APPENDIX C

CASE STUDY: CONTROL OF A 9-STATE AMMONIA REACTOR

C.1 INTRODUCTION

In this section, we present the results of a case study on the control of a 9-state Ammonia Reactor taken from the Benchmark Collection (Benner *et al.* 1995; see also Patnaik *et al.* 1980). The dynamics of the system is described by:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t), \end{aligned}$$

with system matrices as:

$$A = \begin{bmatrix} -4.019 & 5.120 & 0. & 0. & -2.082 & 0. & 0. & 0. & 0.870 \\ -0.346 & 0.986 & 0. & 0. & -2.340 & 0. & 0. & 0. & 0.970 \\ -7.909 & 15.407 & -4.069 & 0. & -6.450 & 0. & 0. & 0. & 2.680 \\ -21.816 & 35.606 & -0.339 & -3.870 & -17.800 & 0. & 0. & 0. & 7.390 \\ -60.196 & 98.188 & -7.907 & 0.340 & -53.008 & 0. & 0. & 0. & 20.400 \\ 0. & 0. & 0. & 0. & 94. & -147.200 & 0. & 53.200 & 0. \\ 0. & 0. & 0. & 0. & 0. & 94. & -147.200 & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 12.800 & 0. & -31.600 & 0. \\ 0. & 0. & 0. & 0. & 12.800 & 0. & 0. & 18.800 & -31.600 \end{bmatrix},$$

$$B = \begin{bmatrix} 0.010 & 0.003 & 0.009 & 0.024 & 0.068 & 0. & 0. & 0. & 0. \\ -0.011 & -0.021 & -0.059 & -0.162 & -0.445 & 0. & 0. & 0. & 0. \\ -0.151 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \end{bmatrix}^{\mathrm{T}},$$

$$C = \begin{bmatrix} 0. & 0. & 0. & 0. & 0. & 0. & 1. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 1. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 1. \end{bmatrix}, \qquad D = \begin{bmatrix} 0. & 0. & 0. \\ 0. & 0. & 0. \\ 0. & 0. & 0. \end{bmatrix}.$$

The controllability, observability, and the asymptotic stability of the system are determined first using, respectively, the MATCONTROL functions **cntrlhs**, **obserhs**, and the MATLAB function **eig**.

The feedback controller is then designed via **Lyapunov stabilization, pole-placement;** and **LQR** and **LQG techniques**. The MATCONTROL functions **stablyapc** and **polercm**, and the MATLAB functions **lqr** and **lqgreg**, are, respectively, used for this purpose.

The **impulse responses** of the system are then compared in each case.

The states of the system are estimated using the MATLAB function **Kalman** and the MATCONTROL function **sylvobsmb**. The relative errors between the estimated and actual states are then plotted in each case.

Finally, the system is identified using the MATCONTROL functions **minresvd** and **minremsvd** and then the identified model is reduced further by the MATCONTROL function **modreds**. The frequency response in each case is compared using the MATCONTROL function **freqresh**.

C.2 TESTING THE CONTROLLABILITY

In order to test the controllability of the system, the MATCONTROL function **cntrlhs (see Chapter 6)**, based on the decomposition of the pair (A, B) to a **controller-Hessenberg** form is used.

```
tol = 1e−13;
info = cntrlhs(A, B, tol)
info = 1
```

Conclusion: The system is controllable.

C.3 TESTING THE OBSERVABILITY

The observability is tested by using the MATCONTROL function **obserhs**, based on the decomposition of the pair (A, C) to a **observer-Hessenberg form (See Chapter 6)** with the **same tolerance** as above.

```
info = obserhs(A, C, tol)
info = 1
```

Conclusion: The System is observable.

C.4 TESTING THE STABILITY

In order to test the asymptotic stability of the system, the eigenvalues of the matrix A are computed using the MATLAB function **eig.** They are:

$$\{-147.2000, -153.1189, -56.0425, -37.5446, -15.5478, -4.6610, -3.3013, \\ -3.8592, -0.3047\}.$$

Conclusion: The system is asymptotically stable but it has a small eigenvalue $\lambda = -0.3047$ (relative to the other eigenvalues).

C.5 LYAPUNOV STABILIZATION

The Lyapunov stabilization technique is now used to move the eigenvalues further to the left of the complex plane. The MATCONTROL function **stablyapc** is used for this purpose. This function requires an upper bound β of the spectrum of A, which is taken as the Frobenius norm of A.

```
beta = norm(A,'fro')
beta = 292.6085
K_lyap = stablyapc(A, B, beta)
```

The feedback matrix $K_{-\text{lyap}}$ is:

$$K_{-\text{lyap}} = 10^2 \begin{bmatrix} -3.3819 & -0.2283 & -56.4126 \\ 5118.1388 & 1207.4106 & 15424.1947 \\ -237858.9775 & -57713.9866 & -997148.5316 \\ -544.7145 & 220.0287 & 15199.8829 \\ 31495.6810 & 7491.5724 & 125946.2030 \\ -4510.0481 & 20403.0258 & -5516.3430 \\ 85.8840 & -495.7330 & 40.1441 \\ -39007.7960 & 182650.2401 & -43969.5212 \\ 38435.8476 & -150078.6710 & 61412.4200 \end{bmatrix}$$

The eigenvalues of the corresponding closed-loop matrix are:

$$\{-292.6085 \pm 644.6016i, -292.6085 \pm 491.8461i, -292.6085 \pm 145.4054i, \\ -292.6085 \pm 49.3711i, -292.6085\}.$$

Note that these close-loop eigenvalues now are much further to the left of the complex plane than the open-loop ones.

C.6 POLE-PLACEMENT DESIGN

It is now desired to move all the above nine eigenvalues to the negative real-axis with equal spacing in the interval $[-||A||_F/9, -||A||_F)]$. The pole-placement technique is used for this purpose. The MATCONTROL function **polercm**, which implements the recursive multi-input pole-placement algorithm (**Algorithm 11.3.1**) is used to do so.

$$\begin{aligned} &\mathrm{eig}_{-\mathrm{rcm}} = -[1:9] * \mathrm{beta}/9 \\ &K_{-\mathrm{rcm}} = \mathrm{polercm}(A, B, \mathrm{eig}_{-\mathrm{rcm}}) \end{aligned}$$

The feedback matrix $K_{-\mathrm{rcm}}$ in this case is:

$$K_{-\mathrm{rcm}} = 10^5 \begin{bmatrix} -0.1088 & 14.0002 & -1358.6004 & 17.6295 & 171.8716 & 1.2245 & 0.0034 & 4.8847 & -5.8828 \\ -0.0153 & 2.1371 & -207.6062 & 2.6939 & 26.2618 & 0.1865 & 0.0005 & 0.7408 & -0.8998 \\ -0.0357 & -0.5495 & -74.6029 & 1.3318 & 9.3685 & 0.0670 & 0.0002 & 0.2666 & -0.3206 \end{bmatrix}.$$

The eigenvalues of the corresponding closed-loop matrix are

$$\{-292.6085, -260.0965, -227.5844, -195.0724, -162.5603, -130.0482, \\ -97.5362, -65.0241 \text{ and } -32.5121\}.$$

C.7 THE LQR AND LQG DESIGNS

Recall (**Chapter 10**) that the LQR design is used to find the optimal control-law

$$u^0(t) = K_{-\mathrm{lqr}} x(t)$$

such that the objective functional $J = \int_0^\infty (x^T Qx(t) + u(t)Ru(t))dt$ is minimized subject to $\dot{x} = Ax + Bu,\ x(0) = x_0$. The gain matrix $K_{-\mathrm{lqr}}$ is obtained by solving the CARE: $XA + A^T X - XBR^{-1}B^T X + Q = 0$. The MATLAB function **lqr** with $R = eye(3)$, $N = zeros(9, 3)$, and $Q = eye(9)$ is used for this purpose.

$$\boxed{K_{-\mathrm{lqr}} = \mathrm{lqr}(A, B, Q, R, N)}$$

The optimal gain matrix $K_{-\mathrm{lqr}}$ is:

$$K_{-\mathrm{lqr}} = 10^{-1} \begin{bmatrix} 0.1187 & 0.0728 & 0.0228 & 0.0012 & -0.0007 & 0.0018 & 0.0003 & 0.0042 & 0.0044 \\ 0.2443 & -0.3021 & 0.0084 & -0.0465 & -0.0673 & -0.0138 & -0.0023 & -0.0464 & -0.0439 \\ -2.8408 & -0.5942 & -0.4540 & 0.0855 & 0.2102 & 0.0061 & 0.0003 & 0.0496 & 0.0378 \end{bmatrix}.$$

The eigenvalues of the corresponding closed-loop system are:

$\{-153.1201, -147.1984, -56.0452, -37.5442, -15.5463, -4.6789, -3.3090,$
$-3.8484, \text{ and } -0.3366\}.$

Note that these closed-loop eigenvalues are quite close to the open-loop ones. Also, $||K_{-\text{lqr}}||$ is much smaller than that of $||K_{-\text{rcm}}||$.

To implement the above control law, one needs to have the knowledge of the state vector $x(t)$; however, in practice only a few of the variables are measured and the remaining ones need to be estimated. There are several ways to do so (see **Chapter 12** and discussions later here in this section). If the Kalman estimator K_{-est} is used for this purpose, the design is called LQG (**Linear Quadratic Gaussian**) design.

The Kalman estimator approximates the state of a stochastic linear system

$$\dot{x}(t) = Ax(t) + Bu(t) + Gw(t) \qquad \textbf{(state equation)}$$
$$y_m(t) = Cx(t) + Du(t) + Hw(t) + v(t) \qquad \textbf{(measured equation)}$$

with known inputs $u(t)$, process noise $w(t)$, measurement noise $v(t)$, and noise covariances

$$Q_n = E[ww^T], \quad R_n = E[vv^T], \quad N_n = E[wv^T]$$

where $E[\cdot]$ denotes the expected value of an stochastic variable. The Kalman estimator has input $(u; y_m)$ and generates the optimal estimates (y_l, x_l) of (y, x) given by:

$$\dot{x_e} = Ax_e + Bu + L(y_m - Cx_e - Du)$$
$$\begin{bmatrix} y_l \\ x_l \end{bmatrix} = \begin{bmatrix} C \\ I \end{bmatrix} x_e + \begin{bmatrix} D \\ 0 \end{bmatrix} u$$

where L is the filter gain determined by solving an algebraic Riccati equation(**See Chapter 12**).

To perform the LQG design, MATLAB functions **kalman** and **lqgreg** are used as follows:

```
sysA = ss(A, B, C, D);
Qn = 1E-3 * eye(3); Rn = 1E-3 * eye(3);
[K_est, L] = kalman(sysA, Qn, Rn)
```

The filter gain matrix L is:

$$L = 10^{-3}\begin{bmatrix} -0.0007 & 1.0703 & 1.6155 & 1.9729 & 2.8499 & 2.1636 & 1.3816 & 0.7990 & 1.5340 \\ 0.0176 & 0.6284 & 0.9780 & 1.1679 & 1.5765 & 1.2251 & 0.7990 & 0.4962 & 0.9469 \\ 0.0349 & 1.2163 & 1.8921 & 2.2676 & 3.0810 & 2.3701 & 1.5340 & 0.9469 & 1.8112 \end{bmatrix}^{T}$$

Using the matrices $K_{-\text{est}}$ and L, the LQG regulator can now be designed. The MATLAB command for finding an LQG regulator is **lqgreg**.

$$\boxed{RLQG = \text{lqgreg}(K_{-\text{est}}, K_{-\text{lqr}})}$$

The resulting regulator $RLQG$ has input y_m and the output $u = -K_{-lqr}x_{-e}$ as shown below:

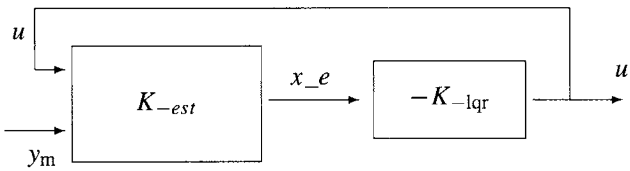

Figure C.1, shows the **impulse response** of thes system in five cases: (i) uncontrolled, (ii) controlled via Lyapunov stabilization, (iii) controlled via pole-placement, (iv) controlled via LQR design, and (v) controlled via LQG design.

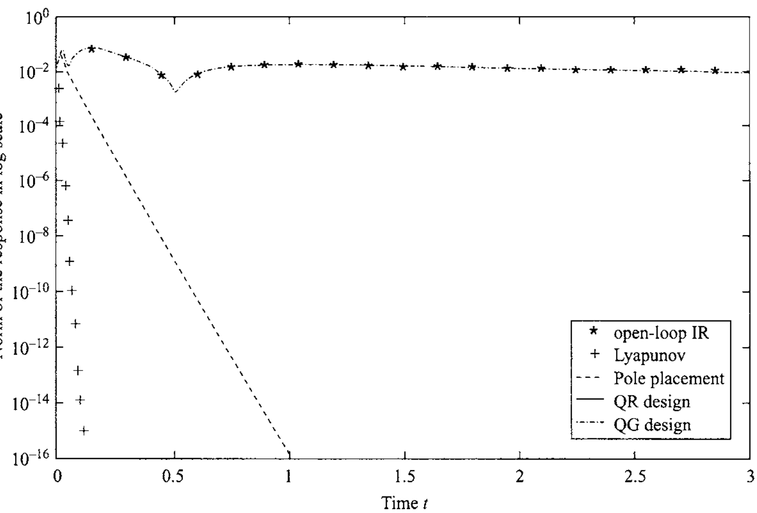

FIGURE C.1: Comparison of the impulse responses.

Note that the LQR and LQG responses cannot be distinguished from each other. Computations were done in **Simulink 4** with **MATLAB 6**.

C.8 STATE-ESTIMATION (OBSERVER): KALMAN ESTIMATOR VS. SYLVESTER EQUATION ESTIMATOR

Our next goal is to compare two procedures for state estimation of the system: the **Kalman estimator approach** and **the Sylvester-observer approach.**

Recall from **Chapter 12** that the Sylvester-equation approach for state-estimation is based on solving the Sylvester-observer equation: $XA - FX = GC$. The MATCONTROL function **sylvobsmb** which implements Algorithm 12.7.2 (A Recursive Block Triangular Algorithm) for this purpose, is used here. Using the data of the case study and the observer eigenvalues as $ev = [-2, -4 \pm 2i, -5, -6, -7]^T$, we obtain

$$[X, F, G] = \text{sylvobsmb}(A, C, \text{ev});$$

$$X = \begin{bmatrix} -5734.5147 & 5470.8582 & -1106.8206 & 52.2506 & 287.3781 & 0. & 0. & 0. & 727.5156 \\ -8.4146 & 13.4543 & -0.0962 & -0.0139 & -0.5931 & 0. & 0. & 0. & -1.2949 \\ 6.8075 & -9.1950 & 0.8500 & -0.0354 & 0.2121 & 0. & 0. & 0. & 0.5729 \\ 0.5214 & -0.8505 & 0.0685 & -0.0029 & 0.0782 & 0. & 0. & 0. & 0.2327 \\ 0. & 0. & 0. & 0. & 1. & 0. & -0.0213 & 0. & 3.1234 \\ 0. & 0. & 0. & 0. & 0. & 1. & 1.5234 & 0. & -7.1875 \end{bmatrix},$$

$$F = \begin{bmatrix} -2. & 0. & 0. & 0. & 0. & 0. \\ -0.0042 & -5. & 0. & 0. & 0. & 0. \\ 0. & 0.2435 & -6. & 0. & 0. & 0. \\ 0. & 0. & -0.4901 & -7. & 0. & 0. \\ 0. & 0. & 0. & -115.4430 & -4. & -2. \\ 0. & 0. & 0. & 0. & 2. & -4. \end{bmatrix},$$

$$G = 10^1 \begin{bmatrix} 0. & 0. & 0. & 0. & 0.6094 & -21.8109 \\ 1367.7293 & -2.4345 & 1.0771 & 0.4375 & 5.8721 & -8.1925 \\ -1793.4391 & 3.0741 & -1.1006 & -0.4058 & -5.3316 & 19.2128 \end{bmatrix}^{\mathrm{T}},$$

where F was chosen to be a stable matrix. The error in the solution X: $\|XA - FX - GC\|_F = 1.2246 \cdot 10^{-11}$.

Figure C.2 shows the comparison of relative errors, between actual and estimated states in two cases: **Kalman estimator** and **Sylvester-equation estimator**. The

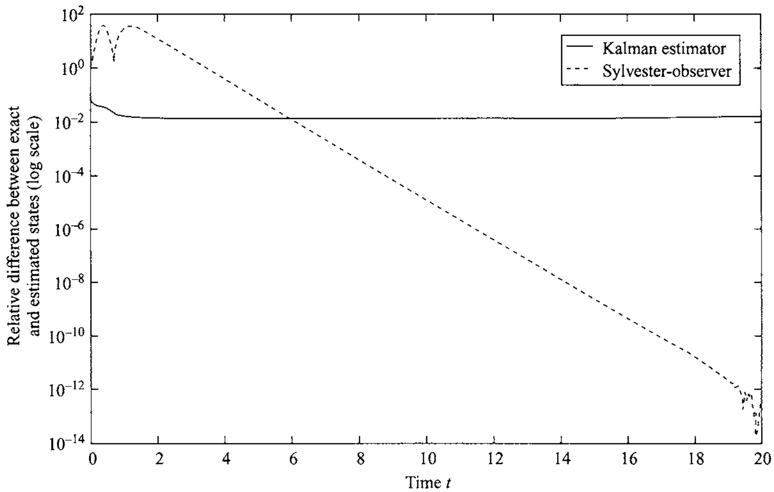

FIGURE C.2: Comparison between Kalman and Sylvester-observer Estimations.

quantity plotted is

$$r(t) = \frac{\|x(t) - \hat{x}(t)\|}{\|x(t)\|}$$

where $\hat{x}(t)$ is the estimate given by the estimator in each case.

The plot shows that error in the Sylvester-observer estimator approaches to zero faster than the Kalman estimator as the time increases.

C.9 SYSTEM IDENTIFICATION AND MODEL REDUCTION

In order to perform system identification tasks, we recall that our system has the transfer function

$$H(s) = C(sI - A)^{-1}B = \sum_{i=1}^{\infty} \frac{CA^iB}{s^i}.$$

The quantities $H_i = CA^iB, i = 1, 2, 3, \dots$ are called the **Markov parameters**; they are usually obtained from input-output experiments. The frequency response is defined by $G(j\omega) = H(j\omega)$ where ω is a nonnegative real number and $j = \sqrt{-1}$.

After directly computing the first 9 Markov parameters, $H_i, i = 1, ..., 9$; *MATCONTROL* functions **minresvd** and **minremsvd** are used to perform system identification:

[A_S,B_S,C_S] = minresvd(4,[H1 H2 H3 H4 H5 H6 H7 H8 H9],1e-8)

The resulting model obtained by **minresvd** is oversized. We, therefore, applyl a model reduction technique to this identified oversized model. The MATCONTROL function **modreds** (see **Chapter 14**) is used for this purpose, obtaining a reduced-order model (A_{-r}, B_{-r}, C_{-r}).

```
N = 4; tol = 1e−13;
[A_s, B_s, C_s] = minresvd (N, H_i, tol)
[A_r, B_r, C_r] = modreds (A_s, B_s, C_s, 9)
```

The frequency response function **freqresh** from MATCONTROL is then invoked to compute frequency responses in a chosen frequency range for all these models: **the original**, the **model identified by the SVD algorithm(Algorithm 9.3.1)**, the **model identified by the modified SVD algorithm (Algorithm 9.3.2)**, and the **model identified by the SVD algorithm followed by the model reduction technique.**

Let $(A_{-sm}, B_{-sm}, C_{-sm})$ denote the system identified by the function **minremsvd**:

```
omega = 1:.1:100;
G = freqresh (A, B, C, omega)
Gs = freqresh (As, Bs, Cs, omega)
Gsm = freqresh (Asm, Bsm, Csm, omega)
Gr = freqresh (Ar, Br, Cr, omega)
```

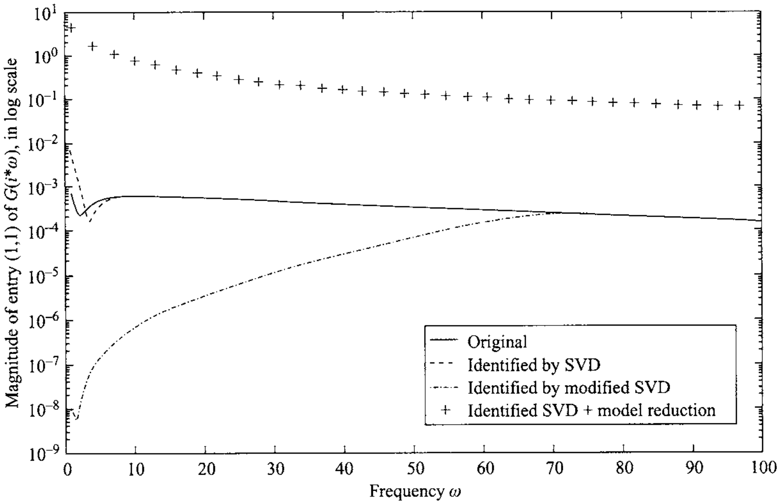

FIGURE C.3: Comparison between Frequency Responses.

Figure C.3 shows a comparison between the magnitude of the entry (1,1) of the original frequency response G, the frequency response $\mathrm{G_s}$ of the system identified by the SVD method, the frequency response $\mathrm{G_{sm}}$ of the system identified by the modified SVD method, and the frequency response G_{-r} of the system $(A_{-\mathrm{r}}, B_{-\mathrm{r}}, C_{-\mathrm{r}})$, which is obtained from $(A_{-\mathrm{s}}, B_{-\mathrm{s}}, C_{-\mathrm{s}})$ followed by model reduction.

References

Benner, P., Laub, A., and Mehrmann, V. *A Collection of benchmark examples for the numerical solution of algebraic Riccati equations I: continuous-time case. Technische Universität Chemnitz-Zwickau,* SPC Report 95–22, 1995.

Patnaik, L., Viswanadham, N., and Sarma, I. *Computer control algorithms for a tubular ammonia reactor.* IEEE Trans. Automat. Control, AC-25, pp. 642–651, 1980.