} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{b}_0\end{bmatrix}}_{\mathscr{B}}. \tag{3.36}$$

It is often possible to invert the matrix of control inputs, $\mathscr{B}$, to solve for the Markov parameters $\mathscr{S}^\delta$. However, $\mathscr{B}$ may be sparsely populated, so that either it is uninvertible, or inversion is ill-conditioned. In addition, $\mathscr{B}$ is large for lightly damped systems, making inversion computationally expensive. Finally, noise is not optimally filtered by simply inverting $\mathscr{B}$ to solve for the Markov parameters.

The OKID method addresses each of these issues. Instead of the original discrete-time system, we now introduce an optimal observer system:

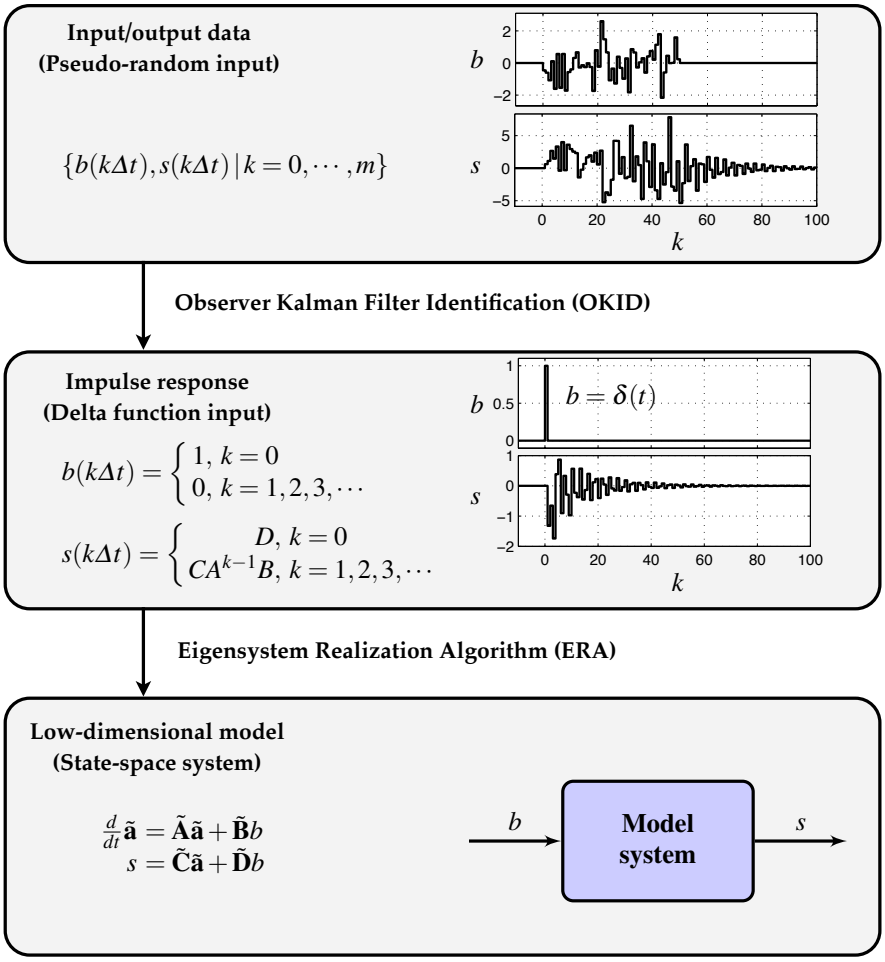

Fig. 3.4 Schematic illustrating the use of OKID followed by ERA to identify a low-dimensional state-space model, based on measurement data. The schematic illustrates the single-input single-output (SISO) case, although both methods are general and handle multiple-input multiple-output (MIMO) systems.

$$\hat{\mathbf{a}}_{k+1} = \mathbf{A}_d\hat{\mathbf{a}}_k + \mathbf{K}_f\left(\hat{\mathbf{s}}_k - \mathbf{s}_k\right) + \mathbf{B}_d\mathbf{b}_k \tag{3.37a}$$

$$\hat{\mathbf{s}}_k = \mathbf{C}_d\hat{\mathbf{a}}_k + \mathbf{D}_d\mathbf{b}_k, \tag{3.37b}$$

which may be re-written as:

$$\hat{\mathbf{a}}_{k+1} = \underbrace{\left(\mathbf{A}_d + \mathbf{K}_f\mathbf{C}_d\right)}_{\bar{\mathbf{A}}_d}\hat{\mathbf{a}}_k + \underbrace{\left[\mathbf{B}_d + \mathbf{K}_f\mathbf{D}_d,\ -\mathbf{K}_f\right]}_{\bar{\mathbf{B}}_d}\begin{bmatrix}\mathbf{b}_k \\ \mathbf{s}_k\end{bmatrix}. \tag{3.38}$$

Recall from above that if the system is observable, it is possible to place the poles of $\mathbf{A}_d + \mathbf{K}_f \mathbf{C}_d$ anywhere we like. However, depending on the amount of noise in the measurements and structural disturbance in our model, there are *optimal* pole locations that are given by the *Kalman filter* (recall Sec. 3.3). We may now solve for the *observer Markov parameters* $\bar{\mathscr{Y}}^{\delta}$ of this system in terms of measured inputs and outputs according to the following algorithm from [152]:

1. Choose the the number of observer Markov parameters to identify, p.
2. Construct the data matrices below:

$$\mathscr{S} = \begin{bmatrix} \mathbf{s}_0 & \mathbf{s}_1 & \cdots & \mathbf{s}_p & \cdots & \mathbf{s}_M \end{bmatrix} \tag{3.39}$$

$$\mathscr{C} = \begin{bmatrix} \mathbf{b}_0 & \mathbf{b}_1 & \cdots & \mathbf{b}_p & \cdots & \mathbf{b}_M \\ \mathbf{0} & \mathbf{v}_0 & \cdots & \mathbf{v}_{p-1} & \cdots & \mathbf{v}_{M-1} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{v}_0 & \cdots & \mathbf{v}_{M-p} \end{bmatrix} \tag{3.40}$$

where $\mathbf{v}_i = \begin{bmatrix} \mathbf{b}_i^T & \mathbf{s}_i^T \end{bmatrix}^T$.
The matrix $\mathscr{C}$ resembles $\mathscr{B}$, except that is has been augmented with the outputs $\mathbf{s}_i$. In this way, we are working with a system that is augmented to include a Kalman filter. We are now identifying the observer Markov parameters of the *augmented* system, $\bar{\mathscr{Y}}^{\delta}$, using the equation $\mathscr{S} = \bar{\mathscr{Y}}^{\delta}\mathscr{C}$.
3. Identify the matrix $\bar{\mathscr{Y}}^{\delta}$ of observer Markov parameters by solving $\mathscr{S} = \bar{\mathscr{Y}}^{\delta}\mathscr{C}$ for $\bar{\mathscr{Y}}^{\delta}$ using the right pseudo-inverse of $\mathscr{C}$ (i.e., SVD).
4. Recover system Markov parameters, $\mathscr{Y}^{\delta}$, from the observer Markov parameters, $\bar{\mathscr{Y}}^{\delta}$.

(a) Order the observer Markov parameters $\bar{\mathscr{Y}}^{\delta}$ as:

$$\bar{\mathscr{Y}}_0^{\delta} = \mathbf{D}, \tag{3.41}$$

$$\bar{\mathscr{Y}}_k^{\delta} = \left[(\bar{\mathscr{Y}}^{\delta})_k^{(1)} \ (\bar{\mathscr{Y}}^{\delta})_k^{(2)} \right] \text{ for } k \geq 1, \tag{3.42}$$

where $(\bar{\mathscr{Y}}^{\delta})_k^{(1)} \in \mathbb{R}^{q \times p}$, $(\bar{\mathscr{Y}}^{\delta})_k^{(2)} \in \mathbb{R}^{q \times q}$, and $\mathbf{s}_0^{\delta} = \bar{\mathscr{Y}}_0^{\delta} = \mathbf{D}$.

(b) Reconstruct system Markov parameters:

$$\mathbf{s}_k^{\delta} = (\bar{\mathscr{Y}}^{\delta})_k^{(1)} + \sum_{i=1}^{k} (\bar{\mathscr{Y}}^{\delta})_i^{(2)} \mathbf{s}_{k-i}^{\delta} \text{ for } k \geq 1. \tag{3.43}$$

Thus, the OKID method identifies the Markov parameters of a system augmented with an asymptotically stable Kalman filter. The system Markov parameters are extracted from the observer Markov parameters by Eq. (3.43). These system Markov

parameters approximate the impulse response of the system, and may be used directly as inputs to the ERA algorithm.

There are numerous extensions of the ERA/OKID methods. For example, there are generalizations for linear parameter varying (LPV) systems and systems linearized about a limit cycle. We will implement the ERA/OKID method in Sec. 6.4 on a turbulent mixing layer experiment. This example will demonstrate the limited usefulness of linear system identification for strongly nonlinear systems.

3.6 Exercises

Exercise 3–1: Show that the following system is controllable but not observable:

$$\frac{d}{dt}\begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} -1 & 1 \\ 0 & -2 \end{bmatrix}\begin{bmatrix} a_1 \\ a_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} b \tag{3.44a}$$

$$s = \begin{bmatrix} 0 & 1 \end{bmatrix}\begin{bmatrix} a_1 \\ a_2 \end{bmatrix}. \tag{3.44b}$$

How might we change the matrix $\mathbf{C} = \begin{bmatrix} 0 & 1 \end{bmatrix}$ to make the system observable?

Exercise 3–2: Develop an optimal LQR controller for the following system:

$$\frac{d}{dt}\begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} -1 & 1 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} a_1 \\ a_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} b. \tag{3.45}$$

(a) In particular, solve for the gain matrix $\mathbf{K}_r$ so that $b = -\mathbf{K}_r\mathbf{a}$ minimizes the cost function:

$$J = \int_0^\infty \left[\mathbf{a}^T(\tau)\,\mathbf{Q}\,\mathbf{a}(\tau) + R\,b^2(\tau)\right] d\tau, \quad \mathbf{Q} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad R = 1. \tag{3.46}$$

(b) Now show that nearby controllers with controller gain $1.1\mathbf{K}_r$ and $.9\mathbf{K}_r$ are suboptimal.

(c) Finally, solve for the optimal $\mathbf{K}_r$ using a genetic algorithm. You will likely need to approximate the cost function J by integrating to a finite but long time until transients decay.

Exercise 3–3: Develop a Kalman filter for the following system:

$$\frac{d}{dt}\begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} -.01 & 1 \\ -1 & -.01 \end{bmatrix}\begin{bmatrix} a_1 \\ a_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} b + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} w_{d_1} \\ w_{d_2} \end{bmatrix} \tag{3.47a}$$

$$s = \begin{bmatrix} 1 & 0 \end{bmatrix}\begin{bmatrix} a_1 \\ a_2 \end{bmatrix} + w_n. \tag{3.47b}$$

(a) Simulate the system with measurement and process noise with forcing $b = \sin(t)$ and plot the Kalman filter prediction of the state. You can compare this to the full-state of the true system by using the same $\mathbf{A}$ and $\mathbf{B}$ matrices above but using $\mathbf{C} = \mathbf{I}$ to output the full state $\mathbf{a}$.

(b) Now, using the same Kalman filter above, increase the process noise (disturbance) by a factor of 5. How does this change the full-state prediction?

(c) For a range of process and measurement noise magnitudes, compute the Kalman filter. How do the eigenvalues of the full-state estimator change with the various noise magnitudes? Is there a relationship?

Exercise 3–4: Consider the following system

$$\frac{d}{dt}\begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} -2 & 1 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} a_1 \\ a_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} b \tag{3.48a}$$

$$s = \begin{bmatrix} 1 & 0 \end{bmatrix}\begin{bmatrix} a_1 \\ a_2 \end{bmatrix}. \tag{3.48b}$$

(a) Compute an LQR controller for the matrix pair **A** and **B**.
(b) Compute a Kalman filter for the matrix pair **A** and **C**.
(c) Now, compute the closed-loop system in Matlab® by implementing LQG control. Show that the closed-loop eigenvalues are the same as the LQR and Kalman filter eigenvalues from above.

Exercise 3–5: Consider the following linear system:

$$\frac{d}{dt}\begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} -1.0 & .001 \\ 0 & -.99 \end{bmatrix}\begin{bmatrix} a_1 \\ a_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} b \tag{3.49a}$$

$$s = \begin{bmatrix} 1 & 0 \end{bmatrix}\begin{bmatrix} a_1 \\ a_2 \end{bmatrix}. \tag{3.49b}$$

(a) Construct a time-series of impulse-response data from this system using a sufficiently small Δt to resolve the dynamics.
(b) How many terms of the time series are required before the rank of the Hankel matrix in Eq. (3.27) saturates? Why is this true?
(c) Use the eigensystem realization algorithm to determine a model from the time-series.
(d) Now, add a small amount of measurement noise to the time-series. How many terms are required to capture the system dynamics now? How does this number change as the noise magnitude is increased?

3.7 Suggested reading

Texts

(1) **Feedback Systems: An Introduction for Scientists and Engineers**, by K. J. Aström and R. M. Murray, 2010 [222].

This is an excellent introductory text that provides a number of motivating examples. The control problem is formulated in state-space, which is beneficial for students with a strong mathematical background.

(2) **Feedback Control Theory**, by J. C. Doyle, B. A. Francis, and A. R. Tannenbaum, 2013 [87].

This text strikes a delicate balance between simple introductory concepts and advanced topics in robust control. The authors largely defined this field, and this book is essential reading.

(3) **Multivariable Feedback Control: Analysis and Design**, by S. Skogestad and I. Postlethwaite, 2005 [251].

This is perhaps the most complete and practically useful guide for real-world engineering control. It strikes a delicate balance between historical, theoretical, and practical advice for the advanced control practitioner.

(4) **A Course in Robust Control Theory: A Convex Approach**, by G. E. Dullerud and F. Paganini, 2000 [93].

This text provides an excellent treatment of the mathematical foundations of linear control theory. There is a considerable focus on computational aspects, including the use of methods from convex optimization.

(5) **Optimal Control and Estimation**, by R. F. Stengel, 2012 [254].

This book provides a comprehensive overview and derivation of optimal control, including advanced methods such as neighboring optimal control. This text covers estimation and forecasting with a subtle balance between dynamical systems and probability.

Seminal papers

(1) **Guaranteed margins for LQG regulators**, by J. C. Doyle, *IEEE Transactions on Automatic Control*, 1978 [86].

This paper turned the world of control upside down. With a simple counterexample, Doyle showed the possible lack of robustness of LQG regulators.

(2) **Principal component analysis in linear systems: Controllability, observability, and model reduction**, by B. C. Moore, *IEEE Transactions on Automatic Control*, 1981 [192].

This paper connects dimensionality reduction with controllability and observability, paving the way towards modern techniques in model reduction.

(3) **Identification of linear parameter varying models**, by B. Bamieh and L. Giarré, *International Journal of Robust and Nonlinear Control*, 2002 [16].

This paper describes how to identify a parameterized family of locally linear models that may be used for gain-scheduled control.