

# Contents

<b>1</b>	<b>Introduction</b>	<b>4</b>
<b>I</b>	<b>Literature, Theory and Background Material</b>	<b>7</b>
<b>2</b>	<b>Literature Review</b>	<b>8</b>
2.1	Stochastic Model Predictive Control . . . . .	8
2.2	Switching Model Predictive Control . . . . .	12
<b>3</b>	<b>Background Theory</b>	<b>16</b>
3.1	Probability Theory . . . . .	16
3.1.1	Discrete Random Variables . . . . .	17
3.1.2	Continuous Random Variables . . . . .	20
3.2	Graph Theory . . . . .	24
3.3	Probabilistic Graphical Models . . . . .	25
3.3.1	Bayesian Networks . . . . .	26
3.3.2	Dynamic Bayesian Networks . . . . .	28
3.4	Control . . . . .	30
3.4.1	Linear Quadratic Regulator Control . . . . .	30
3.4.2	Reference Tracking . . . . .	34
3.4.3	Linear Quadratic Gaussian Control . . . . .	34
3.4.4	Model Predictive Control . . . . .	36
3.5	Matrix Identities . . . . .	37
<b>4</b>	<b>Hidden Markov Models</b>	<b>38</b>
4.1	Markov Models . . . . .	38
4.2	Hidden Markov Models . . . . .	39
4.2.1	Filtering . . . . .	40
4.2.2	Smoothing . . . . .	41
4.2.3	Viterbi Decoding . . . . .	43
4.2.4	Prediction . . . . .	44
4.3	Burglar Localisation Problem . . . . .	45
<b>5</b>	<b>CSTR Model</b>	<b>48</b>

5.1	Qualitative Analysis . . . . .	49
5.2	Nonlinear Model . . . . .	51
5.3	Linearised Models . . . . .	53
<b>II</b>	<b>Single Model Systems</b>	<b>60</b>
<b>6</b>	<b>Inference using linear models</b>	<b>61</b>
6.1	Kalman filter . . . . .	62
6.2	Kalman prediction . . . . .	64
6.3	Smoothing and Viterbi decoding . . . . .	66
6.4	Filtering the CSTR . . . . .	67
<b>7</b>	<b>Inference using nonlinear models</b>	<b>72</b>
7.1	Sequential Monte Carlo methods . . . . .	73
7.2	Particle filter . . . . .	76
7.3	Particle prediction . . . . .	78
7.4	Smoothing and Viterbi decoding . . . . .	78
7.5	Filtering the CSTR . . . . .	79
<b>8</b>	<b>Stochastic linear control</b>	<b>84</b>
8.1	Unconstrained stochastic control . . . . .	85
8.2	Constrained stochastic control . . . . .	88
8.3	Reference tracking . . . . .	95
8.4	Linear system . . . . .	95
8.5	Nonlinear system . . . . .	104
8.6	Conclusion . . . . .	115
<b>III</b>	<b>Multiple Model Systems</b>	<b>117</b>
<b>9</b>	<b>Inference using Linear Hybrid Models</b>	<b>115</b>
9.1	Exact Filtering . . . . .	116
9.2	Rao-Blackwellised Particle Filter . . . . .	117
9.3	Rao-Blackwellised Particle Prediction . . . . .	118
9.4	Smoothing and Viterbi Decoding . . . . .	119
9.5	Filtering the CSTR . . . . .	119
<b>10</b>	<b>Stochastic Switching Linear Control using Linear Hybrid Models</b>	<b>128</b>
10.1	Unconstrained Switching Control . . . . .	130
10.2	Conclusion . . . . .	134
<b>11</b>	<b>Inference using Nonlinear Hybrid Models</b>	<b>135</b>
11.1	Exact Filtering . . . . .	136
11.2	Switching Particle Filter . . . . .	136

11.3	Switching Particle Prediction . . . . .	137
11.4	Smoothing and Viterbi Decoding . . . . .	138
11.5	Filtering the CSTR . . . . .	138
<b>12</b>	<b>Stochastic Switching Linear Control using Nonlinear Hybrid Models</b>	<b>143</b>
12.1	Unconstrained Switching Control . . . . .	144
12.2	Constrained Switching Control . . . . .	146
12.3	Conclusion . . . . .	152
<b>13</b>	<b>Future Work and Conclusion</b>	<b>153</b>
13.1	Parameter Optimisation . . . . .	153
13.2	Augmented Switching Graphical Model . . . . .	153
13.3	Generalised Graphical Models . . . . .	153
13.4	Conclusion . . . . .	154

## **Part I**

# **Literature, Theory and Background Material**

## Part II

# Single Model Systems

# Chapter 8

## Stochastic linear control

In this chapter we consider the stochastic reference tracking problem. It is required to move the states and manipulated variables of the system, shown in (8.1), to the set point  $(x_{sp}, u_{sp})$  by manipulating the input variables  $u$ . It is assumed that  $x_t$  is a latent stochastic variable and  $y_t$  is an observed stochastic variable.

$$\begin{aligned} x_{t+1} &= f(x_t, u_t) + w_{t+1} \\ y_{t+1} &= g(x_{t+1}) + v_{t+1} \end{aligned} \tag{8.1}$$

We assume uncorrelated zero mean additive Gaussian noise ( $w_t$  and  $v_t$ ) in both the state function  $f$  and the observation function  $g$  with known covariances  $W$  and  $V$  respectively. Clearly it is not possible to achieve perfect control (zero offset at steady state) because of the noise terms, specifically  $w_t$ . For this reason we need to relax the set point goal a little bit. We will be content if our controller is able to achieve Definition 8.1.

**Definition 8.1. Stochastic reference tracking goal:** Suppose we have designed a controller and set  $\delta > 0$  as a controller benchmark. If there exists some positive number  $t^* < \infty$  such that  $\forall t > t^*$  the controller input causes  $\mathbb{E}[(x_t - x_{sp})^T Q(x_t - x_{sp}) + (u_t - u_{sp})^T R(u_t - u_{sp})] < \delta$  we will have satisfied the stochastic reference tracking goal given  $\delta$ .

While Definition 8.1 is pleasing from a theoretical point of view, it is not easy to design a controller to specifically satisfy a given  $\delta$ . We again simplify our goal somewhat: we will be content if the controller we design (without a specific  $\delta$  in mind) can satisfy Definition 8.1 for some suitably small resultant  $\delta$ . Intuitively, we would like the mean of the states and inputs to be “close enough” to the set points.

In this chapter we limit ourselves by only considering controllers developed using a single linear model of the underlying, possibly nonlinear, system functions  $f$  and  $g$ . The linearised model control is based upon is shown in (8.2) and is subject to the same noise as (8.1).

$$\begin{aligned} x_{t+1} &= Ax_t + Bu_t + w_{t+1} \\ y_{t+1} &= Cx_{t+1} + v_{t+1} \end{aligned} \tag{8.2}$$

We will endeavour to develop predictive controllers using the graphical models of Chapter 6 and 7.

## 8.1 Unconstrained stochastic control

Our first goal is to solve the problem in (8.3) given the current state estimate  $x_0$ <sup>1</sup>. If the system is controllable then solving (8.3) will satisfy the linear unconstrained stochastic reference tracking goal.

$$\min_{\mathbf{u}} J_{LQG}(x_0, \mathbf{u}) = \mathbb{E} \left[ \frac{1}{2} \sum_{k=0}^{N-1} (x_k^T Q x_k + u_k^T R u_k) + \frac{1}{2} x_N^T P_f x_N \right] \quad (8.3)$$

subject to  $x_{t+1} = Ax_t + Bu_t + w_t$

Note that the future inputs  $\mathbf{u} = (u_0, u_1, \dots, u_{T-1})$  are denoted in boldface to emphasise that it could be a vector of vectors. Inspecting (8.3) we see that this is none other than the LQG control problem of Chapter 3.4.3. Therefore we know what the optimal solution should look like.

We start our analysis using the results of Chapter 6. We immediately realise that the optimal linear state estimator is the Kalman filter. We assume that at every sequential time step we have the current state estimate, supplied by the Kalman filter, and denote this by  $x_0$ . Since we are using the Kalman filter the mean and covariance of the state estimate is well defined.

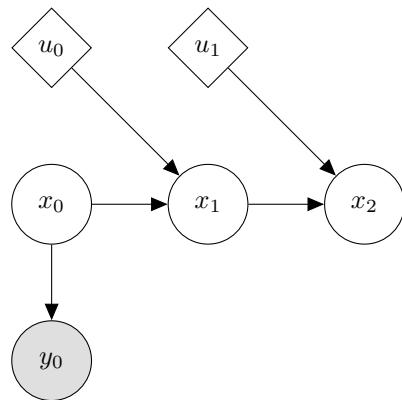


Figure 8.1: graphical model for state prediction

Inspecting Figure 8.1 we note that the state prediction equations derived in Chapter 6.2 are applicable. Thus we can predict the state distributions given the future inputs  $\mathbf{u}$ .

Before we proceed we prove a very intuitive result in Theorem 8.1. We will use this to link the predictive controller derived using the results of Chapter 6 to the LQR controller derived in Chapter 3.4.1.

---

<sup>1</sup>It is customary to assign  $x_0 \leftarrow x_t$  at each time step to simplify the controller optimisation problem's notation.

**Theorem 8.1. Optimisation Equivalence** Suppose we have two real valued convex objective functions  $f(x_0, \mathbf{u})$  and  $g(x_0, \mathbf{u})$  and we are required to minimise them with respect to  $\mathbf{u}$  over the same space where they are both defined:  $\mathbf{u} \in \mathcal{U}$  and  $x_0 \in \mathcal{X}$ . Furthermore, suppose there exists a real number  $k$  such that  $\forall \mathbf{u} \in \mathcal{U}$  we have that  $g(x_0, \mathbf{u}) + k = f(x_0, \mathbf{u})$ . Finally, assume the existence and uniqueness of the global minimiser for each problem. Then the global minimiser  $\mathbf{u}^*$  of  $g(x_0, \mathbf{u})$  is also the global minimiser of  $f(x_0, \mathbf{u})$ .

*Proof.* This proof only holds over functions which are at least twice differentiable. By assumption we know that  $\mathbf{u}^*$  is the minimiser of  $g(x_0, \mathbf{u})$  given  $x_0$ . By the necessary conditions for optimality [25] we know that  $\nabla g(x_0, \mathbf{u}^*) = 0$  and that  $\nabla^2 g(x_0, \mathbf{u}^*)$  is positive semi-definite. Since  $f$  and  $g$  are both twice differentiable and  $g(x_0, \mathbf{u}^*) + k = f(x_0, \mathbf{u}^*)$  it must hold that  $\nabla g(x_0, \mathbf{u}^*) = \nabla f(x_0, \mathbf{u}^*)$  and  $\nabla^2 g(x_0, \mathbf{u}^*) = \nabla^2 f(x_0, \mathbf{u}^*)$ . Since  $\nabla^2 g(x_0, \mathbf{u}^*)$  is positive semi-definite it must be that  $\nabla^2 f(x_0, \mathbf{u}^*)$  is also positive semi-definite. Therefore  $\mathbf{u}^*$  is necessarily a minimum of  $f$ . Since  $f$  is convex the minimum must also be a global minimum.  $\square$

Now we are in a position to show the equivalence between the LQR control problem and the LQG control problem using the results of Chapter 6. Theorem 8.2 shows how this is possible. It is quite reassuring to note that by starting within the framework of graphical models we arrive at the most important contribution of [59] and [60] in an intuitively simple manner.

**Theorem 8.2. LQR and LQG objective function difference** Consider the LQR and LQG objective functions in (8.4) and (8.5) respectively.

$$J_{LQR}(x_0, \mathbf{u}) = \frac{1}{2} \sum_{k=0}^{N-1} (x_k^T Q x_k + u_k^T R u_k) + \frac{1}{2} x_N^T P_f x_N \quad (8.4)$$

with  $x_{t+1} = Ax_t + Bu_t$

$$J_{LQG}(x_0, \mathbf{u}) = \mathbb{E} \left[ \frac{1}{2} \sum_{k=0}^{N-1} (x_k^T Q x_k + u_k^T R u_k) + \frac{1}{2} x_N^T P_f x_N \right] \quad (8.5)$$

with  $x_{t+1} = Ax_t + Bu_t + w_{t+1}$

(8.6)

Suppose  $x_0$  is the state estimate supplied by the Kalman filter given the latest observation in the stochastic case. In the deterministic case we have that  $x_0 = \mathbb{E}[x_0] = \mu_0$  because we exactly observe the state. Given any input sequence  $\mathbf{u} \in \mathcal{U}$ , where  $\mathcal{U}$  is the shared admissible input space, we have that  $J_{LQR}(x_0, \mathbf{u}) + \frac{1}{2} \sum_{k=0}^N \text{tr}(Q\Sigma_k) = J_{LQG}(x_0, \mathbf{u})$  where  $\Sigma_{t+1} = W + A\Sigma_t A^T$  and  $\Sigma_0$  is the covariance matrix of the current state given by the Kalman filter.

*Proof.* Expanding the LQG objective function and noting that  $\mathbf{u}$  is deterministic we have (8.7). Note that the conditional expectations in the expansion originate from the graphical

model in Figure 8.1 due to the first order Markov assumption.

$$\begin{aligned}
J_{LQG}(x_0, \mathbf{u}) &= \frac{1}{2}\mathbb{E} [x_0^T Q x_0 + u_0^T R u_0] + \frac{1}{2}\mathbb{E} [x_1^T Q x_1 + u_1^T R u_1 | x_0] + \dots \\
&\quad + \frac{1}{2}\mathbb{E} [x_{N-1}^T Q x_{N-1} + u_{N-1}^T R u_{N-1} | x_{N-2}] + \frac{1}{2}\mathbb{E} [x_N^T P_f x_N | x_{N-1}] \\
&= \frac{1}{2}\mathbb{E} [x_0^T Q x_0] + \frac{1}{2}u_0^T R u_0 + \frac{1}{2}\mathbb{E} [x_1^T Q x_1 | x_0] + \frac{1}{2}u_1^T R u_1 + \dots \\
&\quad + \frac{1}{2}\mathbb{E} [x_{N-1}^T Q x_{N-1} | x_{N-2}] + \frac{1}{2}u_{N-1}^T R u_{N-1} + \frac{1}{2}\mathbb{E} [x_N^T P_f x_N | x_{N-1}]
\end{aligned} \tag{8.7}$$

We know that  $x_0 \sim \mathcal{N}(\mu_0, \Sigma_0)$  because the current state estimate comes from the Kalman filter. This means that we can evaluate the first expected value in (8.7) using Theorem 3.5 as shown in (8.8).

$$\mathbb{E} [x_0^T Q x_0] = \text{tr}(Q\Sigma_0) + \mu_0^T Q \mu_0 \tag{8.8}$$

Now we turn our attention to the second expected value in (8.7). First note that because we have  $x_0$  and  $\mathbf{u}$  we can use the result from Chapter 6.2 to predict the distribution of  $x_1$ . Therefore we know that  $x_1 \sim \mathcal{N}(A\mu_0 + Bu_0, W + A\Sigma_0 A^T)$ . Now we let  $\mu_1 = A\mu_0 + Bu_0$  and  $\Sigma_0 = W + A\Sigma_0 A^T$ . Then by using Theorem 3.5 as before we have (8.9).

$$\mathbb{E} [x_1^T Q x_1 | x_0] = \text{tr}(Q\Sigma_1) + \mu_1^T Q \mu_1 \tag{8.9}$$

Note that  $\text{tr}(Q\Sigma_1)$  does not depend on  $u_0$  but only on the initial state estimate  $x_0$  which is independent of the future inputs  $\mathbf{u}$ . Notice that we can continue in this manner to simplify the LQG objective function to (8.10).

$$\begin{aligned}
J_{LQG}(x_0, \mathbf{u}) &= \frac{1}{2} \sum_{k=0}^{N-1} (\mu_k^T Q \mu_k + u_k^T R u_k) + \frac{1}{2} \mu_N^T P_f \mu_N + \frac{1}{2} \sum_{k=0}^N \text{tr}(Q\Sigma_k) \\
\text{with } \mu_{t+1} &= A\mu_t + Bu_t \\
\text{and } \Sigma_{t+1} &= W + A\Sigma_t A^T
\end{aligned} \tag{8.10}$$

Now note that except for the last term  $J_{LQG}(x_0, \mathbf{u})$  is exactly the same as  $J_{LQR}(x_0, \mathbf{u})$ . The conclusion follows because  $\frac{1}{2} \sum_{k=0}^N \text{tr}(Q\Sigma_k)$  is independent of  $\mathbf{u}$ .  $\square$

Finally we combine Theorem 8.1 and 8.2 to produce Theorem 8.3.

**Theorem 8.3. Solution of the finite horizon LQG control problem** We wish to solve the LQG control problem within the framework of graphical models. The full problem is shown in (8.11). We assume that  $x_0$  is the current posterior state estimate supplied by the Kalman filter.

$$\min_{\mathbf{u}} V(x_0, \mathbf{u}) = \mathbb{E} \left[ \frac{1}{2} \sum_{k=0}^{N-1} (x_k^T Q x_k + u_k^T R u_k) + \frac{1}{2} x_N^T P_f x_N \right] \tag{8.11}$$

$$\text{subject to } x_{t+1} = Ax_t + Bu_t + w_t$$

The solution of (8.11) is equivalent to solving the LQR problem with initial state equal to the mean of the initial state estimate from the Kalman filter.

*Proof.* We assume that we have the Kalman filter posterior state estimate for  $x_0$ . We use Theorem 8.2 to prove that given  $x_0$  and  $\forall \mathbf{u} \in \mathcal{U}$  we have that  $J_{LQR}(x_0, \mathbf{u}) + \frac{1}{2} \sum_{k=0}^N \text{tr}(Q\Sigma_k) = J_{LQG}(x_0, \mathbf{u})$  with  $\frac{1}{2} \sum_{k=0}^N \text{tr}(Q\Sigma_k) \in \mathbb{R}$  a constant depending only on  $x_0$ . Thus we can use Theorem 8.1 to prove that we only need to solve for the optimal controller input  $\mathbf{u}$  using the LQR objective function to solve (8.11).  $\square$

As we have mentioned before, the separation theorem (sometimes called certainty equivalence) implies that the solution of the LQG control problem is achieved by using the Kalman filter to optimally estimate the current state and then using that state estimate in the optimal LQR controller. It is reassuring that Theorem 8.3 is confirmed by this result. The primary benefit of the graphical model approach is clear: we have solved the LQG problem without resorting to stochastic dynamical programming.

Under some circumstances it is also possible to extend the result of Theorem 8.3 to the infinite horizon case as shown in Theorem 8.4.

**Theorem 8.4. Solution of the infinite horizon LQG control problem** If the linear model of (8.2) is stable then, using, with some minor adjustments, Theorems 8.1 and 8.2 it is possible to show that the infinite horizon LQG problem is solved in a similar manner: the Kalman filter state estimate is used in conjunction with the infinite horizon LQR solution. This result can also be obtained by using the separation theorem.

To clarify why it is important that the linear system (i.e. the matrix  $A$ ) is stable, consider the quantity  $\frac{1}{2} \sum_{k=0}^N \text{tr}(Q\Sigma_k)$ . If it is unbounded the optimisation problem will be ill posed because the minimum will tend to infinity. Inspecting  $\Sigma_{t+1} = W + A\Sigma_t A^T$  we see that  $\|\Sigma_\infty\|$  will be unbounded if  $\|A\Sigma_t A^T\|$  becomes unbounded ( $W$  is a constant) as  $t \rightarrow \infty$ . Note that  $\|\cdot\|$  is some matrix norm. It can be shown that only if the eigenvalues of  $A$  are less than unity i.e. the linear model is stable, then  $\|A\Sigma_{t+1} A^T\| \leq \|A\Sigma_t A^T\|$  which implies that  $\frac{1}{2} \sum_{k=0}^N \text{tr}(Q\Sigma_k)$  is bounded and the optimisation is reasonable.

## 8.2 Constrained stochastic control

The goal of this chapter is to solve the stochastically constrained MPC optimisation problem shown in (8.12). We assume that the underlying system is linear and the probability distributions are Gaussian. From the results of Chapter 6 the probability distributions will be Gaussian if the system dynamics are linear. However, it is well known [39] that MPC is not in general a linear controller. From an analytical point of view this is problematic. We assume that the nonlinearity introduced by the MPC is negligible. We also restrict our analysis to affine constraints. Note that we only include one constraint in the succeeding analysis however, additional constraints are handled in exactly the same way as we will show.

Furthermore, we assume that the current state estimate  $x_0$  is supplied by some observer.

$$\begin{aligned} \min_{\mathbf{u}} \mathbb{E} & \left[ \frac{1}{2} \sum_{k=0}^{N-1} (x_k^T Q x_k + u_k^T R u_k) + \frac{1}{2} x_N^T P_f x_N \right] \\ \text{subject to } & x_{t+1} = Ax_t + Bu_t + w_t \\ \text{and } & \mathbb{E}[d^T x_t + e] \geq 0 \quad \forall t = 1, \dots, N \\ \text{and } & \Pr(d^T x_t + e \geq 0) \geq p \quad \forall t = 1, \dots, N \end{aligned} \tag{8.12}$$

It might seem that the last constraint is a duplicate of the preceding one. Closer inspection reveals their different character. The first inequality constraint requires that the predicted states satisfy the constraint “on average” while the second inequality constraint requires that the predicted states jointly satisfy the constraint with at least some probability  $p$ .

For navigational convenience we supply a link to Theorem 8.8 - when the reader reaches that point the reason will become obvious.

Theorem 8.5 succinctly shows that it is simple to convert the expected value constraint in (8.12) to a linear deterministic constraint.

**Theorem 8.5. Affine expected value constraints** Suppose we have a stochastic variable  $x$  with a known Gaussian distribution. Then the stochastic constraint  $\mathbb{E}[d^T x + e] \geq 0$  simplifies to the deterministic constraint  $d^T \mu + e \geq 0$  (with  $d \in \mathbb{R}^n$  and  $e \in \mathbb{R}$ ) where  $\mathbb{E}[x] = \mu$  is the mean of the stochastic variable.

Furthermore, suppose we have the set of stochastic constraints  $\mathbb{E}[Dx + e] \geq 0$  (with  $D \in \mathbb{R}^{n \times m}$  and  $e \in \mathbb{R}^n$ ). Then  $\mathbb{E}[Dx + e] \geq 0$  simplifies to  $D\mu + e \geq 0$ .

*Proof.* We know that  $x$  is a Gaussian stochastic variable. By Theorem 3.5 we know that  $\mathbb{E}[d^T x + e] = d^T \mu + e$  and  $\mathbb{E}[Dx + e] = D\mu + e$ . This immediately implies the Theorem.  $\square$

It is interesting to pause here for a moment and consider Theorem 8.3 and Theorem 8.5 applied to a problem of the form (8.13). We assume that the current state estimate is available at each time step and that  $\mathbb{E}[x_0] = \mu_0$ .

$$\begin{aligned} \min_{\mathbf{u}} \mathbb{E} & \left[ \frac{1}{2} \sum_{k=0}^{N-1} (x_k^T Q x_k + u_k^T R u_k) + \frac{1}{2} x_N^T P_f x_N \right] \\ \text{subject to } & x_{t+1} = Ax_t + Bu_t + w_t \\ \text{and } & \mathbb{E}[d^T x_t + e] \geq 0 \quad \forall t = 1, \dots, N \end{aligned} \tag{8.13}$$

It is clear that by applying Theorems 8.3 and 8.5 it is possible to rewrite (8.13) as (8.14).

$$\begin{aligned} \min_{\mathbf{u}} & \frac{1}{2} \sum_{k=0}^{N-1} (\mu_k^T Q \mu_k + u_k^T R u_k) + \frac{1}{2} \mu_N^T P_f \mu_N + \frac{1}{2} \sum_{k=0}^N \text{tr}(Q \Sigma_k) \\ \text{subject to } & \mu_{t+1} = A\mu_t + Bu_t \\ \text{and } & d^T \mu_t + e \geq 0 \quad \forall t = 1, \dots, N \end{aligned} \tag{8.14}$$

This implies that the standard deterministic MPC problem (8.14) is equivalent to the stochastic MPC problem with affine expected value constraints (8.13) under the assumptions of linearity and normality. This suggests that if chance constraints are not required the standard deterministic MPC will be sufficient for the control of stochastic processes. The generalisation of (8.13) to multiple expected value constraints is straightforward due to the second part of Theorem 8.5.

Theorem 8.6 is a necessary step before we can convert the chance constraint of (8.12) into a nonlinear deterministic constraint.

**Theorem 8.6. Shortest squared Mahalanobis distance between a hyperplane and a point** Suppose we are given a symmetric positive semi-definite matrix  $S$  and a point  $y$ . The shortest squared Mahalanobis distance between  $y$  and the hyperplane  $b^T x + c = 0$  is given by  $\frac{(b^T y + c)^2}{b^T S b}$ .

*Proof.* It is natural to formulate Theorem 8.6 as an optimisation problem as shown in (8.15).

$$\begin{aligned} \min_x \quad & (x - y)^T S^{-1} (x - y) \\ \text{subject to } & b^T x + c = 0 \end{aligned} \tag{8.15}$$

Note that  $S$  and therefore also  $S^{-1}$  is symmetric. Using conventional calculus we have  $\nabla f(x) = (S^{-1} + S^{-1}{}^T)x - 2S^{-1}y = 2S^{-1}x - 2S^{-1}y$  and  $\nabla g(x) = b^T$ . Using the method of Lagrangian multipliers [25] we have the system of equations (8.16)

$$\begin{aligned} 2S^{-1}x - 2S^{-1}y + \lambda b &= 0 \\ b^T x + c &= 0 \end{aligned} \tag{8.16}$$

This can be rewritten in block matrix form as shown in (8.17).

$$\begin{pmatrix} 2S^{-1} & b \\ b^T & 0 \end{pmatrix} \begin{pmatrix} x \\ \lambda \end{pmatrix} = \begin{pmatrix} 2S^{-1}y \\ -c \end{pmatrix} \tag{8.17}$$

The special structure of the left hand side matrix in (8.17) allows us to analytically compute the inverse (see Theorem 3.13 in Chapter 3.5) as shown in (8.18).

$$\begin{pmatrix} 2S^{-1} & b \\ b^T & 0 \end{pmatrix}^{-1} = \begin{pmatrix} \frac{1}{2}S(I - \frac{bb^T S}{b^T S b}) & \frac{Sb}{b^T S b} \\ \frac{b^T S}{b^T S b} & -\frac{2}{b^T S b} \end{pmatrix} \tag{8.18}$$

To find the arguments which satisfy (8.16) we solve (8.19) which is equivalent to solving the system of linear equations in (8.17).

$$\begin{pmatrix} \frac{1}{2}S(I - \frac{bb^T S}{b^T S b}) & \frac{Sb}{b^T S b} \\ \frac{b^T S}{b^T S b} & -\frac{2}{b^T S b} \end{pmatrix} \begin{pmatrix} 2S^{-1}y \\ -c \end{pmatrix} = \begin{pmatrix} S(I - \frac{bb^T S}{b^T S b})S^{-1}y - c \frac{Sb}{b^T S b} \\ 2(\frac{b^T S}{b^T S b} + \frac{c}{b^T S b}) \end{pmatrix} \tag{8.19}$$

Therefore, the arguments which minimise (8.15) are  $x^* = S(I - \frac{bb^T S}{b^T S b})S^{-1}y - c\frac{Sb}{b^T S b}$ . Substituting this into the objective function we have (8.20).

$$\begin{aligned}
& (x^* - y)^T S^{-1} (x^* - y) \\
&= \left( S \left( I - \frac{bb^T S}{b^T S b} \right) S^{-1} y - c \frac{Sb}{b^T S b} - y \right)^T S^{-1} \left( S \left( I - \frac{bb^T S}{b^T S b} \right) S^{-1} y - c \frac{Sb}{b^T S b} - y \right) \\
&= \left( \frac{Sbb^T y}{b^T S b} + c \frac{Sb}{b^T S b} \right)^T S^{-1} \left( \frac{Sbb^T y}{b^T S b} + c \frac{Sb}{b^T S b} \right) \\
&= \frac{(Sb)^T}{b^T S b} (b^T y + c)^T S^{-1} \frac{Sb}{b^T S b} (b^T y + c) \\
&= \frac{b^T S}{b^T S b} (b^T y + c)^T \frac{b}{b^T S b} (b^T y + c) \\
&= (b^T y + c)^T \frac{b^T S b}{(b^T S b)^2} (b^T y + c) \\
&= \frac{(b^T y + c)^T (b^T y + c)}{b^T S b} \\
&= \frac{(b^T y + c)^2}{b^T S b}
\end{aligned} \tag{8.20}$$

We can conclude that the shortest squared Mahalanobis distance between a point  $y$  and the constraint of (8.15) is  $\frac{(b^T y + c)^2}{b^T S b}$ .  $\square$

In Theorem 8.7 we apply Theorem 8.6 to convert the chance constraints into nonlinear deterministic constraints.

**Theorem 8.7. Gaussian affine chance constraints** Suppose the underlying distribution of a random variable  $x$  is Gaussian with sufficient statistics  $(\mu, \Sigma)$  and we have that  $d \in \mathbb{R}^n$  and  $e \in \mathbb{R}$ . Also, suppose that  $d^T \mu + e > 0$  is ensured.

Then the chance constraint  $\Pr(d^T x + e \geq 0) \geq p$  can be rewritten as the deterministic constraint  $\frac{(d^T \mu + e)^2}{d^T \Sigma d} \geq k^2$  where  $k^2$  is the (constant) critical value of the inverse cumulative Chi Squared distribution with the degrees of freedom equal to the dimensionality of  $x$  such that  $\Pr(\mathcal{X} \leq k^2) = p$  with  $\mathcal{X}$  a Chi Squared random variable. Note that  $p > 0.5$  due to the fact that  $d^T \mu + e > 0$  is ensured.

Furthermore, suppose  $D \in \mathbb{R}^{n \times m}$ ,  $e \in \mathbb{R}^n$  and  $D\mu + e > 0$  which, again, implies that  $p > 0.5$ . Then joint chance constraint  $\Pr(Dx + e \geq 0) \geq p$  can be written  $\frac{(d_i^T \mu + e_i)^2}{d_i^T \Sigma d_i} \geq k^2$  for all  $i = 1, \dots, n$ . Note  $d_i$  is each row in  $D$  and  $e_i$  each corresponding element in  $e$ .

*Proof.* Intuitively Theorem 8.7 posits that if the shortest squared Mahalanobis distance is further away than some threshold  $k^2$  the chance constraint  $\Pr(d^T x + e \geq 0) \geq p$  will be satisfied. Figure 8.2 depicts the idea that the minimum squared Mahalanobis distance must be greater than  $k^2$  i.e.  $\epsilon > 0$ . The use of the Mahalanobis distance is important because it takes into account the uncertainty associated with each component of the random variable  $x$ .

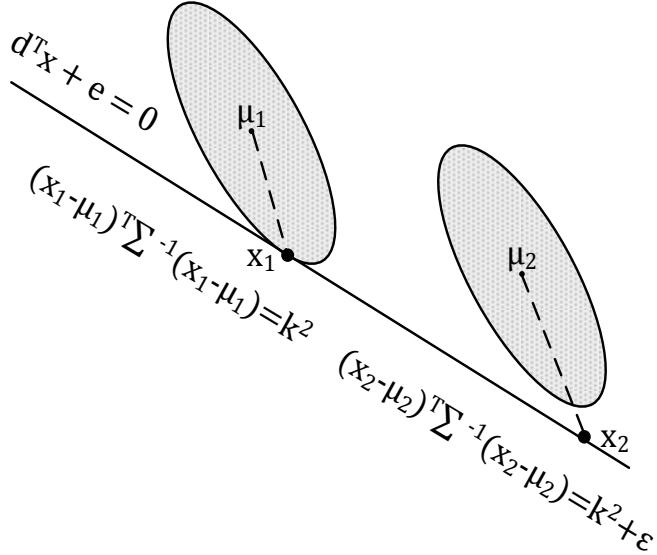


Figure 8.2: The ellipse centred around  $\mu_1$  intersects the constraint while the ellipse centred around  $\mu_2$  is wholly above (satisfies) the constraint. The shortest Mahalanobis distance will not necessarily be perpendicular to the constraint.

Since  $x$  is a Gaussian stochastic variable we have that  $\mathbb{E}[x] = \mu$  and  $\text{var}[x] = \Sigma$ . Let  $\Omega = \{x \in \mathbb{R}^n \mid (x - \mu)^T \Sigma^{-1} (x - \mu) \leq k^2\}$  and  $k^2$  be the critical value such that for the Chi Squared distribution with degrees of freedom equal to the dimension of  $x$  we have that  $\Pr(\mathcal{X} \leq k^2) = p$ . Then it is well known [49] that  $\int_{\Omega} p(x|\mu, \Sigma) dx = p$  where  $p(\cdot|\mu, \Sigma)$  is the multivariate Gaussian probability distribution function of  $x$ . If the shortest squared Mahalanobis distance from the mean of  $x$  is further away from the affine constraint  $d^T z + e = 0$  than  $k^2$  it implies that the curve of the function  $h(z) = (z - \mu)^T \Sigma^{-1} (z - \mu) = k^2$  does not intersect with the constraint - a positive  $\epsilon$  in the context of Figure 8.2. Therefore  $\Omega$  is wholly contained within the feasible region because  $d^T \mu + e > 0$ . This implies, with at least a probability of  $p$ , that the chance constraint will not be violated because the ‘‘confidence ellipse’’ is contained in the feasible region. Therefore, by applying Theorem 8.6 to find the shortest squared Mahalanobis distance we have that  $\frac{(d^T \mu + e)^2}{d^T \Sigma d} \geq k^2 \implies \Pr(d^T x + e \geq 0) \geq p$ .

The generalisation to more than one constraint, which should be jointly satisfied, is shown in Figure 8.3. It is required that the ellipse, up to confidence level  $p$ , is wholly contained in the feasible region  $Dx + e > 0$ . If this requirement is satisfied we will satisfy the joint chance constraint by the same reasoning as before.

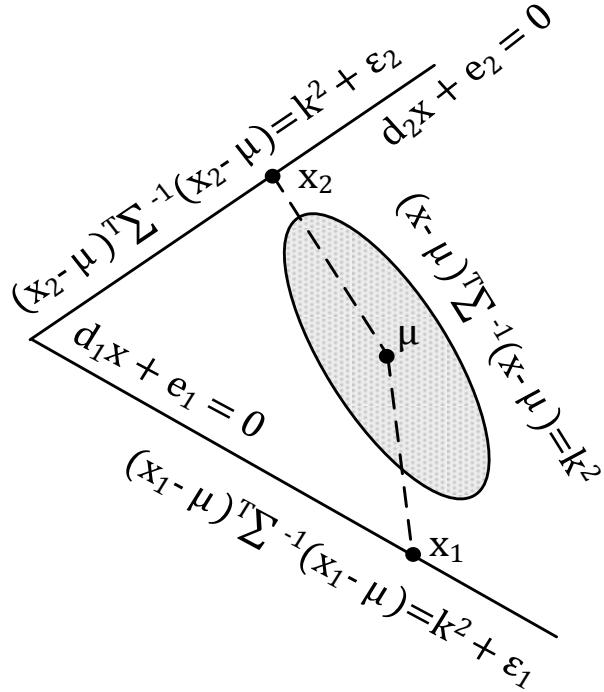


Figure 8.3: Illustration of the fact that Theorem 8.7 holds jointly for multiple constraints. The ellipse centred around  $\mu$  must lie wholly within the feasible region. This is guaranteed if the Mahalanobis constraint is satisfied.

By ensuring that  $D\mu + e > 0$  and that the shortest squared Mahalanobis distance between  $\mu$  and the constraints is bigger than  $k^2$  this is guaranteed. This implies that if for all  $i = 1, \dots, n$   $\frac{(d_i^T \mu + e_i)^2}{d_i^T \Sigma d_i} \geq k^2$  then  $\Pr(Dx + e \geq 0) \geq p$  will be satisfied.  $\square$

In Theorem 8.7 it is important that  $\mu$  lie within the feasible region. If that is not required then it is possible for the ellipse to be wholly outside the feasible region while still satisfying the constraint that the shortest squared Mahalanobis distance is at least  $k^2$ . Clearly the chance constraint will then not be satisfied.

It is interesting to note the striking similarity between the work in [55] and [54] and Theorem 8.7 given that we started analysing the problem within the framework of graphical models. The additional benefit of Theorem 8.7 is that it formulates the chance constraint as a function of the statistical Mahalanobis distance measure. This is useful because it lends a statistical interpretation to Theorem 8.7 even if the underlying distribution is non-Gaussian.

In Theorem 8.8 we combine and elaborate on the work of [59], [55] to yield a QP MPC which exactly satisfies the stochastic MPC in (8.12) given the assumptions of linearity and normality. Note that the dimensionality of the problem is arbitrary.

**Theorem 8.8. Conversion of the stochastic MPC formulation to the standard deterministic QP MPC formulation** Under the assumptions of linearity and Gaussian distributions we can reformulate the stochastic MPC problem shown in (8.12) as a standard

deterministic QP MPC problem shown in (8.21).

$$\begin{aligned}
& \min_{\mathbf{u}} \frac{1}{2} \sum_{k=0}^{N-1} (\mu_k^T Q \mu_k + u_k^T R u_k) + \frac{1}{2} \mu_N^T P_f \mu_N + \frac{1}{2} \sum_{k=0}^N \text{tr}(Q \Sigma_k) \\
& \text{subject to } \mu_{t+1} = A \mu_t + B u_t \\
& \text{and } \Sigma_{t+1} = W + A \Sigma_t A^T \\
& \text{and } d^T \mu_t + e \geq k \sqrt{d^T \Sigma_t d} \quad \forall t = 1, \dots, N
\end{aligned} \tag{8.21}$$

Other deterministic constraints, e.g. on the input, can be added as usual. Note that we have assumed that the initial state estimate  $x_0$  is available in the form of its mean,  $\mu_0$ , and covariance,  $\Sigma_0$ . It is straightforward to include more chance constraints. Since each chance constraint is reduced to an inequality constraint which measures the Mahalanobis distance between the predicted distributions, the resultant feasible points will jointly satisfy all chance constraints.

*Proof.* Let the admissible set of controller inputs  $\mathcal{U}$  be the same for both the stochastic MPC and the deterministic MPC formulations. Furthermore, let the current state estimate  $x_0$  be given. Then by Theorem 8.2 the objective function and equality constraints follow. By Theorem 8.5 the expected value inequality constraint in (8.12) can be reformulated to require that  $d^T \mu_t + e \geq 0$  for each  $t = 1, 2, \dots, N$ . The chance constraint in (8.12) follows from Theorem 8.7 as shown in (8.22).

$$\begin{aligned}
\frac{(d^T \mu_t + e)^2}{d^T \Sigma_t d} \geq k^2 & \implies (d^T \mu_t + e)^2 \geq k^2 d^T \Sigma_t d \\
& \implies d^T \mu_t + e \geq k \sqrt{d^T \Sigma_t d} \quad \forall t = 1, 2, \dots, N
\end{aligned} \tag{8.22}$$

The first line of (8.22) follows because  $\Sigma_t$  is positive semi-definite for all  $t = 1, 2, \dots, N$  and  $d \neq 0$  (otherwise it would not be a constraint), therefore it can be multiplied over the inequality sign like a positive number. By Theorem 8.3 we have that  $\Sigma_{t+1} = W + A \Sigma_t A^T$ , therefore the predicted covariance matrices used in (8.22) are well defined. The second line follows because of the first inequality constraint: we know that  $d^T \mu_t + e \geq 0 \quad \forall t = 1, 2, \dots, N$  and therefore we can square root both sides of the inequality constraint. Thus we have the two inequality constraints  $d^T \mu_t + e \geq 0$  and  $d^T \mu_t + e \geq k \sqrt{d^T \Sigma_t d}$  for each  $t = 1, 2, \dots, N$ . Since  $k > 0$  (otherwise we do not have a meaningful chance constraint) we can condense the two inequality constraints into a single constraint:  $d^T \mu_t + e \geq k \sqrt{d^T \Sigma_t d} > 0$  for each  $t = 1, 2, \dots, N$  from which the conclusion follows.  $\square$

The beauty of Theorem 8.8 is that no new theory is necessary to analyse the stability and convergence results of the new MPC. This is highly desirable because it allows one to merely ‘add’ the last inequality constraint to your existing MPC formulation. Most practical MPCs will have some form of state estimation and thus no new parameters are introduced either. Since the problem is in standard QP form it is straightforward to implement and, even more importantly, it is computationally fast because the problem is trivially convex.

In the work by [59] and [60] it is found that feasibility problems might arise if one uses the predicted covariance estimates in the controller. If the system is unstable the uncertainty in the estimates can grow with time as discussed in Theorem 8.4. This can cause the ellipses used in Theorem 8.7 to become too large to fit inside the feasible region. The approach adopted by [59] and [60] is to only use the one step ahead predicted covariance ( $\Sigma_1$ ) over the entire prediction horizon. This does not ensure feasibility but restricts the growth associated with infeasibility. The drawback of this approach is that it might cause constraint violation because the controller will be more aggressive.

### 8.3 Reference tracking

So far we have only dealt with controllers which drive the system to the origin. The more general situation we are interested in is arbitrary reference point tracking. Fortunately, Chapter 3.4.2 applies without modification because the Stochastic MPC problem was reduced to the *standard* deterministic MPC problem.

### 8.4 Linear system

In this chapter we consider the problem of controlling a linear system using the stochastic MPC formulation of Theorem 8.8. More precisely, we assume the linear model linearised about the unsteady operating point of our CSTR example from Chapter 5 accurately represents the underlying system. For the purposes of illustration we will only use the (somewhat unrealistic) system where both states are measured. However, there is no theoretical reason why we cannot use the system where only temperature is measured. The drawback of using the latter system is that inference, as discussed previously, will be worse. The control goal is to steer the concentration, in the sense of Definition 8.1, to the unsteady operating point ( $C_A = 0.49 \text{ kmol.m}^{-3}$ ) about which the system was linearised. See Chapter 5 for more details.

We repeat the relevant system dynamics in (8.23) using a time step  $h = 0.1$ .

$$A = \begin{pmatrix} 0.9959 & -6.0308 \times 10^{-5} \\ 0.4186 & 1.0100 \end{pmatrix} \quad B = \begin{pmatrix} 0 \\ 8.4102 \times 10^{-5} \end{pmatrix} \quad C = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

$$W = \begin{pmatrix} 1 \times 10^{-6} & 0 \\ 0 & 0.1 \end{pmatrix} \quad V = \begin{pmatrix} 1 \times 10^{-3} & 0 \\ 0 & 10 \end{pmatrix} \quad (8.23)$$

The tuning parameters used in this and all subsequent chapters are shown in (8.24). Note that the magnitude difference between the components of  $Q$ ,  $P_f$  and  $R$  is necessary because the units of concentration and heat input are not scaled.

$$Q = P_f = \begin{pmatrix} 1 \times 10^4 & 0 \\ 0 & 0 \end{pmatrix} \quad R = \begin{pmatrix} 1 \times 10^{-6} \end{pmatrix} \quad (8.24)$$

We assume that a Kalman filter supplies the initial state estimate  $x_0$  and that the mean  $\mu_0$  of that estimate is used for control. Additionally, we assume a zero order hold of 1 min between controller inputs.

It is useful to introduce the measures we will use to quantify the effectiveness of the control techniques used in this and the following chapters. We define the average energy input by  $\frac{1}{N} \sum_{t=0}^N |u_t - u_s|$  and the average concentration error by  $\frac{1}{N} \sum_{t=0}^N |\frac{C_{A,t} - y_{sp}}{y_{sp}}|$ . Additionally, to confirm that the results hold true over multiple simulations we will, at the end of Chapters 8.4 and 8.5, include a Monte-Carlo simulation. The metrics we use will be the time each simulation spent in violation of the constraints and the total state space area (measured within the context of the Mahalanobis distance) in violation of the constraint.

First we illustrate the approach of using only the result of Theorem 8.3 i.e. we apply the LQG regulator to the system. Based on our previous results we know that given the Kalman filter state estimate mean  $\mu_0$  we only need to solve the LQR as shown in (8.25) to solve the LQG problem.

$$\min_{\mathbf{u}} \frac{1}{2} \sum_{k=0}^{N-1} (\mu_k^T Q \mu_k + u_k^T R u_k) + \frac{1}{2} \mu_N^T P_f \mu_N \quad (8.25)$$

subject to  $\mu_{t+1} = A\mu_t + Bu_t$

Figure 8.4 shows that the system does indeed converge, noisily, to the set point. This is not surprising because we have effectively just implemented the very well studied LQG controller. The primary drawback of this method is that there is no easy way to constrain the system. From a practical perspective this can be problematic.

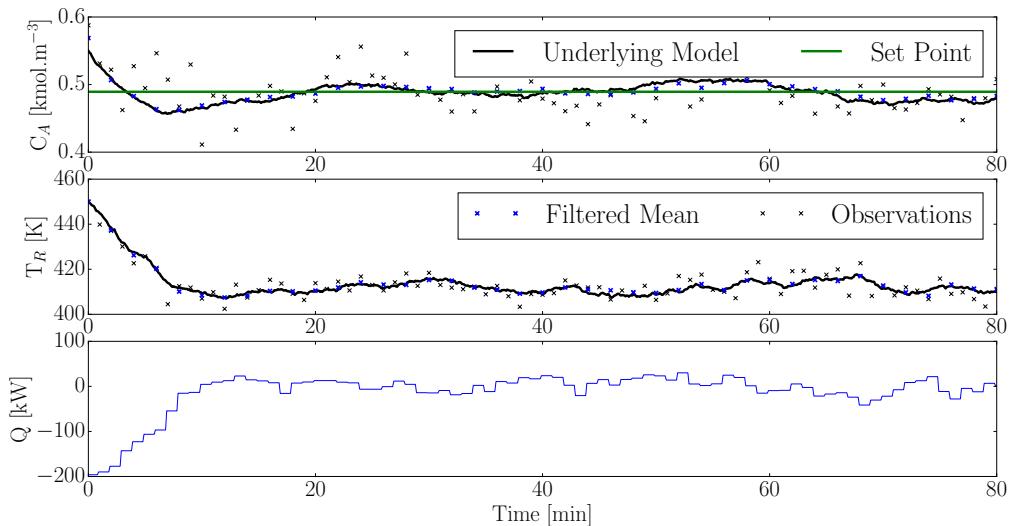


Figure 8.4: Unconstrained LQG regulator tracking with initial condition  $(0.55, 450)$  and measuring both states.

The average heat energy usage (controller input) over the simulation run is 25.29 kW. The

average set point error is 2.23% over the same 80 min time period.

Next we illustrate the approach of using conventional deterministic MPC to control the stochastic system. The MPC formulation is shown in (8.26). Using MPC allows us to easily add state and input constraints; this is a significant improvement over conventional LQG as discussed previously.

$$\begin{aligned}
 & \min_{\mathbf{u}} \frac{1}{2} \sum_{k=0}^{N-1} (\mu_k^T Q \mu_k + u_k^T R u_k) + \frac{1}{2} \mu_N^T P_f \mu_N \\
 & \text{subject to } \mu_{t+1} = A \mu_t + B u_t \\
 & \text{and } \begin{pmatrix} 10 \\ 1 \end{pmatrix}^T \mu_t + 412 \geq 0 \quad \forall t = 1, \dots, N \\
 & \text{and } |u_t| \leq 165 \quad \forall t = 0, \dots, N-1
 \end{aligned} \tag{8.26}$$

We only use a single state constraint in this dissertation but the extension to multiple constraints is straightforward. The prediction and control horizon are equal to each other and set at 150 time steps i.e. 15 minutes into the future. We additionally constrain the magnitude of the inputs.

Due to assumption of normality and linearity and by Theorem 8.3 and 8.5 we can also interpret the deterministic MPC as a stochastic MPC with an affine expected value constraint. Therefore the controller in (8.26) is not inappropriate for control of stochastic CSTR processes under consideration.

In Figure 8.5 we see the reference tracking and controller input for the deterministic MPC. The average heat energy input and set point error over the simulation run is 28.22 kW and 2.43% respectively. Interestingly the average error is not much different but the controller requires more energy to track the set point. This is reasonable because the additional constraints make problem (8.26) a harder problem than (8.25).

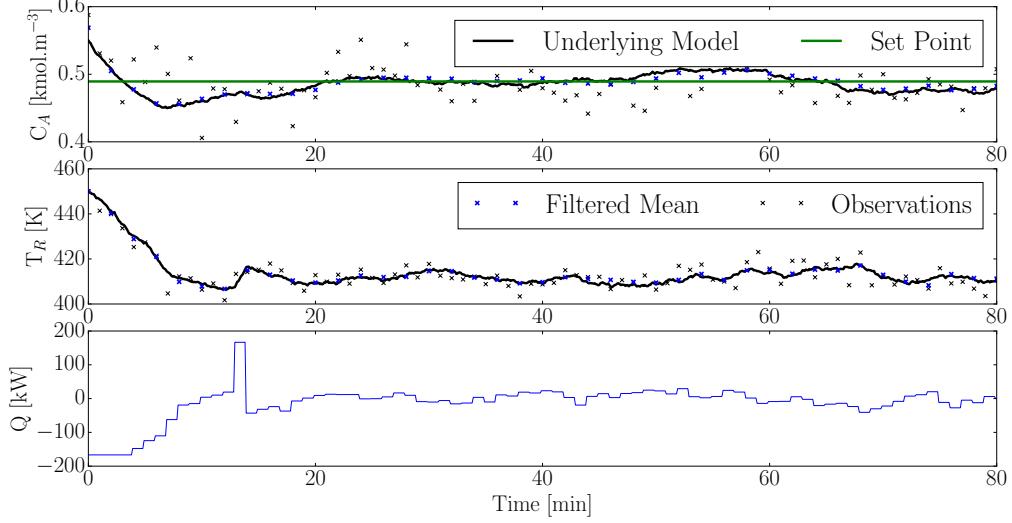


Figure 8.5: Deterministic constrained MPC tracking with initial condition  $(0.55, 450)$  and measuring both states.

Like the LQG controller, it is clear that we have noisy convergence to the set point. As mentioned previously we will never be able to achieve zero set point offset because of the noise term in the system dynamics (8.1). Note that we have constrained the maximum magnitude of the inputs such that  $|u_t| \leq 165$  kW. In the unconstrained case the controller required a maximum absolute input magnitude of over 200 kW; the ability to naturally constrain the input can be practically very useful. The benefit of MPC is apparent here.

In Figure 8.6 we see the corresponding state space trajectory of the system together with the state constraint.

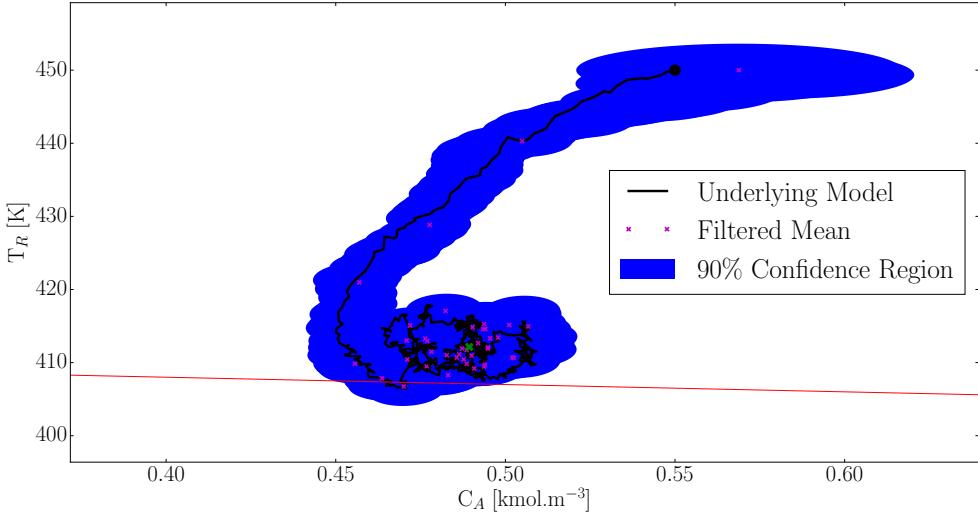


Figure 8.6: Deterministic MPC state space trajectory with initial condition  $(0.55, 450)$  and measuring both states.

While the predicted mean state estimates do not violate the constraint (due to the optimisation constraints) the actual underlying system does. This is clearly seen if one inspects the confidence region around the lower state estimates. The confidence region is deeply violated by the constraint which implies that it is likely that the underlying system might. This is clearly a problem from a control point of view; the deterministic MPC cannot ensure that the constraint is satisfied.

We remedy this situation by introducing the stochastic MPC as discussed in Theorem 8.8 and shown in (8.27) for convenience. Note that  $d^T = (10, 1)$  and  $e = 412$  as before. By consulting a Chi Squared distribution table we set  $k^2 = 4.6052$  which corresponds to the chance constraint  $\Pr(d^T x_t + e \geq 0) \geq 90\% \forall t = 1, \dots, N$ .

$$\begin{aligned} \min_{\mathbf{u}} \quad & \frac{1}{2} \sum_{k=0}^{N-1} (\mu_k^T Q \mu_k + u_k^T R u_k) + \frac{1}{2} \mu_N^T P_f \mu_N + \frac{1}{2} \sum_{k=0}^N \text{tr}(Q \Sigma_k) \\ \text{subject to } & \mu_{t+1} = A \mu_t + B u_t \\ \text{and } & \Sigma_{t+1} = W + A \Sigma_t A^T \\ \text{and } & d^T \mu_t + e \geq k \sqrt{d^T \Sigma_t d} \quad \forall t = 1, \dots, N \\ \text{and } & |u_t| \leq 165 \quad \forall t = 0, \dots, N-1 \end{aligned} \tag{8.27}$$

Note that problem (8.27) is harder than (8.26) due to the added constraint and thus we expect that the system will require greater controller input to satisfy the constraint.

In Figure 8.7 we see that the stochastic MPC is able to track the set point in a similar manner as the LQG controller and deterministic MPC. The total average heat input and set point error over the simulation run is 38.58 kW and 2.58% respectively. This problem is harder than the preceding one due to the additional constraint and thus more energy is required.

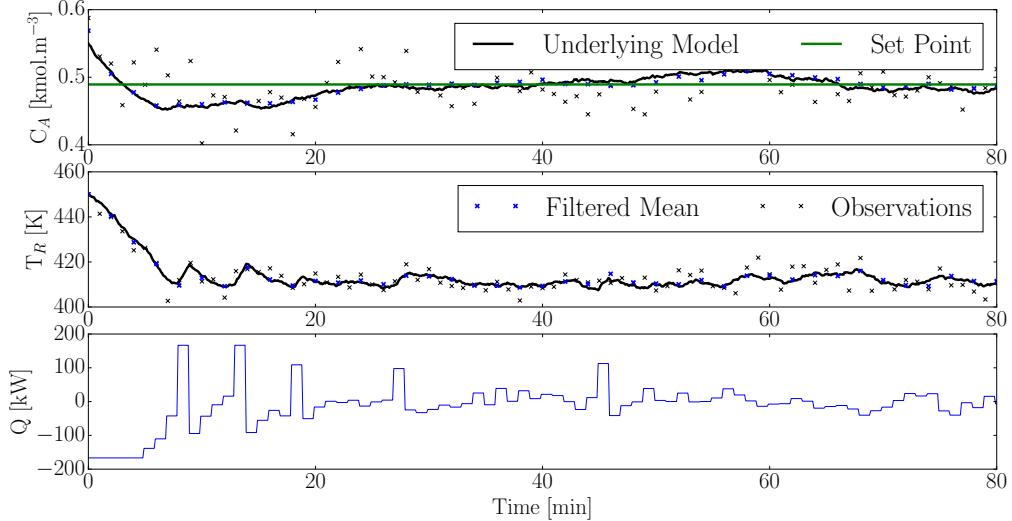


Figure 8.7: Chance constrained MPC tracking with initial condition  $(0.55, 450)$  and measuring both states. A Kalman filter is used for inference and the chance constraint is set at 90%.

However, the benefit of adding the chance constraint is apparent in Figure 8.8. It is clear that the constraint on the underlying state is not violated.

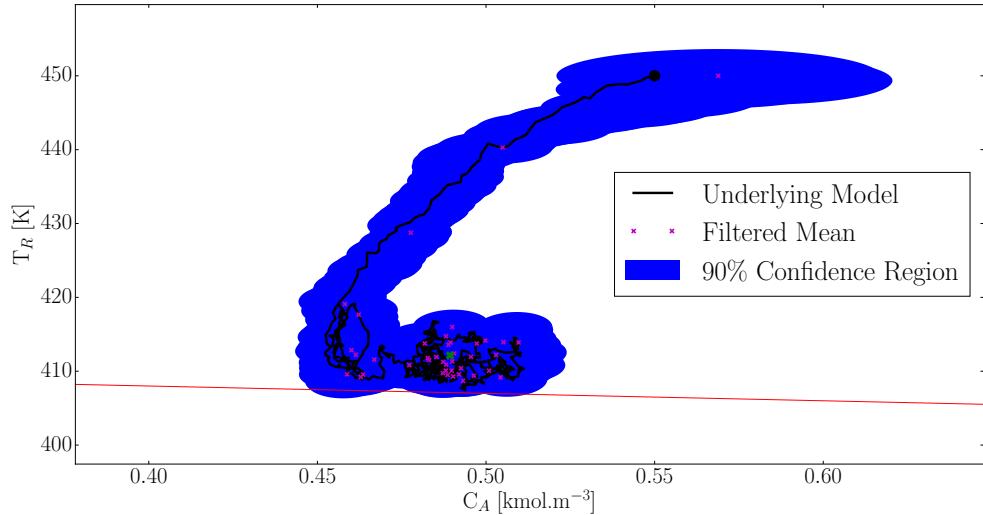


Figure 8.8: Chance constrained MPC state space trajectory with initial condition  $(0.55, 450)$  and measuring both states. A Kalman filter is used for inference and the chance constraint is set at 90%.

Since the chance constraint is only enforced with probability 90% it is possible that the underlying system can come “close” to the constraint. This then has the consequence that the posterior confidence region marginally violates (spills over) the constraint as seen in the

lower regions of Figure 8.8.

It is interesting to investigate what effect increasing the probability that the chance constraint is satisfied will have on the system. To this end we modify the chance constraint of (8.27) such that  $k^2 = 13.8155$  which corresponds to the chance constraint  $\Pr(d^T x_t + e \geq 0) \geq 99\% \forall t = 1, \dots, N$ . We expect that the underlying system will be moved further away from constraint due to this added level of conservativeness.

In Figure 8.9 we see that the stochastic MPC still tracks the set point and Figure 8.10 shows that the expected behaviour is realised.

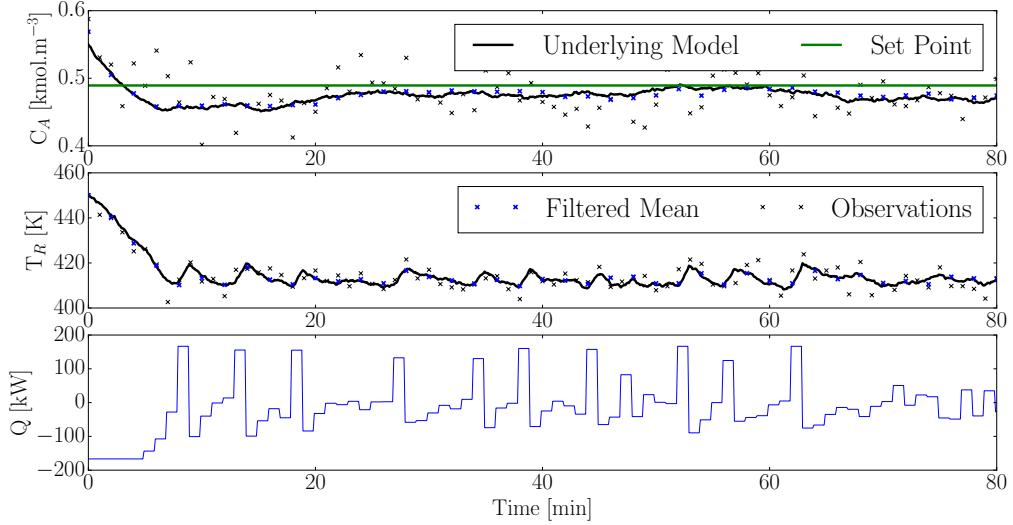


Figure 8.9: Chance constrained MPC tracking with initial condition  $(0.55, 450)$  and measuring both states. A Kalman filter is used for inference and the chance constraint is set at 99%.

The average heat input and average set point error is 3.49% and 58.39 kW. The added conservativeness of the MPC prevents it from attempting to get to the set point as fast as the previous stochastic MPC; consequently there is no set point overshoot in Figure 8.9. This causes the higher average error but the constraints are satisfied more robustly. As before, the control problem is harder and thus requires more energy.

In Figure 8.10 we see the 90% confidence region is above the constraint. Since the probability that the predicted states are close to the constraint is much lower than before we see that the confidence region satisfies the constraint everywhere.

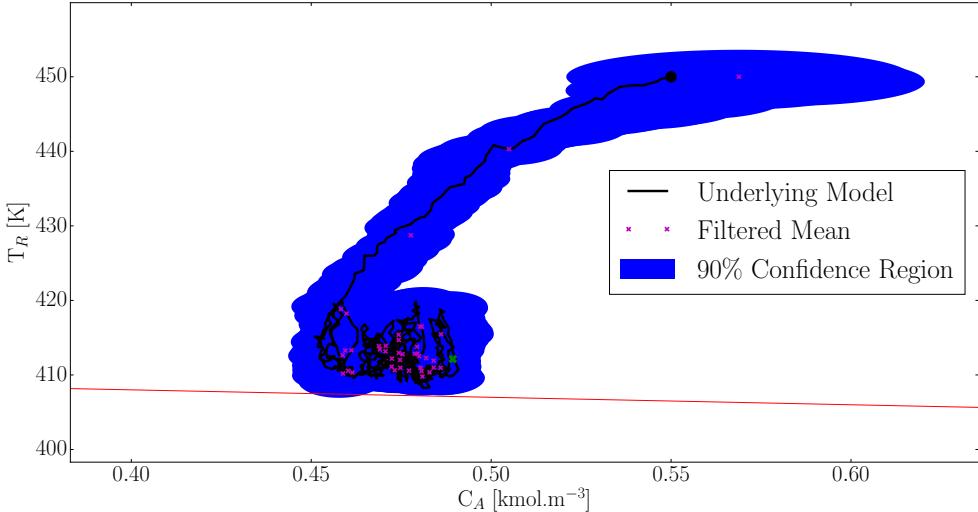


Figure 8.10: Chance constrained MPC state space trajectory with initial condition  $(0.55, 450)$  and measuring both states. A Kalman filter is used for inference and the chance constraint is set at 99%.

It would also be correct to infer that  $k$  can be used as an empirical measure of the inherent stochastic conservativeness of the resulting controller. That is, lower values of  $k$  indicate a more aggressive controller which may violate the chance constraints and higher values of  $k$  indicate a more conservative controller. This can be useful for systems where the normal assumption is not valid but one would still like to enforce chance constraints in some empirical sense.

We have made the strong assumption that the system dynamics remain linear and Gaussian even under the MPC control law which is not necessarily linear [39]. Clearly if the system is far from Gaussian the Gaussian approach to simplifying the chance constraint will not be valid. Fortunately this is relatively simple to check and serves as a good way of measuring controller health. That is, the more Gaussian the filtered distributions are, the better the linear stochastic controller will work.

Kullback-Leibler Divergence was introduced in Theorem 3.8 to estimate the degree to which samples match a given distribution. From Chapter 7 we know that given enough particles a particle filter can accurately represent any distribution. Thus we temporarily replace the Kalman filter with a particle filter and use Theorem 3.8 to estimate the degree of normality of the posterior state distributions.

Figure 8.11 shows the degree to which the underlying distribution is Gaussian. Since we cannot use an infinite amount of particles we compare the sampled Gaussian distribution approximation to a baseline.

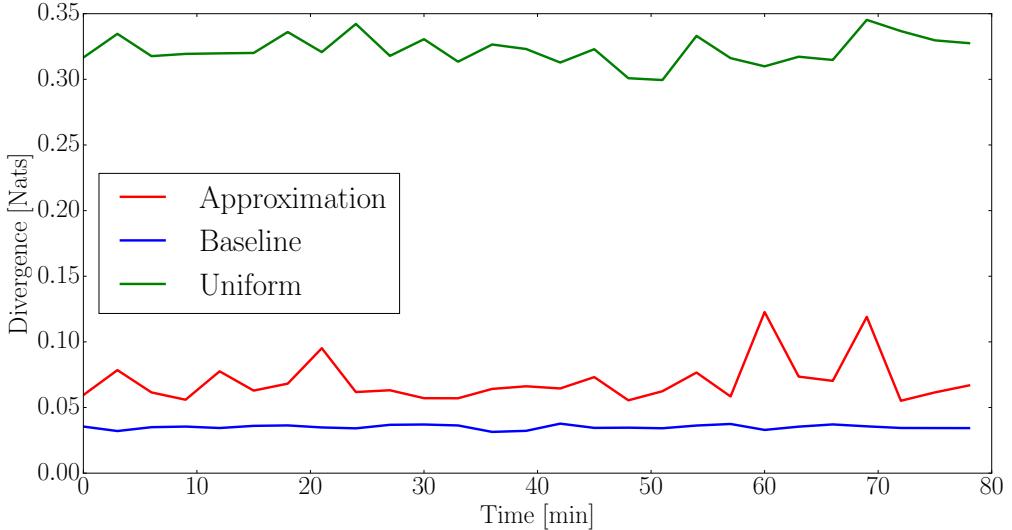


Figure 8.11: Kullback-Leibler Divergence between the assumed Gaussian distribution and different sampled distributions using 5000 particles. The underlying model is linear.

The approximation curve in Figure 8.11 shows how much the samples diverge from the Gaussian distribution approximated using the samples. The baseline curve shows how much the Gaussian distribution diverges from samples of the same distribution. The uniform curve shows how much a Gaussian approximation of a Uniform distribution drawn in the interval  $(\mu_i - 2\sigma_i, \mu_i + 2\sigma_i)$  (for each  $i$  in the dimension of the underlying distribution) diverges; this serves to illustrate the divergence one would expect if attempting to model a distribution which is decidedly not normal. One would expect the baseline curve to tend to zero as the number of particles tends to infinity. Sampling error causes divergence from zero for the baseline curve. Thus we can use the baseline and uniform curves as a crude measure of normality.

In Figure 8.11 we see that the approximation is relatively close to the baseline. Additionally it is far removed from the uniform curve. The average divergence for the baseline, approximation and uniform curve (in nats) is: 0.035, 0.069 and 0.322 respectively. This implies that even though we are using a non-linear control technique the posterior state distributions are still approximately Gaussian.

Finally, in Figure 8.12 we investigate the effect  $k^2$  has on the chance constraint. We expect that by increasing  $k^2$  the system becomes more conservative and less likely to violate the constraint. Under the assumptions of linearity and normality we are assured of this behaviour due to Theorem 8.8. However, as Figure 8.11 indicates our assumptions do not strictly hold. In Figures 8.8 and 8.10 we reached the conclusion that the chance constraint was effective by only considering a single simulation. In Figure 8.12 we show the compilation of over 2000 runs to illustrate the veracity of the claim that the chance constraints are effective.

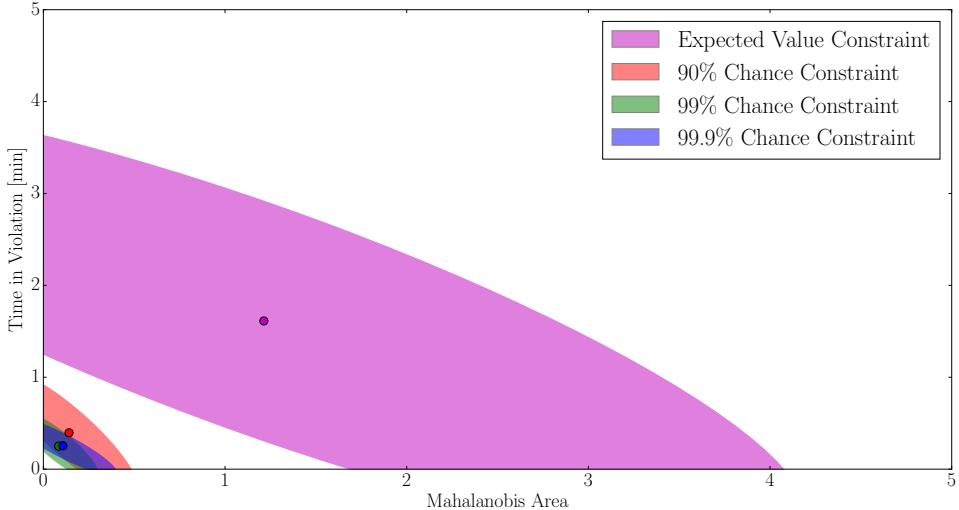


Figure 8.12: Monte-Carlo simulation to investigate the effect increasing  $k^2$  has on the constraint violation characteristics of the MPC controllers. Each region indicates where 90% of the simulations scored. The mean is indicated by a solid point.

The Mahalanobis area dimension indicates the degree to which the constraint was violated: larger numbers imply the constraint was deeply violated in state space. The time in violation dimension indicates the length of time the system violated the constraint. It is clear that the chance constraints dramatically reduce the violation characteristics of the system.

Interestingly, there seems to be diminishing returns on increasing  $k^2$  beyond the 99% chance interval. Feasibility issues plague the system under the more conservative chance constraint levels. Intuitively the predicted ellipses become too big to fit inside the feasible region - especially when projected forward in time. Given that the controller input - the only way to move them inside the feasible region - is constrained the system reaches a performance deadlock. In the work by [59] they also noted this problem and solved it by not letting the confidence ellipses grow as they were projected into the future (see the discussion after Theorem 8.4 and in Chapter 2.2). For this system the issue is not severe enough to warrant further concern.

## 8.5 Nonlinear system

In this chapter we consider the problem of controlling the full non-linear system with a linear model linearised around the unsteady operating point. The control goal is the same as before; the only difference between this chapter and chapter 8.4 is that the underlying plant is non-linear.

The linear control model and noise parameters are the same as (8.23). The control tuning

parameters are the same as (8.24).

As before we first investigate the LQG controller. Figure 8.13 shows the unconstrained reference tracking results.

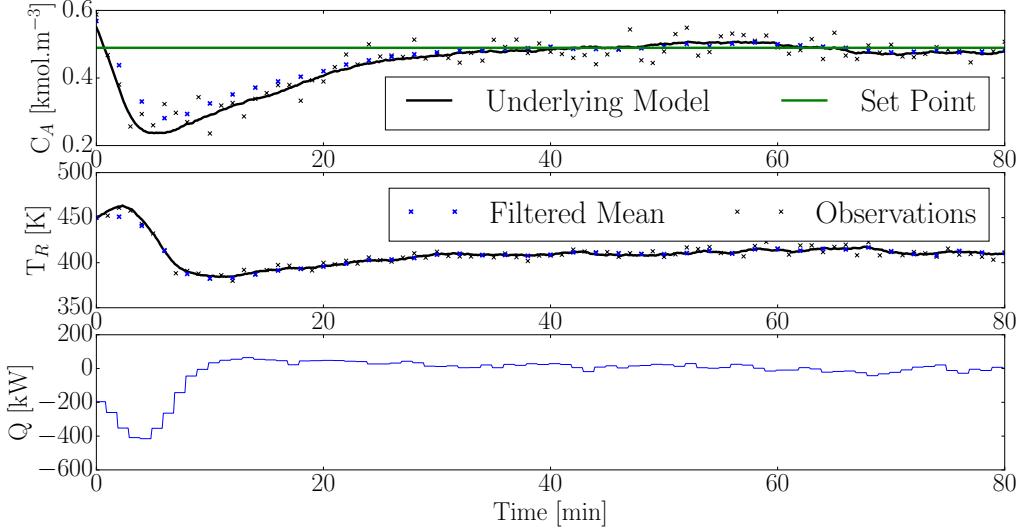


Figure 8.13: LQG regulator tracking with initial condition  $(0.55, 450)$  and measuring both states.

The average energy usage and concentration error was 50.27 kW and 10.94% respectively over the 80 min simulation time. Comparing Figures 8.4 and 8.13 we see that the maximum absolute input energy is much greater with the non-linear underlying dynamics. This is not unexpected because the controller in both cases is linear: one expects that the plant-model mismatch to have a detrimental effect on control.

In the previous chapter we had a linear underlying model and linear control. Figure 8.11 also demonstrated that the posterior state distributions were approximately Gaussian. Thus there was no reason to use non-linear inference algorithms like the particle filter introduced in Chapter 7. However, in this chapter we are using a non-linear underlying model and it might be advantageous to use a more sophisticated inference tool. We investigate using both a Kalman filter and a particle filter for inference. In the setting of the particle filter we approximate the samples as Gaussian and use that for control.

As before we first investigate the deterministic MPC. The control problem is shown in (8.28). Note that the constraints are different due to the expected extra difficulty introduced by the

non-linear underlying model.

$$\begin{aligned} \min_{\mathbf{u}} \frac{1}{2} \sum_{k=0}^{N-1} (\mu_k^T Q \mu_k + u_k^T R u_k) + \frac{1}{2} \mu_N^T P_f \mu_N \\ \text{subject to } \mu_{t+1} = A \mu_t + B u_t \\ \text{and } \begin{pmatrix} 10 \\ 1 \end{pmatrix}^T \mu_t + 406 \geq 0 \quad \forall t = 1, \dots, N \\ \text{and } |u_t| \leq 330 \quad \forall t = 0, \dots, N-1 \end{aligned} \tag{8.28}$$

In Figure 8.14 we see that the deterministic MPC using the Kalman filter for state inference does converge to the set point. The ability to naturally constrain the system is again highlighted in the input: as opposed to the peak  $|420|$  kW required by the LQG controller the MPC manages to control the system while never requiring more than  $|330|$  kW.

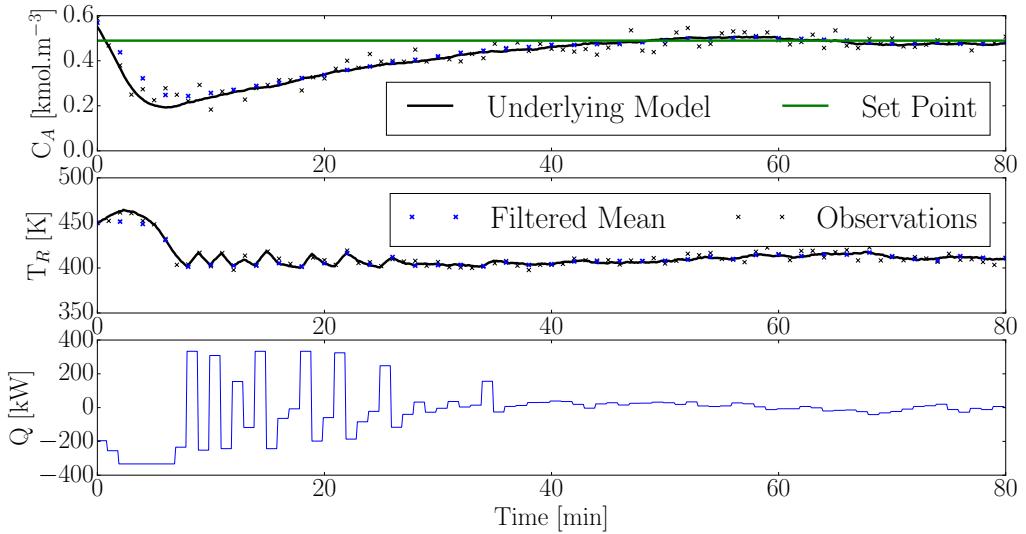


Figure 8.14: Deterministic constrained MPC reference tracking with initial condition  $(0.55, 450)$  and measuring both states. The Kalman filter is used for inference.

In Figure 8.15 we see that the state constraint is violated just like Figure 8.6. We also see a somewhat unrealistic jagged state trajectory but this is just a numerical artefact. The average energy input and concentration error over the simulation run is 88.68 kW and 16.11% respectively. The added constraints explain why the performance is degraded when compared to the LQG controller.

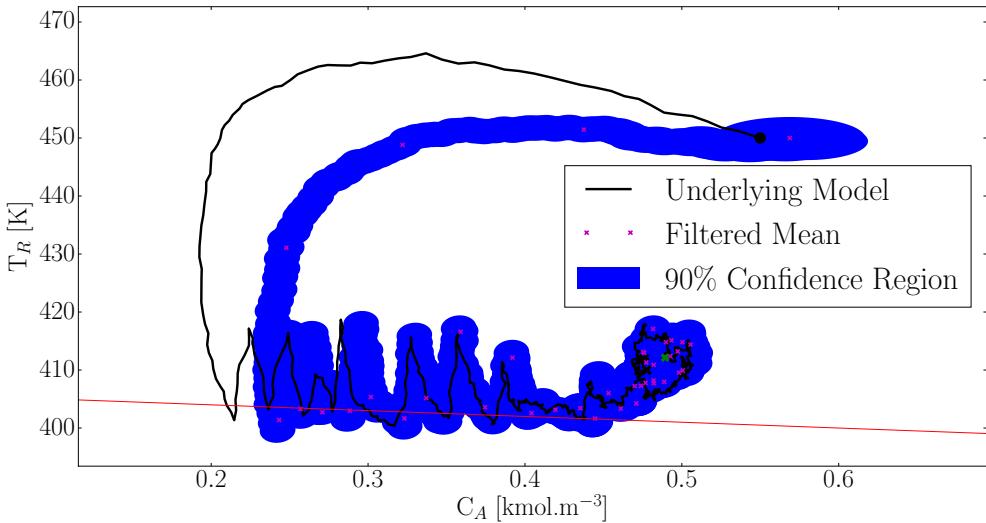


Figure 8.15: Deterministic constrained MPC state space trajectory with initial condition  $(0.55, 450)$  and measuring both states. The Kalman filter is used for inference.

Since we are not using a stochastic MPC the state constraint violation is not surprising in Figure 8.15. However, a more significant issue is the inability of the Kalman filter to accurately track the states throughout the simulation (the underlying system briefly diverges from the state estimates). This can be significantly problematic if a constraint existed in the left hand side of the state space: the controller wouldn't know that it was violating the constraint because the state estimate is poor. This behaviour is caused by the linear model used by the Kalman filter. The state trajectory moves away from the region close to the linearisation point and thus, as explained in Chapter 6, the state estimate becomes poor.

We can remedy this situation by using a more sophisticated inference algorithm. In Figure 8.16 we see the deterministic MPC using a particle filter with 200 particles for inference.

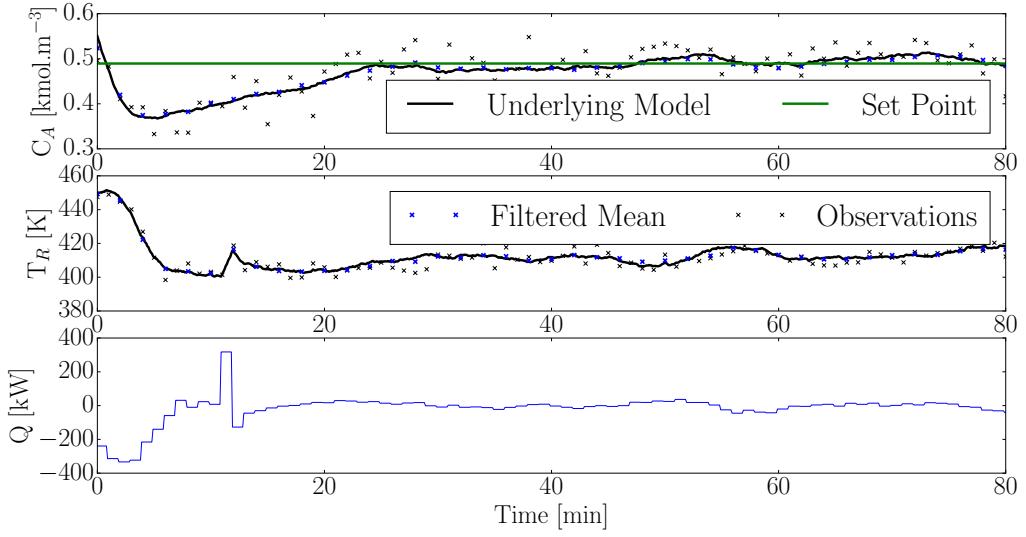


Figure 8.16: Deterministic constrained MPC reference tracking with initial condition  $(0.55, 450)$  and measuring both states. A particle filter with 200 particles is used for inference.

The average energy input and concentration error is 38.83 kW and 5.81% respectively. This is a vast improvement over the same controller where the Kalman filter was used for inference. The benefit of accurate state estimation is apparent here and also in Figure 8.17.

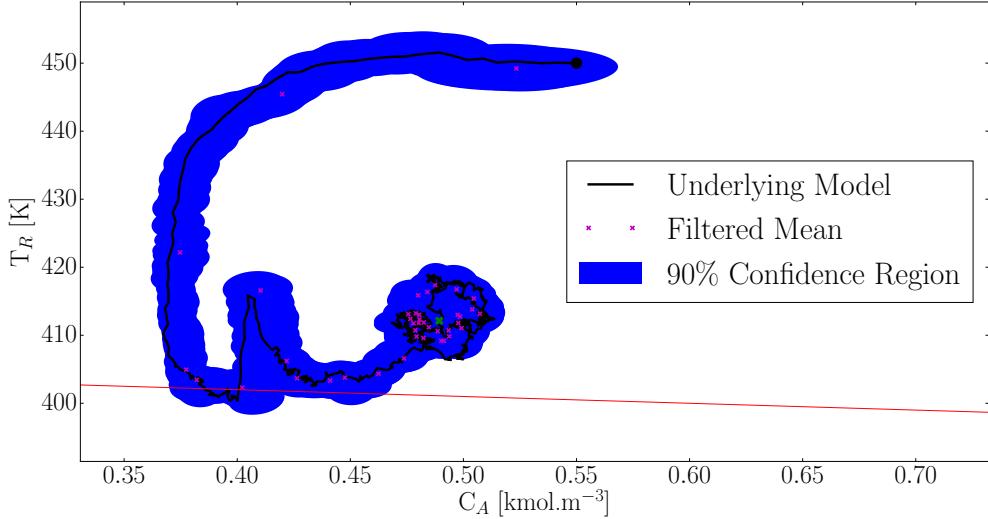


Figure 8.17: Deterministic constrained MPC state space trajectory with initial condition  $(0.55, 450)$  and measuring both states. A particle filter with 200 particles is used for inference.

In Figure 8.15 we saw significant estimation deviation from the true underlying state, while in Figure 8.17 the deviation is negligible. We still have that the state constraint is violated but

this is due to the stochastic nature of the underlying system. It is clear that the particle filter MPC combination is superior to the Kalman filter MPC combination in this case. However, the benefit of using the particle filter should be weighed against the cost of the algorithm especially in higher dimensions where it is known that the particle filter does not perform well (recall the discussion following Figure 7.7).

Now we introduce the chance constrained MPC in (8.29). Note that the constraints are different than (8.27) for the same reason as (8.28). We have  $d^T = (10, 1)$  and  $e = 406$  as before. By consulting the Chi Squared distribution table we set  $k^2 = 4.6052$  which corresponds to the chance constraint  $\Pr(d^T x_t + e \geq 0) \geq 90\% \forall t = 1, \dots, N$  exactly as in the previous chapter.

$$\begin{aligned} \min_{\mathbf{u}} \quad & \frac{1}{2} \sum_{k=0}^{N-1} (\mu_k^T Q \mu_k + u_k^T R u_k) + \frac{1}{2} \mu_N^T P_f \mu_N + \frac{1}{2} \sum_{k=0}^N \text{tr}(Q \Sigma_k) \\ \text{subject to } \quad & \mu_{t+1} = A \mu_t + B u_t \\ \text{and } \Sigma_{t+1} = & W + A \Sigma_t A^T \\ \text{and } d^T \mu_t + e \geq & k \sqrt{d^T \Sigma_t d} \quad \forall t = 1, \dots, N \\ \text{and } |u_t| \leq & 330 \quad \forall t = 0, \dots, N-1 \end{aligned} \tag{8.29}$$

As with the deterministic case we first investigate the system where a Kalman filter is used for inference. Figure 8.18 illustrates that the chance constrained system does indeed converge to the set point. The average energy input and concentration error is 80.75 kW and 14.12%. The average energy usage and average concentration error is reduced compared to the deterministic system.

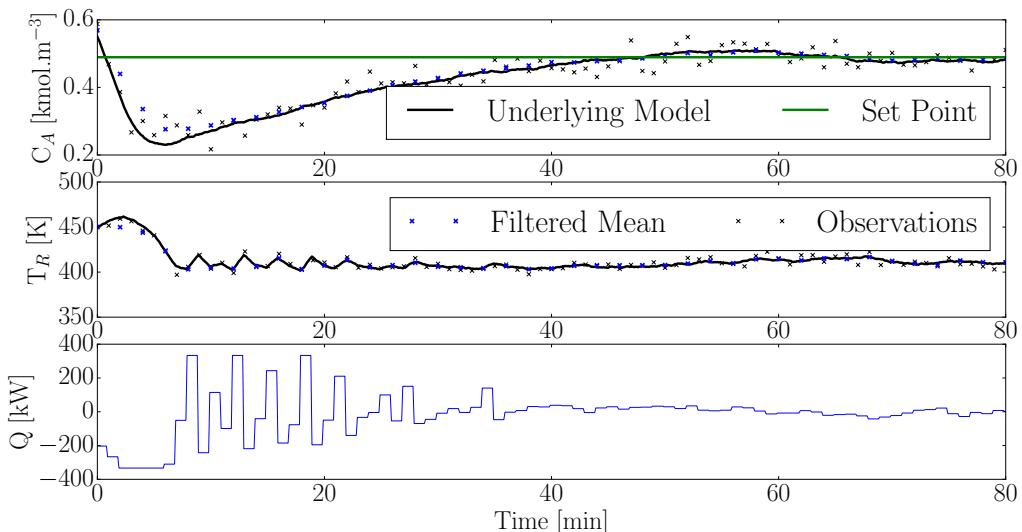


Figure 8.18: Chance constrained MPC tracking with initial condition  $(0.55, 450)$  and measuring both states. A Kalman filter is used for inference and the chance constraint is set at 90%.

The benefit of adding the chance constraint is apparent in Figure 8.19. It is clear that the constraint on the underlying state is not violated as much as in Figure 8.15.

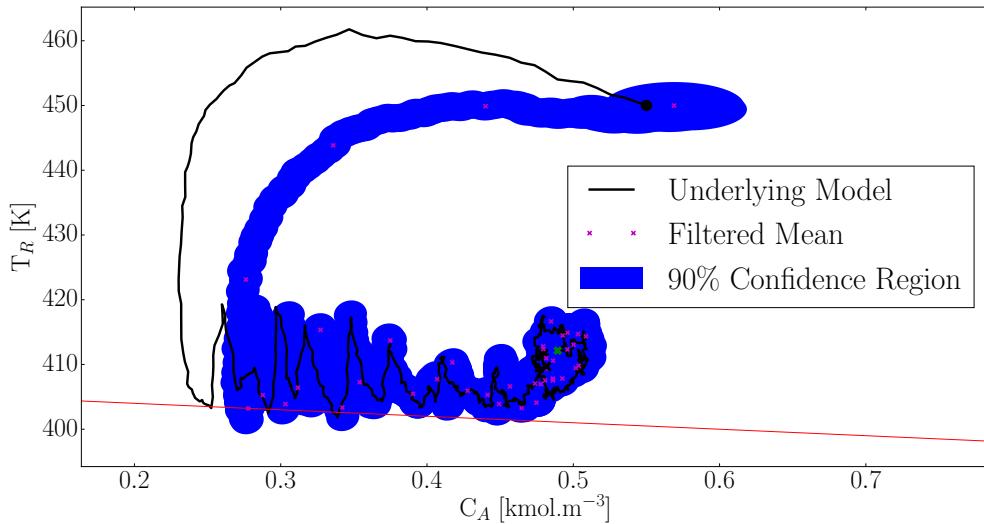


Figure 8.19: Chance constrained MPC state space trajectory with initial condition  $(0.55, 450)$  and measuring both states. A Kalman filter is used for inference and the chance constraint is set at 90%.

It is clear that the non-linearity of the underlying system makes stochastic control difficult. By increasing  $k^2 = 9.21$  which corresponds to changing the chance constraint such that  $\Pr(d^T x_t + e \geq 0) \geq 99\% \forall t = 1, \dots, N$  we hope to increase the minimum distance between the constraint and the underlying state.

Figure 8.20 shows the tracking of the modified system. The average energy input and average concentration error is 69.67 kW and 12.44% respectively. It is quite interesting that the more conservative system performs better with regard to these two metrics than the less conservative system.

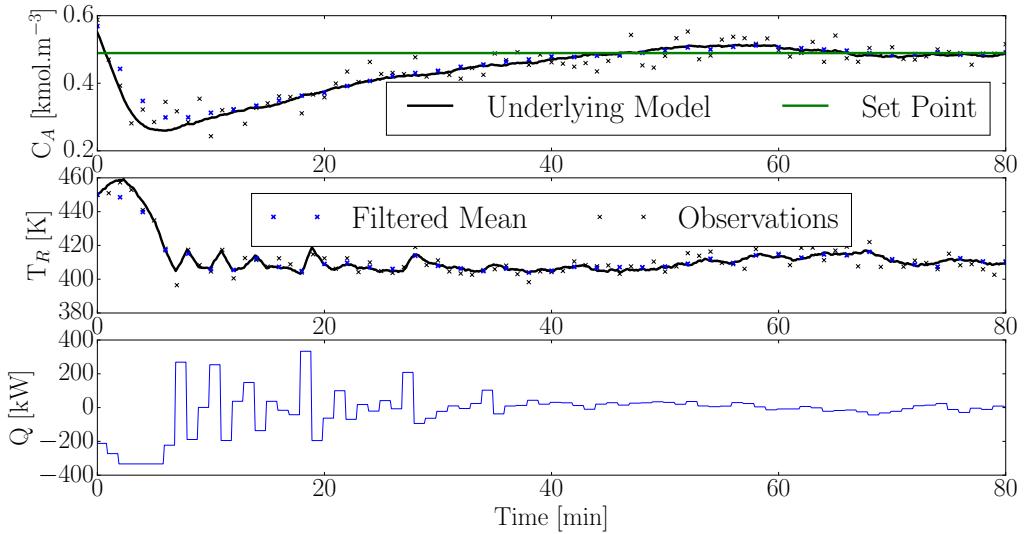


Figure 8.20: Chance constrained MPC tracking with initial condition  $(0.55, 450)$  and measuring both states. A Kalman filter is used for inference and the chance constraint is set at 99%.

In Figure 8.21 we see that the margin of safety is increased although the confidence region still spills over the constraint. However, there is no constraint violation in this case.

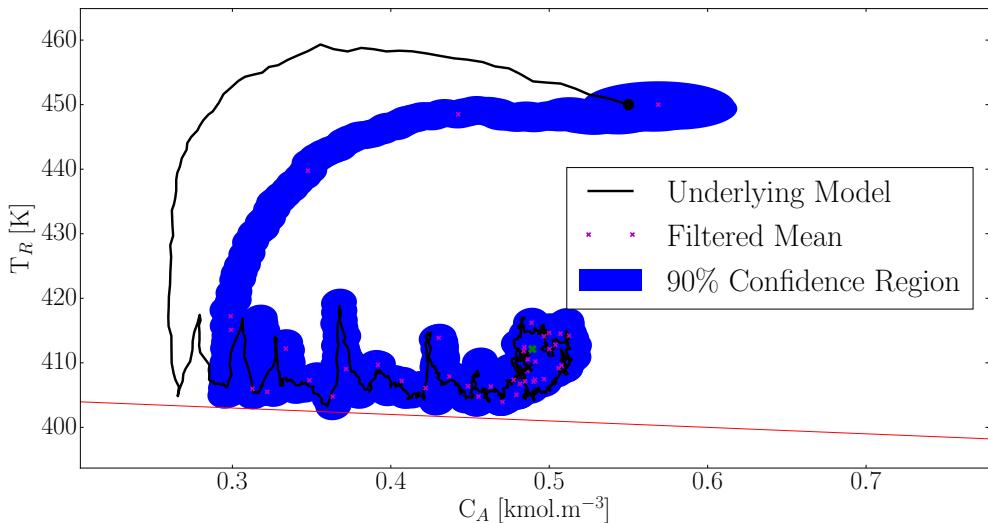


Figure 8.21: Chance constrained MPC state space trajectory with initial condition  $(0.55, 450)$  and measuring both states. A Kalman filter is used for inference and the chance constraint is set at 99%.

The ability of  $k$  to increase or decrease the conservativeness of the constraint lends credibility to its value, at the very least if the system is non-normal, as an empirical measure to include

stochastic robustness to the MPC in an efficient way.

Finally, Figure 8.12 illustrates the effect increasing  $k^2$  (i.e. the threshold for the chance constraint) has on the system. Over 2000 simulations were used to illustrate that these tendencies hold not just for a specific realisation.

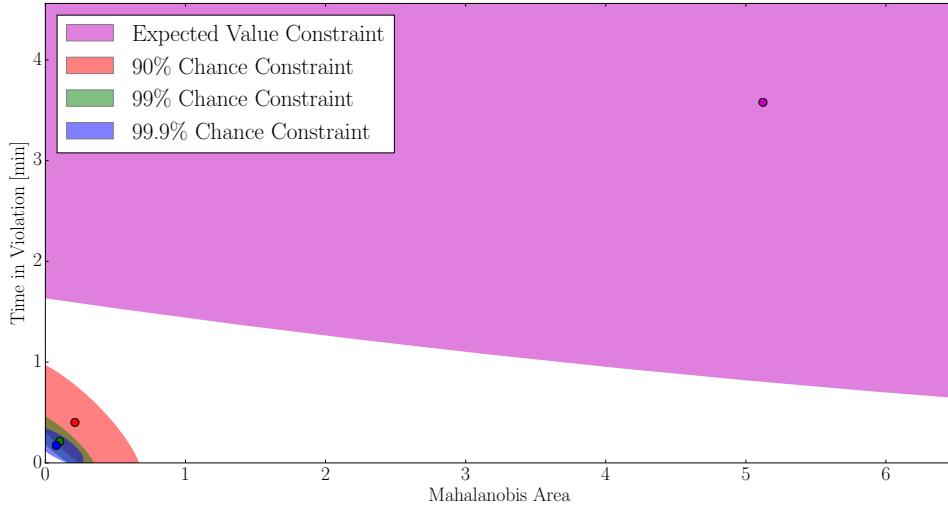


Figure 8.22: Monte-Carlo simulation to investigate the effect increasing  $k^2$  has on the constraint violation characteristics of the MPC controllers. Each region indicates where 90% of the simulations scored. The mean is indicated by a solid point.

It is clear that increasing  $k^2$  causes the constraints to be violated less. It is also clear that the expected value constrained MPC (i.e. the deterministic MPC) performed significantly worse with the nonlinear underlying model when compared to Figure 8.12. The value of the chance constrained MPC is apparent here. Unlike Figure 8.12 there seems to be more value in increasing the chance threshold above the 99% interval.

Figures 8.18 to 8.21 all display the undesirable property originally seen in Figure 8.14: the poor state estimation and associated control problems. We attempt to rectify this situation by using a more sophisticated filter with the chance constrained MPC. We use the MPC as shown in (8.29) with a particle filter using 200 particles. The tracking results are shown in Figure 8.16.

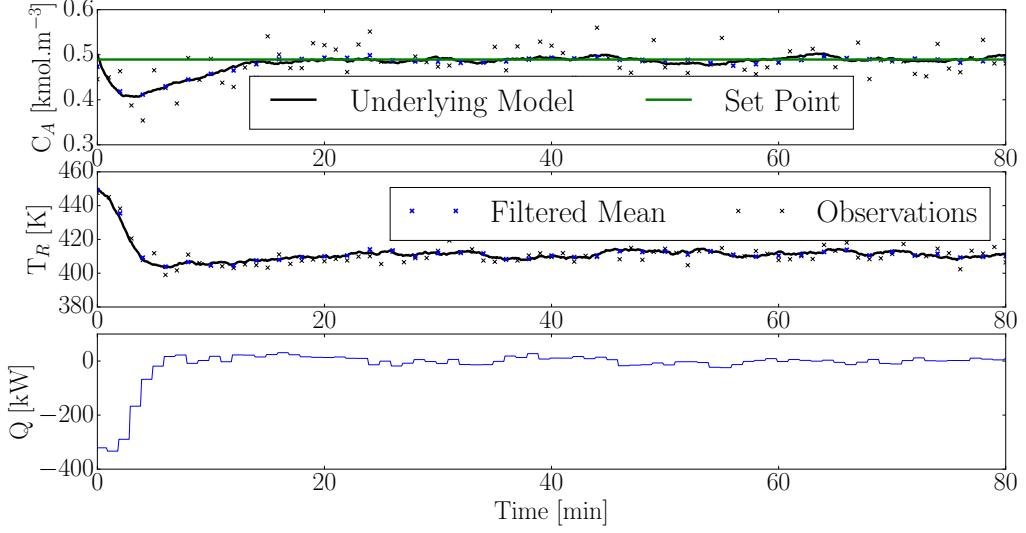


Figure 8.23: Chance constrained MPC tracking with initial condition  $(0.55, 450)$  and measuring both states. A particle filter with 200 particles is used for inference and the chance constraint is set at 90%.

One could see the discussion in Chapter 7.5 as a justification for using the particle filter instead of the Kalman filter for state estimation. Since the underlying model is non-linear we expect the particle filter to outperform the Kalman filter (recall the discussion following Figure 7.7).

The average input energy and average concentration error is 24.26 kW and 2.34% over the course of the simulation. This is a vast improvement over the Kalman filter MPC combination. Clearly the more accurate state estimation is immensely beneficial for control. Figure 8.24 also illustrates that the chance constraint is easily satisfied using the more accurate state estimator.

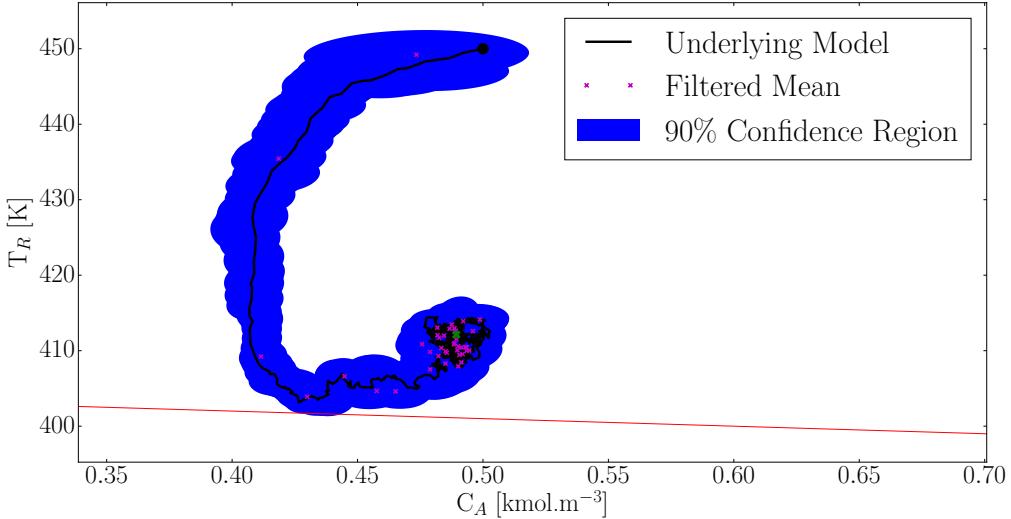


Figure 8.24: Chance constrained MPC state space trajectory with initial condition  $(0.55, 450)$  and measuring both states. A particle filter with 200 particles is used for inference and the chance constraint is set at 90%.

In Figure 8.25 we compare the constraint violation characteristics of the Kalman filter and particle filter based MPCs using over 2000 simulations.

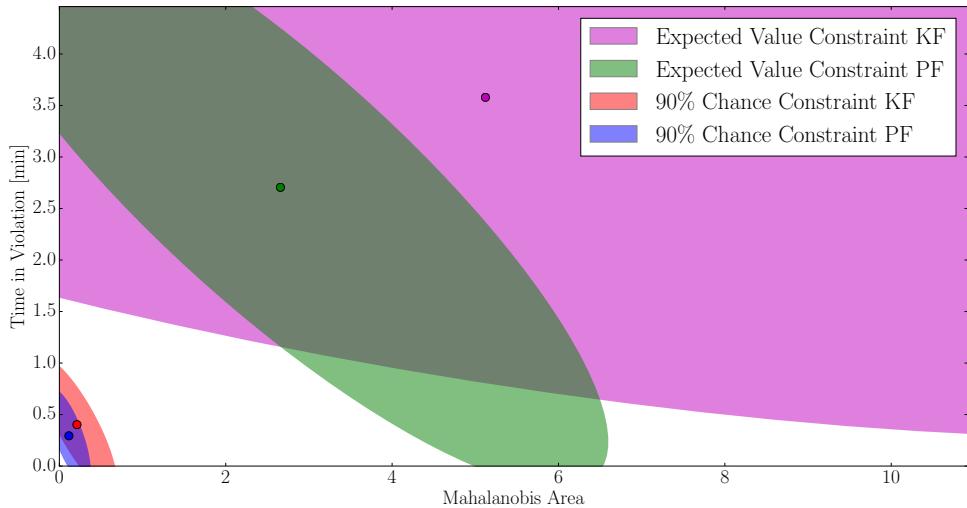


Figure 8.25: Monte-Carlo simulation to investigate the effect increasing  $k^2$  has on the constraint violation characteristics of the MPC controllers. Each region indicates where 90% of the simulations scored. The mean is indicated by a solid point. KF indicates the Kalman Filter MPC and PF indicates the particle filter MPC.

It is clear that the particle filter based MPC outperforms the Kalman filter based MPC.

This is not surprising because, as we discussed for the single realisation results, the state estimation ability of the Kalman filter is inferior due to the nonlinear underlying system.

As before we need to investigate the normal assumption underpinning the theory behind the chance constraint simplification. We investigate it in exactly the same manner as Figure 8.11 using Kullback-Leibler Divergence. To this end, Figure 8.26 shows the degree to which the underlying posterior state distributions are Gaussian given the non-linear underlying system.

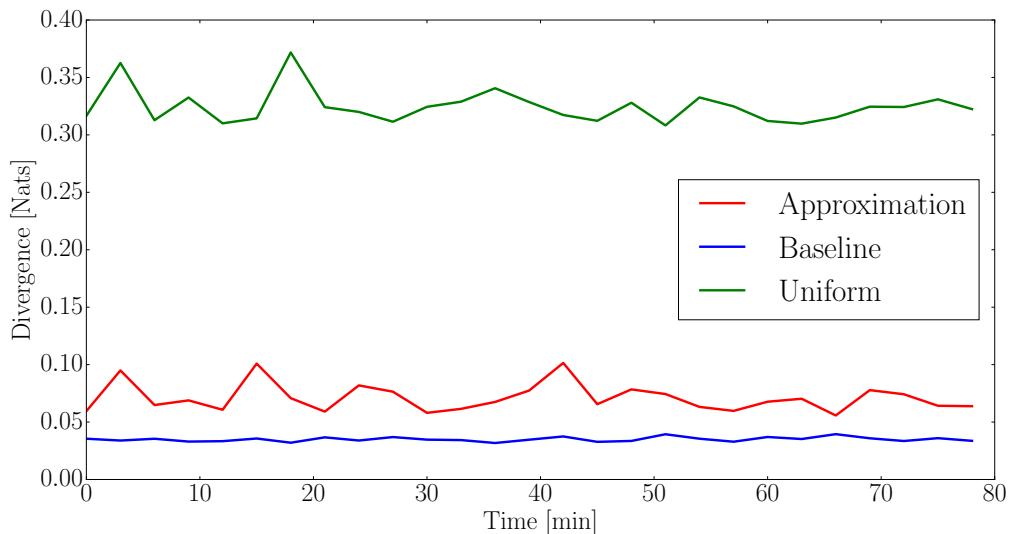


Figure 8.26: Kullback-Leibler Divergence between the assumed Gaussian distribution and actual distribution using 5000 particles. The underlying model is non-linear.

It is a relief that the normal assumption seems to hold almost as well in the non-linear underlying system case. The average divergence for the baseline, approximation and uniform curves are: 0.035, 0.071 and 0.324. It is not surprising that the nonlinearity reduced the degree of normality of the distributions. However, the distribution were not significantly non-Gaussian to raise any concern - the difference between Figures 8.11 and 8.26 is almost negligible.

## 8.6 Conclusion

We have illustrated the benefits gained by designing model predictive controllers within the framework of probabilistic graphical models by showing:

- Under the assumption of normality and linearity it is possible to convert stochastic objective functions into their deterministic equivalents. The analysis is closely related to the work of [59] and [60] but we have shown that these results are immediately obvious from within the framework of Probabilistic graphical models. Thus it is possible to solve LQG objective type problems without resorting to stochastic Dynamic Programming.

2. We have generalised our analysis to stochastic MPC and shown that by using the statistically important metric, the Mahalanobis distance, we arrive at a technique for enforcing chance constraints which is very closely related to the approach taken by [54] and [55]. Under the assumptions of linearity and normality we have shown that constraint satisfaction is ensured. Due to the use of the Mahalanobis distance metric we have provided some theoretical support for the use of the “ellipsoidal approximation” technique if the underlying system is non-linear or not exactly Gaussian.
3. Combining the previous results we have shown that it is possible to write the joint chance constrained stochastic MPC problem as a deterministic MPC problem. Additionally we show that the joint chance constraints can be written in a linear format. The entire optimisation problem can then be written in the standard form for Quadratic Programming optimisation. Standard deterministic MPC solution techniques can then be used to solve the stochastic problem.
4. We have compared the effect different inference techniques have on the quality of the MPC. If the system is linear and Gaussian the Kalman filter is adequate. If there is significant departure from linearity or normality it can be beneficial to use the particle filter.

## **Part III**

# **Multiple Model Systems**

# Bibliography

- [1] Y. Bar-Shalom, X.R. Li, and T. Kirubarajan. *Estimation with applications to tracking and navigation*. John Wiley and Sons, 2001.
- [2] D. Barber. Expectation correction for smoothed inference in switching linear dynamical systems. *Journal of Machine Learning*, 7:2515–2540, 2006.
- [3] D. Barber. *Bayesian Reasoning and Machine Learning*. Cambridge University Press, 2012.
- [4] I. Batina, A.A. Stoervogel, and S. Weiland. Optimal control of linear, stochastic systems with state and input constraints. In *Proceedings of the 41st IEEE Conference on Decision and Control*, 2002.
- [5] A. Bemporad and M. Morari. Control of systems integrating logic, dynamics, and constraints. *Automatics*, 35:407–427, 1999.
- [6] C.M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.
- [7] L. Blackmore, Hui Li, and B. Williams. A probabilistic approach to optimal robust path planning with obstacles. In *American Control Conference*, June 2006.
- [8] L. Blackmore, O. Masahiro, A. Bektassov, and B.C. Williams. A probabilistic particle-control approximation of chance-constrained stochastic predictive control. *IEEE Transactions on Robotics*, 26, 2010.
- [9] M. Cannon, B. Kouvaritakis, and X. Wu. Probabilistic constrained mpc for multiplicative and additive stochastic uncertainty. *IEEE Transactions on Automatic Control*, 54(7), 2009.
- [10] A.L. Cervantes, O.E. Agamennoni, and J.L Figueroa. A nonlinear model predictive control system based on weiner piecewise linear models. *Journal of Process Control*, 13:655–666, 2003.
- [11] R. Chen and J.S. Liu. Mixture kalman filters. *Journal of Royal Statistical Society, Series B*, 62(3):493–508, 2000.

- [12] J.J. Dabrowski and J.P. de Villiers. A method for classification and context based behavioural modelling of dynamical systems applied to maritime piracy. *Expert Systems with Applications*, 2014.
- [13] B.N. Datta. *Numerical Methods for Linear Control Systems - Design and Analysis*. Elsevier, 2004.
- [14] M. Davidian. Applied longitudinal data analysis. North Carolina State University, 2005.
- [15] R. De Maesschalck, D. Jouan-Rimbaus, and D.L. Massart. Tutorial: The mahalanobis distance. *Chemometrics and Intelligent Laboratory Systems*, 50:1–18, 2000.
- [16] J.P. de Villiers, S.J. Godsill, and S.S. Singh. Particle predictive control. *Journal of Statistical Planning and Inference*, 141:1753–1763, 2001.
- [17] N. Deo. *Graph Theory with Applications to Engineering and Computer Science*. Prentice-Hall, 1974.
- [18] M. Diehl, H.J. Ferreau, and N. Haverbeke. Efficient numerical methods for nonlinear mpc and moving horizon estimation. *Control and Information Sciences*, 384:391–417, 2009.
- [19] A. Doucet and A.M. Johansen. A tutorial on particle filtering and smoothing: fifteen years later. Technical report, The Institute of Statistical Mathematics, 2008.
- [20] A.D. Doucet, N.J. Gordon, and V. Krishnamurthy. Particle filters for state estimation of jump markov linear systems. *IEEE Transactions on Signal Processing*, 49(3):613–624, March 2001.
- [21] J. Du, C. Song, and P. Li. Modeling and control of a continuous stirred tank reactor based on a mixed logical dynamical model. *Chinese Journal of Chemical Engineering*, 15(4):533–538, 2007.
- [22] The Economist. In praise of bayes. Article in Magazine, September 2000.
- [23] C. Edwards, S.K. Spurgeon, and R.J. Patton. Sliding mode observers for fault detection and isolation. *Automatica*, 36:541–553, 200.
- [24] H.C. Edwards and D.E. Penny. *Elementary Differential Equations*. Pearson, 6th edition edition, 2009.
- [25] W. Forst and D. Hoffmann. *Optimisation - Theory and Practice*. Springer, 2010.
- [26] O.R. Gonzalez and A.G. Kelkar. *Electrical Engineering Handbook*. Academic Press, 2005.
- [27] N.J. Gordon, D.J. Salmond, and A.F.M. Smith. Novel approach to nonlinear/non-gaussian bayesian state estimation. *IEE Proceedings-F*, 140(2):107–113, 1993.

- [28] R. Isermann and P. Balle. Trends in the application of model based fault detection and diagnosis of technical processes. *Control Engineering Practice*, 5(5):709–719, 1997.
- [29] K. Ito and K. Xiong. Gaussian filters for nonlinear filtering problems. *IEEE Transactions on Automatic Control*, 45(5):910–928, 2000.
- [30] R. J. Jang and C.T. Sun. *Neuro-fuzzy and soft computing: a computational approach to learning and machine intelligence*. Prentice-Hall, 1996.
- [31] D. Koller and N. Friedman. *Probabilistic Graphical Models*. MIT Press, 2009.
- [32] K. B. Korb and A. E. Nicholson. *Bayesian Artificial Intelligence*. Series in Computer Science and Data Analysis. Chapman & Hall, first edition edition, 2004.
- [33] M. Kvasnica, M. Herceg, L. Cirka, and M. Fikar. Model predictive control of a cstr: a hybrid modeling approach. *Chemical Papers*, 64(3):301–309, 2010.
- [34] J.H. Lee, M. Morari, and C.E. Garcia. *Model Predictive Control*. Prentice Hall, 2004.
- [35] U.N. Lerner. *Hybrid Bayesian Networks for Reasoning about Complex Systems*. PhD thesis, Stanford Univesity, 2002.
- [36] P. Li, M. Wendt, H. Arellano-Garcia, and G. Wozny. Optimal operation of distillation processes under uncertain inflows accumulated in a feed tank. *American Institute of Chemical Engineers*, 2002.
- [37] P. Li, M. Wendt, and G. Wozny. A probabilistically constrained model predictive controller. *Automatica*, 38:1171–1176, 2002.
- [38] W.L. Luyben. *Process Modeling, Simulation and Control for Chemical Engineers*. McGraw-Hill, 2nd edition edition, 1990.
- [39] J.M. Maciejowski. *Predictive Control with constraints*. Prentice-Hall, 2002.
- [40] O. Masahiro. Joint chance-constrained model predictive control with probabilistic resolvability. *American Control Conference*, 2012.
- [41] P. Mhaskar, N.H. El-Farra, and P.D. Christofides. Stabilization of nonlinear systems with state and control constraints using lyapunov-based predictive control. *Systems and Control Letters*, 55:650–659, 2006.
- [42] K.P. Murphy. Switching kalman filters. Technical report, Compaq Cambridge Research Lab, 1998.
- [43] K.P. Murphy. *Dynamic Bayesian Networks: Representation, Inference and Learning*. PhD thesis, University of California, Berkeley, 2002.
- [44] K.P. Murphy. *Machine Learning: A Probabilistic Perspective*. MIT Press, 2012.

- [45] N. Nandola and S. Bhartiya. A multiple model approach for predictive control of non-linear hybrid systems. *Journal of Process Control*, 18(2):131–148, 2008.
- [46] L. Ozkan, M. V. Kothare, and C. Georgakis. Model predictive control of nonlinear systems using piecewise linear models. *Computers and Chemical Engineering*, 24:793–799, 2000.
- [47] T. Pan, S. Li, and W.J. Cai. Lazy learning based online identification and adaptive pid control: a case study for cstr process. *Industrial Engineering Chemical Research*, 46:472–480, 2007.
- [48] J.B. Rawlings and D.Q. Mayne. *Model Predictive Control*. Nob Hill Publishing, 2009.
- [49] B. Reiser. Confidence intervals for the mahalanobis distance. *Communications in Statistics: Simulation and Computation*, 30(1):37–45, 2001.
- [50] Y. Sakakura, M. Noda, H. Nishitani, Y. Yamashita, M. Yoshida, and S. Matsumoto. Application of a hybrid control approach to highly nonlinear chemical processes. *Computer Aided Chemical Engineering*, 21:1515–1520, 2006.
- [51] A.T. Schwarm and Nikolaou. Chance constrained model predictive control. Technical report, University of Houston and Texas A&M University, 1999.
- [52] C. Snyder, T. Bengtsson, P. Bickel, and J. Anderson. Obstacles to high-dimensional particle filtering. *Mathematical Advances in Data Assimilation*, 2008.
- [53] S.J. Streicher, S.E. Wilken, and C. Sandrock. Eigenvector analysis for the ranking of control loop importance. *Computer Aided Chemical Engineering*, 33:835–840, 2014.
- [54] D.H. van Hessem and O.H. Bosgra. Closed-loop stochastic dynamic process optimisation under input and state constraints. In *Proceedings of the American Control Conference*, 2002.
- [55] D.H. van Hessem, C.W. Scherer, and O.H. Bosgra. Lmi-based closed-loop economic optimisation of stochastic process operation under state and input constraints. In *Proceedings of the 40th IEEE Conference on Decision and Control*, 2001.
- [56] H. Veeraraghavan, P. Schrater, and N. Papanikolopoulos. Switching kalman filter based approach for tracking and event detection at traffic intersections. *Intelligent Control*, 2005.
- [57] D. Wang, W. Wang, and P. Shi. Robust fault detection for switched linear systems with state delays. *Systems, Man and Cybernetics*, 39(3):800–805, 2009.
- [58] R.S. Wills. Google’s pagerank: the math behind the search engine. Technical report, North Carolina State University, 2006.

- [59] J. Yan and R.R. Bitmead. Model predictive control and state estimation: a network example. In *15th Triennial World Conference of IFAC*, 2002.
- [60] J. Yan and R.R. Bitmead. Incorporating state estimation into model predictive control and its application to network traffic control. *Automatica*, 41:595–604, 2005.
- [61] M.B. Yazdi and M.R. Jahed-Motlagh. Stabilization of a cstr with two arbitrarily switching modes using model state feedback linearisation. *Chemical Engineering Journal*, 155(3):838–843, 2009.