Robust Self-Tuning Control Design under Probabilistic Uncertainty using Polynomial Chaos Expansion-based Markov Models

Yuncheng Du*, Hector Budman, Thomas Duever*****

**Department of Chemical & Biomolecular Engineering, Clarkson University, Potsdam, NY 13699 USA*
(Tel: 315-268-2284; e-mail: ydu@clarkson.edu)
***Department of Chemical Engineering, University of Waterloo, ON N2L 3G1 Canada*
(Tel: 519-888-4567 x36980; e-mail:hbudman@uwaterloo.ca)
****Department of Chemical Engineering, Ryerson University, ON M5B 2K3 Canada*
(Tel: 416-979-5140; e-mail: tom.duever@ryerson.ca)

Abstract: A robust adaptive controller is developed for a chemical process using a generalized Polynomial Chaos (gPC) expansion-based Markov decision model, which can account for time-invariant probabilistic uncertainty and overcome computational challenge for building Markov models. To calculate the transition probability, a gPC model is used to iteratively predict probability density functions (PDFs) of system's states including controlled and manipulated variables. For controller tuning, these PDFs and controller parameters are discretized to a finite number of discrete states for building a Markov model. The key idea is to predict the transition probability of controlled and manipulated variables over a finite future control horizon, which can be further used to calculate an optimal sequence of control actions. This approach can be used to optimally tune a controller for set point tracking within a finite future control horizon. The proposed method is illustrated by a continuous stirred tank reactor (CSTR) system with stochastic perturbations in the inlet concentration. The efficiency of the proposed algorithm is quantified in terms of control performance and transient decay.

Keywords: Uncertainty propagation, predictive control, Markov decision model, dynamic programming

1. INTRODUCTION

Adaptive control provides a systematic approach for automatic adjustment of controller parameters to maintain a desired level of control system performance. The basic idea is to recursively identify a best model of the process from the closed loop input-output data and to subsequently adjust controller parameters based on the identified model and an adaptation law. However, the model cannot be always identified with certainty, since noisy data are used for model calibration and the process can change unpredictably in time, e.g., unmeasured disturbance. This can result in uncertainty in the process model that may deteriorate the control performance.

Markov decision models based control is one of the recently reported approaches for adapative control in the presence of uncertainty (Ikonen, et al., 2016). The controller tuning with Markov model can concern the closed loop performance and account for uncertainty in various system components. The basic idea of Markov models based control is that, using the first principle models of a process, the state variables, e.g., controlled and manipulated variables, are discretized into a finite set of discrete states within their effective dynamic ranges, and the evolution between states is described with transition probabilities (Negenborn, et al., 2005). Based on this predicted evolution in time, a control action can be calculated from an optimization problem defined over a finite future control horizon as done in model predictive control algorithms. Such Markov models based strategy can be used for predicting the outputs in nonlinear dynamic problems in the presence of uncertainty. However, this technique is difficult for real-time implementation, since the formulation of transition probability between states requires numerous simulations, thus it may be computationally prohibitive (Lee & Lee, 2004).

This paper addresses these computational limitations by the use of the generalized Polynomial Chaos (gPC) expansions. The idea is to develop a robust adaptive control algorithm, using a Markov decision model and uncertainty quantification techniques. Our objective is to build a basic framework to integrate the Markov model with uncertainty quantification for nonlinear process control, when only an inaccurate process model is available. The key in this work is to approximate the probability density function (PDF) of uncertainty in a process and propagate it onto manipulated and controlled variables. The PDFs to be calculated online by using gPC models can be discretized into a finite number of discrete states in a Markov model. Based on this discretization results, the transition probability between states can be readily calculated from the PDFs, thus eliminating the need for numerous simulations. Finally, using the transition probability, an optimization that minimizes a sequence of cost in the future control horizon can be defined for online controller tuning. Moreover, since a Markov model is used, the optimization can be converted into an iterative dynamic programming problem to avoid excessive simulation runs within the optimization search.

Since our objective is to adjust control parameters online, it is crucial to propagate uncertainty onto measured quantities in a computationally efficient manner and then build a Markov model in real-time. Although sampling-based methods such as Monte Carlo (MC) simulations could be used, they are time prohibitive for online implementation. Thus, the generalized polynomial chaos (gPC) expansion (Xiu, 2009) is used. The advantage of the gPC is that it can efficiently propagate the probabilistic uncertainty onto the predictions of measured quantities and quickly approximate their corresponding PDFs (Du, et al., 2017), which can be discretized to calculate the transition probability used for controller tuning. The rapid calculation of the transition probability is the key element in the proposed approach, since it is the main challenge to apply Markov models for control.

This paper is organized as follows. Section 2 presents the principal techniques used in this work.The proposed adaptive control strategy is presented in Section 3. The control strategy is illustrated for an endothermic continuous stirred tank reactor (CSTR) in Section 4. Analysis and discussion of the results are given in Section 5 followed by conclusions in Section 6.

2. THEORETICAL BACKGROUND AND PROBLEM FORMULATION

2.1 Process models

Markov models based control typically requires first principle models. Let assume a nonlinear system can be defined as:

$$\dot{\boldsymbol{x}} = f(t, \boldsymbol{x}, \boldsymbol{u}; \mathbf{g}) + v_1(t) \quad (1)$$

$$\boldsymbol{y} = h(t, \boldsymbol{x}) + v_2(t) \quad (2)$$

, where f and h are nonlinear functions and $0 \leq t \leq t_f$. $\boldsymbol{x} \in R^n$ contains the system states (including controlled variables) with initial conditions $\boldsymbol{x}(0) = \boldsymbol{x}_0$ over time domain $[0, t_f]$, $\boldsymbol{u}$ denotes the manipulated variable, $\boldsymbol{y}$ is the process outputs, v_1 and v_2 are random vectors respresenting noise, and $\mathbf{g} \in R^{n_g}$ is an unknown time varying input vector representing the uncertainty in the process. Such an uncertainty is common in chemical processes generally due to materials variablity or imperfect control. The control objective is to find an optimal tuning parameters such that controlled variables can optimally track their set points over a finite future horizon. For instance, a PID controller can be used as follows:

$$\mathrm{u} = \mathrm{u_s} + \mathrm{K_p e} + (\mathrm{K_p}/\tau_\mathrm{i}) \int_0^{t} \mathrm{e\, dt'} + \mathrm{K_p} \tau_\mathrm{d} \frac{\mathrm{de}}{\mathrm{dt}} \quad (3)$$

, where e is the error, i.e., the difference between set point and measurement of controlled variable, K_p, τ_i and τ_d are controller parameters, solved with an adjusting criterion. Although, for simplicity, we have only considered PID controllers in this work, the proposed method can be similarly extended to a state feedback controller, such as certain model predictive control (MPC) formulations (Kothare, et al., 1996; Wan & Kothare, 2002), where the gain matrix elements will be self-tuned.

2.2 Markov decision models

Markov decision models are applicable in processes involving uncertain state transitions and can enable sequential decision making. A first order Markov model is used in this work, for which the future states only depend on the current states. Additionally, it is assumed that dynamic ranges of measured quantities, i.e., $\boldsymbol{x}$ and $\boldsymbol{u}$, can be approximated with a finite sets of values. For example, the state variables $\boldsymbol{x}$ can be discretized into S disjoint regions $\{\chi_i\}$, i.e., $\chi = \cup_{i=1}^{S} \chi_i$ and $\chi_i \cap \chi_j = \phi$, where $i, j = 1, 2, \ldots S$, and each region represents a state. Details about this discretization step will be discussed in Section 3. In this way, the continuous variables such as $\boldsymbol{x}$ in (1) can be described with discrete transitions $\{\chi_i\}$. Since uncertainty such as time vaying input $\mathbf{g}$ and measurement noise are considered in this work, the evolution between χ_i and χ_j is stochastic, i.e., a state χ_i can evolve to χ_j in some time intervals, while at other time intervals χ_i may evolve to $\chi_{j'}$ rather than χ_j. This is conveniently described by a transition matrix $\boldsymbol{P} = \{p_{i,j}\}$, i.e., a $(S \times S)$ matrix, where $p_{i,j}$ is the probability that χ_i can evolve to χ_j. The process outputs $\boldsymbol{y}$ can be discretized in a similar way.

Since the transitions occur under closed loop control, the evoltuion is dependent on the controller parameters such as K_p, τ_i and τ_d in (3). For control implementation, it is necessary to discretize the space defined by a controller. The discretization is relative to state variables $\boldsymbol{x}$, and a set of states of controller parameters can be defined, i.e., $\boldsymbol{c}_\mathrm{a} \in C = \{\boldsymbol{c}_1, \boldsymbol{c}_2, \ldots, \boldsymbol{c}_{n_A}\}$, $a = 1, 2, \ldots n_A$. The discretization of controller parameters is analogous to generating a look-up table, which provides all the possible control actions. Subsequently, n_A transition matrices can be defined, i.e., $\boldsymbol{P}^a = \{p_{i,j}^a\}$, and each matrix provides the transition probability between states at two consecutive time intervals for a particular controller setting $\boldsymbol{c}_\mathrm{a}$.

Using the discretization result, an equivalent Markov model of a continuous process in (1) is defined as:

$$\mathbf{q}(\mathrm{k}+1) = \mathbf{q}(\mathrm{k})\, \mathbf{P}^{\,c_\mathrm{a}(\mathrm{k})} \quad (4)$$

, where k is a discrete time instant, $\boldsymbol{P}^{\,c_a(k)}$ is the transition matrix for a particular set of controller parameter $\boldsymbol{c}_\mathrm{a}$ at k, and $\boldsymbol{q}(k)$ and $\boldsymbol{q}(k+1)$ are the probability that a process occupies a set of states $\{\chi_i\}$ at two consecutive time instants k and $k+1$, respectively.

The next step is to formulate the probability transition matrix. Based on first principle models, the probability is often built by counting the number of observed state pairs $(\{\chi_i\}, \boldsymbol{c}_\mathrm{a})$ that lead to a particular state χ_j, and by normalizing the count with respect to the total number of transitions in each pair as below:

$$p_{i,j}^a = \frac{\#\,(\chi_\mathrm{j} \mid \{\chi_\mathrm{i}\}, \boldsymbol{c}_\mathrm{a})}{\#\,\sum(\{\chi_\mathrm{i}\}, \boldsymbol{c}_\mathrm{a})} = \frac{\#\,(\{\chi_\mathrm{j}(k+1)\} \mid \{\chi_\mathrm{i}(k)\}, \boldsymbol{c}_\mathrm{a}(k))}{\#\,\sum(\{\chi_\mathrm{i}(k)\}, \boldsymbol{c}_\mathrm{a}(k))} \quad (5)$$

, where # indicates the number of active states or the number of active transitions. Note that *active* here means transitions with a nonzero probability.

The computational time to construct the probability transition matrix is a main challenge for Markov model-based control, and the accuracy of transition probability is related to process models. For example, the total number of simulations is $(S{*}n_A{*}m)$, when m samples are used in each state in order to calculate a transition probability, since there are S dicrete states of x and n_A possible control actions in total. Also, the transition matrix may have to be repeatedly calibrated in the presence of uncertainty arising from unpredictable changes, which can complicate the calculations. To accelerate the online calculations, a gPC model is used in this work.

2.3 *Generalized polynomial chaos (gPC) expansion*

A gPC expansion estimates a random variable as a function of another random variable (e.g., ξ) with a prior known PDF (Xiu, 2009). To preserve orthogonality, the basis functions of gPC are selected according to the choice of the distribution of ξ. For a process given in (1), each element g_i (i=1,2,…,n_g) of the uncertain input **g** can be approximated with a gPC model as:

$$\mathrm{g_i} = \mathrm{g_i}(\xi_\mathrm{i}) \tag{6}$$

, where ξ_i is the i^{th} random variable. The random variables $\boldsymbol{\xi}$= {ξ_i} are independent and of equal distributions. Note that ξ_i is assumed to follow a standard distribution here, but elements in {g_i} practically can follow any distributions by including a sufficient number of basis functions in the gPC expansion.

Using gPC, the uncertainties represented by **g**, system states $\boldsymbol{x}$ and manipulated variable $\boldsymbol{u}$ can be approximated in terms of polynomial orthogonal basis functions $\Phi_k(\boldsymbol{\xi})$ as:

$$\mathrm{g_i}(\xi)=\sum_{\mathrm{k}=0}^{\infty} g_{\mathrm{i,k}}\Phi_\mathrm{k}(\xi) \tag{7}$$

$$x_\mathrm{j}(\xi)=\sum_{\mathrm{k}=0}^{\infty} x_{\mathrm{j,k}}(t)\Phi_\mathrm{k}(\xi) \tag{8}$$

$$u_\mathrm{j}(\xi)=\sum_{\mathrm{k}=0}^{\infty} u_{\mathrm{j,k}}(t)\Phi_\mathrm{k}(\xi) \tag{9}$$

, where{$\mathrm{g_{j,k}}$}, {$x_{\mathrm{j,k}}$} and {$u_{\mathrm{j,k}}$} are the gPC coefficients of the j^{th} uncertainty, the j^{th} states x, and the j^{th} manipulated variable, respectively. Also, {$\Phi_k(\xi)$} are multi-dimensional orthogonal polynomial basis functions. Uncertainty {$\mathrm{g_i}$} are assumed to be known approximately, but not accurately. In practice, (7)~(9) are often truncated into a finite number of terms. Note that gPC coefficients {$\mathrm{g_{j,k}}$} in (7) can be estimated with prior knowledge of uncertainty. Using {$\mathrm{g_{j,k}}$}, gPC coefficients {$x_{\mathrm{j,k}}$} and {$u_{\mathrm{j,k}}$} can be calculated by substituting (8) and (9) into (1) and by using a Galerkin projection with respect to each basis function {$\Phi_\mathrm{k}(\xi)$}. For breivty, the steps for the calcualtion of the gPC coefficients is not given, but the details can be found in (Du, et al., 2017; Xiu, 2009; Du, et al., 2016). Once the coefficients of $\boldsymbol{x}$ and $\boldsymbol{u}$ are calculated, their PDFs can be rapidly estimated by sampling from distributions of $\boldsymbol{\xi}$ given in (8) and (9). The ability to quickly estimate the PDFs is the key to accelerate computations of the transition matrix in this work.

3. SELF-TUNING CONTROL DESIGN

The controller parameters is adjusted by a tuning algorithm using a gPC-based finite Markov state model and a dynamic programming in this work. A Proportional-Integral-Derivative (PID) is used for algorithm illustration, since it is one of the most commonly used controller in industry.

3.1 *Markov modelling using gPC approximation*

Since the process model used in this work is assumed to be an inaccurate approximation and includes uncertainty such as **g** in (1), the gPC is used to estimate ranges of the dynamic variables and the transition matrix of $\boldsymbol{x}$. The main feature is to propagate unceratinty in (1) onto measured quantities to build a Markov model without using excessive computation. To implement the algorithm, the PDFs of **g** are approximated with gPC, but the premise is that the exact statistics of the PDFs are unknown, i.e., the gPC coefficients {$\mathrm{g_{j,k}}$} in (7) may not be accurate. For uncertainty propagation, the gPC coefficients of $\boldsymbol{x}$ are solved using Galerkin projection, from which the PDF profiles of $\boldsymbol{x}$ are estimated by sampling from the distribution of the random variables $\boldsymbol{\xi}$ and by substituting these samples into (8). Fig. 1 shows a PDF profile of a measured quantity for illustration.

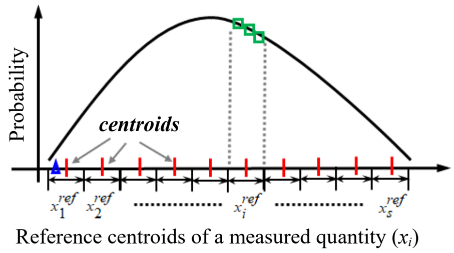

Fig. 1. Markov transition modelling

The next step for building a Markov model is to discretize the state space defined by $\boldsymbol{x}$ in (1). It is assumed that the discrete states in a Markov model can be characterized into S disjoint regions {χ_i}, i.e., $\chi=\bigcup_{i=1}^{S}\chi_i$ and $\chi_i \cap \chi_j=\phi$, where i, j=1, 2, …S. Each region is estimated with a reference centroid $\boldsymbol{x}_i^{ref}$, which represents a state and results in S reference centroids as shown in Fig. 1. To assign each sample in the PDFs to a centroid, a state index is defined with respect to a mapping ($\boldsymbol{x}\rightarrow i$) as:

$$i = arg \min_{i \in S} \left\| \boldsymbol{x} - \boldsymbol{x}_v^{ref} \right\| \tag{10}$$

, where $\boldsymbol{x}_v^{ref}$ is a reference centroid vector, ||·|| represents the Euclidean distance, and i is a state in a Markov model with a minimum distance between measurements and all reference centroids. For instance, for a given measurement (the blue triangle in Fig. 1), the smallest distance can be found with the first centroid $\boldsymbol{x}_1^{ref}$, thus implying that this measurement can be represented as *state* 1 in the Markov model. Due to uncertainty arising from model error and measurement noise, the dynamic ranges of measured quantities $\boldsymbol{x}$ in (1) have to be extended to account for all possible measurements.

Using the discretization results, the next step is to build the probability transition matrix in (5). For each sample in the approximated PDF of $\boldsymbol{x}$, a corresponding state index can be found using (10). For example, Fig. 1 indicates that 3 samples are found to be in the i^{th} reference centroid, and the probability for that state to occur is determined by normalizing 3 with respect to the total number of samples used to approximate the PDF profile. Note that the ability to calculate gPC coefficients and approximate the PDFs at each time instant are the main rationale for using the gPC, since it can significantly reduce the computational time required for building the transition matrix rather than using numerous simulations to calculate transition probabilities. To calculate the gPC coefficients of $\boldsymbol{x}$ over a finite future control horizon, the states at the current time interval are assumed to be measured and used as an initial value for the gPC model, otherwise an observer is required.

For control implementation, the space defined by the controller tuning parameters is discretized into discrete states indexed by $\boldsymbol{c}_a \in C$= {$\boldsymbol{c}_1, \boldsymbol{c}_2, …, \boldsymbol{c}_{n_A}$} using a reference vector $\boldsymbol{c}_A^{ref}$, where a=1, 2, …n_A. These states define a look-up control table. Using the Markov model and the control table, the control problem is formulated with an objective of finding a set of appropriate controller parameters from the look-up table to optimize a tuning criterion, which is discussed below.

In this work, it is assumed for simplicity that the space domain defined by the controller parameters is finite and exact. To ensure stability of controller, the nonlinear model in (1) can be linearized and used off-line to obtain stability constraints for the controller parameters that result in negative eigenvalues.

3.2 Dynamic programming

Using the gPC-based Markov model, the goal is to find an optimal set of controller parameters from the look-up table given by $\boldsymbol{c}_a \in C = \{\boldsymbol{c}_1, \boldsymbol{c}_2, \ldots, \boldsymbol{c}_{n_A}\}$ for minimizing an immediate cost over a finite future control horizon as:

$$\min_{\boldsymbol{c}_i} J(\mathbf{x}_0) = \sum_{k'=1,\cdots,K} \mu_{k'} r(\mathbf{x}(k+k'), \mathbf{u}(k+k')) \tag{11}$$

, where $\boldsymbol{c}_i \in C$ is a vector of decision variables, i.e., controller parameters, and $\boldsymbol{x}(0)$ is the initial conditions at current time k, i.e., measured quantities. $r(\boldsymbol{x}(k+k'), \boldsymbol{u}(k+k'))$ is a loss function that is defined in detail in the following section. The indexes k' in the summation cover the measured quantities in the future control horizon, i.e., $1 \le k' \le K$. The control tuning is similar to model predictive control, but the tuning criterion is based on closed loop information. The weights $\{\mu_{k'}\}$ in (11) penalize the contribution of the cost resulting from each future control horizon k'. The loss function r decides the trade-off between different control objectives, e.g., a larger probability to reach the set point in a short period of time *ver*sus the probability of aggressive movements of the manipulated variables.

Due to the transition between states in a Markov model, the cost at a particular index k' is the summation of the immediate cost r at $k'+1$ and the resulting costs at each future control horizon after $k'+1$, i.e., from $k'+2$ to $k'+K$. This yields an optimization that can be defined recursively by the Bellman equation as follows:

$$\min_{\boldsymbol{c}_i} J^*(\mathbf{x}_0) = \min_{\boldsymbol{c}_i} \{r_{k'=1}(\mathbf{x}_0, \mathbf{u}) + \mu J^*(\gamma(\mathbf{x}_0', \mathbf{u}'))\} \tag{12}$$

, where μ is an optimization weight for a future control horizon, and γ is the loss conditioned on state $(\boldsymbol{x}_0', \boldsymbol{u}')$, i.e., $k'+1$, which has the same tuning mechanism as defined in r explained in next section. The conversion of (12) leads to an iterative dynamic programming problem.

3.3 Adaptive predictive control

Based on the optimization problem (12), it is straightforward to build an adaptive control tuning algorithm. The cost defined in (12) is minimized in a closed loop system with a fixed control horizon, i.e., $0 \le k' \le K$. The optimization can start in both backward and forward manners. For instance, we can start from the last control horizon interval K and calculate the loss r. Then, we can step backward to control horizon K-1, and calculate the corresponding loss. The cost of future horizons for a state and control action pair is now the summation of the immediate loss $r(\boldsymbol{x}(k+K-1), \boldsymbol{u}(k+K-1))$ and the loss from its successor state K. Since the measured quantities $\boldsymbol{x}$ and control actions only include a finites number of states, the optimization in (12) will converge to a minimum J^*.

Since Markov models can provide probabilistic information at each state, the loss function r in (12) can be defined using the transition probability. In addition, r is also dependent on both weighted controlled and manipulated variables as follows:

$$r(\mathbf{x}, \mathbf{u}) = \alpha\{(1-p_{set}^a)(x_{set}-x_{max}^{ref})^2\} + \beta\{(1-p_{u_s}^a)(u_s-u_{max}^{ref})^2\} \tag{13}$$

, where α and β are weights, p_{set}^a is the transition probability of a particular state (reference centroid) that contains the set point of the controlled variable conditioned on a control action $\boldsymbol{c}_a$, x_{set} is the set point of controlled variable. For the manipulated variable, $p_{u_s}^a$ is the transition probability of a specific state that contains the nominal value u_s where the latter may be chosen as the steady state value of u corresponding to the chosen x_{set}. Further, x_{max}^{ref} and u_{max}^{ref} represent a state that have the maximum probability for each future discrete control horizon k,. By minimizing the cost, the tuning of controller is to find a set of controller parameters that can realize the set point tracking in a finite time, while maximizing the transition probability. Note that the cost in (13) will not converge to zero in the presence of a persistent disturbance since u_s will not be the true steady state value corresponding to x_{set}. However, the offset will still converge to zero due to the use of a controller with integral action, i.e. a PI in the current study.

4. CASE STUDY

The adaptive tuning strategy is applied to an endothermic continuous stirred tank reactor (CSTR) system (Du, et al., 2016) with a PI controller, which can be described as:

$$V_r \dot{C}_A = (F/\rho)(C_{A0} - C_A) - V_r k_0 C_A e^{\frac{E}{RT}} \tag{14}$$

$$V_r \rho C_v \dot{T} = FC_p(T_0 - T) - V_r \Delta H k_0 C_A e^{\frac{E}{RT}} + Q \tag{15}$$

$$\dot{Q} = K_p \dot{C}_A - (K_p/\tau_i)e \tag{16}$$

, where K_p and τ_i are the controller gain and integral time constant, respectively. The controller is used to control the outlet reactant concentration C_A by manipulating the external heat Q. To demonstrate the control performance, the inlet concentration C_{A0} is assumed to be the uncertainty, i.e., $\mathbf{g}$ in (1), which is operated around a fixed mean with time-invariant stochastic variations. However, the exact mean and variance of C_{A0} are unknown to the controller. The objective is to solve the optimization problem (12) and execute online tuning of the controller in the presence of the input uncertainty in C_{A0}.

5. RESULTS AND DISCUSSION

5.1 Model formulation with gPC

To formulate stability constraints of controllers, the nonlinear system defined in (14) ~ (16) is linearized using the steady state measurements. To ensure stability, eigenvalues which are functions of controller parameters, must be negative. The application of Galerkin projection for building the gPC model requires integrating the differential equations with respect to an appropriate selected polynomial basis function for a specific random variable. The integral is straightforward for monominal or polynomial terms. However, the integral of non-monominal terms such as the Arrhenius term in (16) needs approximation with a 2nd order Taylor series expansion. Due to the space limitation, details about the stability constraints and the formulation of gPC model with an approxiamtion are not given in the current work, which can be found in our previous work (Du, et al., 2016; Du, et al., 2014).

5.2 Optimal tuning of controller parameters

Using the measurements collected at each time instant *t*, the cost of the objective function (11) is minimized with respect to the tuning parameters of the *PI* controller over a finite future control horizon. As shown in Fig. 2 (a), the inlet concentration is assigned with different values between 0.8 and 1.2 gmol/L. An inset as seen in Fig. 2. (b) is used to illustrate the algorithm. The mean and variance of the inlet concentration profile of C_{A0} in Fig 2 (b) are 1.0 gmol/L and 0.1 gmol/L, respectively. To intentionally introduce mismatch in the gPC model, the mean and variance of C_{A0} are assumed to be 1.1 gmol/L and 0.15 gmol/L, respectively. Based on these values, a gPC model is generated offline which is function of controller parameters and initial values of states. This is used to decide the discrete states in a Markov model and to calculate transition probability over the finite future control horizon. Also, 1% measurement noise is added to the measured quantities.

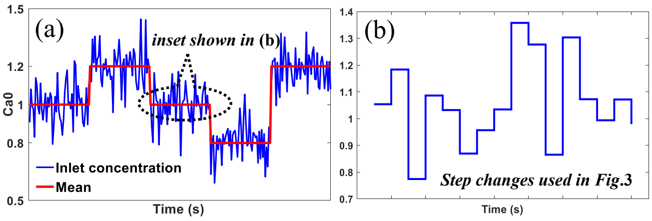

Fig. 2. Profile of inlet concentration (C_{A0})

The control performance was first studied in this work, i.e., set point tracking within finite time. Fig. 3 shows the simulation results of the inlet concentration profile C_{A0} shown in Fig. 2 (b). The control horizon used in this case study is 100, i.e., K=100. The control performance is compared to a robust controller with a set of fixed controller parameters, which was optimized offline as explained in our previous work (Du, et al., 2017). These parameters are K_p=7.5×10^4 and τ_i=0.50. For the adaptive control strategy, these values are used as the initial guesses for tuning of controller parameters. For 16 consecutive step changes of C_{A0} in Fig. 2, Fig. 3 (a) shows the results of the controlled variable (C_A). As seen, both control strategies can realize set point tracking, but the adaptive control algorithm proposed in this work can reach the set point faster, i.e., shorter transient decay. This will be further discussed below. Fig. 3 (b) shows a segment of the simulation results in Fig. 3 (a). It is observed that there are transient excursions occurring at the beginning of each switch between two step-changes in C_{A0}.

To evaluate the control performance, Table 1 shows the results of comparison between the fixed controller *versus* the adaptive controller, in terms of the transient decay time and the integral squared error (ISE) of the controlled variables, for 16 consecutive step changes shown in Fig. 2 (b). For adaptive tuning strategies, two case scenarios were investigated. A fixed transition probability in (13) is used to illustrate the necessity of updating the transition probability in real time. The improvement obtained with the adaptive controller is very significant with around 29% reduction in ISE, as compared to the fixed controller in the second row. Also, as shown in Table 1, the transient decay of the adaptive controller, on average, is about 16 s shorter than a fixed parameters controller.

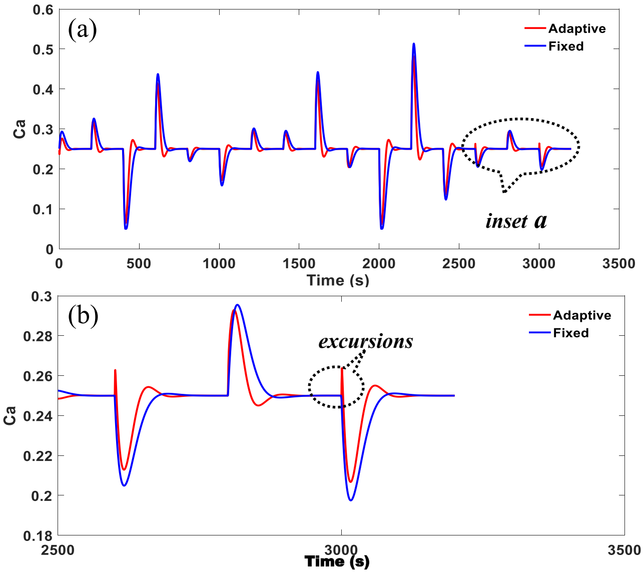

Fig. 3. Illustration of the control performance (noise-free simulation results are used for clarification)

Table 1. Evaluation of the control performance

Method	*ISE* (C_A)×10^{-4}	*Decay* (*s*)
Fixed controller	4.66	97
Adaptive (fixed p^a_{set}*)*	3.71	83
Adaptive controller	3.30	81

In a second study we studied the effect of the horizon length K on the control performance. Two different values of K are used, i.e., K=100 and K=500, respectively. Fig. 4 shows the simulation results of three samples of the inlet concentration.

Similar to the first case study, as seen in Fig. 4 (a), the controlled variable can reach the set point faster and has a smaller variability. Fig. 4 (b) and (c) shows the controller parameters, i.e., K_p and τ_i, when K is 100. For illustration, these parameters are normalized with respect to initial values used in simulations. As seen, the controller parameters exhibit high variations after step changes in C_{A0}., which is expected due to the use of relatively coarsely discretized with Markov models. The controller parameters eventually stabilize at the optimal values as seen in Fig. 4 (b) and (c).

When K is 500, it was found that both controller parameters almost remained constant through the simulations, thus resulting in more conservative control similar to the predictive controllers with long horizon. The simulation results are not shown for brevity. Note that a control horizon of 500 indexes is equivalent to a 50 s simulation in this work. As seen in Fig. 4 (a), the transient decay on average is approximately 80 s for the adaptive self-tuning control strategy, thus resulting in an almost constant controller parameter over a long period of operations. The controller parameters are found to be around $K_p = 7.55\times10^4$ and τ_i=0.31, repectively. The normalized values are K_p'=1.0066 and τ_i'= 0.62.

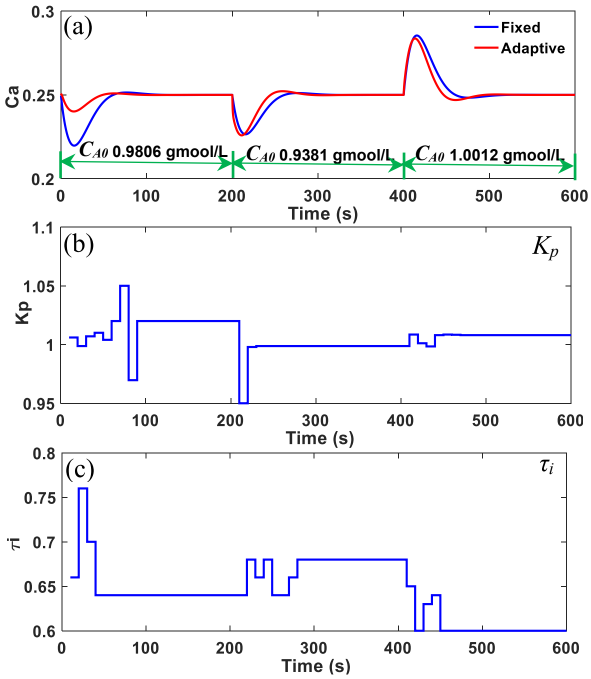

Fig. 4. Normalized tuning parameters of the *PI* controller

Fig. 5 shows the probability of the controlled variable C_A that can be found at a particular discrete state of the Markov model, which contains the set point. The first step change of C_{A0} in Fig. 4 (a) is used, i.e., C_{A0}=0.9806 gmol/L, and K is 100 for the simulations shown in Fig. 5. As seen, the probability to be in the neighbourhood of the set point is smaller during the transients that follow a change in the mean value of the inlet concentration. After approximately 38 s of the simulation, the probability increases and eventually stabilizes around 0.94. the probability is different than one due to measurement noise and the discretization of the continuous states.

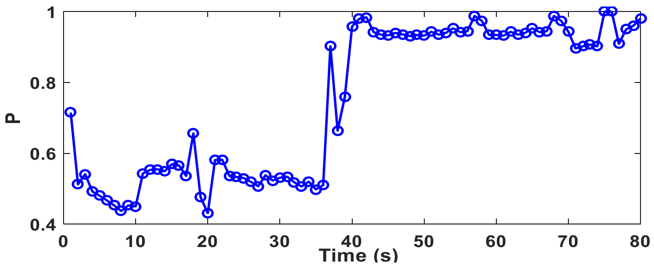

Fig. 5. Profile of transition probability

In terms of computational efficiency, using the gPC model, the calculation of the objective function (11) over a future control horizon is ~0.3 seconds on an Intel® Core™ i7 processor with dual-core. It was also found that the use of a larger number of future control horizons (K=500) has negligible effect on the computational time, which enable the online application.

6. CONCLUSIONS

A methodology is proposed for online adaptive tuning of a PID controller. The tuning procedure is based on a gPC model and a Markov model, which can predict the transitions and its probability between states of the controlled variable. Using the Markov model, the tuning of controller can be formulated as a dynamic programming problem. To overcome computational burden, a gPC model is used to predict the probability density function (PDF) of the measured quantities, which is discretized to rapidly calculate the transition probability. The combination of the Markov model with gPC-based uncertainty propagation technique is attractive for adaptive model predictive control, especially when using inaccurate modelling information.

ACKNOWLEDGMENTS

National Science Foundation (NSF-CMMI-1727487) and Natural Science and Engineering Research Council of Canada (NSERC) are acknowledged for the financial support.

REFERENCES

Du, Y., Budman, H. & Duever, T., 2014. *Integration of fault diagnosis and control by finding a trade-off between the observability of stochastic fault and economics.* Cape Town, South Africa, 2014.

Du, Y., Budman, H. & Duever, T., 2016. Integration of fault diagnosis and control based on a trade-off between fault detectability and closed loop performance. *Journal of Process Control,* Volume 38, pp. 42-53.

Du, Y., Budman, H. & Duever, T., 2017. Comparison of stochastic fault detection and classification algorithms for nonlinear chemical processes. *Computers & Chemical Engineering,* 106(2), pp. 57-70.

Du, Y., Budman, H. & Duever, T., 2017. *Robust self-tuning control under probabilistic uncertainty using generalized polynomial chaos models.* Toulouse, France, The 20th World Congress of the International Federation of Automatic Control, World Congress.

Ikonen, E., Selek, I. & Najim, K., 2016. Process control using finite Markov chains with iterative clustering. *Computers and Chemical Engineering,* Volume 93, pp. 293-308.

Kothare, M., Balakrishnan, V. & Morari, M., 1996. Robust constrained model predictive control using linear matrix inequalites. *Automatica,* 32(10), pp. 1361-1379.

Lee, J. M. & Lee, J. H., 2004. Approximate dynamic programming strategies and their applicability for process control: a review and future directions. *International Journal of Control, Automation, and Systems,* 2(3), pp. 263-278.

Negenborn, R., Schutter, B., Wiering, M. & Hellendoorn, H., 2005. *Learning based model predictive control from Markov decision processes.* Prague, Czech Republic, IFAC 16th Triennial World Congress.

Wan, Z. & Kothare, M., 2002. Robust output feedback model predictive control using off-line linear matrix inequalities. *Jornal of Process Control,* Volume 12, pp. 763-774.

Xiu, D., 2009. Fast numerical methods for stochastic computations: a review. *Communications in Computational Physics,* 5(2-4), p. 242~272.