Contents

1 Introduction **4**

I Literature, Theory and Background Material **5**

2 Literature Study **6**

2.1 Stochastic Model Predictive Control 6

2.2 Switching Model Predictive Control 10

3 Background Theory **13**

3.1 Probability Theory 13

3.1.1 Discrete Random Variables 14

3.1.2 Continuous Random Variables 17

3.2 Graph Theory 21

3.3 Probabilistic Graphical Models 22

3.3.1 Bayesian Networks 23

3.3.2 Dynamic Bayesian Networks 25

3.4 Control 27

3.4.1 Linear Quadratic Regulator Control 27

3.4.2 Reference Tracking 31

3.4.3 Linear Quadratic Gaussian Control 31

3.4.4 Model Predictive Control 33

3.5 Matrix Identities 34

4 Hidden Markov Models **35**

4.1 Markov Models 35

4.2 Hidden Markov Models 36

4.2.1 Filtering 37

4.2.2 Smoothing 38

4.2.3 Viterbi Decoding 40

4.2.4 Prediction 41

4.3 Burglar Localisation Problem 42

5 CSTR Model **45**

5.1 Qualitative Analysis . . . 46
5.2 Nonlinear Model . . . 48
5.3 Linearised Models . . . 50

II Single Model Systems 57

6 Inference using Linear Models 58
6.1 Kalman Filter . . . 59
6.2 Kalman Prediction . . . 61
6.3 Smoothing and Viterbi Decoding . . . 63
6.4 Filtering the CSTR . . . 64

7 Inference using Nonlinear Models 68
7.1 Sequential Monte Carlo Methods . . . 69
7.2 Particle Filter . . . 72
7.3 Particle Prediction . . . 74
7.4 Smoothing and Viterbi Decoding . . . 74
7.5 Filtering the CSTR . . . 75

8 Stochastic Linear Control 79
8.1 Unconstrained Stochastic Control . . . 80
8.2 Constrained Stochastic Control . . . 84
8.3 Reference Tracking . . . 88
8.4 Linear System . . . 89
8.5 Nonlinear System . . . 97
8.6 Conclusion . . . 105

III Multiple Model Systems 106

9 Inference using Linear Hybrid Models 107
9.1 Exact Filtering . . . 108
9.2 Rao-Blackwellised Particle Filter . . . 109
9.3 Rao-Blackwellised Particle Prediction . . . 110
9.4 Smoothing and Viterbi Decoding . . . 111
9.5 Filtering the CSTR . . . 111

10 Stochastic Switching Linear Control using Linear Hybrid Models 120
10.1 Unconstrained Switching Control . . . 122
10.2 Conclusion . . . 126

11 Inference using Nonlinear Hybrid Models 127
11.1 Exact Filtering . . . 128
11.2 Switching Particle Filter . . . 128

11.3 Switching Particle Prediction . . . 129
11.4 Smoothing and Viterbi Decoding . . . 129
11.5 Filtering the CSTR . . . 130

12 Stochastic Switching Control using Non-linear Hybrid Models **135**
12.1 Unconstrained Switching Control . . . 136
12.2 Constrained Switching Control . . . 138
12.3 Conclusion . . . 144

13 Future Work **142**
13.1 Parameter Optimisation . . . 142
13.2 Augmented Switching Graphical Model . . . 142
13.3 Stochastic Inputs Graphical Model . . . 142

14 Conclusion **143**

Part I

Literature, Theory and Background Material

Part II

Single Model Systems

Part III

Multiple Model Systems

Chapter 12

Stochastic Switching Control using Non-linear Hybrid Models

We continue our discussion of switching control from Section 10 here. In this section we use the Switching Particle Filter to identify the best model to use in the stochastic controllers we developed in Section 8. More precisely, let $M_i = (A_i, B_i)$ be the linearised model of the non-linear models (f_i, g_i) as discussed in Section 11. By using the most likely model resulting from the Switching Particle Filter we aim to design a controller which is robust against system faults.

We assume the same scenario as introduced in Section 11.5 i.e. we assume we have 2 non-linear plant models available. Model M_1 corresponds to the healthy CSTR and model M_2 corresponds to the CSTR with denatured catalyst (the faulty model). We will again avail ourselves of the Switching Controller Algorithm repeated here for convenience.

Switching Controller Algorithm:

1. Use a switching filter algorithm, e.g. the SPF, to update the state estimates of the particle population given the current observation. See Section 11 for more details.

2. Select the particle with the highest switching weight. Since each particle corresponds to a certain model we implicitly have the most probable model M_i.

3. Use the mean and covariance information encoded by this particle within the context of the stochastic controller (LQG and MPC) formulation of Section 8. Use the most likely model, M_i from step 2, in this setting.

4. Repeat for the next observation.

Recalling Section 2.2 we note that it is possible to incorporate the model switching into the optimisation problem but at the cost of a significantly increased computational burden. Like Section 10 we simplify the problem by only using a single model in the controller algorithm but we allow this model to change based on the system observations. A coincidental benefit of this approach is that the controller will automatically detect the modelled fault.

Since the underlying Graphical Model in Section 9 and Section 11 is the same, we expect the global trends to be the same as those found in Section 10. For the sake of illustration we exclusively use both state measurements. There is no fundamental reason why one cannot use only one state measurement except that the filter performance will be worse.

For the remainder of this section we assume the control goal is to keep the the system at the unsteady concentration operating point of the healthy model, even in the presence of the denatured catalyst.

12.1 Unconstrained Switching Control

In this section we compare the standard LQG controller (discussed in Section 3.4.3 and 8.1) to the Switching Controller Algorithm implemented within the context of the LQG controller as shown in (12.1). The same control parameters as those found in Section 8.4 are used.

$$\begin{aligned} &\min_{\mathbf{u}} \mathbb{E}\left[\frac{1}{2}\sum_{k=0}^{N-1}\left(x_k^T Q x_k + u_k^T R u_k\right) + \frac{1}{2}x_N^T P_f x_N\right] \\ &\text{subject to } x_{t+1} = A_i x_t + B_i u_t + w_t \text{ (Latent)} \\ &\text{and } y_t = C x_t + v_t \text{ (Observed)} \end{aligned} \tag{12.1}$$

Note that we select $M_i = (A_i, B_i)$ as the most likely model based on the switch weight at each time step. This model is then used in (12.1) and solved using the techniques of Section 8.1

In Figure 12.1 we see the performance of the LQG controller applied to the CSTR system. At 100 minutes the catalyst denatures and the model used to design the controller becomes grossly inaccurate. The inappropriateness of the model also affects the Particle Filter's performance.

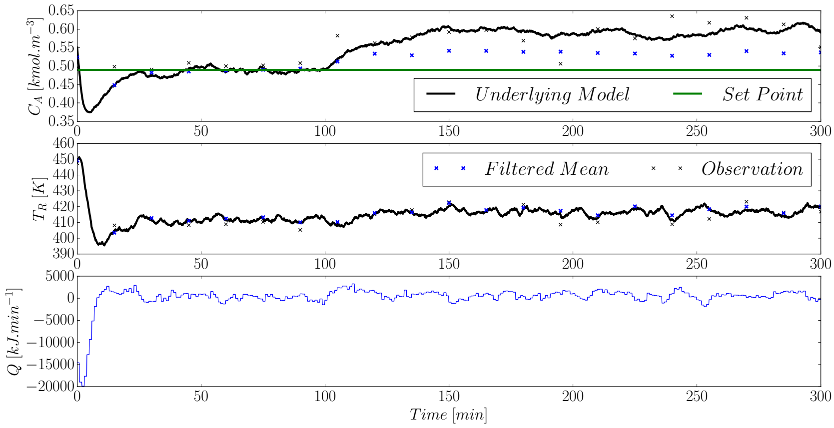

Figure 12.1: Standard LQG controller applied to the CSTR where the catalyst denatures at 100 minutes. The bootstrap Particle Filter was used for inference and the Gaussian approximation of the particles was used.

The average concentration error is 14.31% and the average controller input is 120 kJ/min over the course of the simulation. We can clearly see that there is non-zero set point offset and control is bad in the sense of Definition 8.1. Clearly the standard LQG controller is ineffective in this scenario.

This motivates the use of a controller which intelligently changes the model control is based upon, as discussed previously. In Figure 12.2 we see the set point tracking ability of the Switching Controller Algorithm using the LQG controller. Also note the superior filtering performance of the Switching Particle Filter.

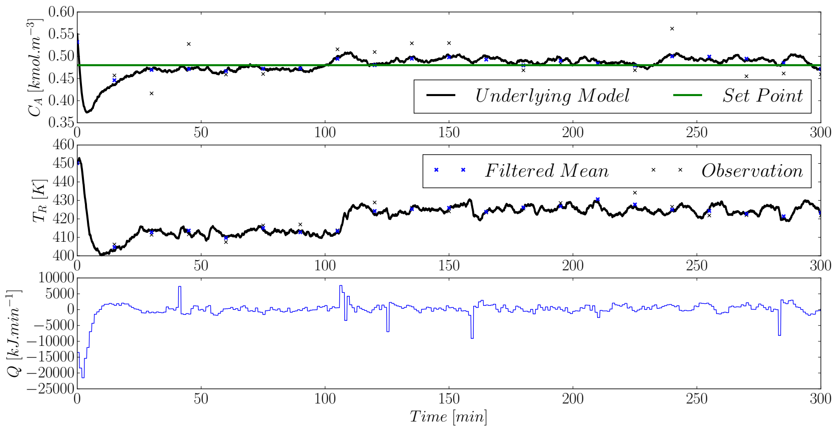

Figure 12.2: Switching LQG controller applied to the CSTR where the catalyst denatures at 100 minutes.

The average concentration error is 2.97% and the average controller input is 129 kJ/min. It is clear that we have set point tracking even after the catalyst denatures. By inspecting Figure 12.3 we see that this is not surprising: the filter correctly identifies when the underlying model changes and then uses the better model for control.

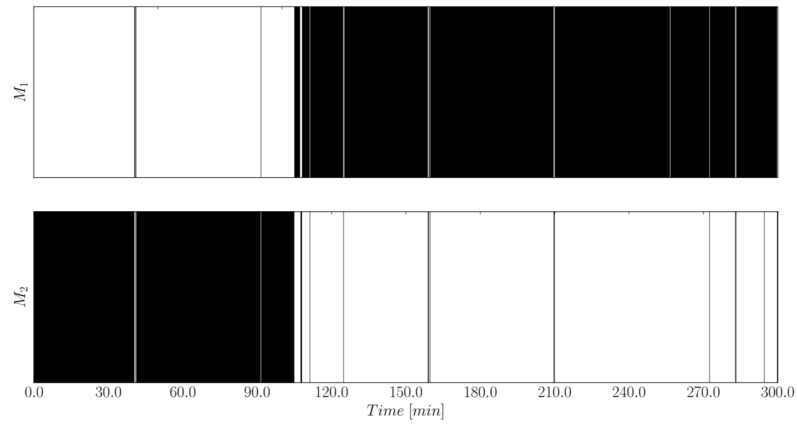

Figure 12.3: Most likely model identified using the Switching Particle Filter within the context of the Switching LQG Controller Algorithm.

However, like in Section 10.1 and 11.5 we see that there is some switching noise. In Section 10.1 this caused controller instability because the filter would switch between the different models too rapidly. Fortunately this is not the case here - the filter only briefly selects the working model M_1 when the system is in the regime of the faulty model M_2. The model oscillations are much less pronounced here.

The cause of this problem is that the models are too similar: the filter cannot clearly distinguish between them at all times. The problem would be exacerbated if only one state measurement was made. When we compared Figures 11.5 and 11.8 we saw that the additional measurement greatly benefited the filter's ability to discern between the models.

12.2 Constrained Switching Control

In this section we extend the Switching Controller Algorithm of Section 12.1 to the stochastic MPC introduced in Section 8.2. We neglected implementing the stochastic MPC using the Rao-Blackwellised Particle Filter of Section 10 because the switching noise (model selection) issues were too pronounced. We use the same parameters and measure both states.

Like in Section 8.4 we first illustrate the performance of the stochastic MPC controller with expected value constraints shown in (12.2) (for some model M_i) and then incorporate chance constraints later.

$$
\begin{aligned}
&\min_{\mathbf{u}} \mathbb{E}\left[\frac{1}{2}\sum_{k=0}^{N-1}(x_k^T Q x_k + u_k^T R u_k) + \frac{1}{2}x_N^T P_f x_N\right] \\
&\text{subject to } x_{t+1} = A_i x_t + B_i u_t + w_t \text{ (Latent)} \\
&\text{and } y_t = C x_t + v_t \text{ (Observed)} \\
&\text{and } \mathbb{E}[\begin{pmatrix}10\\1\end{pmatrix}^T x_t + 400] \geq 0 \ \forall\ t = 1, ..., N \\
&\text{and } |u_t| \leq 15000 \ \forall\ t = 0, ..., N-1
\end{aligned}
\tag{12.2}
$$

Using the results of Section 8.2 we know that (12.2) can be reformulated as a deterministic problem given the (Gaussian) state estimate x_0. The state estimate is derived from either the Particle Filter or Switching Particle Filter using 200 and 500 particles respectively.

In Figure 12.4 we see the set point tracking performance of the (12.2) using the same Particle Filter as used in Section 12.1. Since the Particle Filter only uses the healthy plant model we only use M_1.

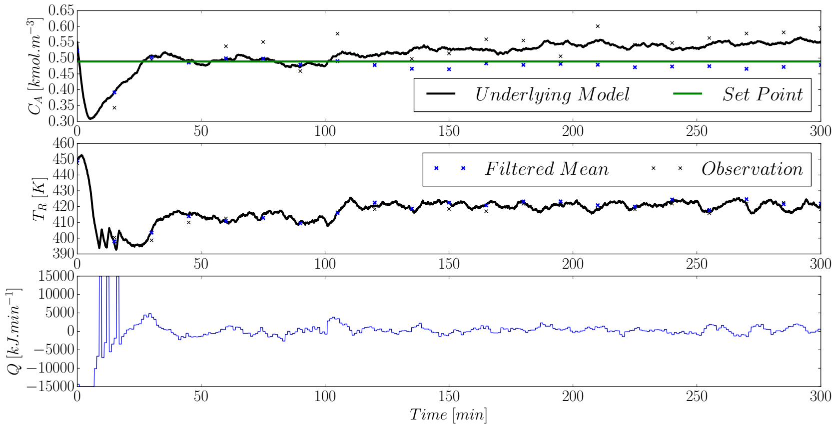

Figure 12.4: Deterministic MPC using a Particle Filter as the state estimator. The initial point is $(0.55, 450)$. An integrating disturbance model was used to estimate the mismatch between the underlying system and the controller model. The catalyst denatures at 100 minutes.

Taking into account the discussion in Section 2.2 on zero offset[1] MPC we expect the controller to be more effective than the corresponding LQG controller of Section 12.1. This is indeed the case. The average concentration error is 8.33% and the average controller input is 145 kJ/min.

Unfortunately we do not observe zero set point offset control but rather zero offset state estimates. Clearly the controller input generated by the MPC, which is based on the healthy

[1]The results of this section implement the constant disturbance model to achieve zero set point offset. To keep notation the same we do not explicitly show it in (12.2) but mention it here.

plant, drives the Particle Filter's predictions to the set point. Note that the Particle Filter's model is only based on the healthy plant. We can see that the classic disturbance model approach [32] to ensure zero set point offset fails here because the underlying (faulty) model is too different from the controller model. Intuitively, we are attempting to control a tricycle (the faulty plant) using a model of a Ferrari.

In Figure 12.5 we see the Switching Controller Algorithm applied within the context of (12.2). The model corresponding to the highest weighted switch at each time step was selected for control.

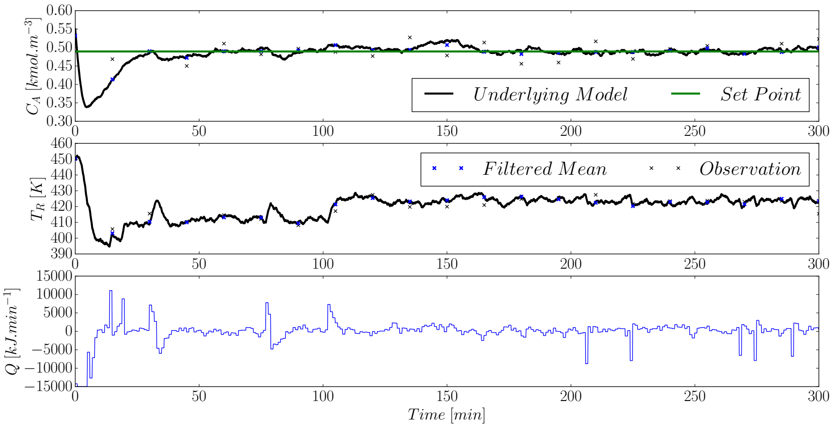

Figure 12.5: The Switching MPC Controller Algorithm applied to the CSTR with catalyst which denatures at 100 minutes.

The average concentration error is 2.82% and the average controller input is 142 kJ/min over the simulation time span. The performance of the switching controller is significantly better than the non-switching case. This is not surprising because, as Figure 12.6 shows, the filter correctly identifies when the plant breaks.

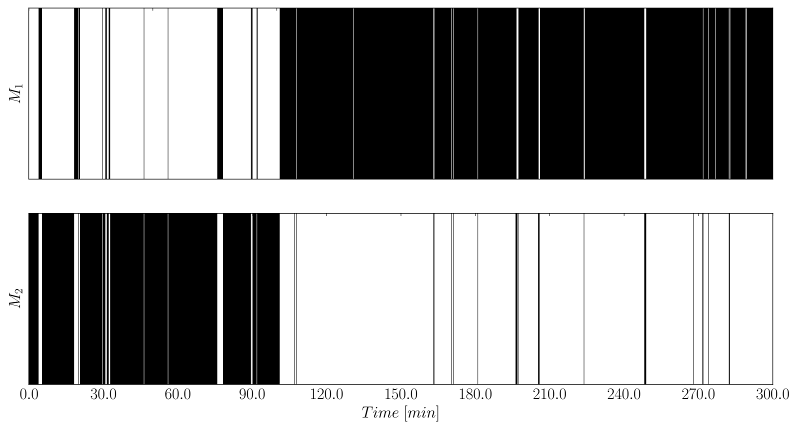

Figure 12.6: Most likely model identified using the Switching Particle Filter within the context of the Switching MPC Controller Algorithm.

Unfortunately the same type of switching noise as found in Section10.1 and 12.1 is present here. This is not surprising because the underlying reasons, as discussed earlier, have not changed. It is also interesting to note that the controller input jumps each time the model switches. This can lead to instability like in Section 10.

While the switching controllers discussed in this section did successfully keep the system at set point it is easy to see that they are not robust against model overlap. If we considered a problem with more than 1 faulty model, e.g. a separate model for the scenario where a pipe bursts, we could have the same controller instability as seen in Section 10. Again it seems intuitively reasonable that by extending the Graphical Model, as discussed in Section 10.1, it is possible to ameliorate this type of problem.

Finally, for completeness we also demonstrate the application of the chanced constrained stochastic MPC within the Switching Controller framework. In Figure 12.7 we see the state space trajectory of the expected value constraint MPC.

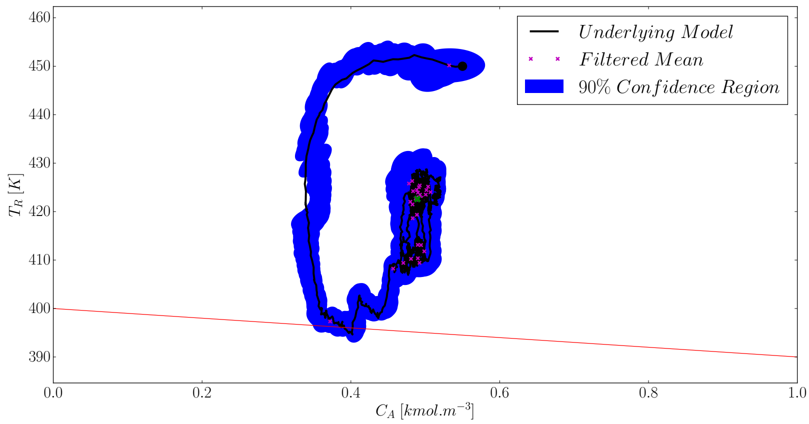

Figure 12.7: State space trajectory of the expected value constrained stochastic MPC using the Switching Controller Algorithm.

Clearly there is a constraint violation - similar to that found in Section 8.4 and 8.5. By extending the MPC problem of (12.2) to the chance constrained (12.3) we attempt to ensure that the constraint is not violated.

$$\begin{aligned}
&\min_{\mathbf{u}} \mathbb{E}\left[\frac{1}{2}\sum_{k=0}^{N-1}\left(x_k^T Q x_k + u_k^T R u_k\right) + \frac{1}{2}x_N^T P_f x_N\right] \\
&\text{subject to } x_{t+1} = A_i x_t + B_i u_t + w_t \text{ (Latent)} \\
&\text{and } y_t = C x_t + v_t \text{ (Observed)} \\
&\text{and } \mathbb{E}[\begin{pmatrix}10\\1\end{pmatrix}^T x_t + 400] \geq 0 \;\forall\; t = 1, ..., N \\
&\text{and } \Pr(\begin{pmatrix}10\\1\end{pmatrix}^T x_t + 400 \geq 0) \geq 0.99 \;\forall\; t = 1, ..., N \\
&\text{and } |u_t| \leq 15000 \;\forall\; t = 0, ..., N-1
\end{aligned} \tag{12.3}$$

The same Switching Controller Algorithm, as discussed previously, is implemented. We have used the 99% chance constraint to highlight the effectiveness of the method compared to the expected value version.

In Figure 12.8 we see that the switching chance constrained MPC successfully tracks the set point.

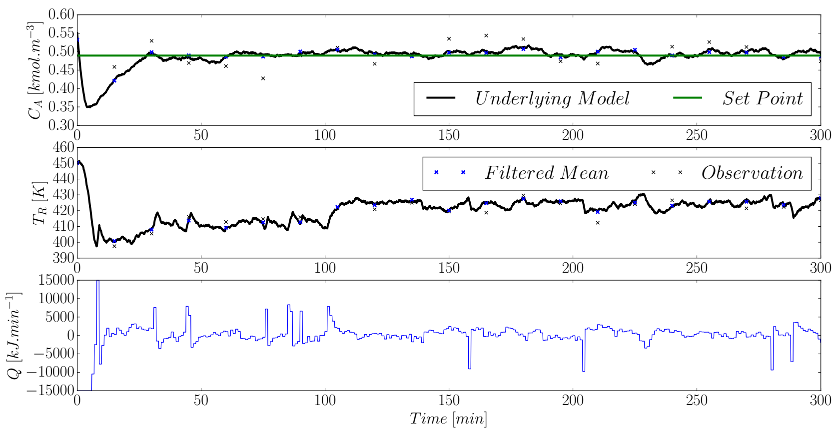

Figure 12.8: The Switching MPC Controller Algorithm applied to the CSTR with catalyst which denatures at 100 minutes. The chance constrained MPC was used.

The average concentration error is 2.97% and the average controller input is 170 kJ/min. In Figure 12.9 we see the familiar model switching diagram. Clearly the controller successfully isolates when the fault occurs.

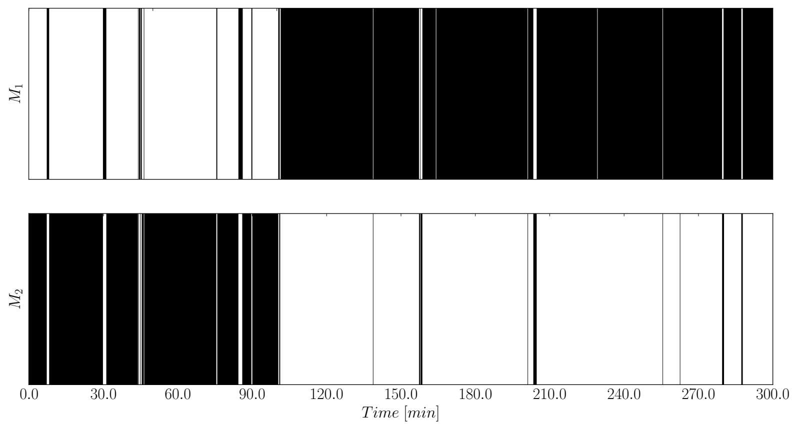

Figure 12.9: Most likely model identified using the Switching Particle Filter within the context of the chance constrained Switching MPC Controller Algorithm.

Finally, in Figure 12.10 we see that the constraint is not violated.

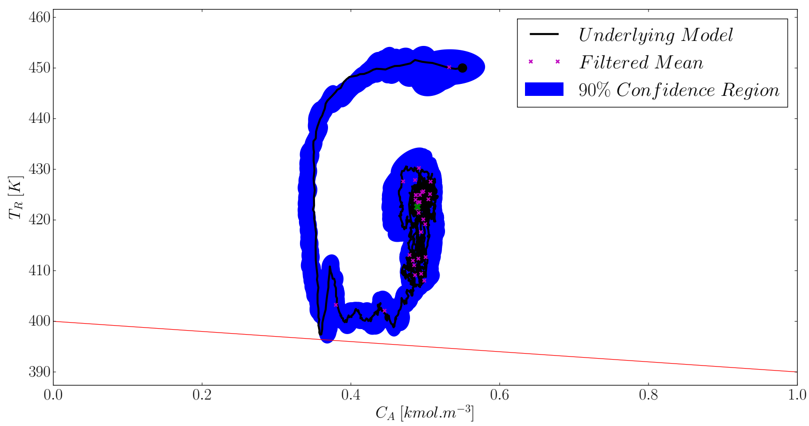

Figure 12.10: State space trajectory of the chance constrained stochastic MPC using the Switching Controller Algorithm.

Since Figure 12.10 only illustrates that the constraint is not violated for a single run we again use a Monte-Carlo technique to justify the assertion that, for this example, the stochastic controller can successfully reduce the constraint violation probability. By simulating 100 runs it was found that the expected value stochastic MPC violated the constraint 2.2 times per run. The chance constrained stochastic MPC violated the constraint 0.2 times per run. If the robustness of the Switching Controller can be improved there is significant upside to its implementation.

12.3 Conclusion

In this section we implemented the Switching Controller Algorithm using the Switching Particle Filter. Both the LQG and stochastic MPC were used in conjunction with the Switching Particle Filter. While the controllers successfully regulated the system the controller/filter combination was not robust against switching noise. Since this noise has the potential to destabilise control more research needs to be done to investigate effective methods to assure stability. The Augmented Switching Kalman Filter Model discussed in Section 10 could be a potential candidate. While stability issues could plague the implementation of such a system the fault detection and superior state estimation ability of the Switching Particle Filter was found to be useful.

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. Stoorvogel, 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*, 62(3):493–508, 2000.

[12] B.N. Datta. *Numerical Methods for Linear Control Systems - Design and Analysis*. Elsevier, 2004.

[13] M. Davidian. Applied longitudinal data analysis. North Carolina State University, 2005.

[14] R. De Maesschalck, D. Jouan-Rimbaus, and D.L. Massart. Tutorial: The mahalanobis distance. *Chemometrics and Intelligent Laboratory Systems*, 50:1–18, 2000.

[15] N. Deo. *Graph Theory with Applications to Engineering and Computer Science*. Prentice-Hall, 1974.

[16] 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.

[17] A. Doucet and A.M. Johansen. A tutorial on particle filtering and smoothing: fifteen years later. Technical report, The Institute of Statistical Mathematics, 2008.

[18] 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.

[19] 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.

[20] The Economist. In praise of bayes. Article in Magazine, September 2000.

[21] C. Edwards, S.K. Spurgeon, and R.J. Patton. Sliding mode observers for fault detection and isolation. *Automatica*, 36:541–553, 200.

[22] H.C. Edwards and D.E. Penny. *Elementary Differential Equations*. Pearson, 6th edition edition, 2009.

[23] W. Forst and D. Hoffmann. *Optimisation - Theory and Practice*. Springer, 2010.

[24] O.R. Gonzalez and A.G. Kelkar. *Electrical Engineering Handbook*. Academic Press, 2005.

[25] 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.

[26] 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.

[27] K. Ito and K. Xiong. Gaussian filters for nonlinear filtering problems. *IEEE Transactions on Automatic Control*, 45(5):910–928, 2000.

[28] R. J. Jang and C.T. Sun. *Neuro-fuzzy and soft computing: a computational approach to learning and machine intelligence*. Prentice-Hall, 1996.

[29] D. Koller and N. Friedman. *Probabilistic Graphical Models*. MIT Press, 2009.

[30] K. B. Korb and A. E. Nicholson. *Bayesian Artificial Intelligence*. Series in Computer Science and Data Analysis. Chapman & Hall, first edition edition, 2004.

[31] 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.

[32] J.H. Lee, M. Morari, and C.E. Garcia. *Model Predictive Control*. Prentice Hall, 2004.

[33] U.N. Lerner. *Hybrid Bayesian Networks for Reasoning about Complex Systems*. PhD thesis, Stanford Univesity, 2002.

[34] P. Li, M. Wendt, H. Arellano-Garcia, and G. Wozny. Optimal operation of distrillation processes under uncertain inflows accumulated in a feed tank. *American Institute of Chemical Engineers*, 2002.

[35] P. Li, M. Wendt, and G. Wozny. A probabilistically constrained model predictive controller. *Automatica*, 38:1171–1176, 2002.

[36] W.L. Luyben. *Process Modeling, Simulation and Control for Chemical Engineers*. McGraw-Hill, 2nd edition edition, 1990.

[37] J.M. Maciejowski. *Predictive Control with constraints*. Prentice-Hall, 2002.

[38] O. Masahiro. Joint chance-constrained model predictive control with probabilistic resolvability. *American Control Conference*, 2012.

[39] 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.

[40] K.P. Murphy. Switching kalman filters. Technical report, Compaq Cambridge Research Lab, 1998.

[41] K.P. Murphy. *Dynamic Bayesian Networks: Representation, Inference and Learning*. PhD thesis, University of California, Berkeley, 2002.

[42] K.P. Murphy. *Machine Learning: A Probabilistic Perspective*. MIT Press, 2012.

[43] N. Nandola and S. Bhartiya. A multiple model approach for predictive control of nonlinear hybrid systems. *Journal of Process Control*, 18(2):131–148, 2008.

[44] 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.

[45] 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.

[46] J.B. Rawlings and D.Q. Mayne. *Model Predictive Control*. Nob Hill Publishing, 2009.

[47] B. Reiser. Confidence intervals for the mahalanobis distance. *Communications in Statistics: Simulation and Computation*, 30(1):37–45, 2001.

[48] 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.

[49] A.T. Schwarm and Nikolaou. Chance constrained model predictive control. Technical report, University of Houston and Texas A&M University, 1999.

[50] C. Snyder, T. Bengtsson, P. Bickel, and J. Anderson. Obstacles to high-dimensional particle filtering. *Mathematical Advances in Data Assimilation*, 2008.

[51] 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.

[52] 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.

[53] 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.

[54] H. Veeraraghavan, P. Schrater, and N. Papanikolopoulos. Switching kalman filter based approach for tracking and event detection at traffic intersections. *Intelligent Control*, 2005.

[55] 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.

[56] R.S. Wills. Google's pagerank: the math behind the search engine. Technical report, North Carolina State University, 2006.

[57] J. Yan and R.R. Bitmead. Model predictive control and state estimation: a network example. In *15th Triennial World Conference of IFAC*, 2002.

[58] 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.

[59] 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.