Contents

Part I

Literature, Theory and Background Material

Part II

Single Model Systems

Part III

Multiple Model Systems

Chapter 10

Stochastic switching linear control using linear hybrid models

In Chapter 2.2 model switching MPC was introduced. In short, a set of models with corresponding binary integer variables are incorporated into the MPC optimisation problem. The optimisation algorithm changes the model it uses for prediction based on the location of the previous predicted state. In this way a number of models can potentially be used for prediction. It is desirable to change models if the system states move far away from the linearisation point of current linear model. It is hoped that the significant computational burden this introduces is offset by the increased predictive accuracy of the controller.

In Chapter 8 we developed efficient stochastic controller algorithms (LQG and MPC) which use a single linear model for control. While it is possible to attempt to extend these algorithms to the aforementioned approach, the computational problems will persist because mixed integer programming is fundamentally more difficult than quadratic programming [25]. From a practical perspective one would like to reduce computational complexity because, especially for large problems, on-line optimisation can become problematic.

In Chapter 9 the Rao-Blackwellised particle filter was introduced. Briefly, the filter uses a set of linear models, $M_i = (A_i, B_i)$ for each model i, to estimate the current state (we assume the system and measurement noise is common across all models as well as the observation matrix). The ability of each model to explain the observations is calculated in a Bayesian sense. This is used to weight the importance of each model's contribution to the current state estimate.

In this chapter we will attempt to combine the ideas of Chapter 2.2, 8 and 9 to create a computationally efficient switching model controller algorithm. We assume that the underlying process dynamics are described by (10.1) where f and g are the nonlinear transition and observation functions of the CSTR process introduced in Chapter 5. It is assumed that x_t is a latent stochastic variable and y_t is an observed stochastic variable.

$$\begin{aligned} x_{t+1} &= f(x_t, u_t) + w_{t+1} \\ y_{t+1} &= g(x_{t+1}) + v_{t+1} \end{aligned} \tag{10.1}$$

We also assume that the models used for inference and control are linear and of the form shown in (10.2) for model $M_i = (A_i, B_i)$.

$$\begin{aligned} x_{t+1} &= A_i x_t + B_i u_t + w_{t+1} \\ y_{t+1} &= C x_{t+1} + v_{t+1} \end{aligned} \tag{10.2}$$

The noise terms retain their meaning from Chapter 8. It is our aim to move the system states from the unstable (nominal) operating point to another operating point. This will clearly cause the system to traverse the state space and necessitate model switching. We first describe the intuition behind the proposed switching controller algorithm and then state the algorithm.

As mentioned before, it becomes desirable to have a mechanism to switch the underlying controller model if the system states move far away from the linearisation point of the current model. However, it is computationally difficult to perform this switching within the framework of the optimisation algorithm because it invariably necessitates the introduction of integer variables. We propose an algorithm which uses the Rao-Blackwellised particle filter to estimate the current state as well as the models which best describe the current observation. Based on the results of Chapter 9 we expect the weight assigned to each model to skew in favour of the models which were linearised closest to the current state. Now we have two options to implement control at each time step[1]:

1. Use only the most likely model (the model with the highest switch weight) for controller prediction i.e. use $A^* = A_{\text{indmax}[s_t]}$.
2. Use the weighted average (from the switch weight) of the models for controller prediction i.e. use $A^* = \sum_{i=1}^{M} s_t^i A_i$.

Since it is not clear which approach is best we investigate both. This "best current model" is then used in the single model controller algorithms discussed in Chapter 8. This approach falls squarely between the purely single model controllers, as discussed in Chapter 8, and the switching model controllers, where the model switching occurs inside the optimisation problem, as discussed in Chapter 2.2. By switching models outside the optimisation problem the scheme will necessarily be more computationally efficient than those found in Chapter 2.2.

Switching controller algorithm:

1. Use a switching filter algorithm, e.g. the Rao-Blackwellised particle filter, to update the state estimates of the particle population given the current observation. See Chapter 9 for more details.

[1]Note that A^* is the model used for control in this explanation.

2. Select the best current model to for control based on the model weights also supplied by the switching filter algorithm.

3. Use the mean and covariance information from the current posterior state estimate and the best current model (from step 2) within the context of the stochastic controller (LQG or MPC) formulation of Chapter 8.

4. Repeat for the next observation.

The astute reader will notice that we are implicitly using the graphical model shown in Figure 10.1 for state estimation (filtering) but the graphical model of Figure 10.2 for model based prediction in step 3.

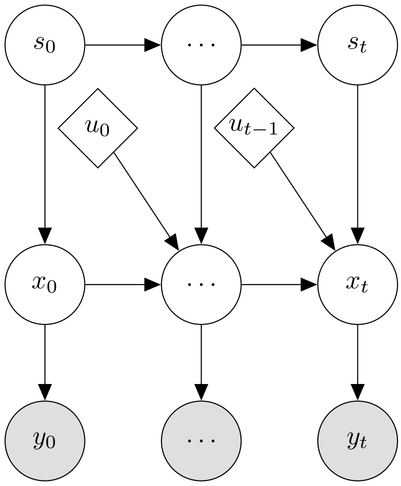

Figure 10.1: Graphical model used for state estimation.

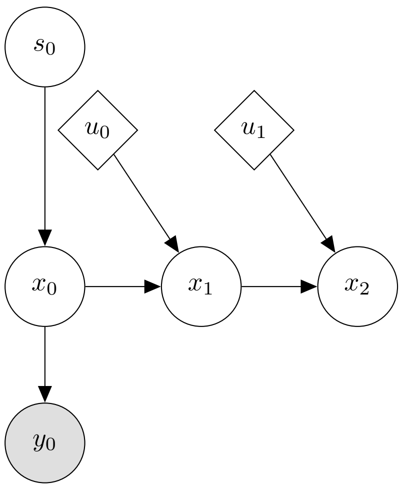

Figure 10.2: Simplified graphical model used for prediction. Within the context of prediction we have that $x_0 \leftarrow x_t$ and $s_0 \leftarrow s_t$ at each successive time step to simplify notation.

We are not using the graphical model associated with Rao-Blackwellised particle prediction (see Chapter 9.3) because that would require that we incorporate stochastic model switching within the optimisation algorithm.

For the remainder of this chapter we assume that we have a bank of M linear models and that we measure both states. Each model is derived by linearising the non-linear CSTR model, found in Chapter 5, around the nominal operating points discussed in the same chapter as well as Chapter 9.5. We also use the switching transition matrices P_2, P_3 found in (9.4) where appropriate. All other parameters are the same as those found in Chapter 8.

10.1 Unconstrained switching control

As mentioned earlier, we will investigate two approaches which can be used to implement the switching controller algorithm. The first approach, used in Chapter 10.1.1, makes use of only the most likely model within the controller. The second approach, discussed in Chapter 10.1.2, makes use of model averaging to construct a model for control.

10.1.1 Most likely model approach

Due to the analysis of Chapter 8.1 we know that it is possible to convert the stochastic optimisation problem (10.3) into the deterministic optimisation problem (10.4) for each linear model (M_1, M_2, M_3) given that we have the current state estimate x_0, the model dynamics are linear and the underlying distributions are Gaussian. Throughout this chapter we make these assumptions[2]. As before, we also denote the mean and covariance of the current state estimate x_0 by $\mathbb{E}[x_0] = \mu_0$ and $\text{var}[x_0] = \Sigma_0$. We use a prediction horizon of $N = 150$ i.e. 15 minutes into the future.

$$\begin{aligned}
&\min_{\mathbf{u}} \mathbb{E}\left[\frac{1}{2}\sum_{k=0}^{N-1}\left(x_k^T Q x_k + u_k^T R u_k\right) + \frac{1}{2}x_N^T P_f x_N\right] \\
&\text{subject to } x_{t+1} = A_i x_t + B_i u_t + w_t
\end{aligned} \tag{10.3}$$

We have that (10.3) is equivalent to (10.4) under the aforementioned assumptions.

$$\begin{aligned}
&\min_{\mathbf{u}} \frac{1}{2}\sum_{k=0}^{N-1}\left(\mu_k^T Q \mu_k + u_k^T R u_k\right) + \frac{1}{2}\mu_N^T P_f \mu_N + \frac{1}{2}\sum_{k=0}^{N}\text{tr}(Q\Sigma_k) \\
&\text{with } \mu_{t+1} = A_i \mu_t + B_i u_t \\
&\text{and } \Sigma_{t+1} = W + A_i \Sigma_t A_i^T
\end{aligned} \tag{10.4}$$

Given this we apply the switching controller algorithm within the context of the LQG controller i.e. given a model M_i from the filter we solve the LQG problem and implement that input. In light of our analysis in Chapter 8 it is clear that the switching controller algorithm is straightforward to implement because it simplifies to M deterministic LQR controllers.

As mentioned before we only use the most likely model for control purposes. By only selecting one model to use for control we dramatically simplify the control problem. It allows us to use the controllers of Chapter 8 directly.

[2]Note that the underlying model is clearly nonlinear but the model used for prediction and inference is linear.

We study 4 control problems using the switching controller algorithm in this chapter. Problems 1 and 2 allow the controller to switch between 3 linear models and problems 3 and 4 allow the controller to switch between 7 linear models. Furthermore, problems 1 and 3 seek to drive the system to the low temperature operating point i.e. a concentration set point of 0.998 kmol.m^{-3} while problems 2 and 4 seek to drive the CSTR to a concentration set point of 0.90 kmol.m^{-3}. In all cases we use the switching LQG controller as shown in (10.3).

We investigate the first problem in Figures 10.3 to 10.5. In Figure 10.3 we see the state space trajectory of the system under control. We expect M_2 to be active initially after which only M_3 should be active.

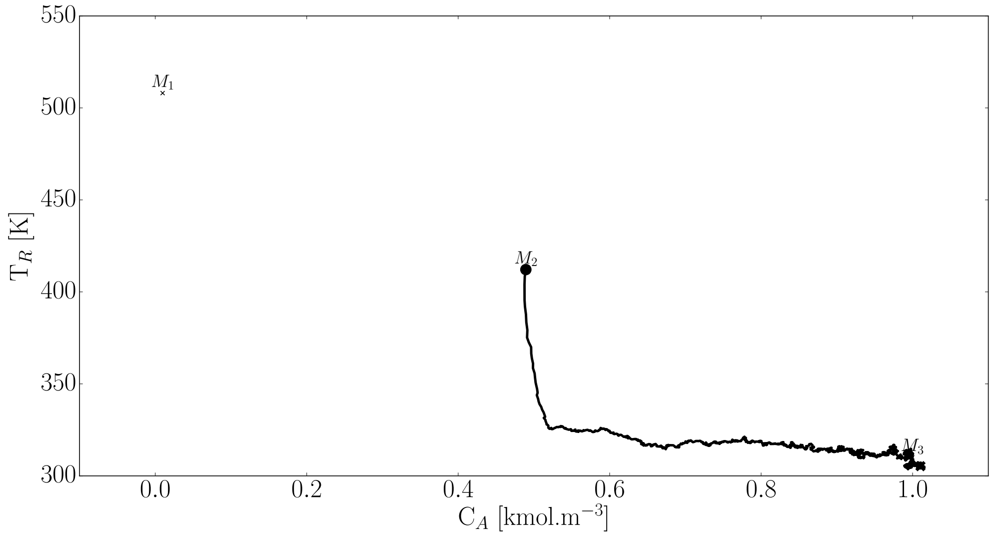

Figure 10.3: State space trajectory of the non-linear CSTR under control of the LQG switching controller algorithm. The initial point was $(0.49, 412)$

Figure 10.4 confirms the behaviour we expected: initially M_2 best explained the observations but M_2 gives way to M_3 throughout the rest of the simulation.

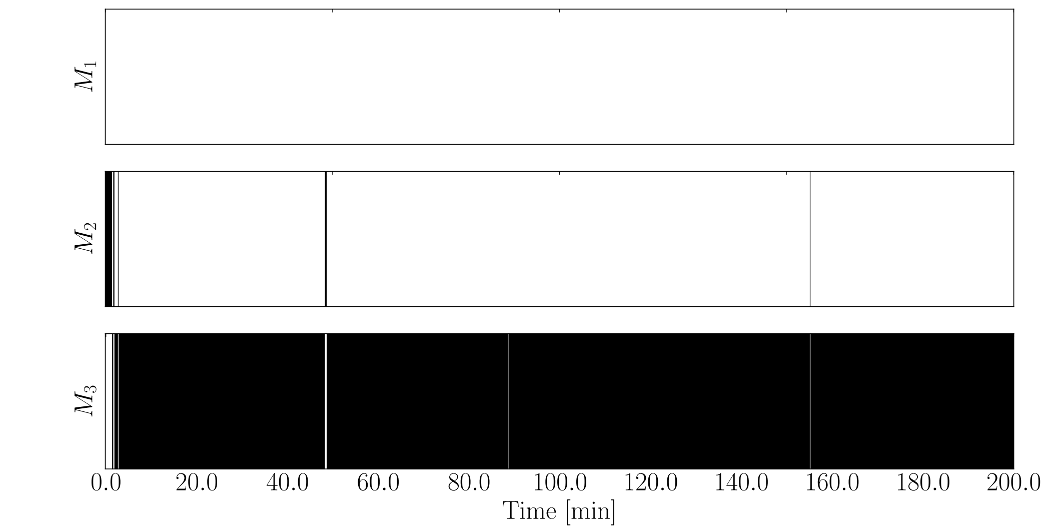

Figure 10.4: Most likely model used for control at each time step over the simulation. Black indicates the model is active.

However, it is clear that there are some problems in Figure 10.4. There does seem to be some slight switching noise. Given that we are using the sticky switching transition matrix P_2 it is clear that model overlap is causing problems. The same problem was identified in Chapter 9.5. In Figure 10.5 we see the set point tracking performance of the switching controller. It is clear that the controller tracks the set point.

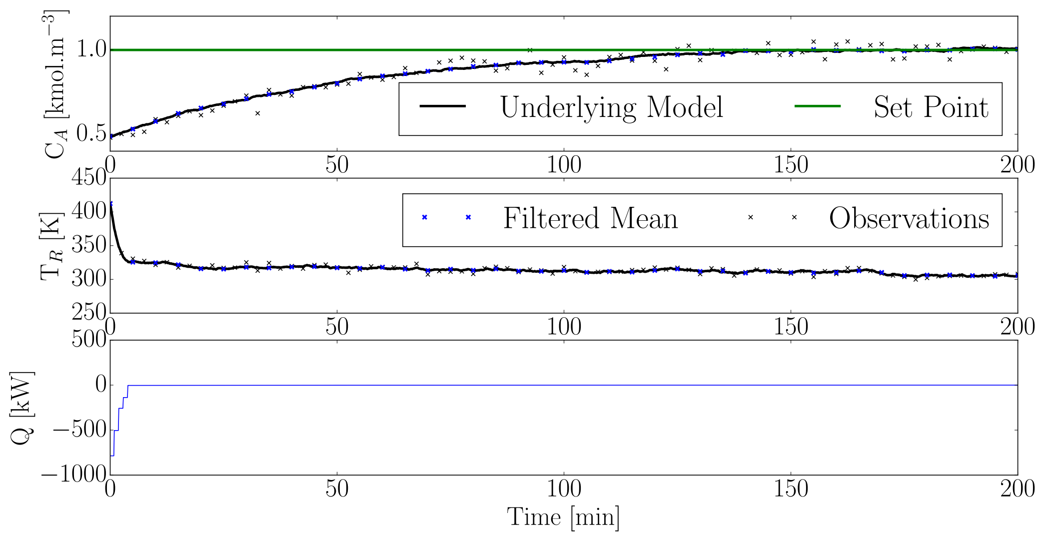

Figure 10.5: Set point tracking and controller input for the LQG 3 model switching controller algorithm. The initial point was $(0.49, 412)$.

Based on Figures 10.4 and 10.5 it would be too easy to surmise that the controller algorithm works. Unfortunately this is not the case in general. In Figures 10.6 and 10.7 we study problem 2: tracking the concentration set point is 0.90 kmol.m^{-3}. In Figure 10.6 we see significant model switching noise.

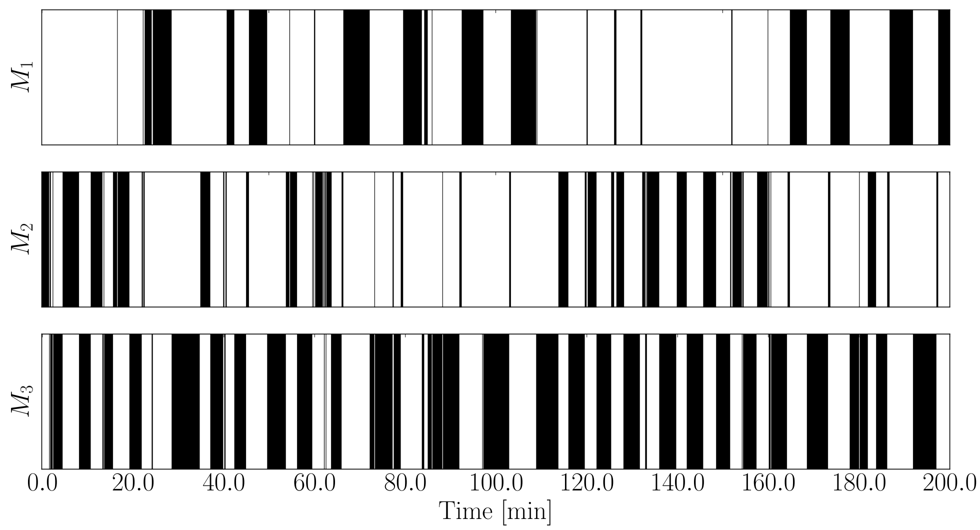

Figure 10.6: Most likely model used for control at each time step over the simulation. Black indicates the model is active.

In Figure 10.7 the detrimental consequence of the switching noise is evident. The controller is completely unstable and oscillates.

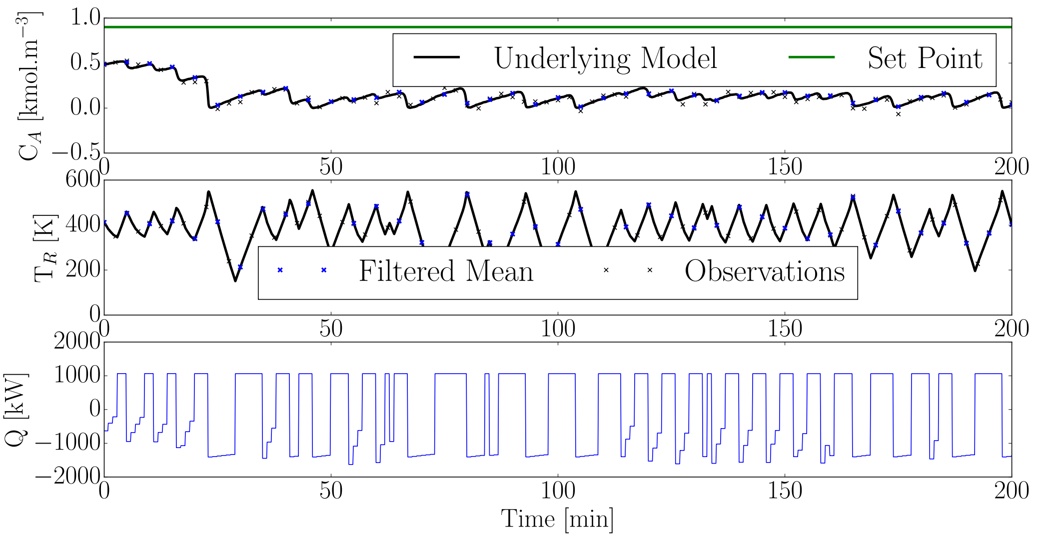

Figure 10.7: Set point tracking and controller input for the LQG 3 model switching controller algorithm. The initial point was $(0.49, 412)$.

It is clear that the oscillations are caused by the filter's inability to stick to a model. There are two major problems with the switching controller algorithm as adopted in this chapter:

1. Fundamentally we are using an inappropriate model for controller prediction during the initial period of the simulation. If we stayed near the current position in state space then the most likely model would predict the future well and thus result in good control. However, we are projecting the controlled states into regions where the current model control is based upon may not a good approximation at all. This is a fundamental problem of our approach - it would be better to incorporate the model switching within the controller prediction (optimisation) process, but this is exactly what we want to avoid due to the computational burden this introduces!

2. The switching noise is problematic because it can cause the controller to use a model which is good locally but inappropriate with regard to the true underlying position of the system in state space. However, from an inference perspective the noise is not undesirable. Switching noise can improve the state estimate accuracy - especially in regions between models. It is also important in allowing the filter to switch punctually: making the switching transition matrix too static retards the sensitivity the filter has to model changes.

We attempt to ameliorate these problems by extending the number of models available to the filter. We use more models to bridge the gap between the current most likely model and where it projects the states during prediction. In Figure 10.8 we illustrate the state space trajectory followed by the system when it has 7 models available for inference and control.

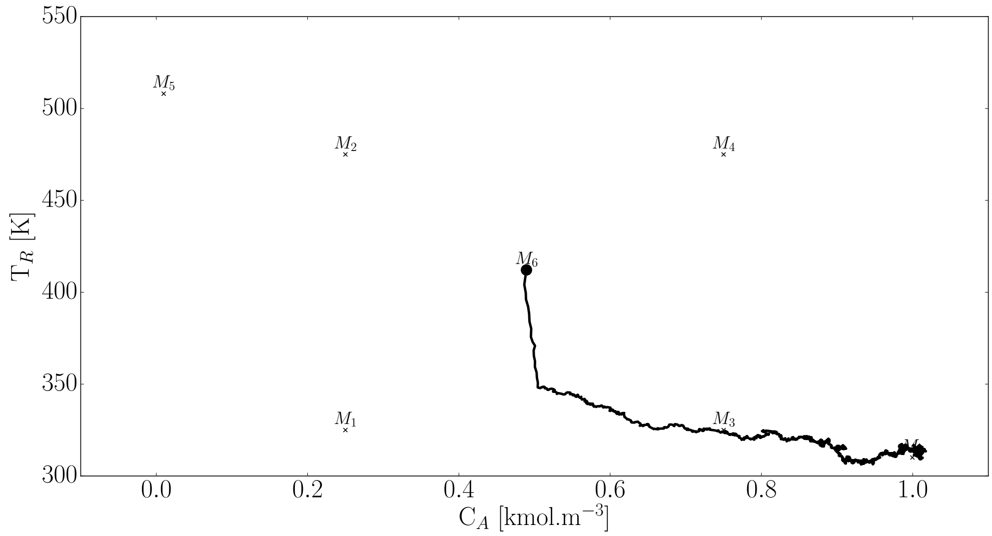

Figure 10.8: State space trajectory of the non-linear CSTR under control of the LQG 7 model switching controller algorithm. The initial point was $(0.49, 412)$

In Figures 10.9 and 10.10 we again attempt to steer the system from the unstable operating point to the low temperature operating point. Figure 10.9 shows which models were active over the simulation window.

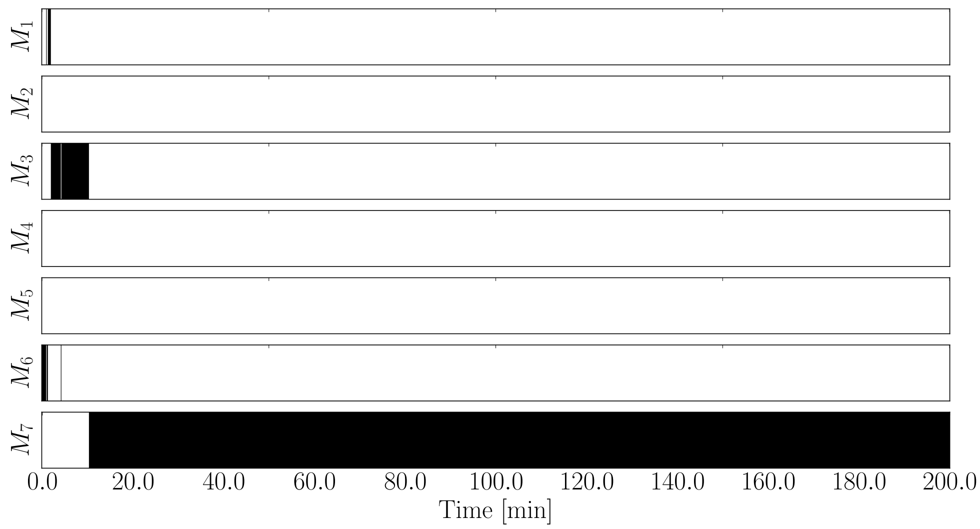

Figure 10.9: Most likely model used for control at each time step over the simulation. Black indicates the model is active.

While there is slight switching noise the model selection is exactly what one would expect. Figure 10.10 shows the controller set point tracking performance over the simulation window.

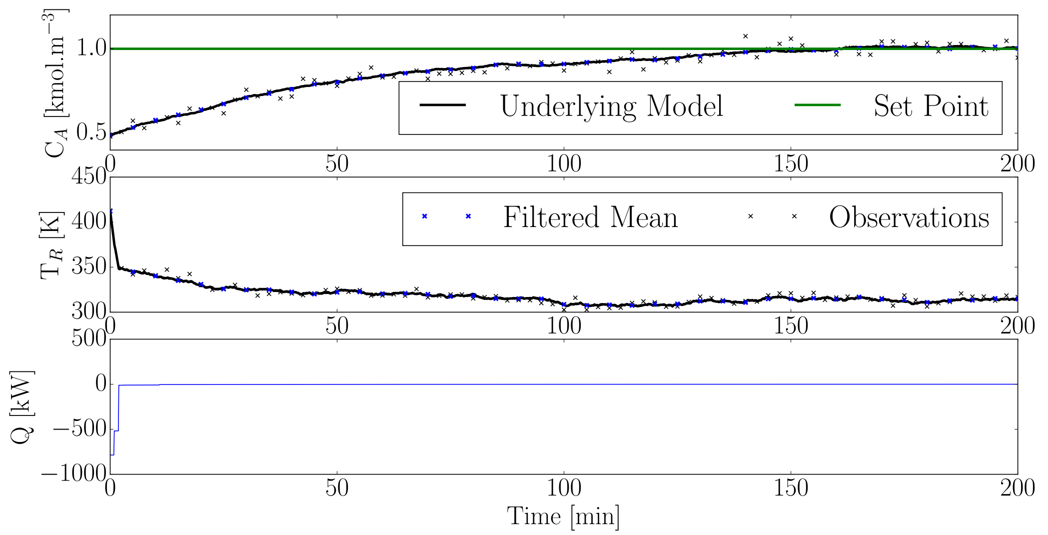

Figure 10.10: Set point tracking and controller input for the LQG 7 model switching controller algorithm. The initial point was $(0.49, 412)$.

Like Figure 10.5 we also have a stable, reference tracking controller. This is not surprising because the model switching/selection was reasonable. In Figure 10.11 and 10.12 we again attempt to steer the system to a concentration set point of 0.9 kmol.m^{-3}. Since we introduced

M_3 to serve as a bridge between M_6 and M_7 (since the set point is between these two models) we expect better performance than in Figure 10.6 and 10.7. Unfortunately this is not the case as may be seen in Figures 10.11 and 10.12.

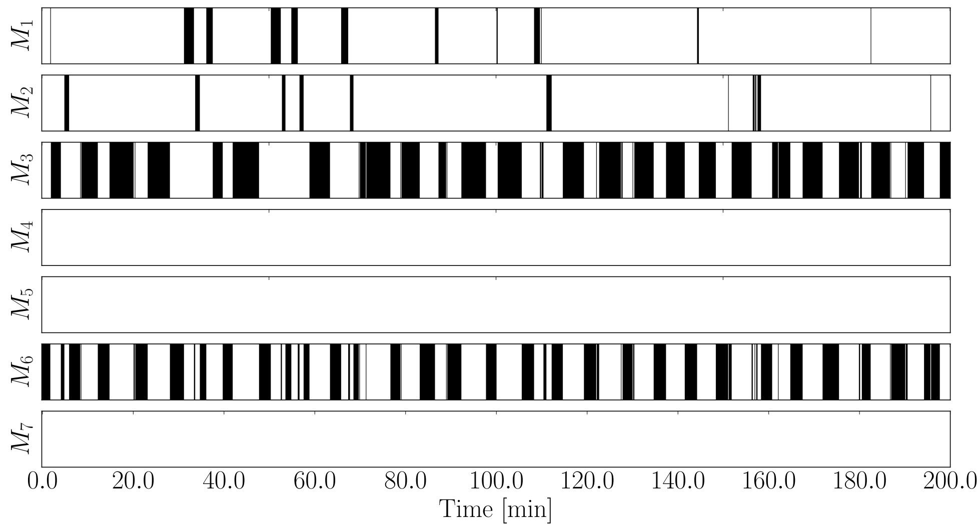

Figure 10.11: Most likely model used for control at each time step over the simulation. Black indicates the model is active.

The same oscillating switching noise and unstable control is present here as there was in Figures 10.6 and 10.7.

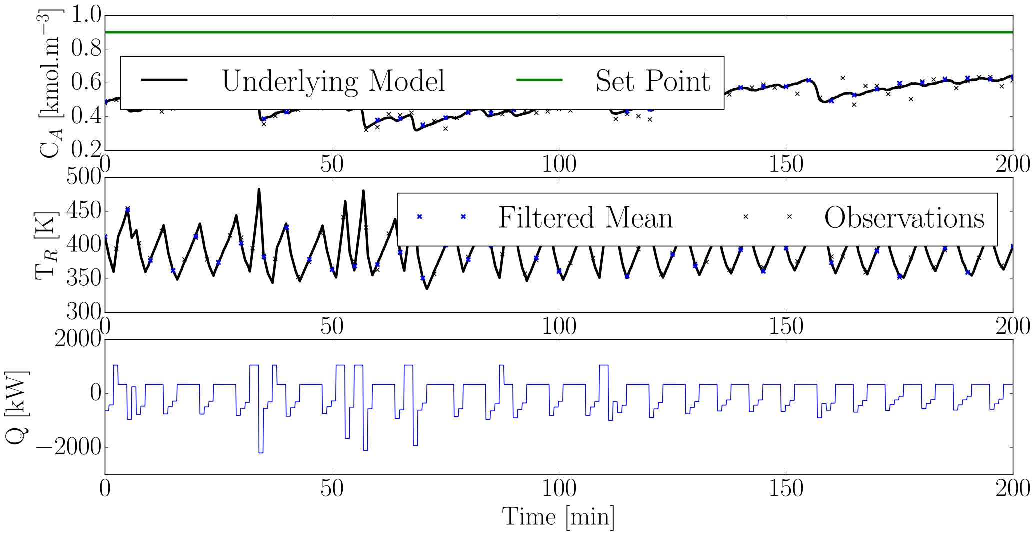

Figure 10.12: Set point tracking and controller input for the LQG 7 model switching controller algorithm. The initial point was $(0.49, 412)$.

The underlying reasons for the instability are the same: we are using models to predict into regions where they are inaccurate and the filter does not robustly enough isolate the model closest to the current system location in state space. The combination of these two problems make effective control impossible.

10.1.2 Model averaging approach

The fundamental problem with the switching controllers of Chapter 10.1.1 is that an inappropriate model was used for prediction. Unfortunately using a weighted average of all the models, based on their probability with respect to the switching variable s_t, will exacerbate this problem. For this reason we do not explore this approach.

10.2 Conclusion

The goal of the switching controller algorithm was to

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] 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 distrillation 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 nonlinear 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.