Preprints of the
9th International Symposium on Advanced Control of Chemical Processes
The International Federation of Automatic Control
June 7-10, 2015, Whistler, British Columbia, Canada



# Model-based On-Line Optimization Framework for Semi-batch Polymerization Reactors

**Tae Y. Jung*, Yisu Nie**, J. H. Lee*, L. T. Biegler*****

** Biomolecular and Chemical Engineering Department (BK21+ Program), Korea Advanced Institute of Science and Technology, Daejeon, Korea, (e-mail : jayhlee@kaist.ac.kr).*
*** The Dow Chemical Company, Freeport, TX 77541, USA (e-mail: ynie@dow.com).*

**** Chemical Engineering Department, Carnegie Mellon University, Pittsburgh, PA 15213, USA (e-mail : biegler@cmu.edu)*

**Abstract:** Until recently, computational complexities and lack of detailed, nonlinear dynamic models have stood as main obstacles to widespread industrial adoption of model-based on-line optimization for operation of batch and semi-batch reactors. With recent advances in both dynamic modeling techniques and nonlinear programming (NLP) solvers, it is conceivable now to use significantly-sized, nonlinear models directly for on-line state/parameter estimation and optimal control calculations. In this study, we propose a framework for doing this. In the proposed framework, nonlinear first principles dynamic model-based optimizations are performed at several time points throughout a batch run over a fixed horizon, in order to estimate the current state of the model and to refine the target batch time and input variables. Here we combine shrinking horizon nonlinear model predictive control (*sh-NMPC*) with expanding horizon least squares estimation (*eh-LSE*). This framework is tested on a large-scale anionic propylene oxide (PO) polymerization process, whose operation considers not only certain end-product specifications but also safety constraints. It is shown that the proposed method is not only computationally feasible (averaging less than 10 CPU seconds at each sampling time) but leads to excellent performance, satisfying the product specification target despite initialization errors and measurement noise.

*Keywords:* Nonlinear Model Predictive Control (NMPC), Least Square Estimation (LSE), Semi-batch process, IPOPT, On-line optimization

## 1. INTRODUCTION

In recent years, batch and semi-batch processing has attracted significant attention of process control researchers owing to its growing importance in process industries and its unique set of challenges. Batch operation should consider path constraints as well as end-point constraints but the use of detailed, nonlinear dynamic models to handle this challenge has been rarely seen in industrial applications (Bonvin et al. (2001)). Main obstacles include the lack of accurate dynamic models and computational complexities associated with nonlinear dynamic models for on-line optimization (Qin and Badgwell (2003)). To overcome these obstacles, several variants of empirical-model-based control schemes have been proposed such as run-to-run end quality control integrated with on-line tracking control (Chin et al. (2000)) and with the relaxation of batch end time to satisfy end-point constraints (Lee et al. (2001)). To reduce the computational complexity, most of these works used linearized or simplified empirical model, which limited their performance and applicability (Lee and Lee (2014)). For example, this kind of simplification may not be acceptable for batch polymerization processes whose behaviour exhibit significant nonlinearity. This is because controllers for such processes have to cover a wide range of operating conditions and cope with highly nonlinear process dynamics. (Seki et al. (2001)). On the other hand, recent advances in dynamic modelling and nonlinear programming (NLP) solvers enable engineers to use large-scale nonlinear differential-algebraic equation (DAE) models directly for on-line optimization (Biegler, 2010).

In this paper, we propose a detailed dynamic model-based framework for on-line estimation and control of batch and semi-batch reactors. Here, shrinking horizon nonlinear model predictive control (*sh-NMPC*) calculates control actions along with control time intervals, on the basis of minimizing the expected batch time. Coupled with this, expanding horizon least squares estimation (*eh-LSE*) finds the current state and/or parameter estimates by minimizing the differences between model predictions and measurements, within the window starting from the initial batch time to the current time. By using the detailed nonlinear model for the entire batch time duration, we can minimize any information loss or approximation error associated with linearization-based or recursive estimation methods (e.g., extended Kalman filter). This framework is tested on a large-scale anionic polymerization of propylene oxide (PO) process for which a detailed first principles

model was developed and reformulated by a null-space projection method (Nie et al. (2013)). This operation should consider not only certain end product specifications such as the number average molecular weight (NAMW), unsaturation value, and unreacted monomer level, but also safety constraints such as the heat removal duty and adiabatic end temperature. We adopt the simultaneous collocation approach to handle potential open-loop instability and path constraints under uncertainty in the optimization. Despite the use of a large-scale, nonlinear DAE model with a full-length horizon, IPOPT finds its solution within a reasonable CPU time - averaging less than 10 CPU seconds at each sampling time. In addition, the controller leads to excellent performance satisfying the end point constraints as well as the path constraints despite the uncertainty. The flexibility to adjust the batch length is seen to be an important feature in this approach.

## 2. DYNAMIC MODEL-BASED FRAMEWORK FOR ON-LINE OPTIMIZATION

In this section, we propose a framework for a model-based on-line optimization of a semi-batch process (Fig. 1). In this framework, the recipe optimization layer calculates optimal batch time and input values to satisfy given end-point constraints and path constraints off-line. Then, as the batch progresses and on-line measurements come in, the shrinking horizon NMPC (sh-NMPC) controller refines the recipe, i.e., adjusts the control intervals (thus the total batch time) and input values to meet the end-point and path constraints, despite the uncertainties. The reason for this two-step approach, as opposed to direct formulation of NMPC without pre-calculation of target recipe, is to ensure more stable and predictable behaviour. Coupled with this, expanded horizon Least Square Estimation (ex-LSE) finds

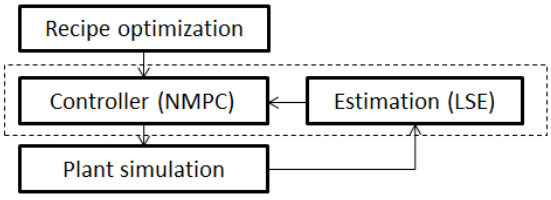


Fig. 1. A framework for model-based on-line optimization of the semi-batch process within the batch run

current state or parameters by minimizing the differences between the model predictions and measurements, within the window starting from initial batch time to the current time.

We adopt the simultaneous collocation approach to convert the dynamic optimization problems of recipe optimization, sh-NMPC and eh-LSE into NLPs. The approach offers certain advantages in terms of treating unstable, nonlinear dynamics as well as path constraints (Biegler (2010)). This method divides the continuous time horizon into finite elements (of fixed length usually). In our case study, the length of each finite element is equal to the overall batch time divided by the number of finite elements (NFEs). The control inputs are assumed to be held constant within each

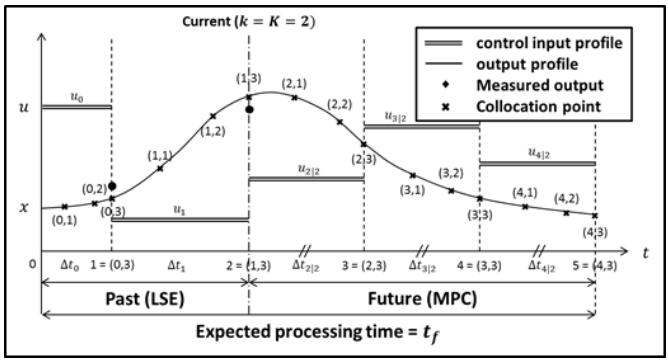


Fig. 2. Schematic diagram for the simultaneous approach-based on-line optimization of semi-batch processes (illustrated for the case of five finite elements and three orthogonal collocation points)

finite element, while the state variables are further discretized using a fixed number of orthogonal collocation points (NCPs). (Fig. 2) The details are explained in the following subsections.

### *2.1 Recipe Optimization*

The process recipe is optimized to minimize the total batch length ($t_f^s$) to satisfy endpoint constraints as well as path constraints. The discretized DAE model also acts as equality constraints. Let z and y be the differential and algebraic state variables respectively and u be the control input variables. Then, given initial estimates of the differential state variables, the recipe optimization problem can be formulated as the following NLP problem using the simultaneous approach. Here, superscript *s* refers to the variables determined by the recipe optimization,

$$\min t_f^s \tag{1a}$$

$$\begin{cases} z_{k+1}^s = f(z_k^s, y_k^s, u_k^s) \\ y_k^s = g(z_k^s, u_k^s) + \xi_k^s \\ h_i(z_k^s, y_k^s, u_k^s) \le 0 \\ h_f(z_{NFE}^s, y_{NFE}^s, u_{NFE-1}^s) \le 0 \\ z_0^s, \ldots, z_{NFE}^s \in \mathbf{Z}, y_0^s, \ldots, y_{NFE}^s \in \mathbf{Y}, u_0^s, \ldots, u_{NFE}^s \in \mathbf{U} \end{cases} \tag{1b}$$

$f(\cdot)$ and $g(\cdot)$ represent (discretized) differential and algebraic equations of our process model, respectively. Also, path constraints and end-point constraints are denoted by $h_i(\cdot)$ and $h_f(\cdot)$, and feasible domains for the state and control variables are denoted by $\mathbf{Z}$, $\mathbf{Y}$ and $\mathbf{U}$. Here, measurement noise $\xi_k^s$ is considered white.

### *2.2 Shrinking Horizon NMPC (sh-NMPC)*

Based on the dynamic set-point profiles determined off-line and real-time updated states from eh-LSE (to be discussed below), we formulate NMPC with a shrinking-horizon to handle the end-point constraints. The objective of NMPC is not only to minimize the batch length (while preserving the feasibility) but also to help track the dynamic set-points given by the recipe optimization. At current time *k=K*, the NLP for *sh-NMPC* is given as follows:

$$\min[\{\sum_{k=K}^{NFE-1}(\Delta t_{k|K}+\phi(z_{k|K},u_{k|K}))\}+\mu P] \tag{2a}$$

$$\begin{cases} \phi(z_{k|K},u_{k|K}) = \| z_{k+1|K}-z_{k+1}^{s}\|_{Q}^{2}+\| u_{k|K}-u_{k}^{s}\|_{R}^{2} \\ z_{k+1|K}=f(z_{k|K},y_{k|K},u_{k|K},\Delta t_{k|K}),\ y_{k|K}=g(z_{k|K},u_{k|K}) \\ h_i(z_{k|K},y_{k|K},u_{k|K})+\lambda\sigma_{h_i}\le 0 \\ h_f(z_{NFE|K},y_{NFE|K},u_{NFE-1|K})+\lambda\sigma_{h_f}\le P,\ P\ge 0 \\ z_{K+1|K},\ldots,z_{NFE|K}\in\mathbf{Z},\ y_{K+1|K},\ldots,y_{NFE|K}\in\mathbf{Y},\ u_{K|K},\ldots,u_{NFE-1|K}\in\mathbf{U} \end{cases} \tag{2b}$$

where $\phi(\cdot)$ is a dynamic set-point tracking error term, included to ensure more stable and predictable closed-loop behavior, and $P$ is a nonnegative variable for the end-point constraint relaxation term to guarantee feasibility of the optimization. $Q$, $R$ are the usual weight parameters for the state and control error terms, respectively and $\mu$ is the weight parameter for the relaxation which should be chosen significantly greater (by several orders of magnitude) than $Q$ and $R$. $\lambda\sigma_{h_i}$ and $\lambda\sigma_{h_f}$ are the standard deviations of the system constraints multiplied by a constant $\lambda$. These represent confidence intervals for the constraints and allow us to consider "back-offs" for the path and end-point constraints, respectively, for added robustness (Heine et al. (2006)). $\Delta t_{K|K},\ldots,\Delta t_{NFE-1|K}$ are the control time intervals corresponding to the future control inputs of $u_{K|K},\ldots,u_{NFE-1|K}$. Note that both present the degrees of freedom in the optimization. The first solution of the control input calculated from the optimization, $u_K=u_{K|K}$, is implemented along with the control time interval, $\Delta t_K=\Delta t_{K|K}$. Since $\Delta t_0,\ldots,\Delta t_{K-1}$ are already determined by the previous optimizations, the expected batch length at current time *k*=*K* can be written as

$$t_{f|K}^{e}=\sum_{k=0}^{K-1}\Delta t_k+\sum_{k=K}^{NFE-1}\Delta t_{k|K}=C+\sum_{k=K}^{NFE-1}\Delta t_{k|K} \tag{2c}$$

Since $C$ is a fixed value at every $k < K$, the objective function (2a) can be re-written as (2a'), which clearly shows that batch length is one of the terms minimized.

$$\min[\{t_{f|K}^{e}+\sum_{k=K}^{NFE-1}\phi(z_{k|K},u_{k|K})\}+\mu P] \tag{2a'}$$

Two points about this formulation are noteworthy. First, we adjust the batch time on-line to meet the end-point constraints. This is because, although estimates improve as batch time goes by, poor estimates at the beginning caused by insufficient measurement information can lead to poor control actions, which may not be compensated to meet the end-product specifications within the fixed batch time. This idea has been suggested in Lee et al. (2001), in order to extend the previous work by Chin et al. (2000), in the context of run-to-run control. The tracking error term is included here to improve robustness. Note that this formulation allows for stretched or shrunk control time intervals. However, in each optimization, future control time intervals, $\Delta t_{k|K}$, are assumed to be equal in order to simplify the computations. Thus, the objective function (2a) can be further simplified as follows.

$$\min[\{(NFE-K)\Delta t_{K|K}+\sum_{k=K}^{NFE-1}\phi(z_{k|K},u_{k|K})\}+\mu P] \tag{2a''}$$

Second, the back-off constraints or confidence margins, denoted as $\lambda\sigma_{h_i}$ and $\lambda\sigma_{h_f}$ in (2), are there to ensure the satisfaction of the constraints despite disturbances and other uncertainties that cannot be compensated for through feedback (Bonvin et al. (2001)). Although the simultaneous approach handles path constraints directly (Biegler (2007)), errors in the state and control variables caused by various uncertainties can lead to violation of important safety constraints. However, determining a reasonable back-off level is challenging because it is dependent on both the state and control variables. Also system nonlinearities make it difficult to calculate the margins using probability distributions. One empirical approach is to use data obtained by closed-loop Monte-Carlo simulation. Here, open-loop simulations will not be effective because it is hard to capture the dependency between the state and control variables.

### *2.3 Expanding Horizon LSE (eh-LSE)*

In concert with the sh-NMPC, the eh-LSE determines current state or parameter estimates by performing another NLP. In this study, we adopt a $l_2$-norm formulation in the objective function as follows:

$$\min\{\sum_{k=0}^{K}\|\frac{y_{k|K}-\hat{y}_k}{\hat{y}_k+\varepsilon}\|_{Q^{-1}}^{2}+\sum_{k=0}^{K-1}\|\frac{u_{k|K}-u_k}{u_k+\varepsilon}\|_{R^{-1}}^{2}\} \tag{3a}$$

$$\begin{cases} z_{k+1|K}=f(z_{k|K},y_{k|K},u_{k|K},\Delta t_k) \\ y_{k|K}=g(z_{k|K},u_{k|K}) \\ z_{0|K},\ldots,z_{K|K}\in\mathbf{Z},\ y_{0|K},\ldots,y_{K|K}\in\mathbf{Y},\ u_{0|K},\ldots,u_{K-1|K}\in\mathbf{U} \end{cases} \tag{3b}$$

Here, we can directly include different control time intervals $\Delta t_K$, which were determined at previous times of problem (2). Also, to make the estimation problem better conditioned, all states and parameters were normalized so that they remained within the same order of magnitude throughout the run. To reduce the effect of nonlinearity in the NLP, this normalization has been done with respect to the measurement data along with a small positive constant $\varepsilon$. Since there is no path constraint in the estimation, we only consider nonnegative constraints for the states and control variables.

One notable feature of *eh-LSE* is that, since the nonlinear model is imposed for the entire batch operation window, we can minimize information loss due to approximations like linearization or recursive Gaussian density construction (Robertson et al. (1996)). As Lee and Lee (2014) claimed, any fixed size moving time window forces the estimator to discard measurement information outside the window or to summarize it in the form of a probability density function for the beginning state. However, if the system dynamics

show significant nonlinearity, this loss may be unacceptably large. For this reason, as the estimation window expands, the estimates generally become more accurate than with EKF. In the past, computational complexity has been a major obstacle to the implementation of such a full-blown nonlinear optimization approach. On the other hand, as shown in Section 3, efficient, large-scale NLP solvers now make the implementation of such an approach a reality.

## 3. POLYMERIZATION CASE STUDY

### *3.1 Polymerization Reactor Model*

We consider a semi-batch polymerization reactor to form polyol product from propylene oxide (PO), as shown in Fig. 3. Here, pure monomer PO continuously fed into the reactor, monomer is hydrolysed, and then each polymer chain in the reactor undergoes initiation, propagation, proton-transfer and cation-exchange reactions. Since this process allows for non-isothermal operation, external heat can be provided in the start-up stage, but later significant heat released from the reactions should be removed for safety reasons.

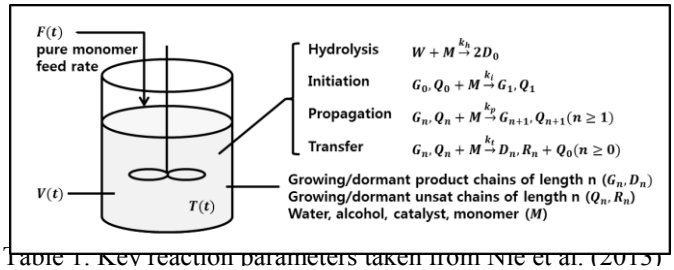


Fig. 3. The schematic diagram of propoxylation reactor

Table 1. Key reaction parameters taken from Nie et al. (2013)

| *Kinetic parameters* ($k_r = A_r exp(-E_r/RT)$) | Hydrolysis | Initiation | Propagation | Transfer |
|---|---|---|---|---|
| $A_r$ [m3/mol·s] | $8.64\times10^{4}$ | $3.964\times10^{5}$ | $1.35\times10^{4}$ | $1.509\times10^{6}$ |
| $E_r$ [J/kmol] | 82.425 | 77.822 | 69.172 | 105.018 |

| *Heat capacity coefficients* ($c_{pi} = A_i + B_iT + C_iT^2 + D_iT^3$) | $A_i$ | $B_i$ | $C_i$ | $D_i$ |
|---|---|---|---|---|
| Feed (f) | 53.347 | 0.51543 | $-1.8029\times10^{-3}$ | $2.7795\times10^{-6}$ |
| Bulk (b) | 1.10 | $2.72\times10^{-3}$ | 0 | 0 |

| *Mass parameters* | $H_2O$ | PO | KOH | Alcohol |
|---|---|---|---|---|
| Molecular weight | 18.02 | 58.08 | 56.11 | 92.09 |
| Initial mass [g] | 0.6 | 5822 | 36 | 178 |

For the model-based optimization and control of the above process, a detailed kinetic model developed by Nie et al. (2013) is used. This is a first-principles dynamic model based on the population balance equations of polymer chains and monomers, overall mass balance and energy balance. A null-space projection method is used to reformulate this model with pseudo-species and quasi-steady states. This reformulated model not only improves computational efficiency significantly but also retains enough accuracy to predict some important variables of the process such as the heat effects due to the chemical reactions. Relevant key parameters in this study are summarized in Table 1 and all other parameters are assumed to be the same as in Nie et al. (2013).

Table 2. Product specification and operating constraints in our polymerization benchmark example

| *Product specification constraints* | |
|---|---|
| Final NAMW | ≥ 3027.84 g/mol |
| Final Unsat. Value | ≤ 0.02mmol/g polyol |
| Final unreacted PO | ≤ 2000 ppm |
| *Operating conditions* | |
| Heat removal duty ≤ Allowed maximum cooling capacity : $r_p(-\Delta H_p) - F(-\Delta H_f) \le UA(T - T_w)/MW_{PO}$ = 430 J/s | |
| Adiabatic end temperature ≤ 192℃ = 465.15K | |

### *3.2 Problem Definition*

In this case study, we want to minimize the overall batch processing time, whose operation should meet both path and end-point constraints under uncertainties. Here, we assume that measurement data for all states are known in real-time. For each run, true values of the initial states and parameters are kept constant and the plant model is the same as the one used in the optimization, except with initial estimates and measurement noise generated randomly. Here normal distributions are used with standard deviations set to 5% of their nominal values, except for overall mass, which is measured more accurately, and has standard deviation set to 0.5% of its nominal value.

Manipulated variables are reactor temperature and monomer feed rate. End-point constraints are product specification constraints which include final number-average molecular weight (NAMW) of the product, maximum unsaturation value and final unreacted PO level, while path constraints are for operating conditions which include maximum heat removal duty level and upper limit of the adiabatic end temperature. The details of these constraints are summarized in Table 2 and other details in recipe optimization are introduced in Nie et al. (2013).

### *3.3 Controller Design and Implementation*

The frameworks introduced in section 2 are applied to our benchmark study and implemented in the GAMS 24.2 in NLP formulation. We choose 24 finite elements and three Gauss-Radau collocation points in order to balance the accuracy and computational load of the model, and to consider reasonable sampling or control time in our process, which is expected to average about 15 minutes.

In this study, we set weight parameters in the NMPC as $Q$=$R$=$I$ and $\mu$=$10^{10}$. Also, to determine back-off constraints in the controller, we performed 200 closed-loop Monte Carlo simulations in the presence of observation noise, with the settings mentioned in subsection 3.2. Based on the 95% confidence interval, we set the back-off constraint level as 40g/mol higher than the original final NAMW lower limit and 0.1℃ lower than the original adiabatic end temperature upper limit. The results shown in subsection 3.4 illustrate that back-off constraints make this operation feasible.

Table 3. CPU time statistics on measurement noise case and perfectly known case

| | | LSE | | | | NMPC | | |
|---|---|---|---|---|---|---|---|---|
| | | Initialization | | After Init. | | Initialization | After Init. | |
| | | Iter # | CPU(s) | Iter # | CPU(s) | CPU | Iter # | CPU(s) |
| Measurement Noise case | Average | 48.9167 | 0.1518 | 33.1667 | 0.451 | 2.8 | 57.4167 | 0.6896 |
| | Max | 123 | 0.207 | 56 | 0.89 | | 171 | 2.324 |
| | Min | 32 | 0.137 | 12 | 0.11 | | 18 | 0.13 |
| Perfectly known case | Average | 40.1667 | 0.1441 | 24.7917 | 0.3541 | 2.972 | 54.375 | 0.6824 |
| | Max | 71 | 0.166 | 31 | 0.577 | | 103 | 1.565 |
| | Min | 32 | 0.137 | 19 | 0.129 | | 26 | 0.129 |

All computations are performed on a laptop equipped with Intel ® Core™ i7-3770 CPU @ 3.40GHz process and 8.00 GB memory. The optimization package IPOPT is used for both the NMPC and LSE optimizations. CONOPT is used for the recipe optimization and initial point settings for the NMPC at the beginning. The discretized DAE model used for the optimizations has 2930 variables, which, when combined with other constraints, translate into 3029 constraints for the recipe optimization. The number of constraints for the NMPC and LSE optimizations is similar. Because of the importance of finding appropriate initial points in solving NLPs, we initialized the NMPC by an element-by-element technique, and the LSE problem by focusing on recent updates in the measurement data.

### *3.4 Simulation Result*

We compared these cases with measurement noise with the deterministic case where the state and control input variables are known. The CPU time resulting in this study is summarized in Table 3. The total computation time for performing both the LSE and NMPC calculations is well below 10 CPU seconds. The presence of measurement noise increases the computation time, especially in solving the first NMPC problem, which is poorly initialized.

To demonstrate the need for adjusting the total batch time on-line, the conventional NMPC formulation similar to the formulation in Helbig et al. (1998), is simulated where the batch time fixed to the value obtained by the recipe optimization. However, since the recipe optimization determines the minimum processing time *without disturbances*, it is clear that enforcing tighter back-off constraints with the same processing time will lead to violation of the end-point constraints (in this case, unsat. value). This can be seen in Fig. 4. To satisfy these back-off constraints we need to relax processing batch time in order to meet both path and end-point constraints.

Our results show that, with the help of on-line optimization, the operation can satisfy both the path and end-point constraints under the uncertainty (Fig. 5). However, compared to the result of the recipe optimization, the overall batch processing time is increased due to the application of the back-off constraints as well as the batch time relaxation (Table 4). Here, the back-off constraint leads to the slightly degraded controller performance because of the lowered maximum adiabatic temperature limit. Also, overachievement of NAMW (40g/mol) by the NMPC makes the processing time longer. However, its effect on the actual value (~8g/mol) is relatively small. The overachievement is due to constraint relaxation as well as estimation error, and better tuning in the controller is expected to tighten it.

## 4. CONCLUSIONS

In this study, we propose a nonlinear dynamic model-based framework for the on-line optimization of the semi-batch polymerization reactor. Based on a nonlinear, first principles model presented in Nie et al. (2013), we develop shrinking horizon NMPC coupled with expanding horizon LSE. The simultaneous approach was used to convert the dynamic optimization problem into an NLP. Despite the large model, IPOPT provides a control action within reasonable CPU time. In addition, the results show the possibility to minimize information lost due to approximation error outside the window.

To implement this framework on an industrial semi-batch polymerization reactor, several future studies are needed. First, more general observability analysis should be addressed, because on-line measurements in batch or semi-batch reactor are generally quite limited (usually just pressure and temperature). Also, process noise/disturbances should be considered. On the other hand, despite the full-length window formulation and path constraint handling, uncertainty in state and control variables may force the controller to violate path or end-point constraints. A promising but still suboptimal approach to handle this issue is to approximate the covariance of the state estimates and use them for the back-off constraints in the optimization. Since the heuristics introduced in this paper are basically limited to capture closed-loop properties, this constraint formulation study will be extended in our future work.

## ACKNOWLEDGEMENT

Financial support from the Saudi Aramco-KAIST $CO_2$ Management Center is gratefully acknowledged.

## REFERENCES

Biegler, L. T. (2010). *Nonlinear Programming: Concepts, Algorithms and Applications to Chemical Engineering, SIAM, Philadelphia, PA*

Bonvin, D., Srinivasan, B., Ruppen, D. (2001). Dynamic optimization in the batch chemical industry. In: *Proc. of CPC6*, AIChE Symp. Series, 326 (98), pp. 255-273

Chin, I. S., Lee, K. S., and Lee, J. H. (2000). A technique for integrated quality control, profile control, and constraint handling for batch processes. *I & EC Research,* 39 (3), pp. 693-705. Heine, T., Kawohl, M.,

 

and King, R. (2006). Robust model predictive control using the unscented transformation. In: *IEEE Intl. Conf. Control Applics*., pp. 224-230.

Helbig, A., Abel, O., and Marquardt, W. (1998). Model predictive control for online optimization of semi-batch reactors. In: *ACC, 1998. Proc.*, 3, pp. 1695-1699

Lee, J., Lee, K. S., Lee, J. H., and Park, S. (2001). An on-line batch span minimization and quality control strategy for batch and semi-batch processes. *Control Engineering Practice,* 9 (8), pp. 901-909.

Lee, J. H., and Lee, J. M. (2014). Progress and Challenges in Control of Chemical Processes. *Ann. Rev. Chemical and Biomolecular Engineering*, 5, pp. 383-404

Nie, Y., Biegler, L. T., Villa, C. M., Wassick, J. M. (2013). Reactor modeling and optimization of polyether polyol processes. *AIChE J.,* 59 (7), pp. 2515-2529.

Nie, Y., Biegler, L. T., Villa, C. M., Wassick J. M., (2014) Reactor Modeling and Recipe Optimization of Ring-Opening Polymerization: Block Copolymers," *I & EC Research*, 53, 18, 7434-7446

Qin, S. J., and Badgwell, T. A. (2003). A survey of industrial model predictive control technology. *Control Engineering Practice,* 11 (7), pp. 733-764.

Robertson, D. G., Lee, J. H.., Rawlings, J. B. (1996). A moving horizon-based approach for least-squares estimation. *AIChE Journal,* 42 (8), pp. 2209-2224.

Seki, H., Ogawa, M., Ooyama, S., Akamatsu, K., Ohshima, M., and Yang, W. (2001). Industrial application of NMPC to polymerization reactors. *Control Engineering Practice,* 9 (8), pp. 819-828.

Table 4. Some significant statistics for both recipe optimization and on-line simulation result

| | NAMW [g/mol] | Unsaturation value [mmol/g polyol] | Unreacted PO [ppm] | Max. adiabatic temp. [C] | Processing time [min] |
|---|---|---|---|---|---|
| Target | 3027.84 | 0.02 | 2000 | 192 | 397.1733 |
| Fixed time | 3028.71 | 0.0218 | 246.07 | 191.9028 | 397.1733 |
| Relaxed time | 3035.84 | 0.0198 | 65.2425 | 191.8976 | 486.7134 |

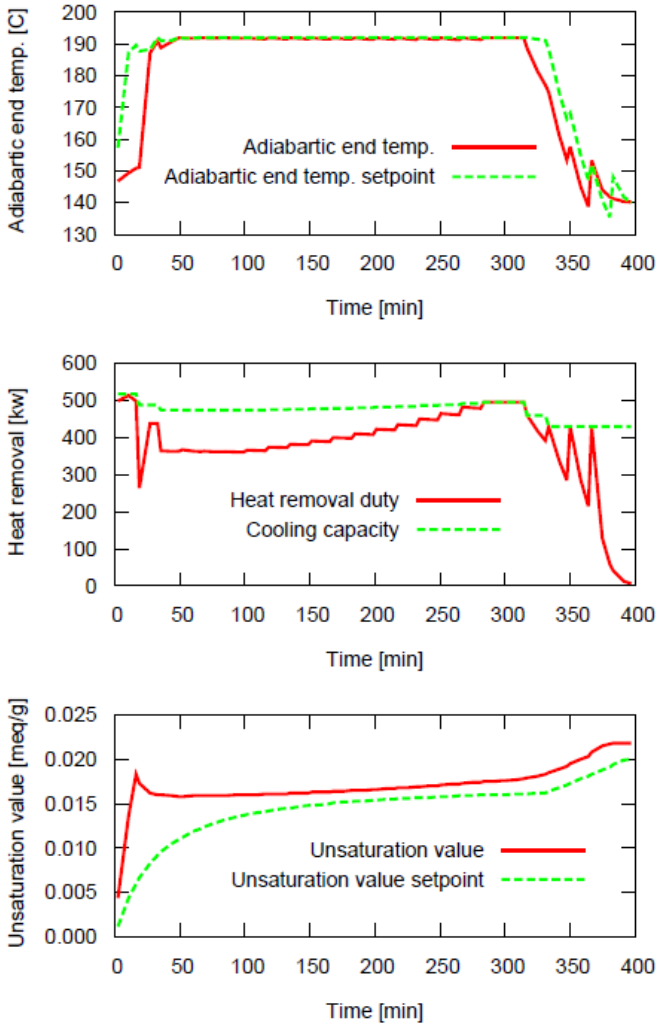


Fig. 4. Simulation profiles of the fixed time formulation

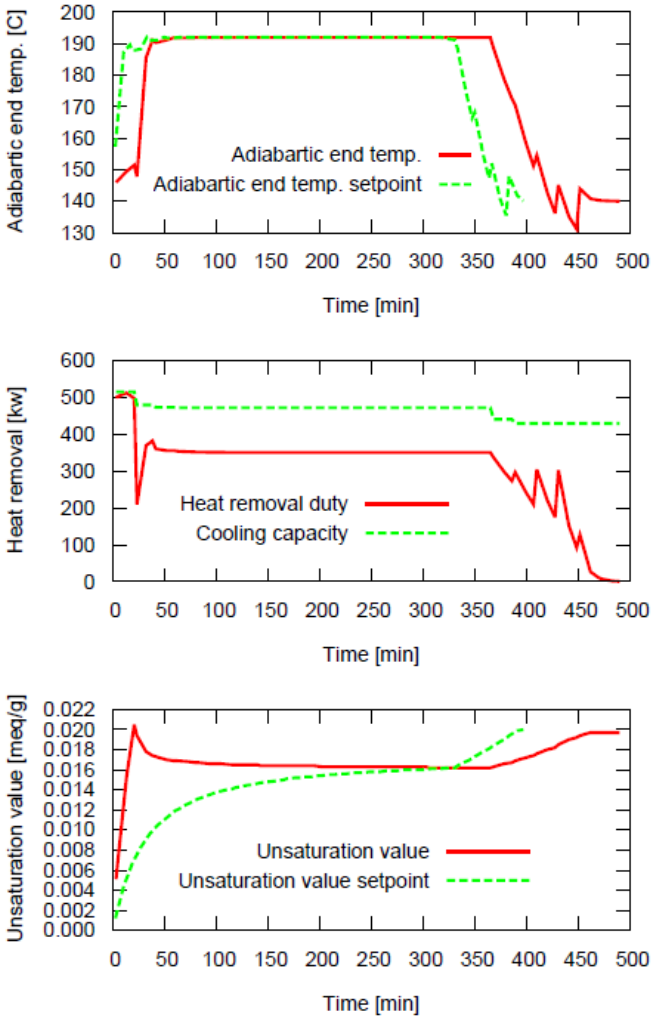


Fig. 5. Simulation profiles of the relaxed time formulation