Data-Driven Modelling and Control of Batch Processes

Data-Driven Modelling and Control of Batch Processes

by

Aswin Chandrasekar

A Thesis Submitted to the School of Graduate Studies in the Partial Fulfillment of the Requirements for the Degree Doctor of Philosophy

Doctor of Philosophy (2024) McMaster University

(Chemical Engineering) Hamilton, Ontario, Canada

TITLE: Data-Driven Modelling and Control of Batch Processes

AUTHOR: Aswin Chandrasekar

(McMaster University, Hamilton, ON)

SUPERVISOR: Dr. Prashant Mhaskar

NUMBER OF PAGES: xiv, 184

karmaṇy-evādhikāras te mā phaleṣhu kadāchana /

mā karma-phala-hetur bhūr mā te saṅgo 'stvakarmaṇi //

You have the right to work only but never to its fruits.

Let not the fruits of action be your motive, nor let your attachment be to inaction.

ABSTRACT

The primary focus of this thesis is to facilitate the modelling of batch processes considering quality variables in the context of model-based quality control and provide direct quality control formulations as opposed to existing techniques which are primarily focused on trajectory control of measured variables. These quality variables, which are unique to batch processes, cannot be measured during a batch run and can only be assessed at the end of the batch. The ability to implement good quality control is dependent on the ability to capture the complex dynamic behavior of batch processes. The first challenge is that of handling process non-linearities and/or multiple phases while being cognizant of the fact that a relatively modest amount of informative data is available, making the recently developed deep-learning-based machine-learning techniques not directly implementable. Yet another challenge/opportunity that exists is in situations where traditional sensors such as thermocouples may not be possible to implement in practice, but feedback may be available through non-traditional sensors such as thermal images.

The present thesis addresses the above-mentioned challenges and demonstrates the approaches on a pilot-scale experimental setup. In the first contribution, a data-driven model-based economic control formulation was initially developed and implemented on a batch Rotational Molding process to achieve product specifications through constraints on the predicted quality variables, while either minimizing the total input consumption or further maximizing the product quality. A Linear Time-Invariant (LTI) State Space (SS) model is used in conjunction with a Partial Least Squares (PLS) quality model to model the process and quality variables. The next contribution presents one way of handling process nonlinearity. An adaptive modelling strategy unique to batch processes was proposed to alleviate this issue and implemented on the Rotational molding process to continuously adapt the LTI SS model during a batch run. The next contribution leveraged the nonlinearity-capturing capabilities of modelling techniques like Neural Networks (NN) while handling the overfitting prob-

lem. The key idea in this approach was to use a subspace identification approach to first determine the state trajectory evolution followed by a Recurrent Neural Network (RNN) to model the non-linear process dynamics. This approach, along with the previous PLS quality model, also exhibited superior dynamic and quality predictions compared to a standard RNN case. In a departure from existing approaches, where the dynamic model and quality model are identified separately, an adaptation of the prediction error minimization framework was proposed where the dynamic model and the quality model are identified simultaneously, resulting in an improved unified model. Finally, the opportunity/challenge of non-traditional sensors was addressed. A framework was proposed where first, the high dimensional output, a thermal image in the specific example batch process, is reduced using a suitable dimensionality reduction technique to a set of latent features, which then would be used as the outputs of the dynamic model in any of the previously discussed approaches. The proposed modelling framework was implemented as a part of a Model Predictive Controller implementation using the thermal images as feedback to produce the desired product.

ACKNOWLEDGEMENTS

I would like to begin by expressing my deepest gratitude to my supervisor, Dr. Prashant Mhaskar. His unwavering support and constant inspiration have been invaluable to me. I cannot thank him enough for the assistance he has provided. His guidance, both as an academic supervisor and on a personal level, has helped me stay motivated and secure since I arrived in Canada and began my journey as an international student. His technical expertise and ability to simplify complex ideas have given me clarity in my thoughts. His cheerful, enthusiastic, and supportive nature has kept me mentally positive even during the most challenging times. I have always felt a profound connection with him, and his humility and modesty have inspired me to lead my life in a similar manner.

Next, I would like to extend my sincere gratitude to Dr. Shahin Sirouspour and Dr. Kamil Khan for agreeing to be my supervisory committee members. Their insightful advice, invaluable questions, and suggestions have significantly enhanced the quality of this research. Their guidance has been critical to the successful completion of my studies.

I would also like to thank my friends for their support and encouragement during challenging times. Finally, my deepest gratitude goes to my parents, Chandrasekar and Shobana, for believing in me and supporting my decisions. Their unconditional love, moral support, and endless encouragement have helped me overcome the toughest moments and given me the strength to push forward.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background and motivation

Batch processes are essential in the chemical process industries and are widely employed for manufacturing high-quality products. Their popularity largely stems from their flexibility, which allows for the production of different product grades by altering initial conditions and input trajectories. However, batch processes are defined by the lack of equilibrium conditions and exhibit highly nonlinear and time-varying dynamics, making traditional approaches for continuous processes inapplicable. Typically, a batch process involves the following main steps:

1. Charging the reactor with ingredients according to a predefined recipe
2. Conducting a transformation process over a set period
3. Terminating the process upon meeting specific criteria

A distinctive feature of batch processes is that along with certain process variables that are measured at every sampling instance during the batch, there are other, 'quality' variables that can only be measured at the batch's end. Additionally, some of the process variables measurements may be unconventional, such as ultrasound or thermal images. There are many examples of such batch processes where the main quality variables of interest are not measured during the batch run but are only measured once after the run has terminated. For example, in a batch crystallization process, the particle size distribution which cannot be measured during a batch run is a key variable of interest that needs to be controlled. Another example is the steelmaking process, where typically it is desired to obtain the final product (steel) with carbon content below a certain user-defined threshold, and this carbon mass fraction is impossible to determine during an ongoing process. The primary objective in a typical batch process is to meet the final product quality requirements Since quality variables,

which are crucial in a batch process, cannot be measured until the batch has ended, it is not possible to directly control the product quality during the operation since direct control methods like PI/PID need the controlled variables to be measured at every time instance.

Model Predictive Control (MPC) has been a standard control technology used for many processes in the industry (Forbes et al., 2015). An MPC has an optimizer that uses a dynamic model of the process to predict the future trajectory of the process variables given an input sequence. For the MPC to function effectively, the model has to capture not only the behavior of the process variables that are measured continuously during the process but also the quality variables. In the context of continuous processes, data-driven models have been increasingly preferred in the past for this purpose in the absence of first principles knowledge or any physics-based information relating to the process variables. There has been research specifically aimed at developing model-based predictive control for batch processes, specifically for the processes mentioned above (Kwon et al., 2014; Shi et al., 2006; Bekker et al., 2000; Rashid et al., 2016). In these studies, however, the key control variables were measured during the process, as they were based on simulation case studies. This approach, though, may not be feasible in real-life scenarios, and therefore, a more appropriate strategy needs to be developed. There are few data-driven approaches that focus on batch processes as well, especially linking the quality variables with the process variables. For example, linear modelling techniques like Partial Least Squares (PLS) regression have been successfully used for modelling and control of batch processes in the past (Flores-Cerrillo and MacGregor, 2002; Flores Cerrillo and MacGregor, 2005). However, these techniques require all batches to be of the same length and moreover, do not distinguish between the inputs and outputs, which is necessary for feedback control. On the other hand, LTI SS models have been used along with PLS models (Corbett and Mhaskar, 2016) for linear dynamic and quality modelling. Hence these models can be used for quality control in batch processes.

One has to be mindful when using the approach of (Corbett and Mhaskar, 2016) to design the control formulation since the optimizer in the controller could easily give an energy-intensive input profile to achieve a product with good qualities. Moreover, this approach with a single linear model for the entire batch does not target modelling process non-linearities, which are essentially handled through feedback. One way to counter the non-linearities is by adapting the model continuously during an ongoing batch. There have been significant contributions in the area of model identification or adaptive model control in the past but in the context of continuous processes. There is a need to distinguish between continuous and batch processes and requires careful consideration while designing a reidentification strategy for batch quality control.

Strong indications of process nonlinearities necessitate a direct approach to address these nonlinearities, such as fitting a nonlinear model itself. Nonlinear machine learning-based models, like neural networks, which have gained increasing popularity in the process modelling and control domain (Hunt et al., 1992; Sjöberg et al., 1995; Tian et al., 2001; Wu et al., 2020), offer an appealing alternative. However, there are still several challenges associated with neural networks, the most common being overfitting. Factors such as hyperparameter tuning, network complexity, training times, and the availability of process data required to train a neural network model all directly or indirectly contribute to the overfitting issue. Hybrid approaches have been explored in the past (Sansana et al., 2021; Ghosh et al., 2019; Bangi et al., 2022), where first principles models are used to constrain neural network training and thus prevent overfitting. However, such information may not always be available. Additionally, while subspace methods are well-established and known for their efficiency in identifying linear models, there remains an opportunity to leverage the underlying rigour of these techniques in fitting neural networks, which will be explored in this thesis.

It is important to note that existing techniques for modelling the process and quality

variables either bypass modelling the dynamics and directly link the process variable trajectory with the quality variables or first model the dynamics and then identify the quality model. Currently, there is no unified optimization-based framework in which both models are identified simultaneously. In batch processes, quality variables are directly correlated with the dynamics of the process variables. Therefore, they should influence dynamic model identification, and conversely, process variables should impact quality model identification concurrently.

Finally, in many industrial batch processes, standard measuring devices such as thermocouples may not be installed due to physical limitations. Instead, unconventional instruments like thermal images or ultrasound provide high-dimensional feedback. Existing techniques for both batch and continuous processes cannot be directly applied for quality control in such scenarios, especially when there's no static (or algebraic) correlation between these high-dimensional feedbacks and the quality variables. Therefore, careful consideration is essential during the design of a modelling strategy, and even during the data collection stage, to effectively utilize all information from a batch process, including these high-dimensional variables, and to effectively use them to model the dynamics and capture the quality variables.

This thesis aims to solve these issues by proposing appropriate modelling and control strategies and implementing them on an actual batch process, specifically the Uniaxial and the Biaxial Rotational Molding processes. It is to be noted that the techniques proposed in this thesis are general to any kind of batch process that has a similar input-output-quality variable structure. The thesis simply demonstrates the working of these approaches by using experiments of example batch processes. In a Rotational Molding process, a specific amount of polymer is first fed into the mold and a hollow plastic product is formed at the end of the process. There are certain inputs and outputs that are measured during the batch, but the product qualities, namely the sinkhole area percentage and the impact strength of the product, can only be measured

once a batch is complete, the product is removed and finally when analysed separately.

In the Uniaxial Rotational Molding process, the internal mold temperature is the measured output, however, the oven temperatures are also recorded. Existing PI control strategies simply aim to maintain the oven temperature at a certain setpoint. The only decision variable here is the time for which the batch is operated, which is presently done based on past experience. Thus, to determine the appropriate operation, various experiments are carried out where the box is kept in the oven for varying times. This makes the 'design' of this process challenging and misses out on improved (and possibly energy efficient) quality control by implementing other variations of the heating input profile.

A model-based approach would be useful in determining the optimal heater input profile, with the ability to implement it in closed-loop to reject disturbances. There are no first principles models available that relate the quality variables of interest with the process variables and hence data-driven approaches can be opted for (note that it is easy to build models between the heater input and the temperature, but a model between the heater input and the quality variables presently does not exist). Additionally, it is known that the process exhibits complex non-linear behaviour due to the existence of multiple phases. So it is important to carefully consider all these challenges while coming up with a suitable framework.

In the Biaxial Rotational Molding process, it is physically impossible to have a thermocouple inside the mold, so no internal mold temperature readings are available. There is a thermocouple that is in a corner of the oven chamber that gives the oven temperature readings. The existing PI-based control strategy aims to maintain this oven temperature at a certain setpoint for a set duration of time. This batch time is the only decision variable here. A thermal camera is available, which can capture images of the mold at every time instance during the process and these thermal images (each image consisting of 90×120 temperatures) act as the outputs of the process. It

would be a laborious task to tune the PI's if a PI-based solution were to be applied in this case like in the previous case, since a PI controller would have to be designed for each of the $90 \times 120 = 10800$ outputs. More importantly, computing one average temperature from the thermal readings and controlling that would result in a missed opportunity in terms of the greater information available in the thermal image. This problem of building dynamic models from thermal images, and using them in the context of quality control problems, has as yet remained unaddressed.

1.2 Thesis outline

Considering the challenges that exist and are yet to be solved in the domain of quality modelling and control as mentioned previously, this thesis aims to address these issues by proposing a suitable framework and demonstrating it on an actual batch experimental setup. The thesis is organized as follows:

In Chapter 2, a data-driven modelling and control framework was devised to enable quality control for batch processes in the absence of first principles/physics-based equations relating the quality variables with the process variables. Particularly, a subspace-based dynamic and quality modelling approach is adopted and the same is integrated within an Economic Model Predictive Control (EMPC) designed for 1) minimizing input consumption, or 2) maximizing the product quality. The working of the proposed modelling and control framework is demonstrated on a lab-scale Uniaxial Rotational Molding process.

Realizing that the LTI SS dynamic model could be improved upon in predicting the output trajectory in the previous example batch process, in Chapter 3, an adaptive modelling strategy is proposed for quality control of batch processes. This could be due to the non-linearities and/or multiple phases occurring in the batch process,

which could be the case in many other batch processes as well. Specifically, the batch nature of the process data is utilized to carefully select only the relevant portions of past data for model re-identification of the subspace-based dynamic and quality models during an ongoing batch. The performance of the proposed re-identification strategy under closed-loop control is compared with existing ones and also with a case with no re-identification strategy.

In some batch processes, the severity of the non-linearity is too high and hence even an adaptive modelling strategy may not be sufficient. In Chapter 4, a non-linear modelling approach is proposed to address the non-linear dynamics. Although the most popular choice would be to use an RNN, the proposed approach explores the possibility of leveraging the robustness of subspace identification algorithms to alleviate the neural network identification problem further to reduce overfitting. In particular, subspace identification is applied first on the batch data to retrieve the Kalman-filtered state trajectories. Then a NARX network is constructed on these states and the inputs to obtain the state equation, along with the output equation obtained directly from the previously implemented subspace algorithm. The quality model is constructed in the same way as mentioned in Chapters 2 and 3. The ability of the proposed hybrid modelling approach to predict the outputs, and mainly the qualities, given an input sequence, is compared with a case where an RNN is directly fit on the input-output trajectories.

In Chapter 5, an alternate train of thought was pursued where, instead of refining the models by handling the non-linearities in the dynamics, both the dynamic and the quality models were refined in a combined optimization framework. Up until now, these models would be identified separately, one after the other i.e., the dynamic model would first get identified, followed by the quality model to relate the qualities with the states obtained from the dynamic model. In the proposed approach, the overall modelling structure is preserved but both models are identified simultaneously.

A prediction error minimization-based optimization formulation is adopted, which results in an NLP problem. The models obtained from the previous approach would be used as the initial guess for the NLP problem for further refining the models. The proposed approach is evaluated on the same Uniaxial Rotational Molding batch process described in the previous chapters and its performance is compared with the previous approach in predicting the process outputs and also mainly the quality variables.

In Chapter 6, the main issue of handling high-dimensional outputs and incorporating them directly within the dynamic and quality modelling framework was addressed. The core modelling strategy is adopted from Chapter 2 with the addition of more model layers, where each layer pre-processes the high dimensional output to finally reduce it to a set of latent variables, which then would represent the dynamic outputs of the process. The proposed modelling strategy is deployed within an EMPC framework on a lab-scale Biaxial Rotational Molding batch process where the high dimensional output is in the form of a thermal image. The performance of the proposed modelling approach is evaluated in its ability to predict the quality variables and also to achieve closed-loop quality control when integrated within an EMPC framework.

Bibliography

Bangi, M. S. F., Kao, K., and Kwon, J. S.-I. (2022). Physics-informed neural networks for hybrid modeling of lab-scale batch fermentation for β-carotene production using saccharomyces cerevisiae. *Chemical Engineering Research and Design*, 179:415–423.

Bekker, J., Craig, I., and Pistorius, P. (2000). Model predictive control of an electric arc furnace off-gas process. *Control Engineering Practice*, 8(4):445–455.

Corbett, B. and Mhaskar, P. (2016). Subspace identification for data-driven modeling

and quality control of batch processes. *AIChE Journal*, 62(5):1581–1601.

Flores-Cerrillo, J. and MacGregor, J. F. (2002). Control of particle size distributions in emulsion semibatch polymerization using mid-course correction policies. *Industrial and Engineering Chemistry Research*, 41(7):1805–1814.

Flores Cerrillo, J. and MacGregor, J. F. (2005). Latent variable MPC for trajectory tracking in batch processes. *Journal of Process Control*, 15(6):651–663.

Forbes, M. G., Patwardhan, R. S., Hamadah, H., and Gopaluni, R. B. (2015). Model predictive control in industry: Challenges and opportunities. *IFAC-PapersOnLine*, 48(8):531–538. 9th IFAC Symposium on Advanced Control of Chemical Processes ADCHEM 2015.

Ghosh, D., Hermonat, E., Mhaskar, P., Snowling, S., and Goel, R. (2019). Hybrid modeling approach integrating first-principles models with subspace identification. *Industrial & Engineering Chemistry Research*, 58(30):13533–13543.

Hunt, K., Sbarbaro, D., Żbikowski, R., and Gawthrop, P. (1992). Neural networks for control systems—a survey. *Automatica*, 28(6):1083–1112.

Kwon, J. S.-I., Nayhouse, M., Christofides, P. D., and Orkoulas, G. (2014). Modeling and control of crystal shape in continuous protein crystallization. *Chemical Engineering Science*, 107:47–57.

Rashid, M. M., Mhaskar, P., and Swartz, C. L. (2016). Multi-rate modeling and economic model predictive control of the electric arc furnace. *Journal of Process Control*, 40:50–61.

Sansana, J., Joswiak, M. N., Castillo, I., Wang, Z., Rendall, R., Chiang, L. H., and Reis, M. S. (2021). Recent trends on hybrid modeling for industry 4.0. *Computers & Chemical Engineering*, 151:107365.

Shi, D., El-Farra, N. H., Li, M., Mhaskar, P., and Christofides, P. D. (2006). Predictive control of particle size distribution in particulate processes. *Chemical Engineering Science*, 61(1):268–281.

Sjöberg, J., Zhang, Q., Ljung, L., Benveniste, A., Delyon, B., Glorennec, P.-Y., Hjalmarsson, H., and Juditsky, A. (1995). Nonlinear black-box modeling in system identification: a unified overview. *Automatica*, 31(12):1691–1724. Trends in System Identification.

Tian, Y., Zhang, J., and Morris, J. (2001). Modeling and optimal control of a batch polymerization reactor using a hybrid stacked recurrent neural network model. *Industrial & Engineering Chemistry Research*, 40(21):4525–4535.

Wu, Z., Rincon, D., and Christofides, P. D. (2020). Process structure-based recurrent neural network modeling for model predictive control of nonlinear processes. *Journal of Process Control*, 89:74–84.

Chapter 2

Economic Quality Model Predictive Control in Batch Processes

The contents of this chapter have been published in Computers & Chemical Engineering Journal.

- Chandrasekar, Aswin, Abhinav Garg, Hassan A. Abdulhussain, Vladimir Gritsichine, Michael R. Thompson, and Prashant Mhaskar. "Design and application of data driven economic model predictive control for a rotational molding process." Computers & Chemical Engineering 161 (2022): 107713.

The following is the authorship contribution statement: Aswin Chandrasekar: Conceptualization, Methodology, Formal analysis, Software, Writing – original draft. Abhinav Garg: Conceptualization, Methodology, Software, Writing – original draft. Hassan A. Abdulhussain: Data curation. Vladimir Gritsichine: Data curation. Michael R. Thompson: Writing – review & editing, Supervision, Project administration, Funding acquisition. Prashant Mhaskar: Conceptualization, Methodology, Writing – review & editing, Supervision, Project administration, Funding acquisition.

Abstract

The present manuscript addresses the problem of economically achieving a user specified set of product qualities in an industrial complex batch process, illustrated through a lab-scale uni-axial rotational molding (also known as Rotomolding) setup. To this end, a data driven Economic MPC (EMPC) formulation is developed and implemented to achieve product specification via constraints on the predicted quality variables. First, a state-space dynamic model of the rotomolding process is built using previous batch data generated in the lab using uni-axial rotomolding setup. The dynamic model captures the internal mold temperature trajectory for an input sequence (combination of two heaters and compressed air). This model is then supplemented by a partial-least-squares quality model, which relates with key quality variables (sinkhole area and impact energy) with the terminal (states) prediction. The complete model is then placed within the EMPC scheme which minimizes the cost associated with inputs and allows the user to specify required product quality via constraints on the quality variables. Results achieved from experimental studies illustrate the capability of the proposed EMPC scheme in lowering the process cost (energy requirements) for two manifestations of the economic objective.

2.1 Introduction

Rotational Molding (otherwise called rotomolding or rotocasting) is an industrial batch process consisting of distinct heating and cooling phases, designed for the fabrication of hollow seamless plastic products (Gomes et al., 2018). A mold is charged with polymer resin and is rotated in an oven where the heater inputs are controlled. This causes the resin to soften and adhere to the walls. Continuous rotation throughout the heating and the cooling phases ensures that the walls of the formed product have uniform thickness and consistent density. Rotomolding is preferred in producing

large products such as traffic cones or water tanks, and can significantly benefit from optimal control over the product quality in order to lower significant waste costs to a company; one poorly formed tank for example can result in tens of kilograms of plastic going to landfill. The main aim in rotational molding is to obtain high quality products of good appearance with desired mechanical properties, which is achieved by controlling the sintering, densification and solidification stages while avoiding degradation due to overheating. A quality control framework capable of restricting the input energy consumption, and determining optimal input trajectory for meeting user-specified bounds on the quality variables is required to minimize wastage, especially for process such as rotomolding where the measurements of quality variables are not available online (Garg et al., 2019).

One approach to achieve the desired quality control objectives is to develop a first principles model (based on laws of physics) that captures the evolution of process variables given an input sequence. This model can then be integrated with a model predictive controller (MPC) to achieve the product quality as per user's requirements. But arriving at the appropriate equations using the knowledge of the discrete particle dynamics, polymer rheology and heat transfer of a process like rotomolding will be a tedious task and may result in a complicated model with a huge number of parameters. A complicated model within an optimizer to solve for the optimal control sequence may often result in sub-optimal solutions, which is contrary to the purpose of a first principles' framework. Thus, this type of model predictive controller (MPC) (Bonvin, 1998) remains unfavourable for rotomolding.

There is a naïve method needing significant experimental effort initially to determine a suitable input trajectory for the desired product quality. Once an input sequence is set, the same profile is implemented, hoping that the user specified qualities can be obtained. This approach is easy to implement since it does not require any sort of a model. However, this approach is vulnerable to disturbances due to its open-loop

nature, and an entirely new set of experiments would be needed to determine optimal process operations for a different specified product quality. There exists another approach which is capable of partially rejecting disturbances, known as trajectory tracking (Abu-Al-Nadi et al., 2005). Yet another approach exists where, an output or key measurable process variable (herein, the internal mold temperature), can be tracked by a predefined set-point trajectory. However, a product with desired qualities may not be obtained even with perfect trajectory tracking because of varying process conditions across batches which in turn causes changing relationship between the measured/tracked variable and the final quality variable.

A solution to this problem is the use of a modeling technique that captures not only the process dynamics but also the dependency of quality variables on the input variables. Partial least squares (PLS) is a popular and widely used technique in modelling dynamic batch processes. PLS projects the key information of process dynamics onto a latent space (Flores-Cerrillo and MacGregor, 2002; Flores Cerrillo and MacGregor, 2005). The structure of the PLS model bears resemblance to linear time-varying (LTV) models which predict deviations from mean process trajectories and relates these to the differences in final product quality. But the PLS approach demands all the batches to be of equal duration, which is rarely the case. A solution is to identify a suitable alignment variable for the training and testing batches, but in doing so, some vital information about the process trajectory might get adjusted unnecessarily. A larger concern is the fact that PLS based approaches do not intrinsically distinguish between input and output variables and as such, the use of PLS techniques for economic control of a process like rotational molding remains an open problem. An alternative yet preferred data-driven modelling technique for this focused work on rotomolding is subspace identification where one builds a model with a certain structure, namely a linear time invariant (LTI) model, based on experimental batch data. The key is that by including previous experiments with varying product quality results, the model captures the dynamics of the system well enough so that it enhances

the performance of the predictive controller and aids the economics of the operation. Data-driven model designs for continuous processes(Ljung, 1998; Kadali et al., 2003; Pour et al., 2010) have been adapted for batch processes, with a batch modeling and control approach based on subspace identification, recently demonstrated (Corbett and Mhaskar, 2016). The proposed approach permits batches of variable duration to be used to build the model without the need for any batch alignment.

This modeling approach was demonstrated recently to achieve desired product qualities in a rotomolding process. The results (Garg et al., 2019) show the performance and success of the approach in achieving best product qualities through validation batches. Very often, to obtain a high quality product, the optimization algorithm tended to produce energy-intensive input profiles, which will foreseeably impact the economics of the process. Hence, a thorough redesign of the optimizer is required, restricting the input usage while meeting product quality requirements in the form of constraints. Persuaded by these challenges, this work presents a modeling and control approach based on subspace identification, to handle user-specific qualities via constraints and illustrates explicitly the ability to achieve economic predictive control in a rotomolding process. For the present study, two manifestations of the economic objective are considered; one that simply 'maximizes' the product quality and the other where total input energy consumption is minimized, while respecting constraints on the quality variables. The key change that has been made for this work is in the objective function which now penalises the excess input usage, in contrast to the previous work, where the objective function penalises sub-standard qualities. This in turn results in significantly different closed-loop behavior and demonstrates the ability of the proposed approach to maintain model validity in a wide range of operating conditions and truly implement economic MPC. The rest of the paper is organized as follows: Section 2.2 describes the rotomolding process and the lab-scale uni-axial rotomolding setup. It also reviews the batch subspace identification approach. The proposed MPC design is presented in Section 2.3 and closed-loop experimental results

are presented in Section 2.4. The concluding remarks are finally given in Section 2.5.

2.2 Preliminaries

A brief overview of the rotomolding process investigated in this work and modeling technique for batch process data based on subspace identification is given in this section. The foundation for the novel design of EMPC is given in this section and its corresponding closed-loop results are presented in Sections 2.3 and 2.4 respectively.

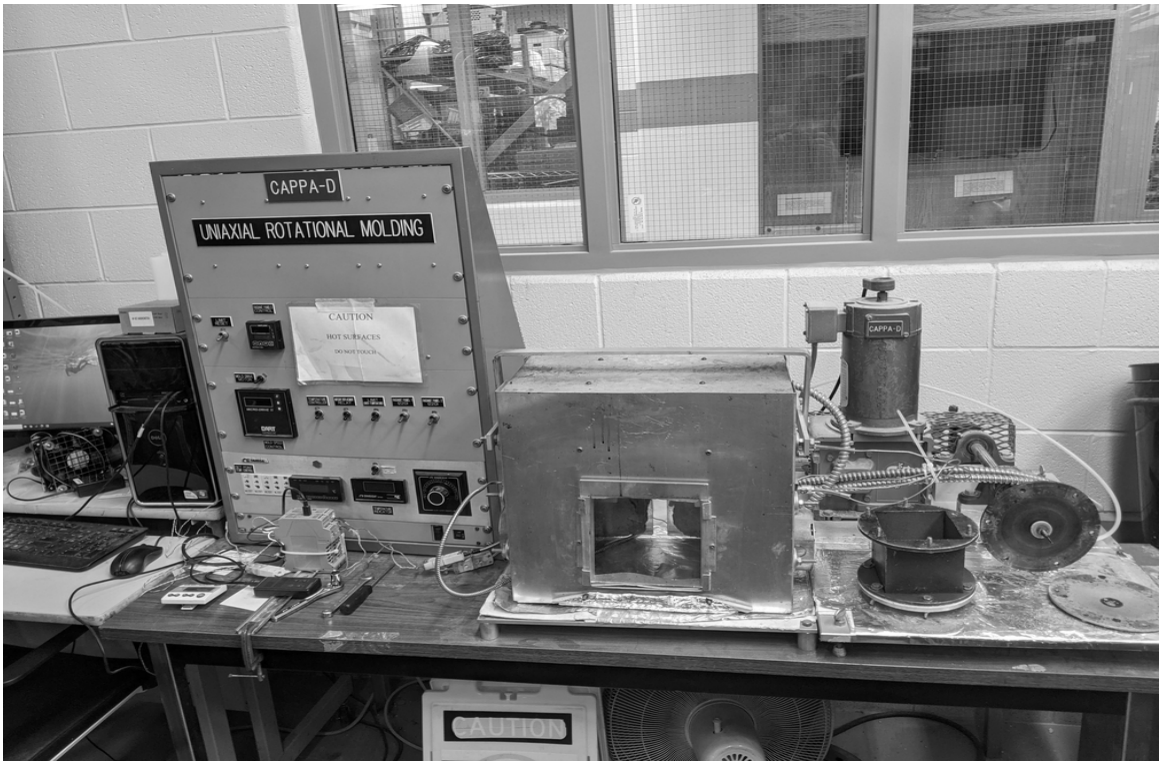

Figure 2.1: Lab-scale setup for rotational molding experiments.

2.2.1 Rotational Molding

The material is a high-density polyethylene powder (ExxonMobil^{TM} HD8660.29) donated by Imperial Oil (Sarnia, ON). Its melt temperature and melt flow index (MFI) of is 129°C and 2 g/10 min, respectively.

Samples are prepared using a laboratory scale uniaxial rotational molding machine (McMaster University; Hamilton, ON). The rotational molding machine is equipped with a custom LabVIEW (National Instruments Corporation; Austin, TX) program that sets the inputs of the compressed air supply, and power to both left and right panels of the oven heater. The controlled variable is the internal mold temperature, which is measured via a K-Type thermocouple mounted directly in the center of the mold. Temperature of the oven is measured by two K-Type thermocouples mounted against the left and right panels. Figure 2.1 shows the rotational molding setup.

The mold is charged with 100 g of polyethylene powder at room temperature. The mold is then rotated at a constant speed of 4 rotations per minute while the oven is removed from the rotomolding unit and pre-heated to 300°C. At time, t=0, the oven is placed over the rotating mold that is filled with powder. A MATLAB (The Mathworks Inc.; Natick, MA) script is running to control the internal air temperature of the oven by adjusting the oven panel temperatures and the supply of room temperature compressed air. When the model reaches an optimal amount of control steps, as determined by its economic predictive controller, the oven is then removed and a cooling unit consisting of fan-forced air at a speed of 2.5 m/s is placed in front of the mold. When the internal mold temperature cools to 80°C, the sample is taken out for quality measurements. The final product cube wall dimensions are 90 × 90 mm with a wall thickness of 3 mm. The characterization techniques for determining product quality are explained in the following sections.

2.2.2 Surface Void Analysis

Surface voids give the appearance of an orange peel, which consist of little surface holes or dimples along the molded part wall. A sample showing substantial surface voids suggests that the polymer particles did not fully sinter during a rotomolding experiment. The integrity of a sample is determined by the completeness of particle sintering, which ideally shows a smooth appearance at the end of molding. These voids are highlighted by rubbing a low viscosity dark-colored grease containing copper and graphite particles onto the faces of a sample. The excess grease is wiped away using a paper towel. Images of the mold face are taken using a digital camera and evaluated using image analysis software (ImageJ; NIH, USA) for a considered area of 40 $\times$ 40 mm. Once the surface void area is evaluated, it is divided by the area of the section (1600 mm^2).

2.2.3 Impact Testing

Four equally sized walls from each molded sample are cut and then frozen at $-40°C$ for 24 hours, in preparation for dart impact testing according to ASTM D5628. The initial height for the dart is 0.762 m (2.5 ft) following staircase method to determine the height at which the sample fails. The staircase method requires that if a sample fails from an impact at its current height, the height being tested next will be 0.1524 m (0.5 ft) lower. Alternatively, if the sample does not fail, then the height is increased by the same step size. The weight of the dart is 6.804 kg (15 lbs). The calculation of the impact energy is based on the height where 50% of the samples will fail, also known as the mean failure height. The mean failure or impact energy is calculated by multiplying the mean failure height by the mass of the dart.

2.2.4 Model Identification of Batch Data

A good process model which picks up the system dynamics, and predicts the evolution of process and quality variables reasonably is necessary for the controller to consistently achieve the desired product quality. The model is determined as follows: a dynamic model state space is built first between the two heaters and compressed air as inputs, and the internal mold temperature as the output. The dynamic state space model is linked with a static PLS model which connects the final product qualities (of the training batches) and the terminal states as predicted by the dynamic model.

A deterministic subspace identification technique, specifically adjusted for batch processes is applied to obtain a dynamic state space process model. This modelling technique involves finding the system order n and the system/coefficient matrices of the above mentioned state space model using input-output batch data.

The following structure is imposed while identification :

$$\mathbf{x}_{k+1}^{d} = \mathbf{A}\mathbf{x}_{k}^{d} + \mathbf{B}\mathbf{u}_{k}, \tag{2.1a}$$

$$\mathbf{y}_{k} = \mathbf{C}\mathbf{x}_{k}^{d} + \mathbf{D}\mathbf{u}_{k}, \tag{2.1b}$$

where $\mathbf{A} \in \mathbb{R}^{n\times n}$, $\mathbf{B} \in \mathbb{R}^{n\times m}$, $\mathbf{C} \in \mathbb{R}^{l\times n}$, $\mathbf{D} \in \mathbb{R}^{l\times m}$ are the associated system matrices and $\mathbf{x}_{k}^{d}$ denotes the state vector.

In contrast to conventional system identification (see, e.g., (Raghavan et al., 2006)), Subspace identification methods enable computation of the model parameters by invoking matrix algebra based decomposition techniques and/or projections, and do not require iterative or complex computations (see, e.g., (Tangirala, 2014; Qin, 2006)). There are variations of the subspace algorithm among which are multivariable output error state space algorithm (MOESP) (Verhagen and Dewilde, 1992), canonical variate analysis (CVA) (Larimore, 1990) and numerical algorithms for subspace state

space system identification (N4SID) (Overschee and Moor, 1992) which differ only in the weighting matrix used at the singular value decomposition step (Van Overschee and De Moor, 1995). All these algorithms were however developed mainly in the context of continuous operation of the processes (see, e.g., (Wang and Qin, 2002; Qin et al., 2005; Pour et al., 2010)). In modeling continuous processes, perturbations are typically imposed around a nominal steady state operating point to generate training data. Continuous process data designed for identification often deviate in small amounts around the steady-state operation of the process and can be readily incorporated in the construction of Hankel matrices within the subspace identification algorithm. However, batch process data is a collection of multiple independent trials which explore the transients of the process. Thus, the subspace algorithm needs to be adjusted for including batch data appropriately. For instance, consider the output measurements of a batch process denoted as $\mathbf{y}^{(b)}[k]$, where k is the sampling instant since the batch trial was initiated, and b denotes the batch index. The output Hankel matrix for a batch b is given by:

$$\mathbf{Y}_{1|i}^{(b)} = \begin{bmatrix} \mathbf{y}^{(b)}[1] & \mathbf{y}^{(b)}[2] & \cdots & \mathbf{y}^{(b)}\left[j^{(b)}\right] \\ \vdots & \vdots & & \vdots \\ \mathbf{y}^{(b)}[i] & \mathbf{y}^{(b)}[i+1] & \cdots & \mathbf{y}^{(b)}\left[i+j^{(b)}-1\right] \end{bmatrix} \tag{2.2}$$

The challenge here is to appropriately utilize data from multiple batches while respecting its batch nature. Concatenating all historical batch data into one 'continuous' data-set for identification would result in misinterpretation since end point of one batch, and the start point of another would be taken as contiguous, when in fact they are not.

Therefore, the following approach (Mhaskar et al., 2019) is applied to form a single pseudo-Hankel matrix from input and output data of all the batches such that the

batch nature of the data is preserved:

$$\mathbf{U}_{1|i} = \begin{bmatrix} \mathbf{U}_{1|i}^{(1)} & \mathbf{U}_{1|i}^{(2)} & \cdots & \mathbf{U}_{1|i}^{(nb)} \end{bmatrix} \tag{2.3}$$

where nb is the number of batches used for training. Similarly, pseudo-Hankel matrices for the output data are formed. Such an approach allows inclusion of batches of different lengths without requiring the determination of an alignment variable or of using a nominal trajectory to mean-center the data, as is required in time-dependent modeling approaches such as PLS.

Once the pseudo Hankel matrices are formed, the deterministic algorithm presented in (Moonen et al., 1989), and appropriately adapted in (Mhaskar et al., 2019) is used for identification of the process. In principle, however, the batch subspace identification algorithm discussed above can be appropriately used with any variation of the subspace identification algorithms. The system matrices are estimated using ordinary least squares once the state trajectories are obtained (see (Mhaskar et al., 2019) for detailed discussion of the algorithm).

2.3 Data Modelling and Predictive Control Design for Rotational Molding Process

An overview of data modelling technique based on subspace identification is presented in this section along with the design for a predictive controller using the obtained model for rotational molding process.

2.3.1 Model Identification

The rotomolding operation includes two heater powers and compressed air supply (a total of three inputs) and enables measuring the internal mold temperature (one output). A database of rotomolding experimental batches is used for model identification. Initial batches in this database were generated using a PI control designed for set-point tracking of the two heaters' temperatures to 300°C. The rest of the batches were generated using a preliminary model (and the control design of Section 2.3.2). In the initial set of closed-loop experiments, there was a mislabeling of input ports between the matlab code and the lab view setup that resulted in a swapping of input connection ports of the heaters with the input connection ports of the controller (thus input 1 port of the heater was misinterpreted as input 2 port of the controller). This resulted in several batches with seemingly strange results, which nonetheless provided sufficient excitation and thus were retained as part of the training set. This error was caught and rectified while working on the consistency of the solution for the current problem statement. Validation of the model was done by splitting the database in a 70:30 fashion for training and testing/validating respectively.

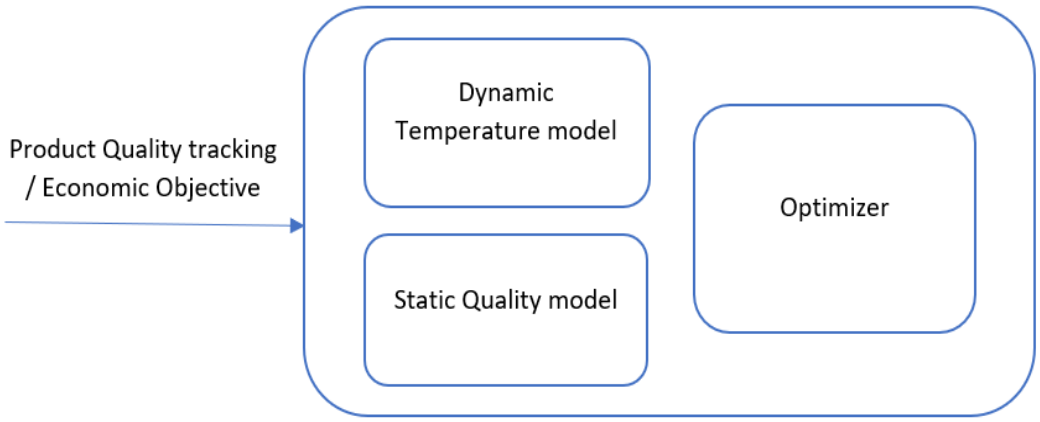

Figure 2.2: Modelling and Control strategy

A state-space model of order 2 was found to sufficiently capture the evolution of

internal mold temperature through the validation process and identified using the proposed technique as discussed in Section 2.2.4. The validation tests show that the training batches contain sufficient information about the process that allows for a good model to be identified. A closer look at the training data reveals sufficient variation in the trajectory during the heating phase, across all the batches, effectively capturing a large region of process operation (for more systematic methods of data identification, see (Genceli and Nikolaou, 1996)). In this work, only the heating cycle has been modelled. For the cooling phase, the mold is kept in front of a fan which is set to operate at full power for the entire time.

In order for the model to function efficiently and produce accurate terminal state predictions, the states of the model have to be suitably initialised for a new batch since initial state estimates directly affect model predictions. In the proposed approach, a state estimation technique is deployed during the initial operation of a new batch. During this initial period, a PI controller is first deployed to control the two heaters and the compressed air supply till the internal mold temperature reaches 130°C, by tracking a defined set-point for the two oven temperatures (fixed at 300°C). Subsequently, a Luenberger observer (see Equation (2.4) is initially run throughout the PI control duration to obtain state estimates. The states, through the observer, essentially obtains information on heating-induced powder adhesion and then melting in the batch. Accuracy of state estimates is evaluated according to the accuracy in predicted output (the internal mold temperature). After 130°C , the controller is switched to MPC and uses the state estimates to suitably control the rest of the sintering (heating) phase.

A Luenberger observer of the following form is deployed during the operation of a batch:

$$\hat{\mathbf{x}}[k+1] = \mathbf{A}\hat{\mathbf{x}}[k] + \mathbf{B}\mathbf{u}[k] + \mathbf{L}(\mathbf{y}[k] - \hat{\mathbf{y}}[k]) \tag{2.4}$$

where $\mathbf{L}$ is the observer gain. The observer gain is designed in a way to ensure that $(\mathbf{A} - \mathbf{LC})$ is stable i.e., poles of $(\mathbf{A} - \mathbf{LC})$ are placed within the unit circle. In MATLAB, this is achieved using the *place* command.

Remark 1. *For this work, a Luenberger observer was used but a Kalman filter may also be used. The focus of this work was more on the optimal control strategy for Rotomolding process and less on the decision of a state estimator. Kalman filter could indeed be more efficient in estimating the states and will be implemented in future applications.*

Since the product quality measurements are not available online during the progress of an experiment, the identified process model alone is not adequate to predict the quality of the sample. The present approach relies on the fact that the essence of the process is effectively captured by the 'states' of the state-space model. Hence the dynamic model is linked with a partial least squares based linear quality model achieved by relating the product quality measurements to the terminal state predictions of the training batches. This augmentation results in an indirect model between the product qualities and the process inputs itself and thus allows for quality control when implementing the predictive controller. The quality model takes the form shown below:

$$Q_{t_f} = \hat{R}_m + \hat{P}_m x[t_{f_{heat}}] + e \tag{2.5}$$

where Q_{t_f} denotes quality measurements at the termination of batch at time t_f, $\hat{P}_m$ and $\hat{R}_m$ are the linear quality model parameters, $x[t_{f_{heat}}]$ are final states as predicted by the state-space model (i.e., at completion of the heating cycle at time $t_{f_{heat}}$) and e represents white noise. Figure 2.5 shows the terminal qualities as predicted from the 55th, 65th and 75th sampling instants for a validation batch. It can be observed from Figure 2.5 that as the batch progresses, and the prediction horizon (the length

of time over which the model needs to predict) decreases, the process model mismatch decreases resulting in improved quality predictions. Recall that the PLS model relates the final states to the final quality, and as such, the states of the particular batch remain unmeasured. Thus the model is trained by using the estimated states (using a Luenberger observer) and the measured quality for the training batches. To validate the model, the state observer is run for the validation batch, and then the estimated states are used to predict the quality using the final states and compared with the measured quality. Intrinsically, the performance of the PLS model depends on the ability of the state space model to accurately predict the terminal states of the batch for a given input sequence. The performance of the quality model in training and validation is shown in the figures below:

Remark 2. *The initial quality model framework aims at solely working with linear models and it is indeed clear that moving away from a linear model may result in improvements. Future work will focus on evaluating non-linear quality models and compare the predictive performance.*

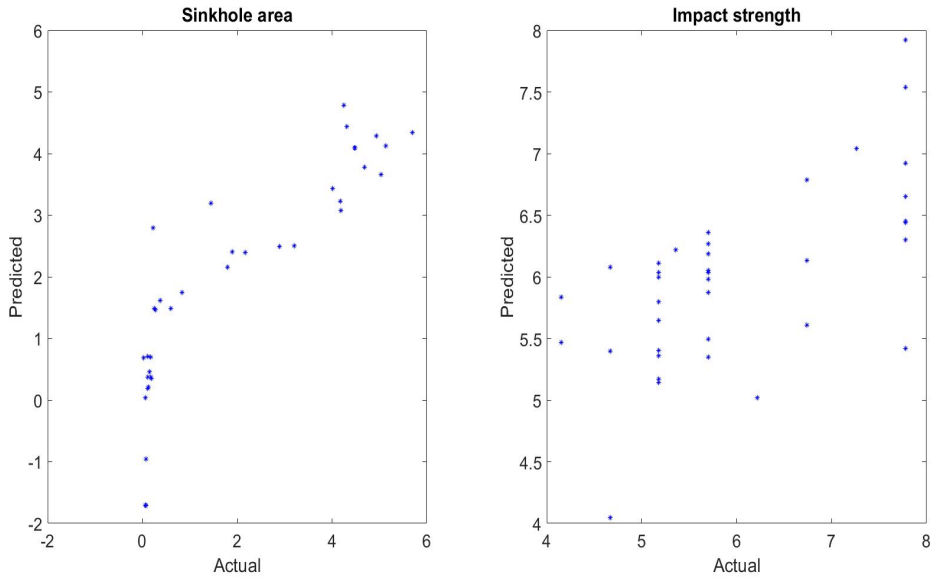

Figure 2.3: Quality model - Training performance

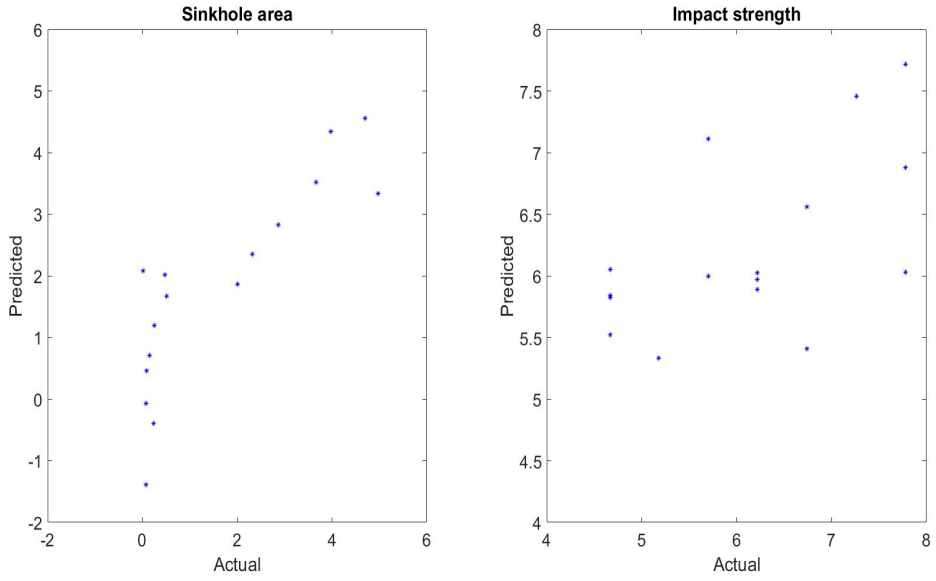

Figure 2.4: Quality model - Testing performance

In summary, the essential rotomolding process dynamics are captured adequately with the state space model augmented with the linear quality model. Note that in this work the accuracy of internal mold temperature predictions as well as the product quality predictions form the criterion for order selection of the state space model.

Remark 3. *Other techniques that could be utilized for dynamic model building are neural networks (NN) and Non-linear Auto-Regressive eXogenous models (NARX). Subspace models are presently being utilized because they are simpler to implement and give direct access to the state trajectory which is connected to the quality variables through the quality model. If one were to use the neural network approach, one would have to draw out the state trajectory to build the quality model, and add additional considerations to alleviate over fitting issues. This remains the focus of future work.*

Remark 4. *Future work can include building rigorous but possibly incomplete first principles models for the rotomolding process and using state space models to predict the dynamics of the residue leftover from the first principles model (Ghosh et al., 2019). In yet another approach for incorporating first principles knowledge (Patel*

et al., 2020) subspace methods have been adapted to include known relationships between certain inputs and outputs. In the context of the rotomolding example, one possibility is to utilize the structure of the process to impose the same structure on the model. For example, in the rotomolding process, the heater inputs first affect the heater temperatures which in turn along with the compressed air input impact the dynamics of the internal mold temperature. Using this information, one can impose a certain hierarchy while building two different state space models in a cascaded fashion.

2.3.2 MPC formulation and implementation

The next step in achieving economic control while achieving product specifications is to incorporate the obtained data-driven model (state space model augmented with a linear quality model as described in the above section) within an EMPC scheme to obtain a product with user-specified qualities. Note that for a model to work well with a predictive controller, it needs to accurately capture and predict the process evolution given an input sequence and this is appropriately illustrated in the previous section.

Recall that the control objective and the way in which the control horizon evolves for control of batch processes are entirely different from that of continuous processes. For the first case, to get the best product qualities, the objective function (2.6a) attempts to simply minimizes the sinkhole area and maximizes the impact energy of the product. For the second case, since the inputs represent heater powers, the objective function is the summation of the input sequence over the control horizon (2.8) (other, more detailed representations of the cost can be readily included). The rest of the constraints listed remain the same for both manifestations of EMPC. The two EMPC formulations are separately run on the Rotomolding setup, i.e., 3 experiments of case-1 and another 3 experiments with case-2 formulation. Furthermore, the duration of

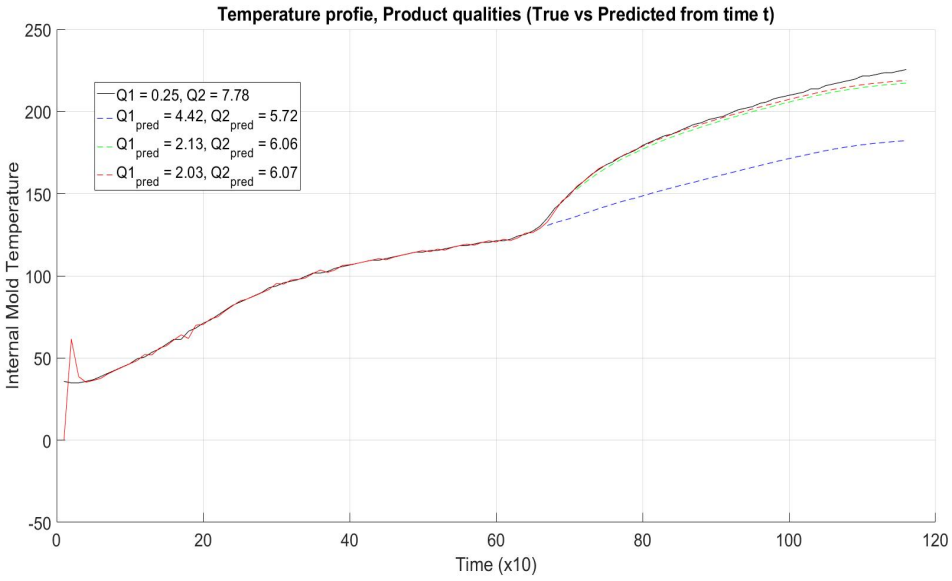

Figure 2.5: Performance of state space and quality models: the black line represents the true process; blue, green and red dashed lines represents the trajectory predicted by the model from times, 65, 70 and 75 respectively; The legends of the graph give information about the true values of the product quality (black) and the product qualities as seen predicted from times 65 (blue), 70 (green) and 75 (red)

the heating cycle in the rotomolding process too is a decision variable and has to be accounted for appropriately while designing the optimisation scheme.

Remark 5. *The adapted subspace identification for multiple batches (Mhaskar et al., 2019) readily allows for inclusion of batches with different duration unlike other dynamic modelling strategies that utilize Partial Least Squares technique (PLS). This allows for the algorithm to explore a sufficient set of operating regions for the process (past experimental batches) and fit a suitable model, and more importantly, utilize it for the purpose of feedback control (since there is no need to find and maintain an alignment variable). The model thus has information necessary for the MPC to be able to optimize the trajectory for different cycle times. An average cycle time from the previous successful experiments may be used but since the trajectory has a direct impact on the terminal product qualities, a framework accommodating for a few candidate batch times is formed in the present application.*

To implement the proposed control scheme on the lab scale rotomolding setup, MATLAB is connected with LabView which in-turn is connected with sensors and actuators, acting as a data acquisition system. The control input is calculated and implemented every $10s$ through the LabView-MATLAB interface. At a sampling instance l, the entire optimal input trajectory is given by the solution to the below mentioned optimization problem, using Equation (2.6e) to compute the state estimate at the l^{th} instant:

EMPC for case 1:

$$\min_{\mathbf{U}_f, l_f} \ \hat{Q}_1 - \hat{Q}_2 \tag{2.6a}$$

$$s.t. \ U_{j,min} \leq u_f[k] \leq U_{j,max}, \ \ \forall \ \ 0 \leq k \leq l_f - l \tag{2.6b}$$

$$|u_f[0] - u[l-1]| \leq \delta, \tag{2.6c}$$

$$|u_f[k] - u[k-1]| \leq \delta, \ \forall \ \ 1 \leq k \leq l_f - l \tag{2.6d}$$

$$\hat{\mathbf{x}}[0] = \hat{\mathbf{x}}[l] \tag{2.6e}$$

$$l_f \in \{t_{switch} + 400, t_{switch} + 500\} \tag{2.6f}$$

$$\mathbf{\Lambda}\hat{Q}_{t_f} \leq \mathbf{\Gamma} \tag{2.6g}$$

$$\hat{\mathbf{x}}[k+1] = \mathbf{A}\hat{\mathbf{x}}[k] + \mathbf{B}\mathbf{u}_f[k] \tag{2.6h}$$

$$\hat{\mathbf{y}}[k] = \mathbf{C}\hat{\mathbf{x}}[k] + \mathbf{D}\mathbf{u}_f[k] \ \ \forall \ \ 0 \leq k \leq l_f - l \tag{2.6i}$$

$$\hat{Q}_{t_f} = R_m + P_m \hat{x}[l_f] \tag{2.6j}$$

with,

$$\hat{Q} = \begin{bmatrix} \hat{Q}_1 & \hat{Q}_2 \end{bmatrix}^T \tag{2.7a}$$

$$\mathbf{\Lambda} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \tag{2.7b}$$

$$\mathbf{\Gamma} = \begin{bmatrix} 2 \\ -5 \end{bmatrix} \tag{2.7c}$$

Objective function for EMPC in case 2:

$$\min_{\mathbf{U}_f, l_f} \sum_{k=0}^{k=l_f-l} u_f[k] \tag{2.8}$$

with the constraints (2.6a) - (2.7c), same as in case 1.

where, $\mathbf{U}_f = [\mathbf{u}_f[0], \mathbf{u}_f[1], ...\mathbf{u}_f[l_f - l]]$ are the set of decision variables which are the two heaters and the compressed air input values at each sampling instance over the control horizon and l_f denotes the termination time of the heating cycle. Based on past experience, the set of admissible termination times, l_f is specified through Equation (2.6f). $\delta = \begin{bmatrix} 60 & 60 & 60 \end{bmatrix}^T$ is the allowed rate of input change specified through Equations (2.6c) and (2.6d), and $U_{j,min}$ and $U_{j,max}$ are the lower and upper bounds on the input variable (Equation (2.6b)) with $U_{j,min}$ and $U_{j,max}$ being $\begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T$ and $\begin{bmatrix} 100 & 100 & 100 \end{bmatrix}^T$ respectively. In addition, t_{switch} denotes the time (in *sec*) at which the controller switches to MPC from PI controller.

Further, Equation (2.6g) represents the user-specified constraints on the quality variables. Finally, Equation (2.6j) represents the quality model which the MPC uses to predict the terminal product quality using the terminal states (of the heating cycle) which in-turn are produced by the dynamic state-space model as specified in the

Equations (2.6h)-(2.6i). In Equation (2.7a), $\hat{Q}_1$ and $\hat{Q}_2$, refers to the estimated values of the two quality variables namely, sinkhole area coverage (%) and impact energy (Kg.m) respectively.

Remark 6. *The quality of a molded product (and the specifications) can, in some cases be characterized through additional quality variables such as viscosity. The proposed control design can readily handle specifications on additional quality variables. This can be achieved by including training data with the additional qualities measured, and by appropriately modifying the above equations in the optimization formulation.*

Remark 7. *Note that if there were no quality constraints in the current formulation, the optimizer would simply choose a 'zero' input sequence (due to the objective function, i.e., minimizing the summation of inputs) since it has no motivation to even meet the minimum required product qualities.*

The above formulated optimization problem is solved using a brute force strategy as two linear programs (with *linprog* in MATLAB) instead of solving as a mixed integer linear program (MILP). This is a viable approach due to the limited choices provided for the batch duration and hence optimizing over this integer variable is done by comparing the solutions corresponding to each of the linear programs and the best one is chosen and implemented. That is, for each of the two choice for batch length, the optimization problem is solved, and the objective function evaluated subject to constraints. Subsequently, out of the two, the solution which has better economics is chosen. Since a constrained linear program is only solved for two different values of batch duration each time, the overall computational complexity remains low. More candidates for batch duration may be included but it might be worth noting that this may result in increased computational complexity and given the inherent plant-model mismatch, the benefit from such a strategy might not be significant. In summary, at each time step, the optimization problem determines the best duration, as well as the best associated input trajectory, and resolves for these variables at the next sampling

instance.

Remark 8. *The system under investigation (Rotomolding) does not permit us to measure the quality variables to be tracked dynamically, i.e., they are only measured offline, once the experiment is done. This is the reason why only the terminal qualities are included in the objective. This in part also distinguishes it from existing type of EMPC formulations while retaining the essence of EMPC since the control action is being chosen to directly maximize the process economics. Eq.*(2.6f) *gives two candidates for the terminal batch time. As explained earlier, two MPCs corresponding to these two terminal batch times are solved at each time step. The fastest solution is indeed considered but the preference is given to the objective function value rather than only the time. Thus if the objective function is lower for a smaller terminal time, then the MPC would implement the corresponding control action and achieve a quicker termination. On the other hand, if better economics can be achieved by running the process a little longer, then that solution would be chosen.*

Furthermore, the optimization problem is setup with a series of relaxations to prevent the code from breaking down during the experimental run due to possible occurrences of infeasibile solutions. That is, first the problem with the original formulation is solved and if the solution turns out to be infeasible then the problem is relaxed by pushing around the values of the quality constraints in Equation 2.6g. This allows for a guaranteed overall feasible and an implementable solution due to the nature of the problem. In the presence of quality and input constraints, the optimization problem could be infeasible due to either the quality simply being unreachable from the current operation in the allotted time period subject to input constraints, or the model is simply not correctly predicting the process dynamics that far ahead in time. For the present application, feasibility tends to be recovered as the process moves to a new operating point, not only because more process output information becomes available and is utilized through the Luenberger observer to update the state

estimates, but also due to the changing nature of the process-model mismatch.

In the current work, the optimization is relaxed in layers. The original constraints for the quality variables are Equation 2.6g. For example, if the original problem is found to be infeasible, then a modified problem where the RHS of the first quality constraint is raised by 1. Further if even this layer fails, then the RHS of the second quality constraint is raised by 1. The formulation keeps advancing to subsequent layers where the RHS of the quality constraints is raised by 1, until the problem is feasible.

Remark 9. *The present application focuses on achieving on-spec product as specified by the user through a set of constraints on the quality variables with a relatively low input energy consumption. Factors like smoothness of input profiles were not focused on while picking the tuning parameters for the EMPC (thus a set value of 60 was utilized in the implementation). If one desires a further smoother profile, the rate constraint can be adjusted appropriately.*

Remark 10. *In the present work, the model is built between the output (internal mold temperature) and three separate inputs, namely power to two heaters and the open/-close command to a solenoid for adding cold compressed air into the oven (though the three inputs together result in a cumulative heating/cooling effect). The positive results of this approach reinforces the current modelling strategy. Alternatively, a principle component analysis can be applied to reduce the three inputs to one input to represent a unified heating/cooling effect. Yet as another option, one could also reduce the two heaters to one heating input via principle component analysis and use this principle heater and the compressed air as the two inputs to the system representing heating and cooling respectively. A state space model can then be built between the input(s) and the output using a subspace algorithm and in turn be integrated within the MPC controller. Furthermore, the economic formulation can be adjusted to incorporate more detailed terms capturing the cost of production, as well as profit related to the quality of the product with appropriate weights for each of the components.*

Remark 11. *Current work assumes a single model for the process and an MPC is appropriately designed and deployed in the experimental setup. The rotomolding process is a multi-stage process with more than one stage during in the heating period (related to sintering and then densification of the polymer). The MPC implementation in the present work is such that it kicks in after the polymer has melted, with the PI being active before that time. Future work could include building separate models for each of the stages and designing a suitable multi-stage MPC for rotomolding.*

Remark 12. *The proposed approach allows the prediction of the terminal product qualities online as the process is evolving. Not only is that required for MPC implementation, but the prediction is also important from a practitioner's standpoint, to enable determining if the input moves prescribed by the controller make sense. This is important in establishing operator confidence in the proposed modeling and control approach by providing a meaningful physical interpretation to the control action.*

Remark 13. *The rotomolding process moves through different phases (and as such, processes are often nonlinear) and a single linear process model may not perform well throughout the process operation. This can be handled by adapting the model to the current operating condition, by checking the model predictability (rather than the controller performance, as is done in most monitoring approaches). Future work will focus on a re-identification framework unique to the rotomolding application.*

2.4 Experimental Implementation Results

The economic model predictive control formulation is evaluated through implementation on the rotomolding experimental setup. The ability to deliver different economic objectives is evaluated by utilizing two different objective functions. In the first case, the ability of the controller to deliver product with the best quality values, without any penalties on the input energy consumption is illustrated. Thus, this represents

a high value product where the profits accrued from higher quality outweigh the production costs. Next we consider the case where there is no additional benefit to improving quality beyond specifications, and thus economic benefit can be achieved through lowering production costs.

Three batch experiments for each of the two above mentioned scenarios were conducted for demonstrating the robustness and efficacy of the EMPC formulation.

The internal mold temperature and input profiles for batches of the first case are shown in Figure 2.6, and the energy consumption and quality measurements are reported in Table 2.1 (which also lists the average energy consumption and quality measurements of the identification batches. These values show that the model built on this training set has adequate 'experience about the process' for performing well under the optimization strategy.)

Compared to the identification batches, the MPC for Case-1, after it is turned on at about the halfway mark, keeps the heaters on for almost the rest of the run at the maximum value, thus increasing the internal mold temperature to values close to 250°C. By doing so, it is able to achieve exceptionally low sinkhole area compared to the training batches. It also achieves the highest measurable impact strength (for the measurement device being used) with the acceptable trade off being increased energy consumption. Compared to the average product qualities in the batches of the training set, Case-1 achieves more desirable sinkhole area percentage and impact strength values, with almost doubled energy consumption values (see Table 2.1). This is reflective of the process economics with the improved quality being far more important than the energy consumption.

The internal mold temperature and input profiles for batches of the second case are shown in Figure 2.7, and the energy consumption and quality measurements are reported in Table 2.1. Recognizing that in this case, only the quality specifications need

to be met and not exceeded, the controller does not drive the temperatures too high, turning down the heat input to a large extent, while making sure that the quality constraints are met. The batches for this case therefore meet quality constraints, except with much lower energy consumption. As it can be inferred from the table, an average energy consumption reduction of 72% is achieved with Case-2 MPC when compared to Case-1. It is also worth noting that Case-2 economic formulation succeeds in achieving product qualities adhering to the given constraints along with a lower energy consumption profile (see Table 2.1) again reflecting the process economics for this case.

Note that the batches under investigation from both formulations satisfied the same quality constraints and that none of the experiments 'failed' or 'relaxed' the quality constraints in the final iteration. Another important point to note is that across various batches the controller does select different batch lengths since the formulation allows for a decision between candidate cycle times.

Determining the internal mold temperature trajectory, i.e., designing a new process for a new product (defined by significantly different quality specifications, or required to be made from very different ingredients) is a very challenging and expensive exercise. The demonstrated ability of the controller to be able to adapt and achieve different quality specifications renders it a very promising design tool. Thus, future work will consider situations where the controller is challenged to produce a significantly new product (or from say, quite different raw material). If initial implementations fail, the data from these new runs could be used to further adapt the model and thus the controller, and in turn rapidly achieve the new desired product.

Remark 14. *Significant work exists (Gomes et al., 2018, 2019), where an alternate technique for quality assessment has been investigated using ultrasonic spectral data as an effort to avoid destructive methods for measuring product quality. Furthermore, image analysis classifier can be built for tracking the product qualities during the course*

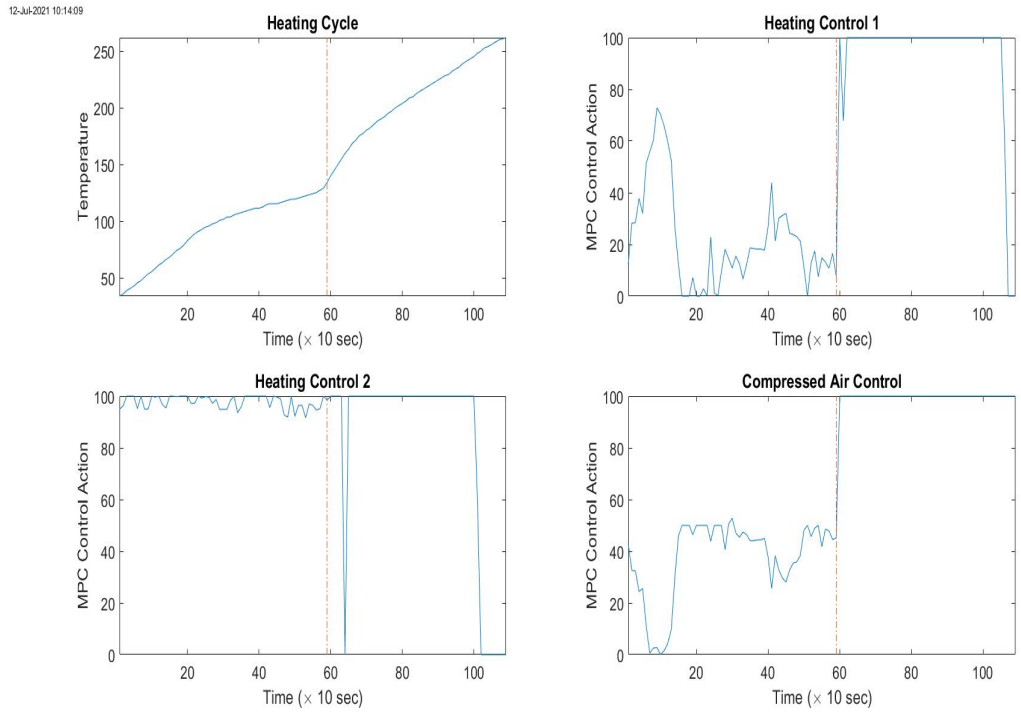

Figure 2.6: Input and Output profiles of a sample batch run with Case-1 EMPC

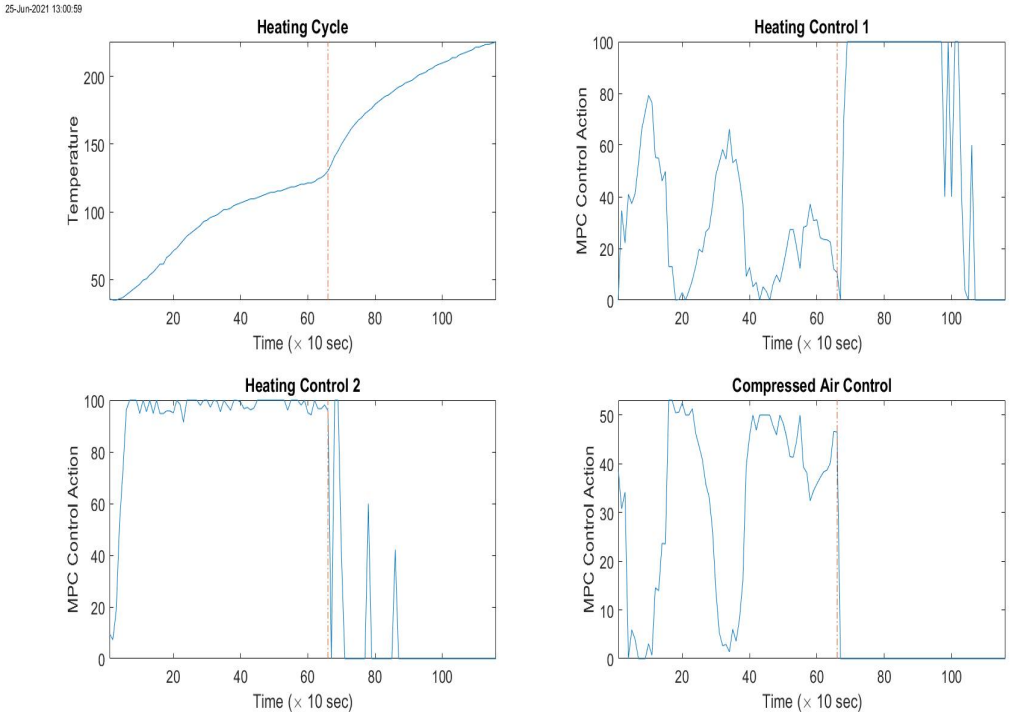

Figure 2.7: Input and Output profiles of a sample batch run with Case-2 EMPC

Table 2.1: Summary of the two economic formulations

Batches $Q1 \leq 2\%$ $Q2 \geq 5Kg.m$	Total Input consumption (PI+MPC)	Input consumption (MPC)	Q_1	Q_2
Average values from training set	$14.91e+03$	$6.94e+03$	2.07	5.84
Economic Case-1, Batch 1	$22.9e+03$	$13.7e+03$	0.08	7.78
Economic Case-1, Batch 2	$22.9e+03$	$13.5e+03$	0.06	7.78
Economic Case-1, Batch 3	$23.4e+03$	$13.7e+03$	0.08	7.78
Economic Case-2, Batch 1	$14.7e+03$	$4.01e+03$	0.37	7.78
Economic Case-2, Batch 2	$13.3e+03$	$3.55e+03$	0.27	6.74
Economic Case-2, Batch 3	$13.9e+03$	$3.79e+03$	0.25	7.78

of the experiment. Techniques like these can be integrated with the modelling approach for optimal control.

Remark 15. *A linear dynamic state space model is coupled with a static linear quality model in the current modelling strategy. When integrated within an MPC framework, this results in a convex problem which can be solved easily with any optimization algorithm. Consequently, a non-linear model may also be developed to replace the existing dynamic as well as the static models and a relevant framework for the MPC can be designed for better quality control.*

2.5 Conclusions

A data driven modelling and economic control strategy is proposed for obtaining product with user specified quality specifications in this article. The efficacy of the framework is demonstrated through a uni-axial rotomolding setup. To this end, a state space model between the inputs (2 heaters + compressed air input) and the output (internal mold temperature) was obtained using subspace identification on past experimental batches and subsequently two economic objectives were formulated and implemented. The quality specifications are given via constraints on quality variables in the optimization formulation. The evolution of the temperature and quality variables predictions are governed by the previously identified state space

model and a linear static model (using PLS) relating the terminal states to the quality measurements (sinkhole area and impact energy). The augmented model (state space dynamic model and PLS quality model) is setup inside the EMPC as a set of equality constraints on the process variables. The experimental results show the ability of the proposed control framework to achieve a) the best product qualities possible, and b) to achieve on-spec product while respecting constraints and minimizing the operating cost.

Rotomolding is a challenging process due to its non-linearities and multi-stage nature and therefore a single model might not be sufficient to capture the dynamics. In the next work, a re-identification algorithm unique to batch processes and in particular, to rotomolding will be focused on. Some aspects for future work can be in the direction of building non-linear dynamic model for the process, incorporating first principles knowledge along with data driven modelling, and developing multi-stage modelling and control strategy.

Bibliography

Abu-Al-Nadi, D. I., Abu-Fara, D. I., Rawabdeh, I., and Crawford, R. J. (2005). Control of rotational molding using adaptive fuzzy systems. *Advances in Polymer Technology*, 24(4):266–277.

Bonvin, D. (1998). Optimal operation of batch reactors—a personal view. *Journal of Process Control*, 8(5-6):355–368.

Corbett, B. and Mhaskar, P. (2016). Subspace identification for data-driven modeling and quality control of batch processes. *AIChE Journal*, 62(5):1581–1601.

Flores-Cerrillo, J. and MacGregor, J. F. (2002). Control of particle size distributions in

emulsion semibatch polymerization using mid-course correction policies. *Industrial and Engineering Chemistry Research*, 41(7):1805–1814.

Flores Cerrillo, J. and MacGregor, J. F. (2005). Latent variable MPC for trajectory tracking in batch processes. *Journal of Process Control*, 15(6):651–663.

Garg, A., Gomes, F. P., Mhaskar, P., and Thompson, M. R. (2019). Model predictive control of uni-axial rotational molding process. *Computers & Chemical Engineering*, 121:306–316.

Genceli, H. and Nikolaou, M. (1996). A new approach to model predictive control and identification. *IFAC Proceedings Volumes*, 29(1):4557–4562. 13th World Congress of IFAC, 1996, San Francisco USA, 30 June - 5 July.

Ghosh, D., Hermonat, E., Mhaskar, P., Snowling, S., and Goel, R. (2019). Hybrid modeling approach integrating first-principles models with subspace identification. *Industrial & Engineering Chemistry Research*, 58(30):13533–13543.

Gomes, F., Garg, A., Mhaskar, P., and Thompson, M. (2018). Quality monitoring of rotational molded parts using a non-destructive technique. In *Proceedings of the ANTEC*.

Gomes, F. P., Garg, A., Mhaskar, P., and Thompson, M. R. (2019). Data-Driven Advances in Manufacturing for Batch Polymer Processing Using Multivariate Nondestructive Monitoring. *Industrial & Engineering Chemistry Research*, page acs.iecr.8b05675.

Kadali, R., Huang, B., and Rossiter, A. (2003). A data driven subspace approach to predictive controller design. *Control Engineering Practice*, 11(3):261–278.

Larimore, W. (1990). Canonical variate analysis in identification, filtering, and adaptive control. In *29th IEEE Conference on Decision and Control*, pages 596–604 vol.2. IEEE.

Ljung, L. (1998). *System Identification: Theory for the User*. Pearson Education.

Mhaskar, P., Garg, A., and Corbett, B. (2019). *Modeling and Control of Batch Processes*. Advances in Industrial Control. Springer International Publishing, Cham.

Moonen, M., De Moor, B., Vandenberghe, L., and Vandewalle, J. (1989). On- And Off-Line Identification Of Linear State Space Models. *International Journal of Control*, 49(1):219–232.

Overschee, P. V. and Moor, B. D. (1992). Two subspace algorithms for the identification of combined deterministic-stochastic systems. *[1992] Proceedings of the 31st IEEE Conference on Decision and Control*, 311(I):75–93.

Patel, N., Nease, J., Aumi, S., Ewaschuk, C., Luo, J., and Mhaskar, P. (2020). Integrating data-driven modeling with first-principles knowledge. *Industrial & Engineering Chemistry Research*, 59(11):5103–5113.

Pour, N. D., Huang, B., and Shah, S. L. (2010). Subspace Approach to Identification of Step-Response Model from Closed-Loop Data. *Industrial & Engineering Chemistry Research*, 49(18):8558–8567.

Qin, S. J. (2006). An overview of subspace identification. *Computers and Chemical Engineering*, 30(10-12):1502–1513.

Qin, S. J., Lin, W., and Ljung, L. (2005). A novel subspace identification approach with enforced causal models. *Automatica*, 41(12):2043–2053.

Raghavan, H., Tangirala, A. K., Bhushan Gopaluni, R., and Shah, S. L. (2006). Identification of chemical processes with irregular output sampling. *Control Engineering Practice*, 14(5):467–480.

Tangirala, A. K. (2014). *Principles of System Identification: Theory and Practice*. Taylor & Francis.

Van Overschee, P. and De Moor, B. (1995). A unifying theorem for three subspace system identification algorithms. *Automatica*, 31(12):1853–1864.

Verhagen, M. and Dewilde, P. (1992). Subspace model identification Part 1. The output-error state-space model identification class of algorithms. *International Journal of Control*, 56(5):1187–1210.

Wang, J. and Qin, S. J. (2002). A new subspace identification approach based on principal component analysis. *Journal of Process Control*, 12:841–855.

Chapter 3

Adaptive Modelling for Quality Control in Batch Processes

The contents of this chapter have been published in Computers & Chemical Engineering Journal.

- Chandrasekar, Aswin, Hassan A. Abdulhussain, Vladimir Gritsichine, Michael R. Thompson, and Prashant Mhaskar. "Adaptive Predictive Control Algorithm for Batch Processes: Application to a Rotational Molding Process." Industrial & Engineering Chemistry Research 61, no. 48 (2022): 17572-17581.

The following is the authorship contribution statement: Aswin Chandrasekar: Conceptualization, Methodology, Formal analysis, Software, Writing – original draft. Hassan A. Abdulhussain: Data curation. Vladimir Gritsichine: Data curation. Michael R. Thompson: Writing – review & editing, Supervision, Project administration, Funding acquisition. Prashant Mhaskar: Conceptualization, Methodology, Writing – review & editing, Supervision, Project administration, Funding acquisition.

Abstract

The present manuscript addresses the problem of handling process nonlinearity in batch process operations and control via a re-identification based subspace identification approach deployed within a Model Predictive Control (MPC) framework. In contrast to existing re-identification algorithms for continuous and batch processes, where all the recent and past experimental data is chosen to re-identify the model, the proposed approach is designed to use the most appropriate subset of the data. In particular, the data for re-identification is determined by first determining the equivalent of a 'locator' index from the training data set, and only using the portion of training batches from the 'locator' to batch termination. The idea is to try and build the model using data pertaining to the state space region that the system is presently passing through. The proposed approach is implemented on a rotational molding lab scale setup coordinated via an existing model monitoring technique deployed within the MPC which detects the model mismatch and triggers the re-identification algorithm. Validation data sets are first used to demonstrate the improved model resulting from the proposed re-identification approach, followed by experimental results demonstrating improved closed-loop performance.

3.1 Introduction

Batch processes are commonly found in many domains including mechanical, biochemical, agricultural and pharmaceutical industries due to the requirement of producing high-value products. Since the production amount is small in batch process, it is important to maintain consistency in obtaining products with excellent quality, and this can only be achieved by deploying a suitable control strategy. Rotational molding, or simply rotomolding, is one such batch process, used in industries for fabricating hollow plastics. In this batch process, a measured amount of polymer resin is charged

into a mold that is rotated slowly in an oven where heaters and compressed air control the heating phase of the process. The heat allows the powdered charge to melt and stick to the walls of the mold, and the constant rotation achieves uniform qualities, like density and thickness, throughout the final product. Once the heating phase is complete, the mold is cooled down using blown air while still rotating to ensure that the polymer evenly solidifies while holding the shape of the mold. From an industry perspective, the aim of this process is to consistently obtain products with desired properties across all batches by regulating various stages of rotomolding and avoiding prolonged heating phase, which can cause product degradation. Oftentimes, rotomolding is applied in production of large scale products, like dinghies, residential water tanks and the fuel tanks of marine vessels and agricultural equipment. Production of these items can be made profitable by using optimal control to significantly reduce the waste costs and, in order to develop such a strategy, one must focus on the modelling aspect associated with the same, especially for this process in which the quality variables that must be tracked are not measurable online and thus not available during processing.

Developing a first principles model based on physics that capture the process evolution can be one of the approaches for obtaining a high fidelity model, which can then be incorporated within an appropriately designed model predictive control (MPC) routine to obtain the desired product qualities. However, coming up with an appropriate set of differential equations describing the heat transfer, particle dynamics and polymer rheology for the rotomolding process is itself a laborious task and may result in a highly complex model with a lot of parameters. What makes this problem almost intractable is the need to develop a model that is able to model and predict not only properties such as the mold temperature, but also the important quality attributes such as surface roughness and impact strength. Some work (Abu-Al-Nadi et al., 2005; Ogila et al., 2017; Hamidi et al., 2016) has been done on building mechanistic models for the rotomolding process but they do not relate the inputs of the system to the

quality variables like sinkhole area and impact strength which are of main interest for our application, and hence are not useful inside a control framework. Hence such control strategies techniques based on mechanistic models may not be preferred. At best, these models can be used to better follow a prescribed set point trajectory for the mold temperature, but are simply not suitable to achieve quality control.

In the absence of a mechanistic model, one approach would be to conduct several experiments to initially determine a good input sequence that would result in a product with particular desired specifications and implement this pre-determined trajectory on a new batch with the hope to obtain a product with similar qualities. In the absence of active feedback, this method is susceptible to disturbances and, if one were to require a different set of qualities, another set of experiments must be run again to determine the optimal process conditions for the same. Another method exists, known as trajectory tracking (Abu-Al-Nadi et al., 2005), which partially rejects disturbances and, in this method, a key output variable of the process can be tracked. However, in a process like rotomolding tracking, a trajectory would not ensure perfect quality tracking due to fluctuations in raw material properties across different batches, thus affecting the connection between the key output variables, such as mold temperature, and the product qualities.

Such a problem can be solved by using a modelling strategy that can accurately describe the connection between the inputs of the system to the product qualities along with the system dynamics. Partial Least Squares (PLS) regression is one such popular and widely used technique that can be applied for modelling a batch process like the rotational molding. In this method, the process information relevant to the dynamics is represented through a reduced dimensional space also called the latent space, through the means of projections (Flores-Cerrillo and MacGregor, 2002; Flores Cerrillo and MacGregor, 2005). The structure of the model obtained using a PLS approach connects the batch trajectory information to the final product quality, which

is quite similar to linear time-varying (LTV) models. One challenge when it comes to identifying a suitable model for batch processes is handling batches of different lengths in a given database. Some work has been done, (Kourti, 2003; Kassidas et al., 1998) where the authors have come up with an alignment variable for the batches so that all the batch sequences are of the same length, but in the process, any information related to the trajectory might get altered. Moreover, PLS approach requires all the batches to be of the same length, which is hardly the case in an industrial setup. Another issue related with PLS based modelling is that the technique does not acknowledge the philosophical difference between the inputs and outputs of the system and moreover the control of rotational molding like-processes using these models remains a loose-end.

Subspace identification is another modelling technique with which one could utilize the available experimental batch data and obtain a linear time invariant (LTI) state space (SS) model describing the dynamics between the inputs and the outputs of the system. In order for the predictive controller (designed to achieve better operating economics) to perform better, a database consisting of past experiments with a wide range of product qualities should be included while identifying the model that is used within the controller. Modelling and control strategy for batch processes based on subspace identification (Corbett and Mhaskar, 2016) has been developed by appropriately adapting the existing data-driven modelling designs for continuous processes (Ljung, 1998; Kadali et al., 2003; Pour et al., 2010; Narasingam and Kwon, 2020). Batches of different duration can be readily incorporated for identifying a model using this proposed modelling solution without requiring any alignment tool.

Even after a satisfactory LTI model is built, there is no guarantee that it will continue to perform well during a new run. This can be due to many process factors like disturbances, non-linearity and/or presence of multiple phases in the system. Some work has been done in the past in adaptive modelling and control strategy (Kher-

admandi and Mhaskar, 2018) in the context of batch processes. In this work, the authors have addressed the adaptive control problem by introducing a new model monitoring metric which gauges the health/predictive capacity of the model to act as a switch to trigger re-identification. This modeling approach was also recently deployed on a lab-scale rotomolding setup to obtain product with desired qualities and the results (Garg et al., 2019) substantiate the ability of this method in procuring product quality consistently. However, the model re-identification approach proposed in (Kheradmandi and Mhaskar, 2018) and utilized in (Garg et al., 2019) does not fully exploit the batch nature of the process when choosing the previous experimental batch data for re-identification purposes, and may end up picking data from past batches that prevent the use of the best information for the current batch.

Motivated by the above considerations, a new algorithm for model re-identification in the context of batch processes is proposed in this manuscript, where a 'locator' is used to decide the portions of batch data best suited for training the model in terms of predictability. In this manuscript, the uni-axial rotomolding process and its lab scale setup is first described in Section 3.2. Subspace identification for batch processes is also reviewed here. We present the implementation, and compare with two scenarios: one without any model monitoring and re-identification routine and another with an existing monitoring and re-identification algorithm (Kheradmandi and Mhaskar, 2018). Then the Economic Model Predictive Control (EMPC) design used for this work is detailed in Section 3.3 and experimental results in Section 3.4. Concluding remarks and a final summary is presented in Section 3.5.

3.2 Preliminaries

A brief overview of the rotomolding process and modeling technique for batch process data based on subspace identification is given in this section. The foundation for the

novel design of EMPC is given in this section and its corresponding closed-loop results are presented in Sections 3.3 and 3.4, respectively.

3.2.1 Rotational Molding

The material used in this study is a high-density polyethylene powder (ExxonMobil^{TM} HDPE HD8660.29) donated by Imperial Oil (Sarnia, ON). The melt temperature and melt flow index (MFI) of the polymer powder are 129°C and 2 g/10 min, respectively.

Samples are prepared using a lab-scale uniaxial rotational molding machine (McMaster University; Hamilton, ON). The rotational molding machine is equipped with a custom LabVIEW (National Instruments Corporation; Austin, TX) program that controls molding conditions by the inputs of the compressed air supply, and power to both left and right heater panels in the oven. The variable of interest is the internal mold temperature, which is measured via a K-Type thermocouple mounted directly in the center of the mold. The wall temperature of the oven is measured by two K-Type thermocouples mounted against the left and right sides.

Samples are prepared by charging 100 g of polyethylene powder into the mold at room temperature. The oven is pre-heated to 300°C, while set aside from the mold. At time, t=0, the mold is rotated at a constant speed of 4 rotations per minute and the oven is placed over the rotating mold that has been charged with powder. A MATLAB (The Mathworks Inc.; Natick, MA) script is run to control the internal air temperature of the mold by adjusting the oven panel power and the duration to which room temperature compressed air is supplied. When the model reaches the optimal amount of control steps, determined by its economic predictive controller, the oven is then removed and the mold is cooled to 80°C by fan-forced air at a speed of 2.5 m/s before the molded polymer part is removed for quality testing. The final cube-shaped part has wall dimensions of 90 × 90 mm with a wall thickness of 3 mm.

The final product is then characterized for its void area and impact strength, which are explained in the below sections.

3.2.2 Surface Void Analysis

Surface voids in rotational molding are a measure of the degree of sintering. A sample showing substantial surface voids suggests that the polymer particles did not fully sinter during the heating cycle. High degrees of sintering are typically correlated with an increase in mechanical properties. The surface voids on the samples are highlighted by rubbing a low viscosity copper based grease onto the faces of a sample to improve their visual contrast from the otherwise white walls. Images of the mold are taken using a digital camera and evaluated using image analysis software (ImageJ; NIH, USA) for a considered area of 40 $\times$ 40 mm. The analysis reports surface void as a percentage by dividing the detected surface void area by the area of the section (1600 mm^2).

3.2.3 Impact Testing

Four walls from each molded sample are cut and cooled at $-40°C$ for 24 hours, in preparation for dart impact testing according to ASTM D5628. The dart is dropped at an initial height of 0.762 m (2.5 ft) and is then adjusted based on the staircase method to determine the mean failure height. If a sample fails due to an impact at its current height, the next height tested will be 0.1524 m (0.5 ft) lower. If the sample does not fail, the height is increased by 0.1524 m (0.5 ft). The weight of the dart is 6.804 kg (15 lbs). The impact energy calculation is based on the height where 50% of the samples fail, also known as the mean failure height. The mean failure height is then multiplied by the mass of the dart to get the impact energy in units of kg-m.

3.2.4 Data Availability

The rotational molding setup has a total of three inputs (two heaters and a compressed air unit) and one measured output, which is the internal mold temperature, and the measurements for all these variables are available at each time step of an experiment. Model identification was done using all the previous experiments which have been stored in a database. A PI strategy designed to track the heater temperatures to 300°C was used to generate the initial set of experiments in this database and, once a sufficient amount of data was collected, the rest of the batches were produced using the modelling and control design as described in section 3.2.5 and 3.3.1. Validation of the model was done on 53 previous experimental batches by splitting the database in a 70:30 fashion for training and testing/validating, respectively.

3.2.5 Model Identification of Batch Data

A deterministic subspace identification routine is applied to come up with a LTI state space model for which the heaters and the compressed air are taken as inputs, and the internal mold temperature is the output. The state space matrices and the system order n are determined using the identification routine on the input-output data and, finally, a PLS model is built to relate the final states of the batch trajectory and the final product quality.

The following structure is imposed while implementing subspace identification :

$$\mathbf{x}_{k+1}^{d} = \mathbf{A}\mathbf{x}_{k}^{d} + \mathbf{B}\mathbf{u}_{k}, \tag{3.1a}$$

$$\mathbf{y}_{k} = \mathbf{C}\mathbf{x}_{k}^{d} + \mathbf{D}\mathbf{u}_{k}, \tag{3.1b}$$

where $\mathbf{A} \in \mathbb{R}^{n\times n}$, $\mathbf{B} \in \mathbb{R}^{n\times m}$, $\mathbf{C} \in \mathbb{R}^{l\times n}$, $\mathbf{D} \in \mathbb{R}^{l\times m}$ are the associated system matrices

and $\mathbf{x}_k^d$ denotes the state vector.

Many subspace identification routines (Van Overschee and De Moor, 1995; Larimore, 1990; Verhagen and Dewilde, 1992) use concepts like projections and decomposition from matrix algebra and were developed mainly for continuous processes (Wang and Qin, 2002; Qin et al., 2005; Pour et al., 2010). Processes in continuous operations typically operate in a small band around the mean operation, but batch operations explore the transients of the process and the data from batch processes consist of multiple such experiments and, hence, the identification algorithms need to be adapted accordingly. The following method (Mhaskar et al., 2019) is used to form a pseudo-Hankel matrix from input and output data of all the batches such that the batch nature of the data is preserved. Consider the output measurements of a batch process denoted as $\mathbf{y}^{(b)}[k]$, where k is the sampling instant since the batch trial was initiated, and b denotes the batch index. The input Hankel matrix for a single batch b and cumulative pseudo-Hankel matrix for all batches is given by Equations 3.2 and 3.3 below:

$$\mathbf{U}_{1|i}^{(b)} = \begin{bmatrix} \mathbf{u}^{(b)}[1] & \mathbf{u}^{(b)}[2] & \cdots & \mathbf{u}^{(b)}\left[j^{(b)}\right] \\ \vdots & \vdots & & \vdots \\ \mathbf{u}^{(b)}[i] & \mathbf{u}^{(b)}[i+1] & \cdots & \mathbf{u}^{(b)}\left[i+j^{(b)}-1\right] \end{bmatrix} \tag{3.2}$$

$$\mathbf{U}_{1|i} = \begin{bmatrix} \mathbf{U}_{1|i}^{(1)} & \mathbf{U}_{1|i}^{(2)} & \cdots & \mathbf{U}_{1|i}^{(nb)} \end{bmatrix} \tag{3.3}$$

where nb is the number of batches used for training.

Similarly, pseudo-Hankel matrices for the output data are formed. After the formation of the pseudo Hankel matrices, model identification is done using the appropri-

ately adapted (Mhaskar et al., 2019) deterministic subspace algorithm (Moonen et al., 1989). A system order of 2 was found to be sufficient to capture dynamics of the internal mold temperature through validation tests. These tests have shown that the training batches were sufficient for the identification routine to derive a linear time invariant model that can capture the dynamics reasonably well.

Since the states in the subspace models are not available for measurement, the models have to be initialised to yield accurate predictions of the state and output trajectory and the terminal states. A Luenberger observer (see Equation (3.4)), a state estimation technique, is utilised during the initial operation of a new experiment. A PI control strategy is used until the measured output temperature crosses 130°C, after which the MPC controller kicks in to control the rest of the sintering phase.

A Luenberger observer of the following form is deployed during the operation of a batch:

$$\hat{\mathbf{x}}[k+1] = \mathbf{A}\hat{\mathbf{x}}[k] + \mathbf{B}\mathbf{u}[k] + \mathbf{L}(\mathbf{y}[k] - \hat{\mathbf{y}}[k]) \tag{3.4}$$

where $\mathbf{L}$ is the observer gain. The observer gain is designed in a way to ensure that $(\mathbf{A} - \mathbf{LC})$ is stable, i.e., poles of $(\mathbf{A} - \mathbf{LC})$ are placed within the unit circle. In MATLAB, this is achieved using the *place* command.

The obtained dynamic model alone is not sufficient to predict the quality of the product, since these quality measurements are not available throughout an experiment. Recall that the product quality measurements are only available at the end of each experiment. Hence, the process model is used to construct a prediction of the entire state trajectory of the experiment using the input moves decided by the predictive controller and then relate the final states to the product quality. This framework allows for us to link the process inputs to the qualities. The quality model (PLS) takes the following form:

$$Q_{t_f} = \hat{R}_m + \hat{P}_m x[t_{f_{heat}}] + e \tag{3.5}$$

where Q_{t_f} denotes quality measurements at the termination of batch at time t_f, $\hat{P}_m$ and $\hat{R}_m$ are the linear quality model parameters, $x[t_{f_{heat}}]$ are final states as predicted by the state-space model (i.e., at completion of the heating cycle at time $t_{f_{heat}}$) and e represents white noise.

It is to be noted that the accuracy of both the state space model and PLS model are taken into account while deciding upon the system order of the state space model. In summary, at the end of the identification step, the process evolution is adequately captured using a state space dynamic model augmented with a PLS quality model.

There has been previous work done on building local linear models (Narasingam and Kwon, 2017; Narasingam et al., 2022; Narasingam and Kwon, 2020) which can appropriately be used to describe the non-linearities of the process. In the published results, the authors utilize a simulation problem, in which all states of the system are measured, i.e., the states are the same as outputs of the system. More importantly, they assume that the measured variables are the system states. They have first formed an 'extended'/'lifted' state space by choosing non-linear functions of the states system and then they have used k-means clustering on the trajectory of the 'lifted' state space for identifying optimal number of regions for building local models. However, in our study, it is important to note that the states are unknown and the number of measured output could be and are different. Due to this, it is not easy to construct a 'lifted' state space since we do not have access to the states in the first place. Moreover, our objective function is to maximize the 'final' product quality. It must be noted that the product qualities (which are the variables of interest) can not be measured online, and can only be measured using destructive methods, once a batch experiment is done and hence only a final product quality measurement is available for each batch.

Since the quality model relates the final states to the product qualities, it would have to choose between the multiple models. Since the switching of the models would be dependent on the process dynamics and not time, it would require special treatment both from a model development perspective and an optimization perspective and remains a subject of future work.

Remark 16. *Recurrent Neural networks (RNNs) or Non-linear Auto-Regressive eXogenous models (NARX) models can be used for describing the non-linear dynamics of the process. Linear time invariant state space models are used in the current work since they are easier to incorporate within an optimizer for predictive control. Moreover, they readily give access to the states of the process, which is useful for the current modelling framework in predicting the quality variables. On the other hand, the basis functions and other considerations, like over-fitting, have to be taken into account while building non-linear models to integrate within the MPC, and this remains one direction of focus for future work.*

3.3 Adaptive predictive control for Rotomolding

An overview of the modelling strategy based on subspace identification is presented in this section along with the design for an adaptive predictive controller for rotomolding.

3.3.1 Model Predictive Controller (MPC)

A data driven model as obtained from Section 3.2.5 is utilised by an adaptive economic control strategy to obtain products with desired specifications. The controller's performance is directly linked to the accuracy of the data driven model in capturing the behaviour of the process. Given below is the optimization formulation inside the MPC scheme:

$$\min_{\mathbf{U}_f, l_f} \quad \sum_{k=0}^{k=l_f-l} u_f[k] \tag{3.6a}$$

$$s.t. \quad U_{j,min} \leq u_f[k] \leq U_{j,max}, \quad \forall \ 0 \leq k \leq l_f - l \tag{3.6b}$$

$$|u_f[0] - u[l-1]| \leq \delta, \tag{3.6c}$$

$$|u_f[k] - u[k-1]| \leq \delta, \ \forall \ 1 \leq k \leq l_f - l \tag{3.6d}$$

$$\hat{\mathbf{x}}[0] = \hat{\mathbf{x}}[l] \tag{3.6e}$$

$$l_f \in \{t_{switch} + 400, t_{switch} + 500\} \tag{3.6f}$$

$$\mathbf{\Lambda}\hat{Q}_{t_f} \leq \mathbf{\Gamma} \tag{3.6g}$$

$$\hat{\mathbf{x}}[k+1] = \mathbf{A}\hat{\mathbf{x}}[k] + \mathbf{B}\mathbf{u}_f[k] \tag{3.6h}$$

$$\hat{\mathbf{y}}[k] = \mathbf{C}\hat{\mathbf{x}}[k] + \mathbf{D}\mathbf{u}_f[k] \ \forall \ 0 \leq k \leq l_f - l \tag{3.6i}$$

$$\hat{Q}_{t_f} = R_m + P_m\hat{x}[l_f] \tag{3.6j}$$

with,

$$\hat{Q} = \begin{bmatrix} \hat{Q}_1 & \hat{Q}_2 \end{bmatrix}^T \tag{3.7a}$$

$$\mathbf{\Lambda} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \tag{3.7b}$$

$$\mathbf{\Gamma} = \begin{bmatrix} 2 \\ -5 \end{bmatrix} \tag{3.7c}$$

where $\mathbf{U}_f = [\mathbf{u}_f[0], \mathbf{u}_f[1], ...\mathbf{u}_f[l_f - l]]$ are the set of decision variables which are the two heaters and the compressed air input values at each sampling instance over the control horizon and l_f denotes the termination time of the heating cycle. Based on past experience, the set of admissible termination times, l_f, is specified through Equation (3.6f). $\delta = \begin{bmatrix} 60 & 60 & 60 \end{bmatrix}^T$ is the allowed rate of input change specified through

Equations (3.6c) and (3.6d), and $U_{j,min}$ and $U_{j,max}$ are the lower and upper bounds on the input variable (Equation (3.6b)), with $U_{j,min}$ and $U_{j,max}$ being $\begin{bmatrix}0 & 0 & 0\end{bmatrix}^T$ and $\begin{bmatrix}100 & 100 & 100\end{bmatrix}^T$, respectively. In addition, t_{switch} denotes the time (in *sec*) at which the controller switches to MPC from PI controller.

Further, Equation (3.6g) represents the user-specified constraints on the quality variables. Finally, Equation (3.6j) represents the quality model which the MPC uses to predict the terminal product quality using the terminal states (of the heating cycle), which in turn are produced by the dynamic state-space model as specified in the Equations (3.6h)-(3.6i). In Equation (3.7a), $\hat{Q}_1$ and $\hat{Q}_2$ denote the two quality variables, namely sinkhole area coverage (%) and impact energy (Kg.m), respectively.

It must be also noted that the control horizon shrinks in the case of control for batch processes, in contrast to that for continuous operation, since a batch only has a fixed total duration. The inputs represent the power of the heaters and compressed air (more specifically, the percentage of the sampling time, 10 seconds, for which the input is to be set in the ON condition). The objective function is to minimize simply the sum of all inputs across all of the control horizon (3.6a), which essentially translates to reducing the total power consumption. It can be noted that other variations of the objective function representing operating costs can be used readily in this formulation as required. Additionally, the batch termination time is also a decision variable and is accounted for while formulating the optimization formulation.

Remark 17. *It must be noted that the proposed control strategy does handle variable batch duration. There are two choices for batch duration and these candidate batch times are given based on past experiments and suggestions from the industry. The MPC at each step, solves the problem for both the batch times separately and simply chooses the best solution based on the economics of the process. Equation* (3.6f) *shows the candidate batch times.*

MATLAB is used for implementing the suggested control strategy through LabView, which acts as a data acquisition system, communicating with the sensors and actuators. Every 10 seconds, which is the sampling interval, the control input is calculated and implemented through the aforementioned software setup. The initial (current) states are computed using the Luenberger observer (Equation (3.6e).

Though the above stated optimization formulation looks like an integer optimization problem, two linear programs are solved using *linprog* in MATLAB at each step corresponding to each of the candidate batch termination times, and the solution corresponding to the one with better economic objective is chosen and implemented. This approach is suitable in this case, since there are only two candidates and the computations are fairly quick, since they are all linear programming problems. However, more number of candidate batch termination times may be added, which may result in increase in computational complexity, but this might not be necessarily helpful due to the existing plant-model mismatch. In summary, the best termination time for the batch is determined by the optimization problem, along with the best input sequence needed to obtain a good product.

A set of layers of relaxations for the optimization problem are provided internally for preventing any breakdowns during an ongoing experiment, which might occur due to infeasibilities. If the original problem is infeasible, then the RHS values in quality constraints (Equation 3.6g) are reduced (essentially allowing for relaxed quality product) and the optimization problem is attempted again. In this way, an implementable solution is achieved. The infeasibilities can arise if the specified set of qualities turn out to be simply unreachable within the termination time during an ongoing experiment within the input constraints. It might also occur initially when there is a significant plant model mismatch. Due to the proposed algorithm, the feasibility lost due to plant model mismatch is alleviated since the model improves and consistent state estimates are obtained using the Luenberger observer as more process information

becomes available.

Remark 18. *The proposed control strategy can handle other measurements for describing the product, like shear viscosity, by including them in the set of quality variables that need to be tracked. However, appropriate training data must be provided for training the new quality model with the new set of qualities and modifying the corresponding equations in the optimizer.*

Remark 19. *Currently, the optimizer focuses only on achieving an input sequence with a low overall energy consumption, along with meeting the user specified quality requirements. The present control strategy can be further modified to take into account the smoothness of the input profiles, as per industrial limitations, by including a penalty term for the rate of change in the input in the objective function of the MPC. Moreover, the objective function can also include additional terms detailing the production costs and profits associated with the product qualities, along with a suitable set of weights for each of the terms.*

Remark 20. *In the current modelling solution, the three inputs are treated as independent variables and, hence, a dynamic model with three inputs and one output is built. Even though this framework produces satisfactory results, in reality, the three inputs, i.e., the two heaters and compressed air, combine to give a unified heating effect. The input space can be reduced further by a dimensional reduction technique such as principal component analysis to obtain the 'principal' input(s) affecting the system, after which a state space model can be built between the resulting new inputs and the output, and can then be utilised within the MPC.*

Remark 21. *As is evident from the temperature profile from Figure 3.3, the rotational molding process is a multi-phase process, which means the dynamics of the process are going to change when moving between phases. Currently, we deploy a PI controller for the first phase and then implement the proposed MPC algorithm for which we have developed a single model using just the second phase data from all the batches. One*

direction for future work can involve building individual models for each of the phases and designing a suitable control strategy.

Remark 22. *The final product quality predictions are accessible with the current modelling strategy at each time step. Although this information is utilised by the MPC anyway, such a framework also allows the operator to relate one's understanding of the physics of the process to the input sequence suggested by the MPC and the corresponding predicted qualities. In this way, one's trust on the predictive controller is established.*

3.3.2 Model Monitoring and Re-identification

Recall that the technique described in Section 3.2.5 determines an LTI state space model. The application of such a model based controller to a nonlinear process, and especially for situations possibly different from the training data, leaves the room (and opportunity) for an improved/re-identified model when considering a new batch. This calls for an appropriately designed process monitoring and model re-identification strategy.

In designing the re-identification strategy, recognizing the distinction between batch operation and continuous processes is key. Traditionally, in continuous processes where the historical data is collected from the same sequence, usually operated across an operating point, re-identification is done using the most recent data to account for the most recent process behaviour. In batch re-identification, past approaches have included utilizing the training batches in conjunction with the data from the current batch to identify a new model (Kheradmandi and Mhaskar, 2018; Garg et al., 2019). While being useful, this approach does not account for the fact that the primary goal of the modelling technique in the context of predictive control is to come up with a model that can accurately *predict*, instead of focusing on the ability to *estimate* the

past behavior. This section presents a re-identification technique specifically targeted to batch operation and control.

An adaptation of an existing model monitoring technique (Kheradmandi and Mhaskar, 2018) is first presented in this section. Note that a candidate input sequence is computed by the MPC, which uses a process model to predict future output trajectory and minimizes the appropriate objective function. Therefore, the performance of the model at each time step is crucial in order to achieve the best closed-loop performance. Oftentimes, model mismatch due to non-linearities, multiple phases in the process, etc. degrades the predictive capability of the identified data driven model. Here in this manuscript, we focus on capturing the validity of the model, which would trigger re-identification and the re-identification algorithm itself, while appropriately accounting for the current state location of an ongoing batch.

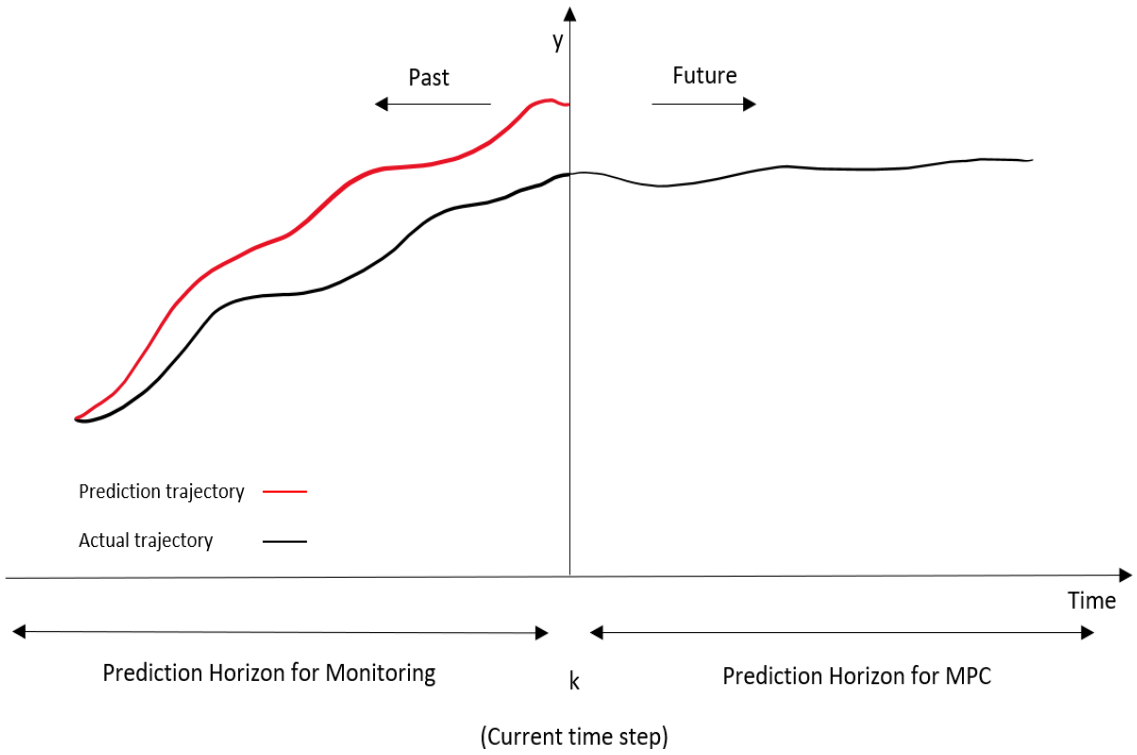

Figure 3.1: Monitoring algorithm (Kheradmandi and Mhaskar, 2018)

The intention of a monitoring tool would be to quantify the drifting of predicted

outputs from the observed outputs. One must realise that it is not possible to monitor the drift for future instances, since the measurements are not yet available. Therefore, the only way would be to march backwards a few time steps and then let the model run until the current time step, but without an observer. The predicted output and the measured output at the current instant are then compared and linked to the re-identification trigger.

The experiment is set up such that all the output measurements (internal mold temperature), observer state trajectories and input sequences are stored. After the first few time steps of the experiment, which would be run through a PI controller, the MPC algorithm kicks in, within which the monitoring algorithm is embedded. In this algorithm, at every time step, the observer-free state sequence and the predicted output sequence from few time steps behind is calculated over a nominal monitoring window, chosen by the user (a value of 5 is taken for this work), until the current time step. This has been illustrated in Figure 3.1. The absolute difference between the predicted output at the current time step and the measured output is compared with a tolerance set by the user. Once this difference hits a specified threshold, re-identification is triggered. To this end, the model error at a time instant k, i.e., the end-point prediction error, $EPPE_k$ is quantified as follows:

$$EPPE_k = \|y_k - \hat{y}_{k|k-w}\| \tag{3.8}$$

where y_k is the measured output at the current time instance k, $\hat{y}_{k|k-w}$ is the observer-free prediction of the output as seen from the $(k-w)^{th}$ time instance, and w is the monitoring window, chosen as 5 for this work.

Once the re-identification is triggered, the following steps are carried out to come up with a new model: In the first step, the current value of the 'locator' is stored. In this case, the locator is just the internal mold temperature, but for a more general

example, the locator can be an appropriate combination of the measured outputs for a case where there are multiple outputs in the system. Once the current value of the locator is noted, the algorithm goes back to the existing past experimental database and, for each batch, marks the times at which the value of the internal mold temperature corresponds to the locator's current value from the ongoing experiment. The algorithm then collects the data only from these marked time instances (see Figure 3.2 for an illustration) until the end of their respective batches, and performs model identification as mentioned in 3.2.5. Finally, once the new model is obtained, the observer gain L is also updated and the entire state trajectory for the current ongoing batch up to the current time instance is re-calculated using the new model and the new observer, thus readying it to be deployed in the MPC.

Remark 23. *There is a subtle difference and a similarity between the locator and the batch alignment variable used in dynamic PLS modelling. The batch alignment technique used in PLS batch identification aligns all of the batch data with respect to an alignment variable, so that all the batch trajectories are of the same length. This is done to accommodate batches of different lengths into the PLS algorithm. Both the batch alignment variable and the proposed locator rely on the existence of a process variable that has a monotonic trend throughout the batch, i.e., continuously increasing or decreasing for the whole duration of the experiment. Note that the 'locator' being defined for this work is philosophically different from a batch alignment variable. The locator is not used to re-align the data but rather just marks the start and end points for a batch, suitable for re-identification of the model. Thus, in contrast to the alignment variable in PLS based approaches, the locator variable enables the use of a controller implementation that allows for achieving batches with potentially different duration.*

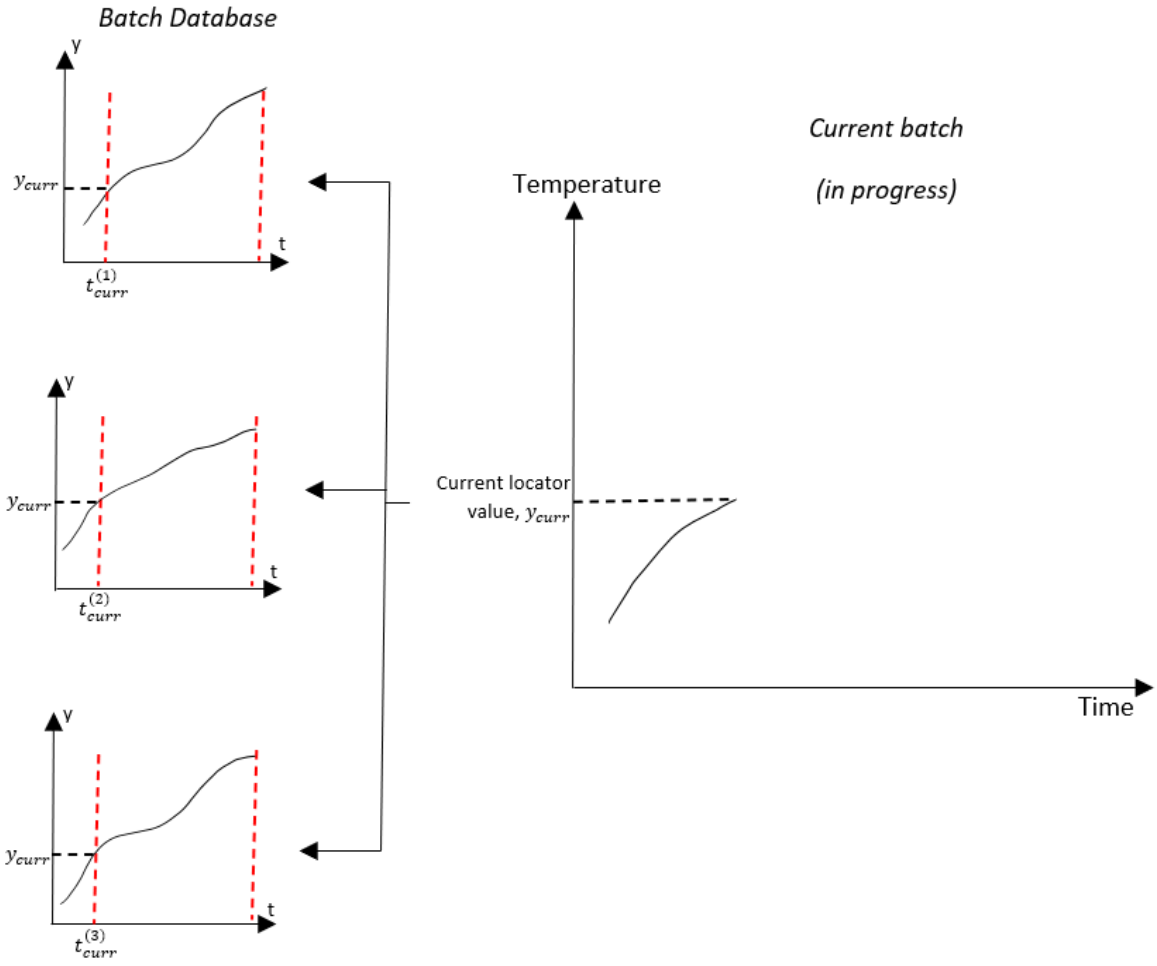

Figure 3.2: Re-identification algorithm

3.4 Results

The proposed re-identification algorithm with the monitoring technique, along with a previously designed EMPC (Chandrasekar et al., 2022), is deployed on the Rotational Molding experimental setup. First, the effect of the proposed algorithm on the output predictions (i.e., the internal mold temperature) and the predicted quality variables is compared to the case with an existing re-identification algorithm (Kheradmandi and Mhaskar, 2018) and also to a case where no re-identification is applied. This comparison is demonstrated on an already existing database of batch experiments

to demonstrate the improved model achieved using the proposed re-identification approach.

Closed loop results on the rotomolding setup, showcasing the better performance of EMPC under the proposed algorithm as compared to when there is no re-identification, are presented next. Two batch experiments for each of the two scenarios were conducted to demonstrate this comparison.

The input and output profiles for a sample rotational molding batch experiment is shown in Figure (3.3). The trajectory for internal mold temperature of a sample batch

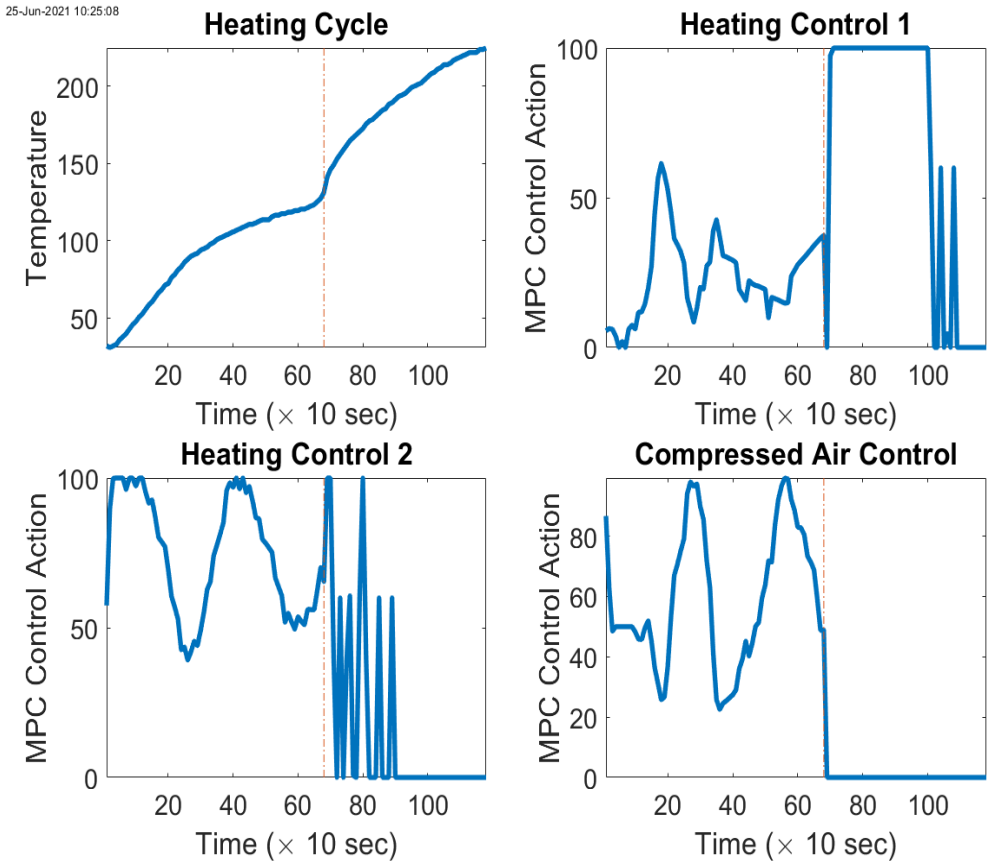

Figure 3.3: Input Output profiles of a representative Rotomolding batch

is also shown in Figure 3.4, in which the accuracy of the output trajectory prediction

given by the model built using the proposed re-identification strategy and an already existing strategy is compared, along with the case when there is no re-identification involved.

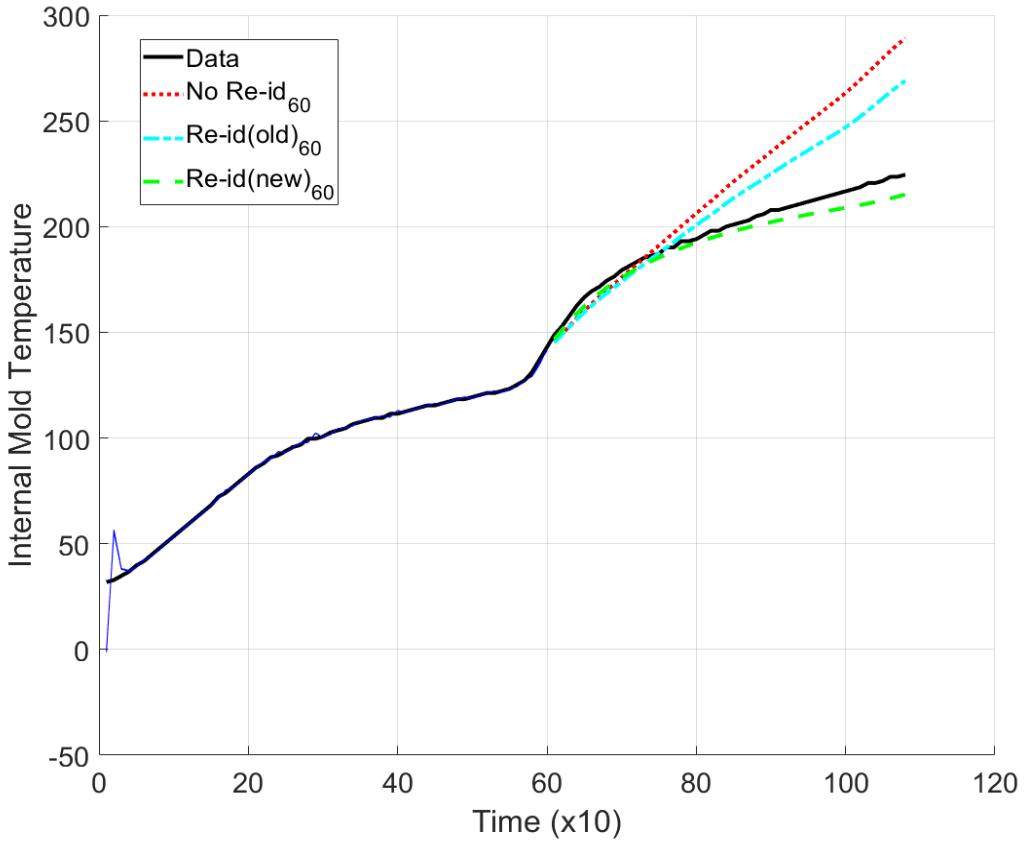

Figure 3.4: Comparison of output trajectory prediction as seen from times $t_1 = 60$ for the three strategies: without any adaptive modelling framework (dotted lines), with an existing re-identification strategy (Kheradmandi and Mhaskar, 2018) (dash-dotted lines), and with the new proposed re-identification algorithm (dashed lines) in comparison with the process trajectory (solid lines)

As it is evident from Figure (3.4), the ability of the model to predict the rest of the batch is much better when a re-identification algorithm is incorporated. Additionally, the strategy proposed in the current manuscript results in better models during re-identification than the ones obtained using an existing strategy, proposed in (Kheradmandi and Mhaskar, 2018). The predicted terminal quality variables are also much

closer to the actual values for the proposed algorithm than the other two cases for a given batch (Table:3.1). This is because the quality model directly relates the terminal state predictions to the quality variables and, hence, the efficacy of this model is dependant on the accuracy of the state sequence itself, which is only guaranteed with a good model.

Table 3.1: Comparison of quality prediction accuracy ($t_1 = 60$ and $t_2 = 70$)

Batch Characteristics	Q_1, Q_2
Measured	$(0.5469, 6.74)$
No-reident batch 1 (predicted from t_1)	$(-2.44, 7.18)$
No-reident batch 2, (predicted from t_2)	$(-1.89, 7.02)$
Reident batch 1 (existing strategy), (predicted from t_1)	$(-2.33, 7.62)$
Reident batch 2 (existing strategy), (predicted from t_2)	$(-1.43, 7.05)$
Reident batch 1 (proposed strategy), (predicted from t_1)	$(2.32, 5.85)$
Reident batch 2 (proposed strategy), (predicted from t_2)	$(1.18, 6.35)$

Now that we have seen that the proposed strategy performs and validates well against the past data, a similar level of performance can be expected when the algorithm is deployed on a new batch experiment under closed loop control. Thus the next half of this section will include the closed loop results obtained with the proposed control algorithm.

Table:3.2 shows the input costs and the quality variables of the obtained product in four different batch experiments, two of which are controlled using the presented EMPC scheme without the proposed re-identification algorithm and the other two, with said algorithm. It can be inferred from this table that the final product qualities obtained in a new rotomolding batch run using the proposed strategy are better than the case where the re-identification strategy is not deployed. Although this may not be evident by comparing the input profiles for the two cases (Figure 3.5), the table shows the clear difference. The re-identified model is able to predict the qualities better and, hence, the MPC is able to take the system far closer to the true and, in turn, better qualities as well. Consequently, energy costs reduction goes from an average of 17.6% for batches without re-identification to 23.3% for batches with re-identification.

Table 3.2: Summary of closed loop results

Batches $Q1 \leq 2\%$ $Q2 \geq 5Kg.m$	Total Input consumption (PI+MPC)	Input consumption (MPC)	Percentage energy reduction	Q_1	Q_2
Average training set values	14.91×10^3	6.94×10^3	-	2.07	5.84
No reident case, Batch 1	15.6×10^3	5.76×10^3	17 %	1.34	4.67
No reident case, Batch 2	14.9×10^3	5.66×10^3	18.2 %	0.55	6.74
Reident case, Batch 1	14.6×10^3	5.46×10^3	21.3%	0.47	6.74
Reident case, Batch 2	14.3×10^3	5.18×10^3	25.3%	0.30	7.78

Coming up with an ideal internal mold temperature profile and a good input profile for different quality specifications and varying process conditions has remained very challenging. The proposed algorithm allows the ability of the MPC to continuously adapt the model as the batch proceeds, and the results have shown its ability to achieve better product quality, along with lowered energy consumption, which is a significant advancement compared to the current industrial practice. It is recognized that a better model is only achieved when a part of the process is covered and, consequently, the MPC has to work with possibly a sub-par model during the initial time steps. A better model for the MPC may yield better products and, hence, future work will consider developing a non-linear model for the overall system.

Remark 24. *The proposed re-identification algorithm seems to be computationally intensive. But it must be noted that during the experiment, the re-identification step takes about a maximum of 0.5 seconds and the rest of the MPC calculation at each time step takes no more than 0.06 seconds. With the control interval being 10 seconds, there are no issues with implementing this on the experimental setup.*

Remark 25. *It is important to note that, though there are underlying relaxed problems which are solved in case of failure/infeasibility, given the quality constraints, the proposed EMPC in all of the experimental data under comparison here has solved the complete optimization problem and did not relax any of the constraints, showing that user requirements can be theoretically met. Although, the controller does choose the appropriate termination time as it cannot be guaranteed to be the same for all the batches and, hence, sometimes, two experiments that run with the same formulation*

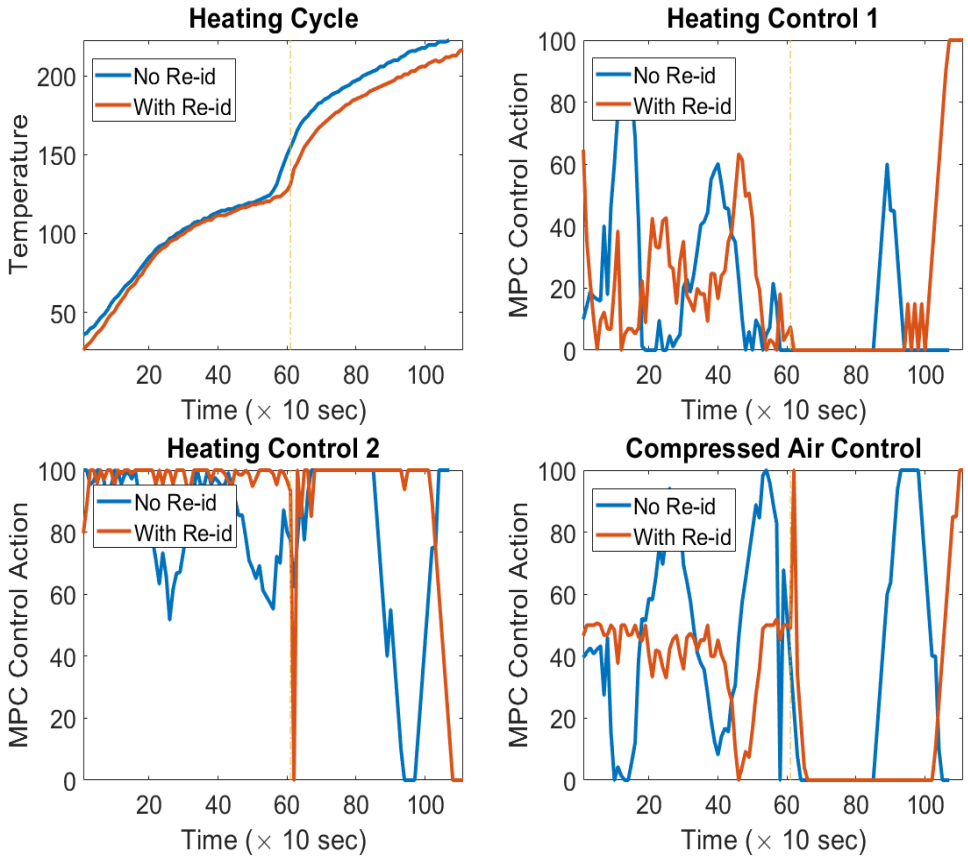

Figure 3.5: Input Output profiles of batches with and without the re-identification algorithm deployed

might have different batch lengths due to disturbances or plant-model mismatch, which eventually gets accounted for under the re-identification algorithm.

Remark 26. *In the present work, the key quality variables, sinkhole area and impact strength are measured through destructive means, i.e., the sides of the product are cut, subject to physical deformation and analysed separately. However, there are alternate ways to get these measurements for quality assessment without physically involving with the product (Gomes et al., 2019), such as utilizing ultrasonic spectral data. Also, currently, a thermocouple from inside the oven gives the internal mold temperature readings. In a more practical and an industrial setting, a thermocouple may not be inserted in the oven, but a heat camera may be available. Hence, an image analysis*

tool can be developed to track the quality variables during an experiment, and in turn used for quality control, and this remains the subject of future work.

Remark 27. *In the present framework, a linear time invariant model for the dynamics of the internal mold temperature, coupled with a PLS model for the quality variables, is used for describing the entire process, and the entire model within the MPC results in a convex optimization problem which is easily solvable. One direction for future work can focus on constructing non-linear models for the dynamics of the process or the qualities, or both, and designing a suitable control framework for superior quality control.*

3.5 Conclusions

Oftentimes, due to plant-model mismatch or inherent non-linearities, a single linear model might not be able to capture all of the process dynamics. The present rotomolding application poses a similar problem and, hence, demands an adaptive modelling solution where the model is continuously improved during the process. The current study provides one such suitable approach specifically designed for batch processes, along with the open-loop and closed-loop results, where a comparison to a case with no such adaptive framework is also made. Subspace identification is applied on all the past data for coming up with an LTI dynamic model between the inputs and the output, and a PLS model is built between the terminal states as predicted by the LTI model and the product quality measurements. The evolution of the internal mold temperature and the product qualities are cumulatively predicted by both the aforementioned models. This whole framework sits inside an EMPC scheme as equality constraints, along with the quality requirements, which are also given as a set of constraints. The experimental results demonstrate the method's improved performance as the process progresses and, in turn, achieve better quality products

with better economics compared to both the open-loop control, and MPC without re-identification.

Bibliography

Abu-Al-Nadi, D. I., Abu-Fara, D. I., Rawabdeh, I., and Crawford, R. J. (2005). Control of rotational molding using adaptive fuzzy systems. *Advances in Polymer Technology*, 24(4):266–277.

Chandrasekar, A., Garg, A., Abdulhussain, H. A., Gritsichine, V., Thompson, M. R., and Mhaskar, P. (2022). Design and application of data driven economic model predictive control for a rotational molding process. *Computers & Chemical Engineering*, 161:107713.

Corbett, B. and Mhaskar, P. (2016). Subspace identification for data-driven modeling and quality control of batch processes. *AIChE Journal*, 62(5):1581–1601.

Flores-Cerrillo, J. and MacGregor, J. F. (2002). Control of particle size distributions in emulsion semibatch polymerization using mid-course correction policies. *Industrial and Engineering Chemistry Research*, 41(7):1805–1814.

Flores Cerrillo, J. and MacGregor, J. F. (2005). Latent variable MPC for trajectory tracking in batch processes. *Journal of Process Control*, 15(6):651–663.

Garg, A., Gomes, F. P., Mhaskar, P., and Thompson, M. R. (2019). Model predictive control of uni-axial rotational molding process. *Computers & Chemical Engineering*, 121:306–316.

Gomes, F. P., Garg, A., Mhaskar, P., and Thompson, M. R. (2019). Data-Driven Advances in Manufacturing for Batch Polymer Processing Using Multivariate

Nondestructive Monitoring. *Industrial & Engineering Chemistry Research*, page acs.iecr.8b05675.

Hamidi, A., Farzaneh, S., Nony, F., Ortega, Z., Khelladi, S., Monzon, M., Bakir, F., and Tcharkhtchi, A. (2016). Modelling of sintering during rotational moulding of the thermoplastic polymers. *International Journal of Material Forming*, 9(4):519–530.

Kadali, R., Huang, B., and Rossiter, A. (2003). A data driven subspace approach to predictive controller design. *Control Engineering Practice*, 11(3):261–278.

Kassidas, A., MacGregor, J. F., and Taylor, P. A. (1998). Synchronization of batch trajectories using dynamic time warping. *AIChE Journal*, 44(4):864–875.

Kheradmandi, M. and Mhaskar, P. (2018). Model predictive control with closed-loop re-identification. *Computers & Chemical Engineering*, 109:249–260.

Kourti, T. (2003). Multivariate dynamic data modeling for analysis and statistical process control of batch processes, start-ups and grade transitions. *Journal of Chemometrics*, 17(1):93–109.

Larimore, W. (1990). Canonical variate analysis in identification, filtering, and adaptive control. In *29th IEEE Conference on Decision and Control*, pages 596–604 vol.2. IEEE.

Ljung, L. (1998). *System Identification: Theory for the User*. Pearson Education.

Mhaskar, P., Garg, A., and Corbett, B. (2019). *Modeling and Control of Batch Processes*. Advances in Industrial Control. Springer International Publishing, Cham.

Moonen, M., De Moor, B., Vandenberghe, L., and Vandewalle, J. (1989). On- And Off-Line Identification Of Linear State Space Models. *International Journal of Control*, 49(1):219–232.

Narasingam, A. and Kwon, J. S.-I. (2017). Development of local dynamic mode decomposition with control: Application to model predictive control of hydraulic fracturing. *Computers & Chemical Engineering*, 106:501–511.

Narasingam, A. and Kwon, J. S.-I. (2020). Application of koopman operator for model-based control of fracture propagation and proppant transport in hydraulic fracturing operation. *Journal of Process Control*, 91:25–36.

Narasingam, A., Son, S. H., and Kwon, J. S.-I. (2022). Data-driven feedback stabilisation of nonlinear systems: Koopman-based model predictive control. *International Journal of Control*, pages 1–12.

Ogila, K. O., Shao, M., Yang, W., and Tan, J. (2017). Rotational molding: A review of the models and materials. *Express Polymer Letters*, 11(10):778–798.

Pour, N. D., Huang, B., and Shah, S. L. (2010). Subspace Approach to Identification of Step-Response Model from Closed-Loop Data. *Industrial & Engineering Chemistry Research*, 49(18):8558–8567.

Qin, S. J., Lin, W., and Ljung, L. (2005). A novel subspace identification approach with enforced causal models. *Automatica*, 41(12):2043–2053.

Van Overschee, P. and De Moor, B. (1995). A unifying theorem for three subspace system identification algorithms. *Automatica*, 31(12):1853–1864.

Verhagen, M. and Dewilde, P. (1992). Subspace model identification Part 1. The output-error state-space model identification class of algorithms. *International Journal of Control*, 56(5):1187–1210.

Wang, J. and Qin, S. J. (2002). A new subspace identification approach based on principal component analysis. *Journal of Process Control*, 12:841–855.

Chapter 4

A Data-driven Hybrid Modelling Approach for Batch Processes

The contents of this chapter have been published in Computers & Chemical Engineering Journal.

- Chandrasekar, Aswin, Susu Zhang, and Prashant Mhaskar. "A hybrid hubspace-RNN based approach for modelling of non-linear batch processes." Chemical Engineering Science 281 (2023): 119118.

The following is the authorship contribution statement: Aswin Chandrasekar: Conceptualization, Methodology, Formal analysis, Software, Writing – original draft. Susu Zhang: Software. Prashant Mhaskar: Conceptualization, Methodology, Writing – review & editing, Supervision, Project administration, Funding acquisition.

Abstract

The manuscript addresses the problem of developing a modelling strategy that can accurately capture the dynamics of a non-linear batch process, demonstrated on a uni-axial rotational molding process. To this end, this work presents a strategy that utilizes the nonlinear modeling capability of a recurrent neural network (RNN) model, while leveraging the rigor of subspace-based approaches. The proposed modeling strategy is as follows: a subspace identification algorithm is first utilized on all the training batches to obtain the state sequence for these batches. Then the state sequences from all the training batches and the corresponding input sequences are given as the output and the input to a neural network, respectively. This step essentially builds a non-linear state space model, albeit using the state trajectory identified by the subspace model. The output equation obtained from the above-mentioned subspace identification step along with the newly obtained non-linear state equation is now ready to be used for predicting the output trajectory of the system. The results from the experiments illustrate the improved prediction performance of a model obtained by the proposed hybrid Subspace-RNN approach in comparison to both a subspace-based approach and an RNN-based approach.

4.1 Introduction

Batch processes are utilized in various fields, such as mechanical, biochemical, agricultural, and pharmaceutical. Deployment of a good control strategy is essential to producing on-spec product. Model Predictive Control (MPC) has been a standard control technology used for many processes in the industry (Forbes et al., 2015). An MPC has an optimizer which uses a dynamic model of the process to predict the future trajectory of the process variables given an input sequence. For the MPC to function effectively, the model has to capture not only the behavior of the process variables

that are measured continuously during the process, but also the so-called quality variables (the variables that characterize product specification) which are often measured at the end of the process. For the motivating example, the mold temperature for the rotational molding process that is measured continuously is the process variable, whereas the sinkhole area along with the impact strength of the product that are measured only after the end of the batch, are the quality variables of the system.

One method to achieve a high-fidelity model that captures the evolution of the process is to develop a first principles model. For many processes, it might be challenging and time consuming to come up with an accurate first principles model simply because the physics involved might be too complex with huge number of parameters, or in the instance of quality variables, not captured (as is the case of the motivating example in this paper). For the Rotomolding process, the batch process considered for this work, there has been some attempts in the past (Abu-Al-Nadi et al., 2005; Ogila et al., 2017; Hamidi et al., 2016) on first principles modelling but they do not capture the relationship between the inputs of the system and variables like sinkhole area and impact strength which are the key quality variables of interest, and hence are not useful and not preferred for model based control. These models, at most, can be utilized to improve adherence to a predetermined mold temperature trajectory, but they are not suitable for accomplishing quality control.

As an alternative to first principles models, linear modelling techniques like Partial Least Squares (PLS) regression have been successfully used for modelling and control of batch processes in the past (Flores-Cerrillo and MacGregor, 2002; Flores Cerrillo and MacGregor, 2005). These techniques, requiring equal length batches, describe the process by projecting the input output information onto a latent space and building correlations to the quality variables in that space. One method to handle uneven batch lengths is to set an alignment variable for the batches in the database (Kourti, 2003; Kassidas et al., 1998), such that all the batches are first adjusted to have the

same length. These techniques work really well for monitoring, but are inherently not designed to distinguish between inputs and outputs (for the purpose of feedback control).

On the other hand, linear state space models obtained from subspace identification do not face the issues that the previously discussed techniques face. With this method, the batch data need not be aligned while building a linear time-invariant (LTI) state space (SS) model. Quality variables of the batch system can also be linked to these models (Corbett and Mhaskar, 2016) by modifying existing modelling designs for processes (Ljung, 1998; Kadali et al., 2003; Pour et al., 2010; Narasingam and Kwon, 2020). However, LTI SS models are by definition linear dynamic models and the modeling approach stands to gain from a careful inclusion of nonlinear modeling ability.

Previous work has been done in developing adaptive modelling and control strategy (Kheradmandi and Mhaskar, 2018; Chandrasekar et al., 2022a) for batch processes, where the authors primarily use an LTI SS model and use the information from an ongoing experiment to correct the model using their suggested algorithms. However, since the model gets better only in the later stages of a batch run, it should be realised that there is still room for improvement, and this can be achieved by using a non-linear model instead.

Non-linear machine learning based models like neural networks that have become increasingly popular in the process modelling and control domain (Bhat and McAvoy, 1990; Hunt et al., 1992; Sjöberg et al., 1995; Shaw et al., 1997; Zhang et al., 1998; Tian et al., 2001; Wu et al., 2020) provide an attractive alternative, albeit with some challenges. Recurrent Neural Networks (RNN) and Non-linear Auto Regressive eXogenous (NARX) models are some of the common methods that have been explored in the area of process modelling and control in particular over the last few years. Ren et al. have reviewed and demonstrated Model Predictive Controllers (MPC) based on

Recurrent Neural Networks (RNN) and Encoder-Decoder structures on an example Continuous Stirred Tank Reactor (CSTR) system (Ren et al., 2022). However, it is accurately pointed out that factors like the complexity of the model, initial guess to the solver, and the type of the solver itself can affect the computational time taken by the MPC to compute the optimal action. Moreover, a sizeable amount of process data is required to train a neural network model since even a seemingly simple neural network might contain a significant number of parameters that need to be tuned. With increasing number of parameters, the immediate concern is the time taken for training the network and more importantly, the risk of over-fitting remains, which could manifest negatively in the context of feedback control.

There remains an opportunity to model process non-linearity using machine learning techniques, but in a way that does not result in overfitting. Motivated by these considerations, the present work focuses on coming up with an RNN model scaffolded onto a subspace identification technique. A modelling approach that combines the robustness of subspace approaches with neural networks in the proposed method hasn't been explored before in the past. There were many types of hybrid modelling techniques in the past, where some prior knowledge regarding the process would be incorporated while developing a data driven model (Sansana et al., 2021). It could be used in restricting the structure of a data-driven model (Patel et al., 2020), or can get aid from a data-driven model in predicting the true process behaviour in a series or a parallel configuration (Thompson and Kramer, 1994). In a parallel approach, the residuals between the first principles model and the true data are calculated and then a black box model is fit for the residuals. Lee et al. have adopted a parallel hybrid approach to model dynamics of intracellular signaling pathway which is only partially known (Lee et al., 2020). Ghosh et al. (Ghosh et al., 2019) have demonstrated superior predictive performance of such a parallel hybrid modelling approach on a batch polymethyl methacrylate polymerization process. On the other hand, for a series approach, the black box model is used to calculate the parameters for the first principles

model and then the first principles model is used for predictions. Bangi et al. and Shah et al. have demonstrated the superior prediction performance using a series configuration-based hybrid modelling on hydraulic fracturing process and industrial fermentation process respectively (Bangi and Kwon, 2020; Shah et al., 2022). The strength of such modelling strategies became evident when working with low volume of data.

There is a hybrid model of an alternate series configuration where a first principles model is used to reinforce the relationship between inputs and the outputs while fitting a data-driven model (Aguiar and Filho, 2001). Such a configuration allows the user to fit high-fidelity models using low volume of data(Sansana et al., 2021). The approach proposed in this manuscript follows a similar idea, where we use a robust linear modelling strategy (subspace identification) to generate extra information (states) about the process and then fit a neural network model. On the other hand, physics informed neural networks are a type of hybrid models where first principles knowledge about the process is incorporated via constraints (formed by differential equations) while training a neural network on the process data. Bangi et al. (Bangi et al., 2022) have used such networks for modelling a batch fermentation process. In many instances (like the rotomolding setup), there might not be a first principles model available so the suggested hybrid strategy from previous works may not be applicable. Second, in these strategies, one cannot validate the individual models of the hybrid model (first principles and data driven) separately since there is no knowledge on which portion of the data belongs to the first-principles behaviour and which portion of it belongs to the data-driven, in contrast to our work, where the subspace model which is fit first, can be verified directly with the data, and because of the argument given in Remark 5, we can proceed to fit the RNN model of our hybrid approach and verify it with the states given by the subspace model in the first step. This will be clear towards the end of the manuscript once the proposed methodology has been established.

In particular, while the subspace technique is utilized to first determine state trajectories from the training batches, a neural network is used to capture the state transition. The predictive performance of the proposed technique is demonstrated using data from a uni-axial rotomolding batch process and compared with a pure RNN approach and a pure subspace-based approach. The rest of the manuscript is organized as follows: Section 4.2 of this paper explains the uni-axial rotational molding process and its lab-scale setup in detail, followed by a review of the subspace identification approach for batch processes, and recurrent neural network models. The proposed approach is presented in Section 4.3. The predictive performance of subspace based linear state space models, recurrent neural networks and the proposed subspace-RNN models are compared in Section 4.4. Concluding remarks are presented in Section 4.5.

4.2 Preliminaries

This section briefly introduces the background of the rotomolding process.

4.2.1 Rotational Molding Process

A lab-scale uni-axial rotational molding machine is used to produce plastic molded products. The inputs (heater powers and compressed air supply) and the outputs of the rotational molding setup are monitored and manipulated through LabVIEW and MATLAB programs. A powder of high-density polyethylene is charged into the mold at room temperature while the oven is pre-heated to 300°C. The oven is then placed over the mold which is now set to rotate at a constant rotational speed. There are three inputs, two heaters and a compressed air unit, and one output, internal mold temperature in the oven. Once the heating phase is complete, a fan is used to air-cool the mold containing the product before the product is removed for quality testing

where the sinkhole area and the impact strength of the product are assessed.

The degree of sintering in rotational molding is measured via surface voids. When the polymer particles does not undergo complete sintering during the process, the resultant product tends to show considerable amount of surface voids. ImageJ is used to perform the surface void analysis on the images of the mold taken by a digital camera. Another quality that needs to be quantified is the strength of the product and one way to measure this is via dart impact testing. The impact energy calculation is based on mean failure height which is defined as the height from which 50% of the samples fail. This height is then multiplied by the mass of the dart to get the impact energy in units of kg-m. Specific details relating to these quality variables in the context of this work can be found in our previous work (Chandrasekar et al., 2022b,a).

4.2.2 Data Measurement

53 batches of rotomolding experiments are available with each batch lasting around 15-18 minutes. The input and output data is collected (and also the control action is implemented) every 10 seconds. A PI strategy designed to track the heater temperatures to 300°C was used to generate the initial set of experiments in this database and, once a sufficient amount of data was collected, the rest of the batches were produced using the control strategy described in (Chandrasekar et al., 2022b,a). Essentially, an Economic Model Predictive Controller has been formulated, minimizing the input energy consumption while meeting the product quality requirements, posed as constraints in the optimization problem, along with the model and input constraints. Validation of the model was done by splitting the database in a 70:30 fashion for training and testing/validating, respectively.

4.2.3 Subspace Identification and Partial Least Squares for Quality Modelling

Recall that a state space model is of the following form:

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k), \\ \mathbf{y}_k &= \mathbf{g}(\mathbf{x}_k, \mathbf{u}_k), \end{aligned} \tag{4.1}$$

An LTI state space model is a special case where the functions f and g are linear operators on the state and input variables and takes the following structure:

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k, \tag{4.2a}$$

$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{D}\mathbf{u}_k, \tag{4.2b}$$

where $\mathbf{A} \in \mathbb{R}^{n \times n}$, $\mathbf{B} \in \mathbb{R}^{n \times m}$, $\mathbf{C} \in \mathbb{R}^{l \times n}$, $\mathbf{D} \in \mathbb{R}^{l \times m}$ are the associated system matrices and $\mathbf{x}_k$ denotes the state vector.

In the present manuscript, the LTI state space model is identified between the inputs and the output (process variables) using a deterministic subspace identification routine as presented in (Moonen et al., 1989) and adapted for batch processes in (Corbett and Mhaskar, 2016). The order n of this state space model is determined by assessing the fit on the testing data. Eventually, a linear quality model is built between the terminal states of a batch and the quality of the product obtained in that particular batch, using PLS. This way the quality variables get linked to the process variables. In this batch system, the two heaters and the compressed air are the inputs the internal mold temperature is the sole output and the sinkhole area percentage along with impact strength of the product are the quality variables.

Matrix algebra concepts like decomposition and projection techniques were used in the past to come up with different variations of the subspace algorithm (Larimore, 1990; Van Overschee and De Moor, 1995; Verhagen and Dewilde, 1992). However these were developed within the context of continuous processes alone (Wang and Qin, 2002; Pour et al., 2010; Qin et al., 2005). Batch processes are fundamentally different from continuous processes since the latter often operates around a 'process mean' but there is no such mean for batch operation since the process traverses across wide range of conditions. Furthermore, batch data is a collection of many experiments which are not continuous to each other, unlike the case with continuous process data, due to which the existing subspace algorithm needs to be modified to account for such form of data. (Mhaskar et al., 2019) details a method to respect the batch nature of the data while using the subspace algorithm by forming a pseudo-Hankel matrix of inputs and outputs of individual batches first and then concatenating them horizontally to get the cumulative pseudo-Hankel matrix for all the batch data. The equations for forming these matrices for the inputs are given by Equations 4.3 and 4.4 below:

$$\mathbf{U}_{1|i}^{(b)} = \begin{bmatrix} \mathbf{u}^{(b)}[1] & \mathbf{u}^{(b)}[2] & \cdots & \mathbf{u}^{(b)}\left[j^{(b)}\right] \\ \vdots & \vdots & & \vdots \\ \mathbf{u}^{(b)}[i] & \mathbf{u}^{(b)}[i+1] & \cdots & \mathbf{u}^{(b)}\left[i+j^{(b)}-1\right] \end{bmatrix} \tag{4.3}$$

$$\mathbf{U}_{1|i} = \begin{bmatrix} \mathbf{U}_{1|i}^{(1)} & \mathbf{U}_{1|i}^{(2)} & \cdots & \mathbf{U}_{1|i}^{(nb)} \end{bmatrix} \tag{4.4}$$

where U is the Input-Hankel matrix, $u[k]$ is the input at time step k, (b) denotes the particular batch, nb is the number of batches used for training, i is the number of Hankel rows, $j^{(b)}$ is the number of Hankel columns which changes for each batch since

each batch can be of different length.

Likewise, the above equations are also used to form the output pseudo-Hankel matrices. Following (Mhaskar et al., 2019), the deterministic subspace technique (Moonen et al., 1989) is implemented using these input and output Hankel matrices to obtain the state sequence and the model matrices. For the present manuscript, a deterministic subspace algorithm is used for identifying the state sequences and the system matrices. A system order of 2 fit and the performance of the state sequence directly obtained from the subspace algorithm was found to be sufficient, when used along with the output equation 4.2b and compared with testing data. It must be noted that due to the batch nature of the data, the noise is negligible compared to the process transients. If we recall deterministic-stochastic algorithm where the term containing the correlation of the noise and the inputs can be ignored due to the previous argument and thus the problem reduces to a deterministic identification algorithm. While the results are not presented in the manuscript, both the algorithms were tested and produced very similar results.

Subsequently, the quality model is built using the previously obtained LTI SS model to generate the terminal states of each batch and then linking them with the quality measurements using PLS. Such a modelling strategy involving two models is necessary because the state space model alone cannot predict the quality variables, since the measurements for these variables are only available after the end of a batch and not at every time step. It is to be noted that the accuracy of both the state space model and PLS model are taken into account while deciding upon the system order of the state space model. The PLS model for the quality variables is as follows:

$$Q_{t_f} = \hat{R}_m + \hat{P}_m x[t_{f_{heat}}] + e \tag{4.5}$$

where Q_{t_f} denotes quality measurements at the termination of batch at time t_f, $\hat{P}_m$

and $\hat{R}_m$ are the linear quality model parameters, $x[t_{f_{heat}}]$ are final states as predicted by the state-space model (i.e., at completion of the heating cycle at time $t_{f_{heat}}$) and e represents white noise.

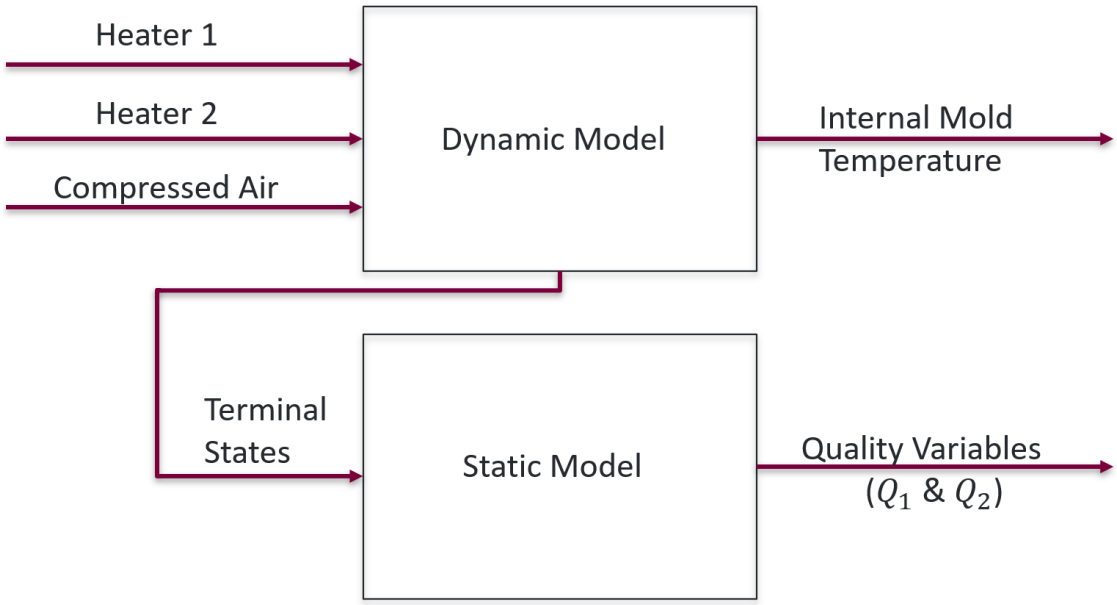

Figure 4.1: Model structure

The PLS model, along with the state space model thus predict the product qualities given an input sequence to a batch. The structure of the modelling approach is shown in Figure 4.1, which essentially uses two models to capture both, the dynamics of the process variable (in this case the internal mold temperature) and also the quality variables. One could also however, use a one-model approaches using a Partial least squares (PLS) regression model. As discussed in the introduction section, a PLS model can be fit to directly relate the quality variables to the input-output trajectories, but the batches needs to be aligned prior to fitting a model since all batches needn't be of the same length. Subspace techniques adapted for batch processes do not require batches to be of the same length. They give a compact model as well using the notion of state space representation with the 'states' storing the information of the process dynamics. Hence it is intuitive to connect the final 'states' of a batch sequence to the

product quality using a PLS model. This is one way to efficiently utilize the dynamic relation between the input and the output and relate it to the quality variables and this method. In the subsequent sections, we will demonstrate that the dynamic model can be improved by replacing the LTI SS model with a non-linear dynamic model.

4.2.4 Recurrent Neural Network

Artificial neural networks are nonlinear models with parameters where the functions are determined by use of inner layers with nodes in each of them, all of which are connected hence representing a 'network'. The nodes act as activation/kernel/filter functions which transform the incoming variable into higher dimensional space, and the connections between the nodes are assigned weights, acting like a normalising factor. The simplest neural network can be thought of as a 3-layered network, where the first contains the inputs, second is a hidden layer with number of nodes selected as a tuning parameter, and the last layer is the output.

Mathematically, the relation between a signal entering and leaving the node is represented as follows:

$$X^{(n)} = F_n(WX^{(n-1)}) \tag{4.6}$$

where, $X^{(n)}$ represents the signals coming out after n^{th} layer of nodes, W represents weight matrix going from $X^{(n-1)}$ to $X^{(n)}$ and F_n contains series of activation functions of all the nodes in the n^{th} layer. Some of the activation functions which are commonly used include hyperbolic tangent, rectified linear unit and sigmoid.

ANNs are not inherently designed to capture dynamics. One way to model dynamic chemical process data using neural networks is by manually including lagged terms of the process inputs and the outputs in the input layer (feature space) of the ANN. Recurrent neural network (RNN), which is a modification of ANN, doesn't require these lagged terms to be included as it allows for dynamic behaviour of the inputs to

be readily captured due to its inbuilt structure. As the name suggests, the nodes in this network have a 'recurrent' nature, or in other words, receive feedback from themselves. We will be referring to Elman type of RNN throughout the rest of the manuscript and use the same type of RNN when compared with proposed approach. Elman type of RNN is shown in Figure 4.2. This compensates for the lagged terms mentioned discussed before, thereby allowing the model to capture temporal dynamics.

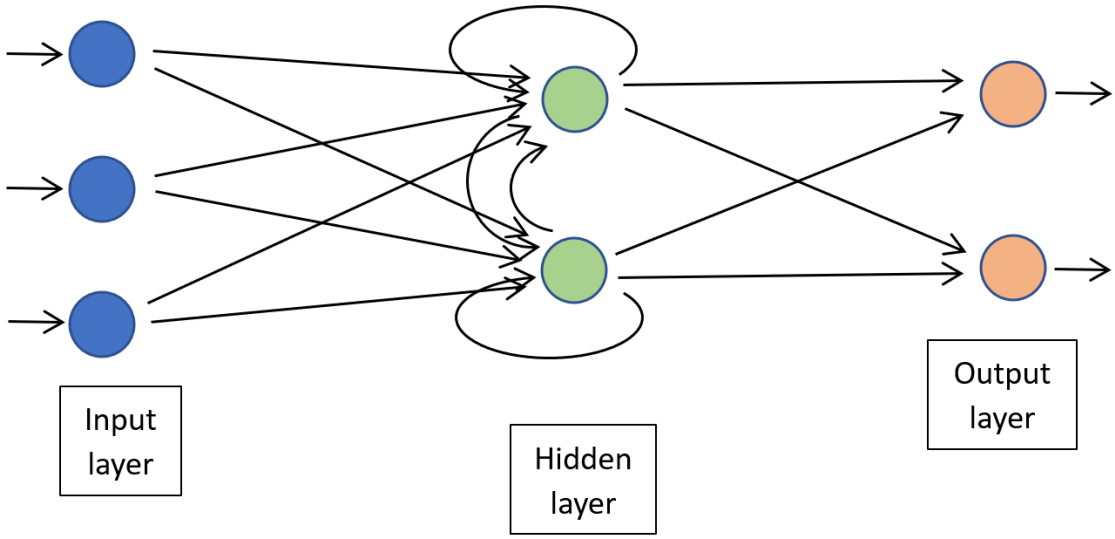

Figure 4.2: A schematic of a simple recurrent neural network with one input layer, one output layer and one hidden layer

The dynamics of the signals passing through the hidden nodes (also called the hidden states) of an Elman type recurrent neural network can be described by the following equation:

$$\begin{aligned} h_t &= f(h_{t-1}, x_t) \\ o_t &= g(h_t) \end{aligned} \tag{4.7}$$

where f is the activation function that is applied on the previous hidden layer values and the inputs to give the current hidden layer value. g is the activation function that actually links two consecutive hidden layers. o here is the values of the next hidden layer in the sequence and for a single hidden layer network, o would be the same as

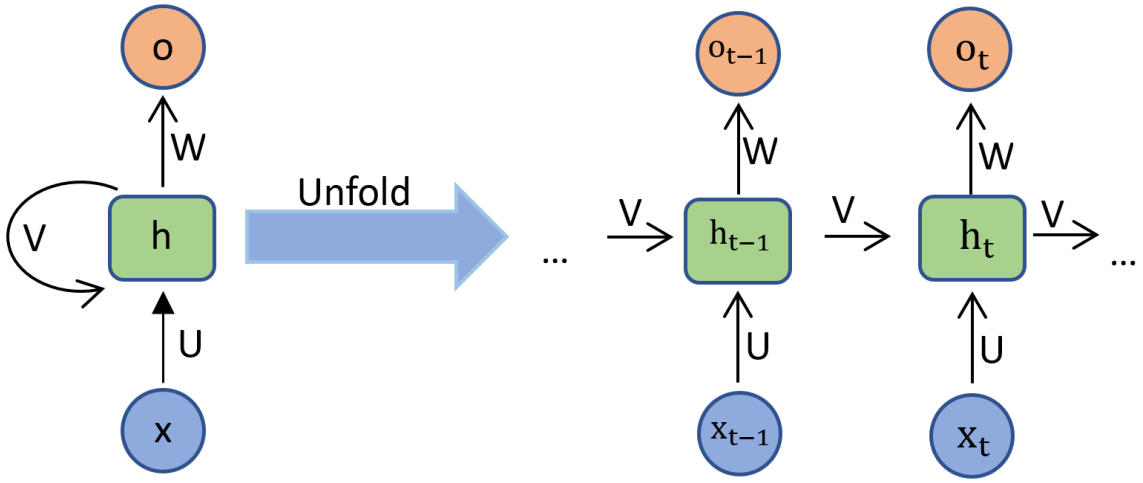

Figure 4.3: Temporal dynamics of signals passing through each node in an RNN model. Here W and U are the respective weighting matrices in the network (which are also parameters of the model). o is a representative variable which acts as the input signal for the next layer in the network.

the outputs y and g would be the output activation function.

RNNs have been used to describe process dynamics in many works although it is interesting to know that most applications of neural networks have been for soft sensing, fault detection (Farhi et al., 2021), modelling (Barton and Himmelblau, 1997; Shahnazari et al., 2019; Song et al., 2020) and process monitoring (Gopaluni et al., 2020) rather than control itself. It should be noted that these applications focus on the overall performance of the model on a given data but do not question the robustness of the modelling techniques and hence do not address issues related to neural networks like overfitting. It can be noticed that Equation (4.1) which is a general state space representation and Equation (4.7) look very similar. In fact if the neural network just had one hidden layer, it would have exactly the same structure as a non-linear state space model. There has been some work done (Kumpati et al., 1990; Zamarreño and Vega, 1999; Zamarreno et al., 2000) on building non-linear state space representations of time series data by adapting a typical neural network structure.

With increasing model complexity, the data requirement increases, the possibility

of several sub-optimal solutions for the parameter estimation problem (Ding et al., 2019) remains. It is worth mentioning that there have been some work regarding model reduction in the context of predictive control, (Narasingam and Kwon, 2020; Prasad and Bequette, 2003; Lee et al., 2000) where a complex and/or a non-linear model is reduced to a simpler model by using latent methods like Principal Component Analysis (PCA) on the internal structure of the given model. However, it must be noted that in (Narasingam and Kwon, 2020), where they have utilised the concept of Koopman operators, the states of the system are known and the user already has an idea of the type of non-linearity to be expected and hence the states are transformed using a priori known non-linear transformations. (Prasad and Bequette, 2003) have built a non-linear state space model using an RNN, and iteratively tried to reduce the number of neurons in the hidden layer using PCA. Recent progress notwithstanding, overfitting remains a major challenge especially in highly intricate non-linear modelling techniques such as neural networks.

4.3 Subspace-RNN modelling: Design and implementation

An overview of the proposed hybrid modelling strategy based on subspace identification is presented in this section. The proposed idea utilizes the strength of robustness of subspace identification in conjunction with the nonlinear modeling capability of nueral networks. The key idea is as follows: first, any of the subspace identification routines (Moonen et al., 1989; Qin, 2006), adapted to batch data (Mhaskar et al., 2019), is applied on the input-output data. The algorithm generates the system matrices A,B,C,D and the state sequence for the data. An important thing to note here is that the obtained state sequence is the fit state sequence, and if the data were non-linear, the fit state sequence would deviate from a state sequence formed using

the appropriate initial condition and the previously estimated system matrices. This is because the subspace algorithm identifies the Kalman state sequence, which obeys equation 4.8.

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k + \mathbf{K}(\mathbf{y}_k^{data} - \mathbf{y}_k), \tag{4.8a}$$

$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{D}\mathbf{u}_k, \tag{4.8b}$$

where $\mathbf{A} \in \mathbb{R}^{n\times n}$, $\mathbf{B} \in \mathbb{R}^{n\times m}$, $\mathbf{C} \in \mathbb{R}^{l\times n}$, $\mathbf{D} \in \mathbb{R}^{l\times m}$ are the associated system matrices and $\mathbf{x}_k^d$ denotes the Kalman state vector and $\mathbf{y}_k^{data}$ is the measured output at k^{th} time step.

The filter term is what makes the equation inherently non-linear, since the non-linearity in state sequence comes from possible nonlinearity observed in the measurements. The measured output therefore, is potentially some arbitrary non-linear function of the inputs and the state equation can be re-written as a non-linear function of just the state and the input.

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k + \mathbf{K}(\mathbf{g}(\mathbf{u}_k) - \mathbf{y}_k), \tag{4.9a}$$

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k), \tag{4.9b}$$

Now we only need to fit an RNN with 1 lag to the state sequence obtained from subspace identification and the input sequence to arrive at a non-linear state equation represented by Equations 4.9. Finally this non-linear state equation represented by the RNN and the output equation together can be used to predict the output sequence, given an input sequence. Note that fitting a regular RNN to the data is much more

involved since one would also have to tune the lag parameters for the hidden layers and the inputs and check for overfit accordingly.

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k), (RNN) \tag{4.10a}$$

$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{D}\mathbf{u}_k, (State\ Space\ model) \tag{4.10b}$$

Remark 28. *It should be noted that subspace identification with respect to Kalman states was discussed in Section 4.2.3 suggesting the use of deterministic-stochastic identification of the LTI SS model. However, due to the batch nature of the process data, the magnitude of the noise is negligible compared to the process transients and hence both, deterministic and deterministic-stochastic identification give the same set of results and the same has been tested and checked.*

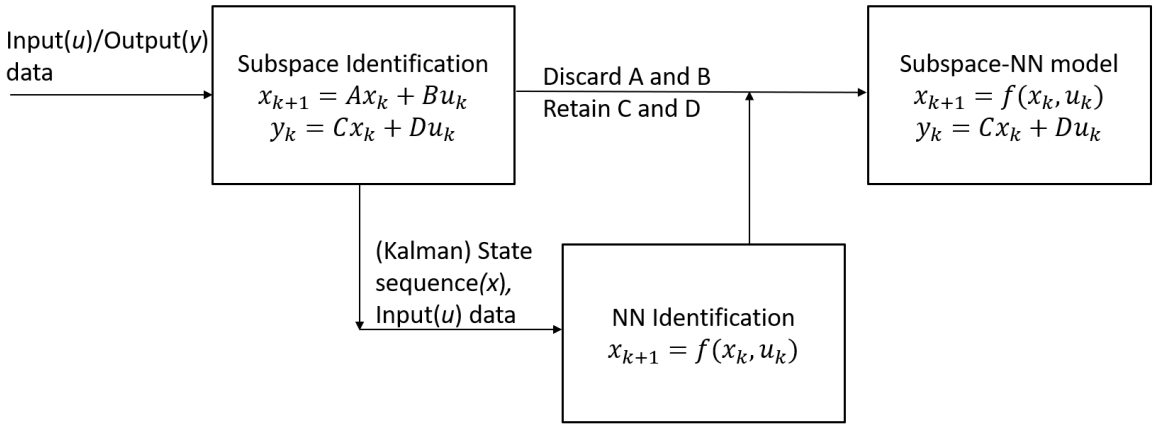

Figure 4.4: Proposed Hybrid modelling strategy

The whole exercise of replacing the right hand side of the Kalman filter equation is to get an expression only in terms of the states and the input and omit the dependence on the measurements. The model needs to be eventually incorporated with an MPC for control purposes and hence such a form capable of predicting multiple steps ahead

is desired. It must be noted that Equations (4.8) may be modified to get multi-step ahead predictions, but doing so will require the future measurement values, which won't be available in reality. Hence there is a need to replace this term with a non-linear function that depends only on the inputs. The schematic of the proposed approach is shown in Figure 4.4.

Remark 29. *Equation* (4.5) *is the quality model which related the final states to the final product quality whereas the output equation relates the state and the input to the output at a given time step. The proposed hybrid approach sits inside the "dynamic model" box in Figure 1. This along with the quality model, i.e., Equations* (4.10) *and* (4.5) *together help in predicting the quality variables along with output and state trajectories, given an input sequence.*

It is worth mentioning that this method is more general than some of the modelling applications in the context of control, involving Koopman operators (Narasingam and Kwon, 2020), Dynamic mode decomposition (Proctor et al., 2016) and PCA (Lee et al., 2000) and could be computationally less expensive as well. This is in the sense that the nonlinearity in the dynamics is not supposed or assumed, but simply mapped to be consistent with what is found with the Kalman filter, and induced by the nonlinearity in the observed data.

Remark 30. *There has been prior work on hybrid modelling techniques (Bhutani et al., 2006; Bangi and Kwon, 2020) where the key idea was to support the existing first principles models using data driven techniques to correct for minor mismatches. However, it must be noted that the present work does not deal with first principles models and the term hybrid refers to the combination of subspace identification and neural networks.*

Remark 31. *One can argue that there are other types of RNN, such as NARX type, that can be constructed which could result in fewer number of parameters. NARX type*

of RNN was indeed trained, tested and compared with the others, and it was found that its performance was even worse (the results are not shown for the sake of brevity). The other argument could be to possibly improve the RNN performance by increasing the network but as we can see from Table 4.2, the performance of an RNN model is not better than the proposed approach already with a higher number of parameters. Increasing the network size for the RNN may guarantee better performance but the same can be done for the proposed hybrid approach as well. The key is to recognize that the proposed hybrid approach performs better than the pure RNN approach using a comparable (or lower) number of parameters.

4.4 Results

The performance of the proposed hybrid modelling approach is evaluated on batch data from the rotational molding setup and compared with two other modelling approaches: 1) Linear Time Invariant State Space model (obtained through subspace identification) and 2) Elman-type Recurrent Neural Network. Specifically, the performance is evaluated on the accuracy in predicting the output trajectories, and the final product qualities of the batches in the testing set.

Before presenting the results, it is important to show the distinction between the states obtained directly from subspace identification routine and the state trajectory obtained using the state space model. The difference is shown through the output trajectory calculated by each of these state sequences when the identification routine is applied on a single batch (Figure 4.5). One of the steps in subspace identification routine results in producing the Kalman-filtered state sequence X of the given data. Model of order 2 is chosen to fit the data. The yellow line in Figure 4.5 represents the output trajectory formed using this state sequence along with the equation $Y = CX + DU$. Subsequently, the red line in Figure 4.5 is formed by running a Luenberger

observer for the first few time steps along with the state space model to let the states converge and then letting the model matrices (A, B, C, D) predict the rest of the output trajectory using Equation (4.2). The visible difference between the two lines is because the states obtained directly from subspace identification in fact follow Equation (4.8) and not Equation (4.2). Since the yellow line closely matches the data, this is the motivation to use the subspace obtained state sequences for identifying a neural network. Note that a single batch is chosen to illustrate the point that the difference between the kalman state produced sequence and the system matrices based dynamics exists regardless. In other words, if data and model built using multiple batches was chosen, one could argue that the mismatch exists due to the system dynamics being averaged over the multiple batches- demonstrating it with a single batch clearly points to the fact that the Kalman state sequence is in essence a 'closed-loop' (in the sense of the measurements being used to generate the state sequence) sequence, motivating the use of this state sequence to build and fit a neural network to capture the nonlinearity.

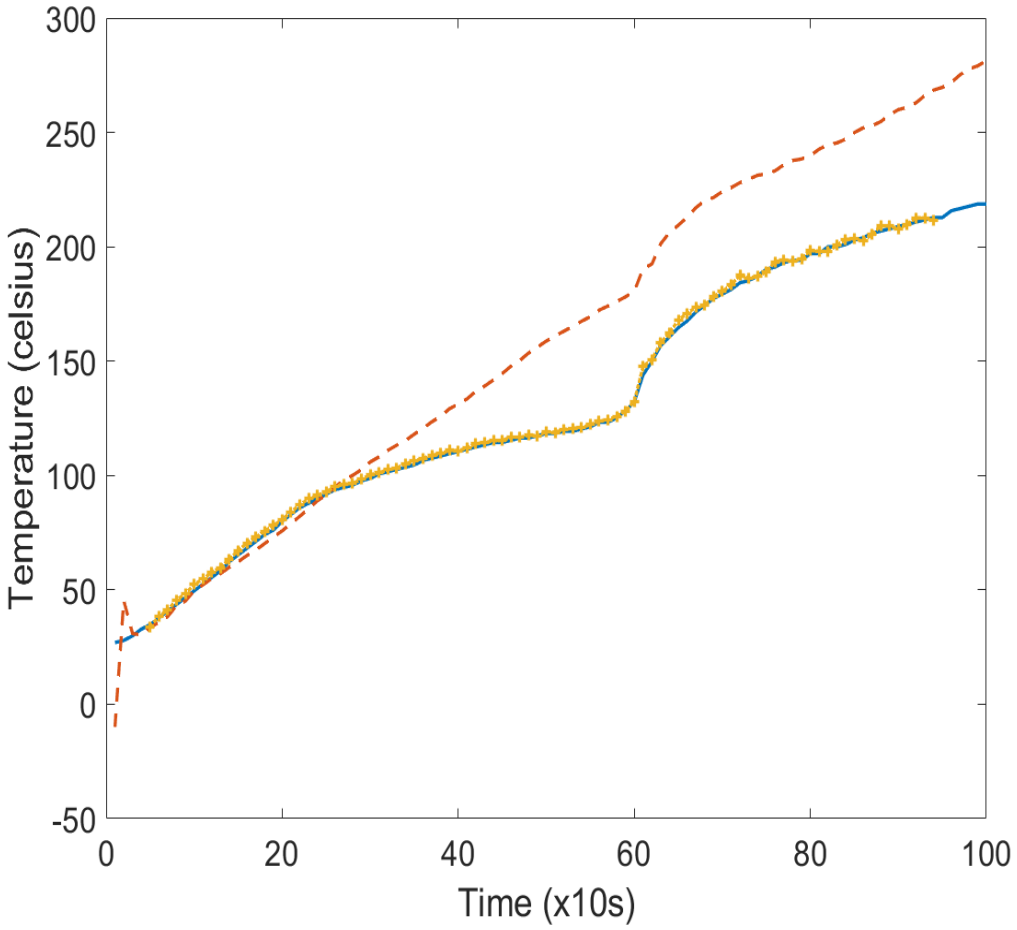

Figure 4.5: Trajectory comparison: 1. Measured output (blue -), 2. Output trajectory obtained with an initial observer, and predicted using the model matrices and input trajectory (red - -), 3. Output trajectory estimated using the subspace state trajectory (yellow +)

Next, we show the ability of the proposed strategy to predict the process output as well as the quality variables in Figure 4.6. An LTI SS model of order 2 is chosen for comparison. MATLAB is used for creating the neural network models. The regular Elman-type RNN is built using ***layrecnet*** function in MATLAB with 1 hidden layer and 10 neurons with tan-sigmoid activation function and with a lag of 1 for the hidden layer. The NN module of the hybrid model on the other hand is built using ***narxnet*** to form a NARX type of RNN with 1 hidden layers with 10 neurons with an output(here

the states x) lag of 1. In particular, the ***narxnet*** model is trained in closed loop by closing the model structure using ***closeloop*** function. ***layrecnet*** models inherently inherently possess a closed structure. It should be noted that the quality model for the regular RNN case is built using the same concept (a PLS regression model) as discussed in the final paragraphs of Section 4.2.3, the 'states' here being the neurons of the hidden layer.

Remark 32. *The quality model can be potentially improved further by using another neural network instead of PLS regression model but it must be noted that there are only* 53 *batches, or,* 53 *data points available for training and it would not make sense to fit a network to such less amount of data. However, the same argument cannot be made for the dynamic model since each batch has roughly around* 100 *time steps implying that there would be* 99 *data points from each batch for the neural network (with state lag of* 1*) to train on.*

To validate the models under comparison, the first initial data points are used to initialise the states (in the case of LTI SS model) or the hidden layers (in the case of neural networks). More specifically, a Luenberger observer is used for the LTI SS model. This is done because the initial conditions of the states are unknown, and the observer uses the output data from the first few instances to get the states closer to their true values. The same is done using the ***preparets*** command in MATLAB for preparing the neural network with the initial data points. Since there is always a factor of uncertainty involved while building neural networks of any size, the networks are fit multiple times and for different train-test splits and the average testing errors are evaluated.

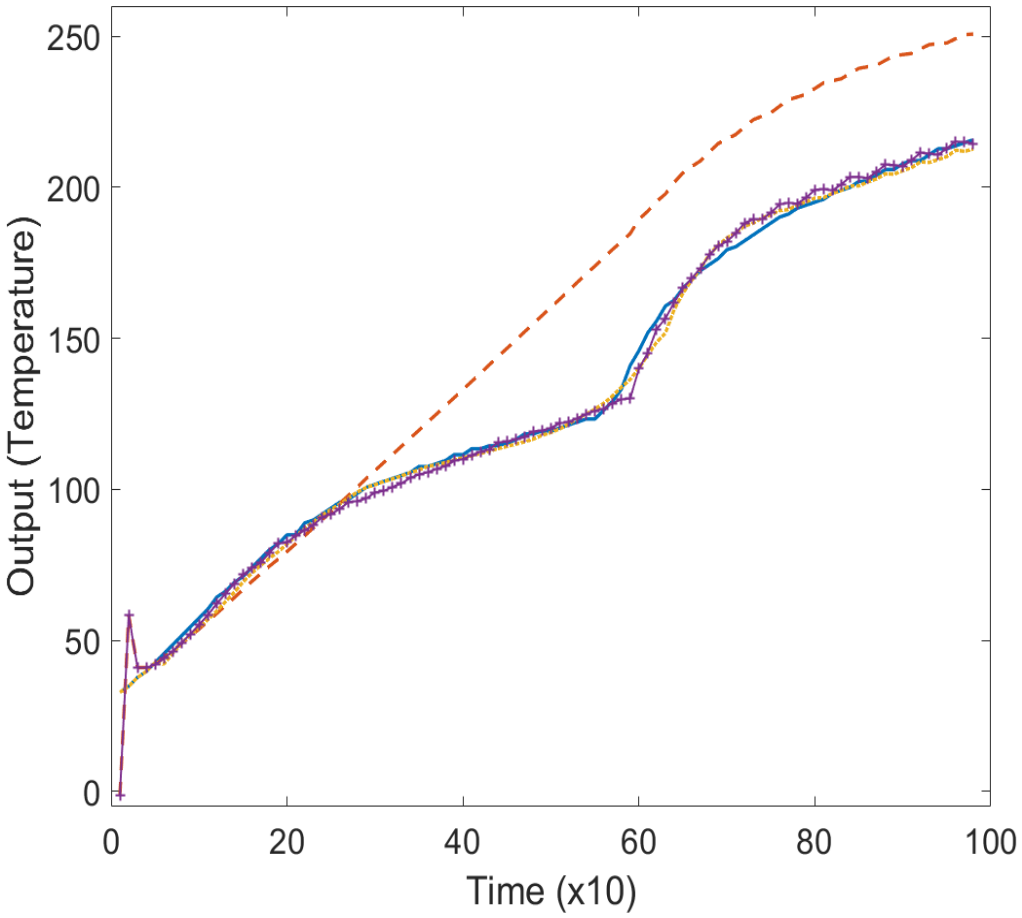

Figure 4.6: Demonstration of the proposed modelling to capture the dynamics of the process output and the quality variables on an illustrative batch. The two measured quality variables for this batch (Blue line) are 1.8, 4.7. Quality variables as predicted by LTI SS model (Dashed red line) are -3.8, 8.2. The same as predicted by an RNN (Dotted yellow line) are 2.2, 6.2, and by the proposed hybrid approach (Crossed purple line) are 2.4, 5.6.

The figure clearly shows there exists room for improvement over the LTI SS model. Moreover, it can also be seen that though the regular RNN approach reasonably captures the dynamics of the output variable, it falls a little short in accurately predicting the quality variables, and the difference in the overall performance across all the batches is shown next.

Table 4.1 shows the consistency in quality predictions as a sample batch progresses.

In particular, the table shows the quality variable predictions by different models from different times t. For example, the column $t = 10$ indicates that the input-output data up until $t = 10$ is used to initialise the model with the internal states and the rest of the trajectory is predicted only using the model. This level of consistency is essential when the model is deployed within a predictive control framework where multiple-step ahead predictions play an important role in deciding the optimal control sequence.

Table 4.1: Error in quality variable predictions as seen from time 't' with different models, for a sample batch

Quality error percentage	$t = 10$	$t = 20$	$t = 30$
LTI SS	59%	14%	31%
Hybrid Subspace-RNN	15%	14%	16%

Table 4.2 shows the number of parameters involved with each modelling strategy, the average error percentage (for the output) and the RMSE (for the qualities) associated with each of them, when applied on the testing data set.

Table 4.2: Summary of the performance of the modelling approaches

	LTI SS	RNN (Elman)	Hybrid Subspace-RNN
Output (Percentage error)	19%	13%	8.6%
Qualities (RMSE)	4.11	11.79	2.32
No. of parameters	15	151	90

It is important to have an idea on the computational burden of using models since these models are eventually to be used within an optimizer while implementing closed loop control. The models trained in this work, however are not extremely complex and both, the proposed hybrid approach and a traditional RNN approach take around $7 * 10^{-3}$ seconds to generate an output sequence by evaluating the function for a given input sequence. Although the difference between the two approaches could get more evident when the complexity of the models are further increased to describe

the dynamics of complex processes since the proposed approach would result in a relatively simpler model than a traditional RNN approach.

Remark 33. *There could be a concern that the output equation, relating y_k and x_k, u_k can also be replaced with a neural network. However, we can see from Figure 4.5 that the output equation from the subspace model already performs well given the Kalman filtered/close-to-accurate state sequence. The figure was an example for one such batch, but the same was tested with all the training and testing batches and the result was the same. Hence we do not really require to modify this equation any further.*

Remark 34. *Overfitting occurs when an unnecessarily bigger model is used to describe the process. Overfitting results in excellent fits and poor validation. We can see from Table 4.2 that the proposed hybrid approach not only uses a smaller model, i.e., 40% fewer parameters but also gives a better performance than a traditional RNN approach, which indicates that the proposed approach is less prone to overfitting than the traditional approach.*

Remark 35. *There has been recent development of faster and efficient computational techniques especially in the domain of Natural Language Processing (NLP) using transformer networks. These networks, due to their structure, are capable of efficiently understanding the context within the input sequence and unlike RNN, the algorithms to train these networks can be parallelized. Sitapure et al. have presented a Time Series Transformer (TST) model based MPC (Sitapure and Kwon, 2023) and have demonstrated its working on a batch crystallization process. Although, TSTs, just like any high fidelity model, require huge amount of good quality data, there is a scope for TST based MPC in an industrial setting. TST based MPC can also be investigated in future for the rotational molding example considered in our manuscript to model the quality variables.*

Remark 36. *In the context of predictive control, the RNN portion of the proposed hybrid model can be further improved by various techniques that have been listed in*

the literature. Wu et al. have applied the ensemble approach, where they first train each RNN model on a different subset of the available data and then use the average predicted value from all the RNN models (Wu et al., 2019a,b). This is done to account for variability and local optimality that is usually observed while solving non-complex optimization problems such as the neural network identification. However, this might require a large amount of data since we would be fitting many RNN models to subsets of the existing data set. Moreover, it is well known that a substantial amount of data set is required to build a neural network of any sort. We currently have around 50 batches, each containing around 100 time steps of data. We do not have the access to a simulation of our system and hence cannot generate generous amounts of datasets which can aid us in identifying RNNs.

4.5 Conclusions

In this manuscript, a novel modelling strategy combining the concepts of subspace identification for linear state space models, and recurrent neural network models, is proposed for modelling non-linear process dynamics. In particular, subspace identification first captures the dynamics of the Kalman states of the process, which can be thought of as a dimensionality reduction step on the input-output space. Since a Kalman filter equation cannot be used for multi-step ahead predictions, it is replaced by a neural network. A NARX type of RNN model is then built on the input-state dynamics rather than the input-output dynamics, which forms the state equation for the resulting non-linear state space model. The output equation for the state space model is retained from subspace identification. The open-loop prediction results demonstrate the benefit of using the proposed hybrid approach over a regular RNN in terms of the prediction accuracy of the output variable and the quality variables, and also the complexity in the network.

Bibliography

Abu-Al-Nadi, D. I., Abu-Fara, D. I., Rawabdeh, I., and Crawford, R. J. (2005). Control of rotational molding using adaptive fuzzy systems. *Advances in Polymer Technology*, 24(4):266–277.

Aguiar, H. C. and Filho, R. M. (2001). Neural network and hybrid model: a discussion about different modeling techniques to predict pulping degree with industrial data. *Chemical Engineering Science*, 56(2):565–570. 16th International Conference on Chemical Reactor Engineering.

Bangi, M. S. F., Kao, K., and Kwon, J. S.-I. (2022). Physics-informed neural networks for hybrid modeling of lab-scale batch fermentation for β-carotene production using saccharomyces cerevisiae. *Chemical Engineering Research and Design*, 179:415–423.

Bangi, M. S. F. and Kwon, J. S.-I. (2020). Deep hybrid modeling of chemical process: Application to hydraulic fracturing. *Computers & Chemical Engineering*, 134:106696.

Barton, R. and Himmelblau, D. (1997). Online prediction of polymer product quality in an industrial reactor using recurrent neural networks. In *Proceedings of International Conference on Neural Networks (ICNN'97)*, volume 1, pages 111–114 vol.1.

Bhat, N. and McAvoy, T. (1990). Use of neural nets for dynamic modeling and control of chemical process systems. *Computers & Chemical Engineering*, 14(4):573–582.

Bhutani, N., Rangaiah, G., and Ray, A. (2006). First-principles, data-based, and hybrid modeling and optimization of an industrial hydrocracking unit. *Industrial & engineering chemistry research*, 45(23):7807–7816.

Chandrasekar, A., Abdulhussain, H. A., Gritsichine, V., Thompson, M. R., and Mhaskar, P. (2022a). Adaptive predictive control algorithm for batch processes:

Application to a rotational molding process. *Industrial & Engineering Chemistry Research*, 61(48):17572–17581.

Chandrasekar, A., Garg, A., Abdulhussain, H. A., Gritsichine, V., Thompson, M. R., and Mhaskar, P. (2022b). Design and application of data driven economic model predictive control for a rotational molding process. *Computers & Chemical Engineering*, 161:107713.

Corbett, B. and Mhaskar, P. (2016). Subspace identification for data-driven modeling and quality control of batch processes. *AIChE Journal*, 62(5):1581–1601.

Ding, T., Li, D., and Sun, R. (2019). Sub-Optimal Local Minima Exist for Neural Networks with Almost All Non-Linear Activations. *arXiv e-prints*, page arXiv:1911.01413.

Farhi, N., Kohen, E., Mamane, H., and Shavitt, Y. (2021). Prediction of wastewater treatment quality using lstm neural network. *Environmental Technology & Innovation*, 23:101632.

Flores-Cerrillo, J. and MacGregor, J. F. (2002). Control of particle size distributions in emulsion semibatch polymerization using mid-course correction policies. *Industrial and Engineering Chemistry Research*, 41(7):1805–1814.

Flores Cerrillo, J. and MacGregor, J. F. (2005). Latent variable MPC for trajectory tracking in batch processes. *Journal of Process Control*, 15(6):651–663.

Forbes, M. G., Patwardhan, R. S., Hamadah, H., and Gopaluni, R. B. (2015). Model predictive control in industry: Challenges and opportunities. *IFAC-PapersOnLine*, 48(8):531–538. 9th IFAC Symposium on Advanced Control of Chemical Processes ADCHEM 2015.

Ghosh, D., Hermonat, E., Mhaskar, P., Snowling, S., and Goel, R. (2019). Hybrid

modeling approach integrating first-principles models with subspace identification. *Industrial & Engineering Chemistry Research*, 58(30):13533–13543.

Gopaluni, R. B., Tulsyan, A., Chachuat, B., Huang, B., Lee, J. M., Amjad, F., Damarla, S. K., Kim, J. W., and Lawrence, N. P. (2020). Modern machine learning tools for monitoring and control of industrial processes: A survey. *IFAC-PapersOnLine*, 53(2):218–229. 21st IFAC World Congress.

Hamidi, A., Farzaneh, S., Nony, F., Ortega, Z., Khelladi, S., Monzon, M., Bakir, F., and Tcharkhtchi, A. (2016). Modelling of sintering during rotational moulding of the thermoplastic polymers. *International Journal of Material Forming*, 9(4):519–530.

Hunt, K., Sbarbaro, D., Żbikowski, R., and Gawthrop, P. (1992). Neural networks for control systems—a survey. *Automatica*, 28(6):1083–1112.

Kadali, R., Huang, B., and Rossiter, A. (2003). A data driven subspace approach to predictive controller design. *Control Engineering Practice*, 11(3):261–278.

Kassidas, A., MacGregor, J. F., and Taylor, P. A. (1998). Synchronization of batch trajectories using dynamic time warping. *AIChE Journal*, 44(4):864–875.

Kheradmandi, M. and Mhaskar, P. (2018). Model predictive control with closed-loop re-identification. *Computers & Chemical Engineering*, 109:249–260.

Kourti, T. (2003). Multivariate dynamic data modeling for analysis and statistical process control of batch processes, start-ups and grade transitions. *Journal of Chemometrics*, 17(1):93–109.

Kumpati, S. N., Kannan, P., et al. (1990). Identification and control of dynamical systems using neural networks. *IEEE Transactions on neural networks*, 1(1):4–27.

Larimore, W. (1990). Canonical variate analysis in identification, filtering, and adaptive control. In *29th IEEE Conference on Decision and Control*, pages 596–604 vol.2. IEEE.

Lee, D., Jayaraman, A., and Kwon, J. S. (2020). Development of a hybrid model for a partially known intracellular signaling pathway through correction term estimation and neural network modeling. *PLoS Computational Biology*, 16.

Lee, K. S., Eom, Y., Chung, J. W., Choi, J., and Yang, D. (2000). A control-relevant model reduction technique for nonlinear systems. *Computers & Chemical Engineering*, 24(2-7):309–315.

Ljung, L. (1998). *System Identification: Theory for the User*. Pearson Education.

Mhaskar, P., Garg, A., and Corbett, B. (2019). *Modeling and Control of Batch Processes*. Advances in Industrial Control. Springer International Publishing, Cham.

Moonen, M., De Moor, B., Vandenberghe, L., and Vandewalle, J. (1989). On- And Off-Line Identification Of Linear State Space Models. *International Journal of Control*, 49(1):219–232.

Narasingam, A. and Kwon, J. S.-I. (2020). Application of koopman operator for model-based control of fracture propagation and proppant transport in hydraulic fracturing operation. *Journal of Process Control*, 91:25–36.

Ogila, K. O., Shao, M., Yang, W., and Tan, J. (2017). Rotational molding: A review of the models and materials. *Express Polymer Letters*, 11(10):778–798.

Patel, N., Nease, J., Aumi, S., Ewaschuk, C., Luo, J., and Mhaskar, P. (2020). Integrating data-driven modeling with first-principles knowledge. *Industrial & Engineering Chemistry Research*, 59(11):5103–5113.

Pour, N. D., Huang, B., and Shah, S. L. (2010). Subspace Approach to Identification of Step-Response Model from Closed-Loop Data. *Industrial & Engineering Chemistry Research*, 49(18):8558–8567.

Prasad, V. and Bequette, B. W. (2003). Nonlinear system identification and model reduction using artificial neural networks. *Computers & chemical engineering*, 27(12):1741–1754.

Proctor, J. L., Brunton, S. L., and Kutz, J. N. (2016). Dynamic mode decomposition with control. *SIAM Journal on Applied Dynamical Systems*, 15(1):142–161.

Qin, S. J. (2006). An overview of subspace identification. *Computers and Chemical Engineering*, 30(10-12):1502–1513.

Qin, S. J., Lin, W., and Ljung, L. (2005). A novel subspace identification approach with enforced causal models. *Automatica*, 41(12):2043–2053.

Ren, Y. M., Alhajeri, M. S., Luo, J., Chen, S., Abdullah, F., Wu, Z., and Christofides, P. D. (2022). A tutorial review of neural network modeling approaches for model predictive control. *Computers & Chemical Engineering*, 165:107956.

Sansana, J., Joswiak, M. N., Castillo, I., Wang, Z., Rendall, R., Chiang, L. H., and Reis, M. S. (2021). Recent trends on hybrid modeling for industry 4.0. *Computers & Chemical Engineering*, 151:107365.

Shah, P., Sheriff, M. Z., Bangi, M. S. F., Kravaris, C., Kwon, J. S.-I., Botre, C., and Hirota, J. (2022). Deep neural network-based hybrid modeling and experimental validation for an industry-scale fermentation process: Identification of time-varying dependencies among parameters. *Chemical Engineering Journal*, 441:135643.

Shahnazari, H., Mhaskar, P., House, J. M., and Salsbury, T. I. (2019). Modeling and fault diagnosis design for hvac systems using recurrent neural networks. *Computers & Chemical Engineering*, 126:189–203.

Shaw, A. M., Doyle, F. J., and Schwaber, J. S. (1997). A dynamic neural network approach to nonlinear process modeling. *Computers & Chemical Engineering*, 21(4):371–385.

Sitapure, N. and Kwon, J. S.-I. (2023). Exploring the potential of time-series transformers for process modeling and control in chemical systems: an inevitable paradigm shift? *Chemical Engineering Research and Design*, 194:461–477.

Sjöberg, J., Zhang, Q., Ljung, L., Benveniste, A., Delyon, B., Glorennec, P.-Y., Hjalmarsson, H., and Juditsky, A. (1995). Nonlinear black-box modeling in system identification: a unified overview. *Automatica*, 31(12):1691–1724. Trends in System Identification.

Song, X., Liu, Y., Xue, L., Wang, J., Zhang, J., Wang, J., Jiang, L., and Cheng, Z. (2020). Time-series well performance prediction based on long short-term memory (lstm) neural network model. *Journal of Petroleum Science and Engineering*, 186:106682.

Thompson, M. L. and Kramer, M. A. (1994). Modeling chemical processes using prior knowledge and neural networks. *AIChE Journal*, 40(8):1328–1340.

Tian, Y., Zhang, J., and Morris, J. (2001). Modeling and optimal control of a batch polymerization reactor using a hybrid stacked recurrent neural network model. *Industrial & Engineering Chemistry Research*, 40(21):4525–4535.

Van Overschee, P. and De Moor, B. (1995). A unifying theorem for three subspace system identification algorithms. *Automatica*, 31(12):1853–1864.

Verhagen, M. and Dewilde, P. (1992). Subspace model identification Part 1. The output-error state-space model identification class of algorithms. *International Journal of Control*, 56(5):1187–1210.

Wang, J. and Qin, S. J. (2002). A new subspace identification approach based on principal component analysis. *Journal of Process Control*, 12:841–855.

Wu, Z., Rincon, D., and Christofides, P. D. (2020). Process structure-based recurrent

neural network modeling for model predictive control of nonlinear processes. *Journal of Process Control*, 89:74–84.

Wu, Z., Tran, A., Rincon, D., and Christofides, P. D. (2019a). Machine learning-based predictive control of nonlinear processes. part i: theory. *AIChE Journal*, 65(11):e16729.

Wu, Z., Tran, A., Rincon, D., and Christofides, P. D. (2019b). Machine-learning-based predictive control of nonlinear processes. part ii: Computational implementation. *AIChE Journal*, 65(11):e16734.

Zamarreno, J., Vega, P., Garcia, L., and Francisco, M. (2000). State-space neural network for modelling, prediction and control. *Control Engineering Practice*, 8(9):1063–1075.

Zamarreño, J. and Vega, P. (1999). Neural predictive control. application to a highly non-linear system. *Engineering Applications of Artificial Intelligence*, 12(2):149–158.

Zhang, J., Morris, A., Martin, E., and Kiparissides, C. (1998). Prediction of polymer quality in batch polymerisation reactors using robust neural networks. *Chemical Engineering Journal*, 69(2):135–143.

Chapter 5

Integrated Dynamic and Quality Modeling Framework for Batch Processes

Authors: Aswin Chandrasekar, Prashant Mhaskar

The content of this chapter is sent to the Chemical Engineering Research and Design Journal and is currently under revision.

The following is the authorship contribution statement: Aswin Chandrasekar: Conceptualization, Methodology, Formal analysis, Software, Writing – original draft. Prashant Mhaskar: Conceptualization, Methodology, Writing – review & editing, Supervision, Project administration, Funding acquisition.

Abstract

This manuscript considers batch process operations and addresses the challenge of identifying a model that synergistically captures the dynamic input-output behavior of continuously measured variables along with the quality variables measured only at batch termination. To this end, an optimization-based framework is developed to identify one model that captures both the dynamics between the inputs and the continuously measured output variables, measurements of which are available at every time step, and the relation between the dynamic "state" information and the terminal quality measurements. Existing approaches either do not identify the dynamic and the quality model simultaneously, or they simply connect the whole trajectory of the process variables with the qualities and do not address the dynamic relationship between the inputs and the process variables. The improved modelling performance of the model obtained from this approach is demonstrated using data from a Uni-axial Rotational Molding process, and compared with existing modelling approaches.

5.1 Introduction

Batch processes constitute a category of processes characterized by an operation conducted over a finite duration. Widely employed in various industries such as pharmaceuticals, biotechnology, specialty chemicals, agriculture, and microelectronics, batch processes offer several advantages. These include the ability to produce multiple related products using the same equipment, adapt to variations in feed, and accommodate changes in product specifications. The relatively low volume of each batch, along with the lack of correlation between batches, makes batch processes particularly valuable in the production of low volume, high-value products. This is attributed to the capability to test product specifications after each batch and discard products that do not meet specifications to satisfy the goal of achieving tight quality control.

This goal is frequently challenged by the influence of feedstock variations and other disturbances on the process evolution. To counteract these disturbances and guide the process towards the intended quality, continuous inputs (such as heat addition and removal, glucose feed concentration, pH of the reactor, feed and bleed rates in a bioreactor to name a few) are applied to the process.

In many batch processes, trajectory-tracking, a simple open-loop control policy is employed, where the same predetermined "best" input trajectory is applied for each batch BONVIN et al. (2006). This approach relies on the assumption that by replicating historically successful actions, the desired product quality will be achieved. Open-loop policies are advantageous for their ease of implementation, lack of requirement for online measurements or process models, and adaptability from lab-scale procedures. However, due to the absence of a feedback mechanism, this approach is incapable of effectively handling disturbances.

To address disturbances and enhance product quality control, closed-loop approaches are necessary. Yet, applying closed-loop quality control to batch processes comes with inherent challenges. Firstly, variables crucial for product quality are often not measured online, making their direct use in quality control impractical. These variables might be either too expensive to measure in real-time or physically impossible to measure continuously. Typically, quality variables are measured in a laboratory setting after the completion of the batch.

More advanced control approaches depend on having a model for the process Shi et al. (2006); Nayhouse et al. (2015); Valappil and Georgakis (2003); Bonvin and Srinivasan (2013); Rafizadeh et al. (2003); Aumi and Mhaskar (2010). While some studies assume the existence of an accurate first-principles dynamic model, the practicality of this assumption is limited Bonvin (1998). Approaches that work for continuous operation, such as identifying a single model via perturbations around a nominal operating condition, do not remain viable for in batch processes due to the wide range of operating

conditions they traverse, requiring special considerations in model identification.

Statistical batch control methods and related model identification can be broadly classified into two categories: batch-to-batch approaches and within-batch approachesFlores-Cerrillo and MacGregor (2005); Chin et al. (2000); Lee and Lee (2006); Kwon et al. (2015). Batch-to-batch modelling approaches assume that disturbances influencing batch quality are correlated from one batch to the next. Models are built to predict the quality of the next batch from the qualities of previous batches and their corresponding input trajectories. While valuable when there is a correlation between batches, these approaches are not applicable when batches are independent events and do not address disturbances within individual batches. This motivates the need for within-batch control, which utilizes measurements taken during the batch in conjunction with a model to reject disturbances as they arise throughout the batch.

Within-batch approaches can be further categorized as trajectory-tracking and quality-control methods. Trajectory-tracking control regulates a process variable to track a historically successful trajectory, often achieved by replacing traditional PI controllers with model-based control. Empirical dynamic models, including local linear ARX models Aumi et al. (2013) and neural networks Sjöberg and Agarwal (2002); Hosen et al. (2011), have been attempted in this context. Another approach involves latent variable methods Yu and Flores-Cerrillo (2013); Golshan et al. (2011); Flores Cerrillo and MacGregor (2005), particularly principal component analysis (PCA), where the correlation between subsequent measurements describes the evolution of the process. However, the drawback of all trajectory-tracking approaches is the potential change in the relationship between the controlled variable and quality variables.

In the context of direct quality control, a model that establishes the relationship between past observations, current observations, the candidate input trajectory, and quality is sought. One commonly used method is Projection to Latent Spaces (PLS) Flores-Cerrillo and MacGregor (2004); Corbett et al. (2015); Wan et al. (2012). In

PLS, trajectories of process measurements and inputs are correlated with quality during the model-building phase. Subsequently, during the control phase for a new batch, the model is applied to available process measurements to determine an appropriate input. An inherent challenge in the control step is that, at certain points in the batch, fewer measurements may be available compared to those used in model fitting. This is because, part way through the batch, only process measurements from the past and the current sampling instant are accessible. Moreover, this type of modelling requires all batches in the training dataset to have the same duration, which may not be the case in many scenarios. Various approaches, such as missing-data algorithms Flores-Cerrillo and MacGregor (2004), batch alignment algorithms Wan et al. (2014) and building independent dynamic models Corbett et al. (2015); Corbett and Mhaskar (2016) to estimate the necessary values, have been explored to address this challenge. It's worth noting that estimating future process measurements essentially poses a dynamic modelling problem.

The independent dynamic model approach, as detailed in Corbett and Mhaskar (2016) utilizes a subspace-based methodology and involves constructing a Linear Time-Invariant (LTI) state-space (SS) model to capture the dynamic relationship between the process variable dynamics and the manipulated variables in a batch, along with a Partial Least Squares (PLS) regression model that establishes the connection between the states derived from the LTI SS model and the final product qualities of that batch. Note that, this approach addresses some of the limitations of previous PLS-based approaches by first identifying the dynamic model and then the quality model separately. However, the dynamic model identification is not cognizant of the variation in the quality variables and solely focuses on the input-output relationship. It only stands to reason that an improved model is likely to be found if both the dynamic and quality models were identified together. A unified technique to identify the dynamic model together with the quality model presently does not exist.

The proposed solution does borrow from the framework of prediction error minimization (PEM) framework Ljung (1998). In a PEM framework, an optimization problem is typically solved to compute the model parameters, where the objective function captures the Euclidean distance between the model's predicted outputs and the actual outputs, with model equations serving as constraints. Even when trying to identify LTI SS models, the optimization turns into a nonlinear programming (NLP) problem. Due to its non-convex nature, an iterative approach is generally adopted, requiring a good initial guess. For a linear state-space model with numerous inputs, outputs, and states, a substantial number of parameters need to be initialized, which requires careful consideration. To avoid manual selection of the initial guess, subspace identification is used to obtain an initial model, which then serves as the initial guess Ljung (1995). However, a unified framework that identifies a dynamic and quality model simultaneously presently does not exist.

Motivated by the above, an optimization framework is formulated that simultaneously identifies a dynamic and quality model and its performance is demonstrated on data from a lab-scale Uni-Axial Rotational Molding process. The description of a typical batch process is given in Section 5.2.1 and the rotational molding process is described in Section 5.2.2. A previous approach based on subspace identification Corbett and Mhaskar (2016) is briefly reviewed in Section 5.2.3. Prediction error minimization framework is reviewed in Section 5.2.4. The proposed prediction error minimization-based optimization framework is presented in Section 5.3. Finally the performance of the proposed method is compared to the existing approaches using experimental data obtained from Uni-axial Rotational Molding process in Section 5.4 .

5.2 Preliminaries

5.2.1 Batch Process Description

Consider a batch process characterized by the following state space description, with quality variables associated with the final state of the sequence:

$$x[k+1] = f(x[k], u[k]) \tag{5.1a}$$

$$y[k] = h(x[k], u[k]) \tag{5.1b}$$

$$Q = g(x_f) \tag{5.1c}$$

where $x[k] \in \mathbb{R}^{n_x \times 1}$ denotes the states of the system at the k^{th} sampling instance, with $t[k] \equiv k\Delta$, where Δ is the sampling interval, $x_f \equiv x[k_f^b]$ being the state vector at the final sampling step of the b^th batch of the process, k_f^b (allowing for the fact that batches could be of different lengths), $y[k]$ corresponds to the measured output variables, $u[k]$ corresponds to the measured input variables, and finally $Q \in \mathbb{R}^{n_q \times 1}$, being the terminal quality variables, measured only at k_f^b. n_x and n_q denote the number of states of the system, and the number of quality variables respectively.

The above notation captures a typical batch process, where certain variables are measured at every time step during a batch run (denoted as u and y), while other variables, referred to as quality variables (Q), are only measured once the batch run is completed. This is because these quality variables often require separate lab analysis of the final product formed during the batch and thus cannot be measured in real time. In the context of state space description (Equations (5.1), since the 'states' at any given time contain the information about the input-output dynamics until that

time, it is reasonable to relate the quality variables, which are measured only at the end of a batch, to the final states of the process.

Remark 37. *It is worth noting that, due to the batch nature of the data, most of the noise present is negligible compared to process transients. In a batch process, the system undergoes significant transitions between the start and end points, which are markedly different. In contrast, in a continuous mode, the system largely operates around a steady state. Consequently, in the presence of nominal noise, the signal-to-noise ratio (SNR) is lower in continuous mode due to the system's operation, whereas in batch processes, the noise becomes negligible relative to the process transients.*

The identification problem for batch processes boils down to identifying the functions f, h and g in Equations (5.1). However, for this work, linear models will be considered for the dynamics and also the quality variables, which are given by the following :

$$x[k+1] = Ax[k] + Bu[k] \tag{5.2a}$$

$$y[k] = Cx[k] + Du[k] \tag{5.2b}$$

$$Q = P + Rx_f \tag{5.2c}$$

where $\mathbf{A} \in \mathbb{R}^{n_x \times n_x}$, $\mathbf{B} \in \mathbb{R}^{n_x \times n_u}$, $\mathbf{C} \in \mathbb{R}^{n_y \times n_x}$, $\mathbf{D} \in \mathbb{R}^{n_y \times n_u}$ are the associated system matrices and $\mathbf{x}_k$ denotes the state vector. n_x, n_y, n_u represent the number of states, number of outputs and number of inputs respectively. Q denotes quality measurements at the termination of batch at time t_f, P and R, are the linear quality model parameters, x_f are final states as predicted by the state-space model. The model matrices along with the states will be identified as part of the identification procedure.

Remark 38. *It is crucial to evaluate the proposed approach against potential alter-*

natives. As highlighted in the introduction, one could directly fit a linear model to connect the quality variables with the entire input-output trajectory. However, this method necessitates aligning the batches beforehand, as their lengths might vary. In contrast, state-space representations for input-output dynamics eliminate the need for batches to be of the same length and offer a compact model where the 'states' encapsulate information about the process dynamics. Therefore, linking the final 'states' of a batch sequence to the product quality via a linear model is a logical and efficient method to leverage the dynamic relationship between the input and the output and associate it with the quality variables.

5.2.2 Motivating Example: Rotational Molding Process

A lab-scale uni-axial rotational molding machine is used to produce plastic molded products. The three inputs, u in Equation (5.2) (two heater powers and one compressed air supply), and the sole output, y in (5.2) (Internal Mold temperature), of the rotational molding setup are monitored and manipulated through LabVIEW and MATLAB programs. A powder of high-density polyethylene is charged into the mold at room temperature while the oven is pre-heated to 300°C. The oven is then placed over the mold which is now set to rotate at a constant rotational speed. Once the heating phase is complete, a fan is used to air-cool the mold containing the product before the product is removed for quality testing where the sinkhole area, $Q(1)$ in Equation (5.2), and the impact strength, $Q(2)$ in Equation (5.2), of the product are assessed.

The degree of sintering in rotational molding is measured via surface voids. When the polymer particles does not undergo complete sintering during the process, the resultant product tends to show considerable amount of surface voids. ImageJ is used to perform the surface void analysis on the images of the mold taken by a digital

camera. Another quality that needs to be quantified is the strength of the product and one way to measure this is via dart impact testing. The impact energy calculation is based on mean failure height which is defined as the height from which 50% of the samples fail. This height is then multiplied by the mass of the dart to get the impact energy in units of kg-m. Specific details relating to these quality variables in the context of this work can be found in our previous work Chandrasekar et al. (2022b,a).

35 batches of rotomolding experiments have been conducted, each lasting approximately 15-18 minutes. Data, including input and output information, along with control actions, is recorded every 10 seconds during these experiments. Initially, a Proportional-Integral (PI) strategy aimed at maintaining the heater temperatures at 300°C was employed to carry out the initial batch of experiments and gather sufficient data. Subsequently, the remaining batches were executed using the control methodology described in our previous work Chandrasekar et al. (2022b,a). The model's performance was assessed by partitioning the dataset into a 70:30 ratio for training and testing/validation purposes, respectively.

5.2.3 Subspace Identification and Quality Modelling for Batch Processes

The proposed modelling approach builds an LTI state space model of the structure shown in Equations (5.2a) - (5.2b).

In this manuscript, the LTI SS model between the inputs and the measured process outputs is identified using a deterministic subspace identification routine Moonen et al. (1989) adapted for batch processes Corbett and Mhaskar (2016). The order n_x of this state space model is determined by evaluating the fit performance on the testing data.

Past work utilized matrix algebra concepts like decomposition and projection techniques to develop variations of the subspace algorithm Larimore (1990); Van Overschee and De Moor (1995); Verhagen and Dewilde (1992). However, these developments were tailored for continuous processes Wang and Qin (2002); Pour et al. (2010); Qin et al. (2005). Batch processes differ fundamentally from continuous processes, as the latter often operate around a 'process mean,' a concept not applicable to batch operations that traverse a wide range of conditions. Additionally, batch data comprises multiple experiments that are not contiguous to each other, unlike continuous process data. Consequently, the existing subspace algorithm needed modification to accommodate such data forms. The batch nature of the data while using the subspace algorithm is accounted for Mhaskar et al. (2019) by initially forming a pseudo-Hankel matrix of inputs and outputs for individual batches and then concatenating them horizontally to obtain the cumulative pseudo-Hankel matrix for all batch data. The equations for forming these matrices for the inputs are provided by Equations (5.3) and (5.4) below:

$$\mathbf{U}_{1|i}^{(b)} = \begin{bmatrix} \mathbf{u}^{(b)}[1] & \mathbf{u}^{(b)}[2] & \cdots & \mathbf{u}^{(b)}\left[j^{(b)}\right] \\ \vdots & \vdots & & \vdots \\ \mathbf{u}^{(b)}[i] & \mathbf{u}^{(b)}[i+1] & \cdots & \mathbf{u}^{(b)}\left[i+j^{(b)}-1\right] \end{bmatrix} \tag{5.3}$$

$$\mathbf{U}_{1|i} = \begin{bmatrix} \mathbf{U}_{1|i}^{(1)} & \mathbf{U}_{1|i}^{(2)} & \cdots & \mathbf{U}_{1|i}^{(nb)} \end{bmatrix} \tag{5.4}$$

where U is the Input-Hankel matrix, $u[k]$, is the input at time step k, (b) denotes the particular batch, nb is the number of batches used for training, i is the number of Hankel rows, $j^{(b)}$ is the number of Hankel columns which changes for each batch since each batch can be of different length.

Similarly, the equations mentioned above are utilized to construct the output pseudo-Hankel matrices. The deterministic subspace technique Moonen et al. (1989) is applied using these input and output Hankel matrices to derive the state sequence and the model matrices.

Due to the batch nature of the data and reiterating a point made in Section 5.2.2, there is negligible effect of noise on the process data. If we recall the deterministic-stochastic algorithm Qin (2006), the term containing the correlation of the noise and the inputs can be ignored due to the argument above, reducing the problem to a deterministic identification algorithm. Although the results are not presented in the manuscript, both algorithms were tested and produced very similar results.

Once the LTI SS model is identified using the subspace approach mentioned above, this LTI SS model is used to generate the terminal states of each batch. These terminal/final states are then linked with the quality measurements using Partial Least Squares (PLS) regression. Such a modelling strategy involving two models is necessary because the state space model alone cannot predict the quality variables since measurements for these variables are only available after the end of a batch and not at every time step. The PLS model for the quality variables is as shown in Equation (5.2c).

In summary, the dynamic model predicts the output behaviour for a given input sequence, and the PLS model, utilizing the terminal predicted states, predicts the product qualities. The structure of the modelling approach is illustrated in Figure 5.1, which essentially employs two models to capture both the dynamics of the process variable and the quality variables.

5.2.4 Prediction Error Minimization methods for State Space Models

Prediction error minimization (PEM) methods are an intuitive way to formulate an optimization problem to fit a model of a specific structure to the data at hand. They form the backbones of all the estimating algorithms in the literature so far. One of the key components of a PEM is an objective function, which is the sum of squared errors between the predicted outputs from the model and the actual output values. The model between the inputs and the outputs appears as the constraints of the optimization problem. PEM methods offer flexibility while identifying a model; depending on the requirements, one could impose constraints on the structure of the model either based on intuition or first principles knowledge. For identifying linear state space models, especially for closed-loop dynamic process data, PEM methods also sometimes show better performance over other traditional methods like subspace identification. This is because subspace identification cannot handle the bias that exists as a result of a feedback mechanism Ljung and McKelvey (1996) when the amount of data we have is finite. Of course, in batch processes, some of these concerns are no longer relevant since even when in closed-loop, batch process control often entails driving a variable through a set point trajectory (instead of holding it constant).

It must be noted that PEM formulation for identifying LTI SS models results in an NLP problem that is non-convex in nature and might have multiple local optima. These problems are solved using iterative methods which require an initial guess. The initial guess has a direct impact on the final solution given by the optimizer and hence a good initial guess is desired. Nevertheless, in the presence of a reasonable initial guess, PEM can be used as a way to fine-tune the model further. The subspace identification algorithm is used for generating the initial model parameter estimates for this purpose Ljung (1995).

The following formulation is adopted for identifying and/or fine-tuning an LTI SS model:

$$\min_{A,B,C,D,x_0(b)} \sum_{b=1}^{n_b} \sum_{k=1}^{k_f^b} (y_{b,k} - \hat{y_{b,k}})^T (y_{b,k} - \hat{y_{b,k}}) \tag{5.5a}$$

$$s.t. \;\; \hat{x}_{b,0} = x_0(b) \;\; \forall \; b = 1, 2, ..n_b \tag{5.5b}$$

$$\hat{x}_{b,k+1} = A\hat{x}_{b,k} + Bu_{b,k} \;\; \forall \; k = 1, 2, ..n_t(b) \;\; and \;\; \forall \; b = 1, 2, ..n_b \tag{5.5c}$$

$$\hat{y}_{b,k} = C\hat{x}_{b,k} + Du_{b,k} \;\; \forall \; k = 1, 2, ..n_t(b) \;\; and \;\; \forall \; b = 1, 2, ..n_b, \tag{5.5d}$$

where $y_{b,k}$ and $u_{b,k}$ denote the inputs and the outputs respectively at time instance 'k' for a batch b. A, B, C, D, the model matrices, along with $x_0(b)$ which are the initial states for each batch sequence, form the solution to the formulated optimization problem. k_f^b denotes the final time step of a batch b and n_b denotes the number of batches in the database.

The objective function captures the predicted output errors across all the time steps in each batch, across all batches. Equation (5.5b) initializes the state for each batch, which is also to be estimated as part of the optimization problem. Equation (5.5c) and (5.5d) represent the LTI SS model constraints on the states and the outputs at each time step 'k' for each batch 'b'.

5.3 Proposed Identification Framework

In this section, we present the proposed optimization-based modelling framework. The overall structure of the modelling approach is shown in Figure 5.1.

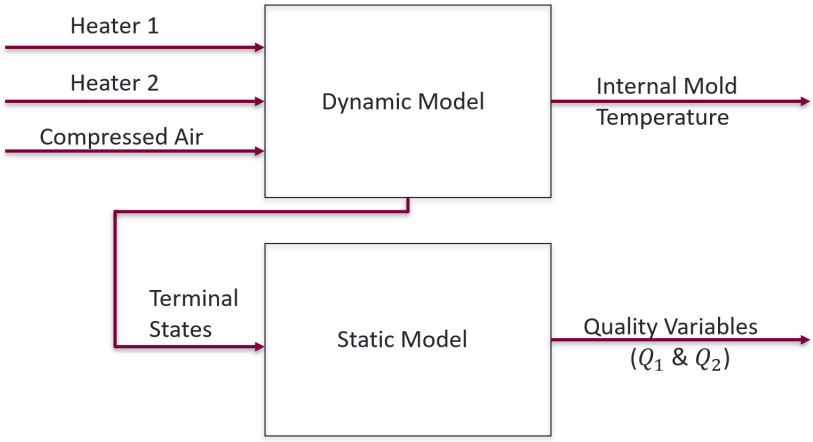

Figure 5.1: Model structure

The dynamic model relates the frequently measured process variables (inputs and outputs) of the process through a linear time-invariant state space model as shown in Equations (5.6).

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k, \tag{5.6a}$$

$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{D}\mathbf{u}_k, \tag{5.6b}$$

where $\mathbf{A} \in \mathbb{R}^{n_x \times n_x}$, $\mathbf{B} \in \mathbb{R}^{n_x \times n_u}$, $\mathbf{C} \in \mathbb{R}^{n_y \times n_x}$, $\mathbf{D} \in \mathbb{R}^{n_y \times n_u}$ are the associated system matrices and $\mathbf{x}_k$ denotes the state vector. n_x, n_y, n_u represent the number of states, number of outputs and number of inputs respectively.

The final states for each batch sequence in the database are related to the quality measurements of the particular batch through a linear static model as shown in Equation (5.7).

$$\mathbf{Q} = \mathbf{P} + \mathbf{R}\mathbf{x}_{n_t} \tag{5.7}$$

where Q denotes quality measurements at the termination of batch at time n_t, P and R are the linear quality model parameters, $\hat{x}_{n_t}$ are final states as predicted by the state-space model.

The following is the optimization framework:

$$\min_{A,B,C,D,P,R,x_0(b)} \sum_{b=1}^{n_b} \left[\lambda_1 \cdot (Q_b - \hat{Q_b})^T (Q_b - \hat{Q_b}) + \lambda_2 \cdot \sum_{k=1}^{n_t(b)} (y_{b,k} - \hat{y_{b,k}})^T (y_{b,k} - \hat{y_{b,k}})\right] \tag{5.8a}$$

$$s.t. \;\; \hat{x}_{b,0} = x_0(b) \;\; \forall \; b = 1, 2, ..n_b \tag{5.8b}$$

$$\hat{x}_{b,k+1} = A\hat{x}_{b,k} + Bu_{b,k} \;\; \forall \; k = 1, 2, ..n_t(b) \;\; and \;\; \forall \; b = 1, 2, ..n_b \tag{5.8c}$$

$$\hat{y}_{b,k} = C\hat{x}_{b,k} + Du_{b,k} \;\; \forall \; k = 1, 2, ..n_t(b) \;\; and \;\; \forall \; b = 1, 2, ..n_b \tag{5.8d}$$

$$\hat{Q_b} = P + R\hat{x}_{b,n_t(b)} \;\; \forall \; b = 1, 2, ..n_b, \tag{5.8e}$$

where Q_b denotes the quality measurements for a batch b, and $y_{b,k}$ and $u_{b,k}$ denote the inputs and the outputs respectively at time instance 'k' for a batch b. A, B, C, D, P, R, along with $x_0(b)$ which are the initial states for each batch sequence, form the solution to the formulated optimization problem. $n_t(b)$ denotes the final time step of a batch b.

The above is a Non-Linear Programming (NLP) problem, and hence it requires a good initialization of the model parameters. Using the subspace-based quality modelling approach for batch processes as described in Section 5.2.3 we first get a primary solution for the models. Then we use this solution as an initial guess to start the optimization problem to further refine both models.

Remark 39. *It is recognized that the previous model identification approach described in Section 5.2.3 is sub-optimal since the dynamic model is identified separately without taking into account the quality measurements as stated in Section 5.1. However, it must be noted that this is one such approach that allows the identification of the models in the specific structure (Figure 5.1) that has proven to be successful under closed-loop control in the past Garg and Mhaskar (2017); Chandrasekar et al. (2022b) demonstrating the model's adequate performance. At the time of writing this manuscript, there exists no other method to simultaneously identify dynamic and quality models, either separately or together. Instead of randomly initializing the model parameters, the previous approach was chosen to initialize the model.*

The optimizer minimizes the objective function, i.e., the sum of squared errors of the quality variable predictions across all batches and also of the output variable predictions across all batches and for all time instances in each batch. λ_1 and λ_2 values can be set accordingly to compensate for the difference in scales of the absolute values of the quality variables and the output variables. Equation (5.8b) initializes the first state for each batch sequence, Equations (5.8c) and (5.8d) represent the LTI state space model equations and finally Equation (5.8e) represents the linear quality model, relating the quality variables to the final states of a batch sequence.

Remark 40. *In this work, the focus was to develop a general modelling framework that allows for estimating both the dynamic and the quality models. One can potentially improve the modelling by replacing one or more equations from Equations* (5.8c) - (5.8e) *with its non-linear version. The focus of the present implementation was to show the improved modelling accuracy with a combined approach over a decoupled approach (Section 5.2.3) for identifying the two models. A direction of future work could focus on implementing the non-linear version of this model, possibly using Neural Networks.*

Remark 41. *Existing Neural Network structures may not be able to handle the re-*

lationship between the inputs and both the outputs and the quality variables. At best, Recurrent Neural Networks (RNNs) can identify the dynamic relationship between the inputs and the outputs. A future direction for this work could focus on using this approach to adapt the RNNs to incorporate quality variables along with inputs and outputs.

Remark 42. *Note that any physics-based dynamic information related to the process variables and/or any information relating the quality variables with the process variables can be readily incorporated in this framework as additional constraints to the optimizer or as an additional loss term in the objective function (see Patel et al. (2020) for one such implementation in identifying dynamic models). This is the same idea used in what are commonly known as Physics Informed Neural Networks (PINNs).*

5.3.1 State Observer for Validating LTI SS Models

Before presenting the results, it is important to note that an observer/state estimator is required for testing the unified model. This necessity arises because the initial states for the testing data are unknown. Consequently, a Luenberger observer (Equation (5.9)) is employed in this work. It is utilized for the first few initial time instances to update the states using the input and output measurements during these instances. The state estimates are obtained using the Luenberger observer as follows:

$$\hat{\mathbf{x}}[k+1] = \mathbf{A}\hat{\mathbf{x}}[k] + \mathbf{B}\mathbf{u}[k] + \mathbf{L}(\mathbf{y}[k] - \hat{\mathbf{y}}[k]), \tag{5.9a}$$

$$\hat{\mathbf{y}}[k]) = \mathbf{C}\hat{\mathbf{x}}[k] + \mathbf{D}\mathbf{u}[k], \tag{5.9b}$$

where A, B, C and D are the LTI SS model matrices. The observer, L is chosen in

such a way that the error in the states decays to zero as time progresses. In particular, the poles of the matrix, $A - LC$ are distinctly chosen to be within the unit circle to ensure the desired convergence. The convergence of the output predictions implies that the states have been estimated, and at this point, the model is ready to start predicting the rest of the trajectory using the most recent state estimate on its own without an observer.

Remark 43. *For this work, a Luenberger observer was used but a Kalman filter may also be used. However, as noted in Section 5.2.3, due to the batch nature of the process, the noise is negligible compared to the process transients. Hence a deterministic approach is opted for, and a Luenberger observer was chosen and also found to perform sufficiently well.*

5.4 Model evaluation on the motivation example

We present the open-loop modelling results in this section. *fmincon* with the interior-point algorithm in MATLAB was used for solving the optimization problem in the proposed approach. The performance of the model obtained through the proposed approach is evaluated on batch data from the rotational molding setup described in Section 5.2.2 and compared with an existing modelling approach described in Section 5.2.3 for identifying the same model structure. Specifically, the performance is evaluated on the accuracy in predicting the output trajectories, and the final product qualities of the batches in the testing set. For both approaches, first, the LTI SS model is used to generate the state trajectory along with the predicted output trajectory. Then the linear quality model is used on the final states from the previously obtained state trajectory to obtain the predicted quality variables.

A system order of $n_x = 2$ was chosen for the LTI SS model in both the approaches after achieving a reasonable prediction performance on the outputs and the quali-

ties. For the proposed approach, the values of λ_1 and λ_2 in Equations (5.8) were set to 100 and 5 respectively. A grid search was performed to select the best values of λ's that resulted in the least combined errors in the outputs and the qualities. Specifically, λ_1 was explored in the grid $0.5, 1, 10, 100$ and λ_2 was explored in the grid $0.002, 0.005, 0.01, 0.1, 0.5, 1, 5, 10$.

We compare the prediction performance of the model obtained from two approaches, as described in Sections 5.2.3 and 5.3. As mentioned in Section 5.3.1, first a Luenberger observer is used along with the LTI SS model for the states to converge. In practice, one can keep using the observer till the end of sequence, to keep correcting for the states using the measurements. However, the true test of a dynamic model is in its ability to predict the outputs and the quality variables accurately on its own without an observer for as long as possible and in the context of batch processes, till the end of the batch sequence. For this purpose, a few time instances within the length of the batch data are chosen until which the observer would run. First, the comparison of the performance of the two models on a sample batch is shown in Figure 5.2. In the figure and the tables below, "Approach 1" refers to the approach described in Section 5.2.3 and "Approach 2" refers to the proposed approach described in Section 5.3. Here, the time until which the observer is run, the observer time is chosen to be 40. We can see that the proposed approach not only predicts the output variable better than the old approach but also does the same for the quality variables.

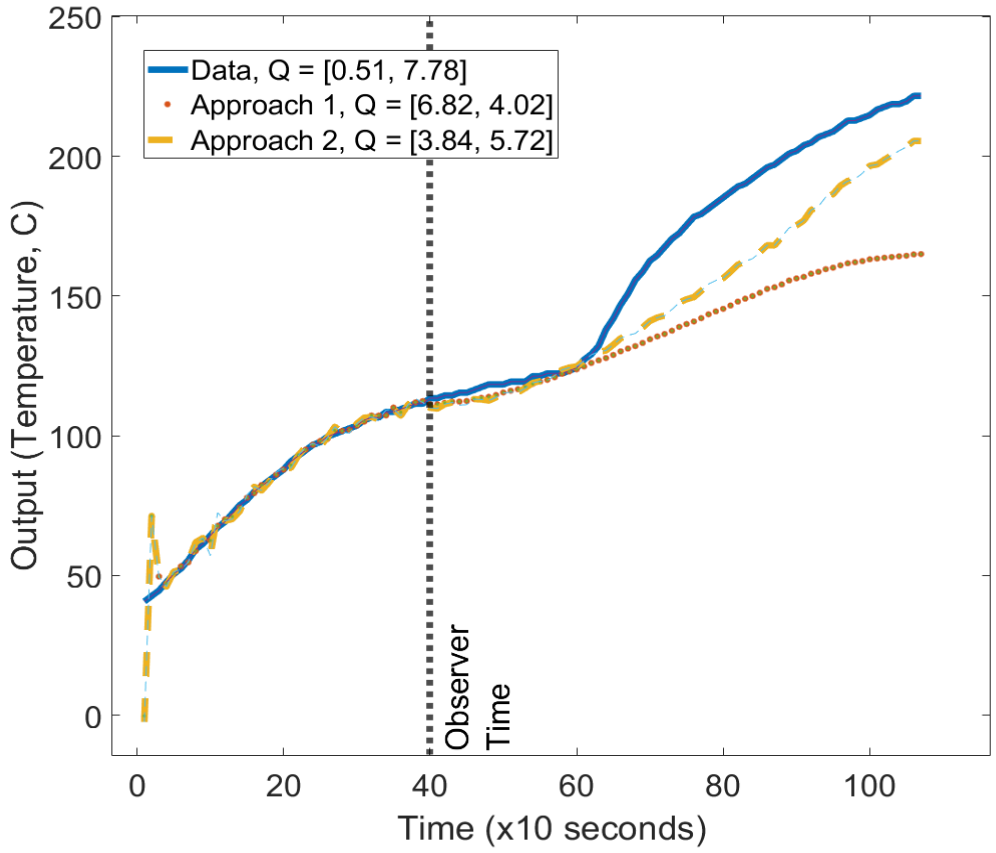

Figure 5.2: Demonstration of the proposed modelling to capture the dynamics of the process output and the quality variables on an illustrative batch. The two measured quality variables for this batch (Blue line) are 0.51, 7.78. Quality variables as predicted by model obtained through 'Approach 1' (Dotted red line) are 6.82, 4.02. The same as predicted by the model obtained through 'Approach 2' (Dashed yellow line) are 3.84,5.72.

Next, we tabulate the average Root Mean Squared Errors (RMSE) obtained for the output variable across all batches and over all time instances within each batch and the RMSE for the quality variables across all batches. Three observer times are chosen, $k = 10$, $k = 40$ and $k = 60$ and the model performance on the outputs (as shown in Table 5.1) and the qualities (as shown in Table 5.2) are compared for these three cases, for both training and testing data.

Table 5.1: Summary of the Output variable prediction performance

RMSE	Approach 1		Approach 2	
	Train	Test	Train	Test
$k = 10$	25.5	25.7	14.5	16.8
$k = 40$	17.4	21.1	13.1	17.3
$k = 60$	16.1	18.2	10.9	13.7

Table 5.2: Summary of the Quality variable prediction performance

RMSE	Approach 1		Approach 2	
	Train	Test	Train	Test
$k = 10$	2.21	2.72	0.87	1.50
$k = 40$	2.05	2.39	1.04	1.32
$k = 60$	2.03	2.25	0.86	1.48

It can be observed that the proposed approach, "Approach 2" not only performs well in predicting the dynamic output variable but also mainly the quality variables on both training and testing data, for all the three observer time instances considered. Moreover, it is important to keep in mind the computational burden of using an NLP optimizer since these models are developed for subsequently implementing closed-loop control. Approach 1, took at most 1 second, whereas the proposed approach took 13 seconds. However, the objective function didn't change for the last $3-4$ seconds by a significant amount, so it could be stated that the proposed approach converged within 9 seconds.

Remark 44. *In the context of closed-loop control, the computation times mentioned above for identifying the models do not matter since the identification is done offline and the model is simply used in a Model Predictive Control (MPC) framework. However, it might become important if one were to use an adaptive modelling strategy where the model is re-identified frequently during a batch operation. Then it would be crucial for the identification procedure to be completed within the sampling interval.*

For this process, the proposed method converges within the control/sampling interval, which is 10 *seconds as mentioned in Section 5.2.2, and hence this approach can be used in an adaptive model-based control setting for this process.*

Remark 45. *Once the dynamic and quality models are identified, they can now be used within an existing economic closed-loop control framework for quality control in batch processes (see, e.g., Chandrasekar et al. (2022b)). The objective of the controller could be to meet the quality constraints set by the user while minimizing the input consumption. Improved prediction of the quality variables is expected to result in improved control, and an experimental demonstration of the proposed approach remains the subject of future work.*

Remark 46. *The proposed model identification framework (and the resultant control application) can benefit many applications, from bioreactors to batch crystallization processes where the quality variables such as detailed assays or the particle size distribution (PSD) are only available at the end of the batch. In such cases, the improved model achieved using the integrated model identification is expected to yield significant benefits. In particular, lack of observability is known to be a recurring issue in crystallization control- developing a synergistic model using the online measurements along with the terminal PSD could possibly enable PSD control of crystallization processes, and remains a direction of future work.*

5.5 Conclusions

In this manuscript, an optimization-based optimization framework to identify a data-driven dynamic and quality model for batch processes to relate the process variables and the quality variables is proposed. The intuition about the quality measurements playing an important role in deciding the process dynamics is proven right with the comparison of a previously segregated approach with the proposed approach. In the

previous approach, the input and the output variables are related using an LTI SS model obtained from subspace identification. Then the state trajectories of all the batches are generated using the LTI SS model, and the final state variable of each batch is related to the quality variables of that particular batch using a PLS model. Instead of identifying these two models separately, the proposed approach aims to optimize the model parameters together under a common optimization framework. Particularly, in the previous approach, the LTI SS model is identified using subspace identification and the linear quality model is obtained using PLS regression. In the proposed approach, a prediction error minimization-based optimization framework is developed for both the dynamic and the quality models, which results in an NLP problem. The previous approach is used to generate the initial model estimates, which are fed to the optimizer as the initial guess. Finally, the models obtained from both these approaches are compared on batch data from a lab scale Uni-axial Rotational Molding process.

Bibliography

Aumi, S., Corbett, B., Mhaskar, P., and Clarke-Pringle, T. (2013). Data-based modeling and control of Nylon-6, 6 batch polymerization. *IEEE Transactions on Control Systems Technology*, 21(1):94–106.

Aumi, S. and Mhaskar, P. (2010). Robust model predictive control & fault-handling of batch processes. *American Control Conference (ACC), 2010*, pages 4415–4420.

Bonvin, D. (1998). Optimal operation of batch reactors—a personal view. *Journal of Process Control*, 8(5-6):355–368.

Bonvin, D. and Srinivasan, B. (2013). On the role of the necessary conditions of

optimality in structuring dynamic real-time optimization schemes. *Computers & Chemical Engineering*, 51:172–180.

BONVIN, D., SRINIVASAN, B., and HUNKELER, D. (2006). Control and optimization of batch processes. *IEEE Control Systems Magazine*, 26(6):34–45.

Chandrasekar, A., Abdulhussain, H. A., Gritsichine, V., Thompson, M. R., and Mhaskar, P. (2022a). Adaptive predictive control algorithm for batch processes: Application to a rotational molding process. *Industrial & Engineering Chemistry Research*, 61(48):17572–17581.

Chandrasekar, A., Garg, A., Abdulhussain, H. A., Gritsichine, V., Thompson, M. R., and Mhaskar, P. (2022b). Design and application of data driven economic model predictive control for a rotational molding process. *Computers & Chemical Engineering*, 161:107713.

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. *Industrial & engineering chemistry research*, 39(3):693–705.

Corbett, B., Macdonald, B., and Mhaskar, P. (2015). Model Predictive Quality Control of Polymethyl Methacrylate. *IEEE Transactions on Control Systems Technology*, 23(2):687–692.

Corbett, B. and Mhaskar, P. (2016). Subspace identification for data-driven modeling and quality control of batch processes. *AIChE Journal*, 62(5):1581–1601.

Flores-Cerrillo, J. and MacGregor, J. F. (2004). Control of batch product quality by trajectory manipulation using latent variable models. *Journal of Process Control*, 14(5):539–553.

Flores-Cerrillo, J. and MacGregor, J. F. (2005). Iterative learning control for final

batch product quality using partial least squares models. *Industrial & engineering chemistry research*, 44(24):9146–9155.

Flores Cerrillo, J. and MacGregor, J. F. (2005). Latent variable MPC for trajectory tracking in batch processes. *Journal of Process Control*, 15(6):651–663.

Garg, A. and Mhaskar, P. (2017). Subspace identification and predictive control of batch particulate processes. In *2017 American Control Conference (ACC)*, pages 505–510. IEEE.

Golshan, M., MacGregor, J. F., and Mhaskar, P. (2011). Latent variable model predictive control for trajectory tracking in batch processes: Alternative modeling approaches. *Journal of Process Control*, 21(9):1345–1358.

Hosen, M. A., Hussain, M. A., and Mjalli, F. S. (2011). Control of polystyrene batch reactors using neural network based model predictive control (nnmpc): An experimental investigation. *Control Engineering Practice*, 19(5):454–467.

Kwon, J. S.-I., Nayhouse, M., Orkoulas, G., Ni, D., and Christofides, P. D. (2015). Run-to-run-based model predictive control of protein crystal shape in batch crystallization. *Industrial & Engineering Chemistry Research*, 54(16):4293–4302.

Larimore, W. (1990). Canonical variate analysis in identification, filtering, and adaptive control. In *29th IEEE Conference on Decision and Control*, pages 596–604 vol.2. IEEE.

Lee, J. H. and Lee, K. S. (2006). Iterative learning control applied to batch processes: An overview. *IFAC Proceedings Volumes*, 39(2):1037–1046.

Ljung, L. (1995). *System identification toolbox: User's guide*. Citeseer.

Ljung, L. (1998). *System Identification: Theory for the User*. Pearson Education.

Ljung, L. and McKelvey, T. (1996). Subspace identification from closed loop data. *Signal processing*, 52(2):209–215.

Mhaskar, P., Garg, A., and Corbett, B. (2019). *Modeling and Control of Batch Processes*. Advances in Industrial Control. Springer International Publishing, Cham.

Moonen, M., De Moor, B., Vandenberghe, L., and Vandewalle, J. (1989). On- And Off-Line Identification Of Linear State Space Models. *International Journal of Control*, 49(1):219–232.

Nayhouse, M., Tran, A., Kwon, J. S.-I., Crose, M., Orkoulas, G., and Christofides, P. D. (2015). Modeling and control of ibuprofen crystal growth and size distribution. *Chemical Engineering Science*, 134:414–422.

Patel, N., Nease, J., Aumi, S., Ewaschuk, C., Luo, J., and Mhaskar, P. (2020). Integrating data-driven modeling with first-principles knowledge. *Industrial & Engineering Chemistry Research*, 59(11):5103–5113.

Pour, N. D., Huang, B., and Shah, S. L. (2010). Subspace Approach to Identification of Step-Response Model from Closed-Loop Data. *Industrial & Engineering Chemistry Research*, 49(18):8558–8567.

Qin, S. J. (2006). An overview of subspace identification. *Computers and Chemical Engineering*, 30(10-12):1502–1513.

Qin, S. J., Lin, W., and Ljung, L. (2005). A novel subspace identification approach with enforced causal models. *Automatica*, 41(12):2043–2053.

Rafizadeh, M., Solgi, R., and Abbaszadeh, M. (2003). Constrained model predictive control of mma polymerization reactor based on genetic algorithm optimization. In *Proceedings of 2003 IEEE Conference on Control Applications, 2003. CCA 2003.*, volume 1, pages 464–469. IEEE.

Shi, D., El-Farra, N. H., Li, M., Mhaskar, P., and Christofides, P. D. (2006). Predictive control of particle size distribution in particulate processes. *Chemical Engineering Science*, 61(1):268–281.

Sjöberg, J. and Agarwal, M. (2002). Trajectory tracking in batch processes using neural controllers. *Engineering Applications of Artificial Intelligence*, 15(1):41–51.

Valappil, J. and Georgakis, C. (2003). Accounting for batch reactor uncertainty in the nonlinear mpc of end-use properties. *AIChE Journal*, 49(5):1178–1192.

Van Overschee, P. and De Moor, B. (1995). A unifying theorem for three subspace system identification algorithms. *Automatica*, 31(12):1853–1864.

Verhagen, M. and Dewilde, P. (1992). Subspace model identification Part 1. The output-error state-space model identification class of algorithms. *International Journal of Control*, 56(5):1187–1210.

Wan, J., Marjanovic, O., and Lennox, B. (2012). Disturbance rejection for the control of batch end-product quality using latent variable models. *Journal of Process Control*, 22(3):643–652.

Wan, J., Marjanovic, O., and Lennox, B. (2014). Uneven batch data alignment with application to the control of batch end-product quality. *ISA Transactions*, 53(2):584–590.

Wang, J. and Qin, S. J. (2002). A new subspace identification approach based on principal component analysis. *Journal of Process Control*, 12:841–855.

Yu, H. and Flores-Cerrillo, J. (2013). Latent Variable Model Predictive Control for Trajectory Tracking in Batch Processes: Internal Model Control Interpretation and Design Methodology. *Industrial & Engineering Chemistry Research*, 52(35):12437–12450.

Chapter 6

Quality Control in Batch Processes with High Dimensional Feedback Data

Authors: Aswin Chandrasekar, Kevork Baghdassarian, Farshad Moayedi, Hassan Abdulhussain, Vladimir Gritsichine, Michael R. Thompson, Prashant Mhaskar

The content of this chapter is sent to the Journal of Process Control and is currently under revision.

The following is the authorship contribution statement: Aswin Chandrasekar: Conceptualization, Methodology, Formal analysis, Software, Writing – original draft. Kevork Baghdassarian: Data curation. Farshad Moayedi: Data curation. Hassan A. Abdulhussain: Data curation. Vladimir Gritsichine: Data curation. Michael R. Thompson: Writing – review & editing, Supervision, Project administration, Funding acquisition. Prashant Mhaskar: Conceptualization, Methodology, Writing – review & editing, Supervision, Project administration, Funding acquisition.

Abstract

This manuscript addresses the problem of leveraging thermal images for modelling and feedback control, specifically tailored for terminal quality control of batch processes. The primary objective, common in many batch processes, is to produce products with quality variables aligning with user specifications, available for measurement only at batch termination, precluding the direct use of classical control strategies. Furthermore, in many instances, traditional online sensors such as thermocouples may not be available, but instead spectral inputs like thermal images or acoustic data may be more readily available for feedback control. The challenge is to not only use the non-traditional sensor data for building a dynamic model but also to use that model for terminal quality control. The proposed approach involves a multi-layered modelling strategy. Initially, a dimensionality reduction technique is employed to condense the high-dimensional image into a set of representative outputs. Subsequently, subspace identification (SSID) is applied to develop a Linear Time-Invariant (LTI) State Space (SS) model between the inputs and the reduced outputs. Finally, a Partial Least Squares (PLS) model is constructed linking the terminal states of a batch (identified using SSID) with the product qualities obtained for that specific batch. This model is then incorporated into a Model Predictive Control (MPC) formulation. The effectiveness of the MPC is illustrated by showcasing its capability to generate products of high quality by deploying the MPC on a bi-axial lab-scale rotational molding setup.

6.1 Introduction

Most industrial processes, irrespective of their domain, share the overarching aim of producing high-quality goods. Batch operations are preferred, notably in pharmaceutical or biochemical applications, where typically the emphasis is on meeting tighter

quality requirements. This focus on quality necessitates the establishment of an effective control regimen for the process, a crucial task in ensuring consistent product quality.

To this end, Model Predictive Controllers (MPCs) have increasingly been deployed in various industrial applications. An MPC relies on an underlying dynamic process model, enabling the controller to predict future process states and make optimal control decisions. It's important to highlight that constructing a reliable model can be challenging, especially in the pursuit of a first principles model. A more significant challenge, and opportunity, arises when incorporating non-traditional sensors such as sound or image data. In such cases, the model development and the resulting MPC framework must be adjusted to appropriately preprocess and handle high-dimensional output data before implementation in a closed-loop system.

Previous efforts have explored the effective use of high-dimensional data for feedback control, with a predominant focus on image-based modelling within the context of applications like monitoring and fault detection (Gopaluni et al., 2020). Image-based models have been developed (Duchesne et al., 2012) for monitoring, modelling, and control of various processes like lumber board processing, snack food processing, and froth flotation to name a few. In their applications, the images were processed using Multi-Resolution Multi-Variate Analysis (MR-MIA), which essentially captures the spatial and spectral features of the images. However, in these applications, the images contained the subject of interest itself, and the spatial features detected, coincidentally corresponded to the quality of interest. For example, in froth flotation (Liu and MacGregor, 2008), the key variable of interest was the size of the bubble which could be calculated using the MR-MIA technique directly on the images containing the bubbles. This might not always be the case and there are other processes where the quality of the product is determined by other non-visual parameters, for example, tensile strength.

In another effort, an image-based MPC has been developed for a level control problem (Oyama et al., 2022), where the images of the tank containing the liquid are taken and used for feedback control. Here again, the height of the liquid which is the key variable of interest that needs to be controlled, is directly seen in the images and hence, calibrated and used under closed-loop control accordingly. However, oftentimes it happens that the variables of interest are not just the physical properties that are present in the images like height or width. Moreover, in batch processes, the key variables of interest, i.e., the quality variables can only be measured and are relevant only after a batch has concluded and hence one may not be able to directly calibrate the images at every time step in a batch with a quality measurement.

Another work (Narasingam and Kwon, 2017) involved applying Dynamic Mode Decomposition (DMDc) directly to data generated using a complex system of Partial Differential Equations (PDEs), constructing a dynamic relationship between inputs and known specific output points in the spatial data. An MPC was designed to track setpoint concentrations at these specific locations, assuming the presence of process states in the data and a known mapping between states and outputs at certain locations. Similarly, (Lu and Zavala, 2021) use DMDc on thermal images in a system with multiple heating inputs spread across the spatial field, employing a desired reference thermal image as an MPC setpoint. This approach relies on the availability of such a high-dimensional thermal image setpoint. Other approaches have been adopted (Tan et al., 2024; Masti and Bemporad, 2021) where an Auto-Encoder and Recurrent Neural Network are trained together to perform the reduction and to extract the dynamic information from the process states. Additionally, research has been conducted on a comparable framework, albeit with linear models, which assumes that the process states are unknown (Yu et al., 2023). This is a more realistic scenario and generally applicable to any process data with high dimensional output feedback. However, all these methods were tested on simulation case studies and also assume a substantial volume of available dynamic data, a condition not always met in practice, partic-

ularly for batch processes where experimental data is limited due to material and energy costs, making it challenging to conduct numerous experiments.

The motivating (and representative) example for the present work is a pilot-scale Bi-axial Rotational Molding setup, a batch process used for crafting hollow plastic parts. The system features a single heater as the input, and the mold rotates bi-axially within the oven. Because of this irregular motion, standard temperature sensors like thermocouples cannot be used to monitor the process. A thermal imaging camera, situated outside the oven, is better suited to capturing temperatures inside the process as images of the mold and oven interior. The camera images are the only continuously measured output in this case — through a window slit. Notably, due to the slightly irregular rotation of the mold and the limited point of view of the camera, the mold cannot always be completely seen in collected images. There are two quality variables linked to the molded product: the sinkhole area percentage and impact strength, measurable only through destructive means after the experiment concludes. The process thus encapsulates the challenges with terminal quality control of batch processes and enables the opportunity to demonstrate the results experimentally.

To address the challenge of image-based modelling and control of batch processes, and demonstrate experimentally through the rotational molding process, a modelling strategy is devised as follows: Initially, a Convolutional neural network (CNN) - based classifier is trained on all batch images to ascertain whether the box is within the camera frame. Subsequently, for the images where the box is detected, the high-dimensional image data is condensed into a representative (lower-dimensional) set of variables. The intent of the lower dimensional set is not just to reasonably capture the dynamics of the mold temperature but more importantly, to contain the information essential for estimating the final product quality. Before this, the image is further cleaned and reduced using a pre-trained object detection model called the YOLOv3 (You Only Look Once (Redmon and Farhadi, 2018)). Subsequently, the re-

duced set of images containing the mold is used to fit a Principal Component Analysis (PCA) Model, yielding a set of latent variables. Next, an LTI SS model is constructed between the heater input and the previously computed latent variables as the output(s). Additionally, a PLS model is identified between the states of the subspace model and the quality measurements. This comprehensive model is integrated into an MPC framework designed to produce products with user-specified qualities while adhering to specific input constraints. The efficacy of this framework is demonstrated through closed-loop experiments conducted on the rotational molding setup. The rest of the manuscript is organized as follows: Section 6.2.2 describes the Bi-axial rotational molding process and a few auxiliary modelling techniques used in the proposed modelling framework, such as PCA and CNN. Section 6.3 then describes the proposed modelling approach along with the closed-loop control formulation in Section 6.3.2. The modelling performance along with the comparison of the closed-loop performance of the MPC with existing operational strategies are shown in Section 6.4.2. Concluding remarks are presented in Section 6.5.

6.2 Preliminaries

This section first describes the class of processes considered, followed by the experimental bi-axial rotational molding setup, followed by a brief review of the existing techniques used in developing the proposed modelling and control approach.

6.2.1 Batch Process Description

Consider a process described by the following state space description:

$$\mathbf{x[k+1]} = \mathbf{f(x[k], u[k])} \tag{6.1a}$$

$$\mathbf{y_{hd}[k]} = \mathbf{h(x[k], u[k])} \tag{6.1b}$$

$$\mathbf{q_f} = \mathbf{g(x_f)} \tag{6.1c}$$

where $x[k] \in \mathbb{R}^{n_x \times 1}$ denotes the states of the system at the k^{th} sampling instance, with $t[k] \equiv k\Delta$, where Δ is the sampling interval, $x_f \equiv x[k_f]$ being the state vector at the final sampling step of the batch process, k_f, $y_{hd}[k] \in \mathbb{R}^{n_{hd} \times 1}$ corresponds to the high dimensional measured output variables, and finally $q_f \in \mathbb{R}^{n_q \times 1}$, being the terminal quality variables, measured only at k_f. n_x, n_{hd} and n_q denote the number of states of the system, the number of output variables in the high dimensional space, and the number of quality variables respectively.

Typically in batch processes, there are some variables that are measured at every time step during a batch run (u, y_{hd}), and some variables that can be measured only once (q_f), after a batch run is completed. This is usually because these "quality" variables require a separate lab analysis of the final product formed during the batch and hence cannot be measured during the batch whatsoever. This is one of the key distinctions between a continuous process and a batch process. In a data-driven approach, it is a reasonable assumption that the quality variables are related to the final states of the process since the states of a system are supposed to contain the input-output dynamic information of the entire trajectory which directly affects the product quality.

Consequently, batch process data consists of many batches, with each batch containing input-output trajectories, and one set of quality variables. Moreover, each batch may be run for different duration lengths, so the length of these input-output trajectories may differ from batch to batch.

It is worth noting that, due to the batch nature of the data, most of the noise present in the data is negligible compared to process transients. A system undergoes a significant transition in a batch process where the start and end points are significantly different, compared to when a process is run in continuous mode, where the system hovers around a steady state for the most part. Consequently, in the presence of nominal noise, the signal-to-noise (SNR) ratio is lower in continuous mode due to the same reason, compared to a batch process where the noise value becomes negligible compared to the process transients.

For the set of batch processes that this work focuses on, the output variables y_{hd} are unusually high dimensional. In such cases, traditional input-output modelling techniques suffer from the 'curse of dimensionality', where the number of observations is much lower than the number of features. Moreover, current techniques do not directly build dynamic models between high dimensional outputs with the inputs nor link the quality variables also with it. The proposed approach in this manuscript is designed in a way to handle these issues and also to give a generalized procedure for modelling such batch processes for predictive control purposes. Note that the proposed approach doesn't assume/need that the functions f, h and g are known in Equation (6.1) as it identifies the relationship between the variables u, y_{hd} and q_f as part of the procedure.

6.2.2 Motivating Example: Rotational Molding Process

A pilot-scale biaxial rotational molding machine for the production of plastic molded items is utilized to demonstrate the proposed modelling and control framework. The system's sole input (u in Equation (6.1)) is the heater power, and the thermal image output (y_{hd} in Equation (6.1)) from the rotational molding setup are both monitored and controlled through LabVIEW and MATLAB programs. The thermal images are

captured through a narrow slit by a FLIR E5 infrared camera that is positioned outside the oven. The infrared camera has a thermal sensitivity of 0.1°C and a resolution of 90 *times*120 pixels. It is important to note that though the output at each time step is a 2-D array of 90 *times*120 variables, this will be flattened to a 1-D array of $90 \cdot 120 \times 1$, in which case, $n_h d = 10800$ from Equation (6.1).

High-density polyethylene powder (ExxonMobile HD 8660.29) is introduced into the mold at room temperature, while the oven is pre-heated, using a PI controller, to 300°C. Subsequently, the mold is positioned inside the oven, before it is set to rotate. Following the completion of the heating phase, the mold is moved to a cooling chamber where the mold containing the polymer is cooled with compressed air. The product is then removed from the mold and taken for quality testing, to assess the sinkhole area percentage ($q_f(1)$) and the impact strength ($q_f(2)$) as defined in Equation (6.1)) It must be noted that the mold containing the polymer doesn't consistently fall within the camera's field of view due to the continuous rotation of the mold, while the camera maintains a fixed position outside the oven. Sample images from the process are shown in Figure 6.1.

Figure 6.1: Sample images from a Rotomolding batch. The leftmost image shows an image with the mold in the frame. As the mold always undergoes constant rotation, the mold doesn't always sit in the frame of the image, as shown in the middle and the rightmost images.

The assessment of sintering in rotational molding relies on the examination of surface voids. When polymer particles do not undergo complete sintering during the process,

the resulting product often exhibits a significant number of surface voids. Copper grease is added to the samples to showcase the voids and ImageJ is employed to conduct surface void analysis on images of the mold, captured by a digital camera. Additionally, the strength of the product needs quantification, and one method for this is pendulum impact testing. Samples are cut into 63.5 x 12.7 mm strips (2.5 x 0.5 in) and impact testing is done on an HIT25P pendulum impact Tester with a 5 J hammer from Zwick/Roel to get the impact energy of the samples according to ASTM D256.

The key objective in the process is to determine (and adjust under closed-loop control as appropriate) the heating provided to the mold to achieve the desired product quality at batch termination.

6.2.3 Dimensionality Reduction Techniques (Principal Component Analysis)

PCA is a well-established technique (Wold et al., 1987; Abdi and Williams, 2010) that has been used in many engineering and other applications over the years (Dunia et al., 1996; Cho et al., 2005; Sanguansat, 2012; Paul and Al Sumam, 2012). These methods aid in assessing the redundancy within a high-dimensional dataset. Specifically, PCA is a linear reduction technique that transforms the data into a different coordinate system through scaling and rotations. This transformation is designed to capture the maximum variation in the data through fewer dimensions. The following equation depicts the relationship between various matrices encountered in a PCA model:

$$T = X * P \tag{6.2}$$

where T is the score matrix, which represents the latent variables, X is the original

data matrix and P is the transformation matrix which transforms the data to the score.

6.2.4 Convolutional Neural Networks

Convolutional Neural Network (CNN) is a type of artificial neural network designed for processing and analyzing visual data, such as images and videos. It was first designed to recognize hand-written numerical digits (Le Cun et al., 1989). Since then and with significant improvements in the computational capacities over the years, CNNs have proven to be highly effective and have been extensively used in various computer vision tasks, including image classification, object detection, and image recognition (Hijazi et al., 2015; Rawat and Wang, 2017; Zhiqiang and Jun, 2017). Some key components of CNNs include:

- Convolutional Layers: CNNs use convolutional layers to automatically and adaptively learn spatial hierarchies of features from the input data. Convolution involves applying filters (also known as kernels) to the input image to detect patterns or features. These filters slide over the input image to create feature maps.

- Pooling Layers: Pooling layers are used to downsample the spatial dimensions of the input data. Common pooling operations include max pooling and average pooling, which help reduce the computational complexity and the number of parameters in the network while retaining important information.

- Activation Functions: Activation functions introduce non-linearity into the network, allowing it to learn complex relationships in the data. Common activation functions include Rectified Linear Unit (ReLU), Sigmoid, and Hyperbolic Tangent (Tanh).

- Fully Connected Layers: After extracting features through convolution and pooling layers, the network often ends with one or more fully connected layers. These layers process the high-level features and make predictions based on them.

CNNs have played a crucial role in the advancement of computer vision applications, contributing to breakthroughs in image recognition tasks, object detection, and even tasks like image generation.

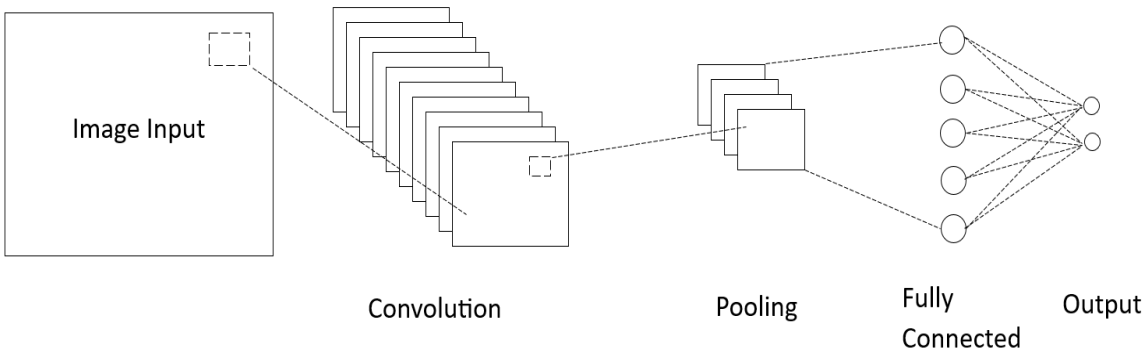

Figure 6.2: Convolutional Neural Network Architecture

6.3 Image Based Modeling and Quality Control of Batch Processes

In this section, the proposed modeling approach along with all the necessary techniques required to make the proposed approach applicable to the current Rotational Molding setup are presented.

6.3.1 Modelling approach

The overall modelling approach is discussed in this section. As mentioned in Section 6.2.1, a batch process has a set of input variables (u), output variables (y_{hd}) and

quality variables (q_f) (Refer to Equation (6.1)). In particular, this approach focuses on cases with high dimensional output y_hd. In preparation for use within a model predictive control strategy, the main objective is to enable predicting q_f from an arbitrary sampling time k_p given $u[k], k = 1, \ldots, k_f$ and $y_{hd}[k], k = 1, \ldots k_p$. Consider the eventual closed-loop control of the process at a time k_p. The model should be able to use the inputs implemented upto that point in time, the measured outputs upto that point in time, and a set of future candidate inputs to predict the final quality. Such a model can be used by a Model Predictive Controller to in turn determine the appropriate control action.

This is achieved by first determining latent variables corresponding to the high dimensional output (Section 6.3.1.1). Next, an LTI SS model is obtained by performing subspace identification tailored for batch data on the representative outputs, y and the inputs, u as described in Section 6.3.1.2. Finally, a PLS-based quality model is built between the product qualities Q_f and the final states x_f calculated using the previously obtained LTI SS model. The proposed approach is shown in Figure 6.3.

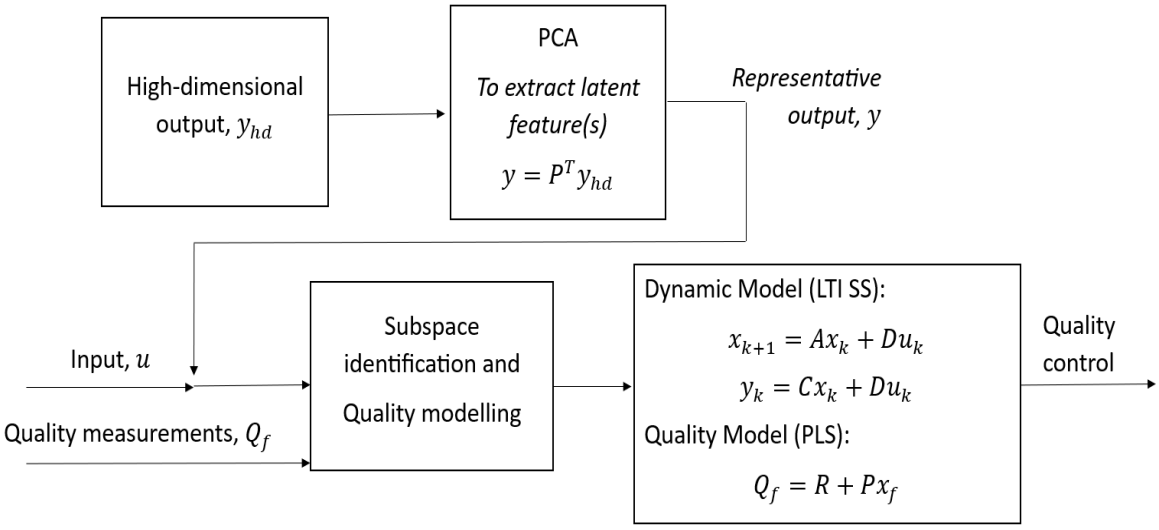

Figure 6.3: Proposed layered modelling strategy

6.3.1.1 Determining Latent Variables

For images, with the high dimensional output originally in the form of a 2-D ($l \times b$) vector or a matrix at each sampling time, all the images are first flattened out into 1-D vectors ($lb \times 1$) and then undergo dimensionality reduction. In the present manuscript, we use PCA as follows.

$$y[k] = P^T \cdot y_{hd}[k] \tag{6.3}$$

where $y[k] \in \mathbb{R}^{n_{lv} \times 1}$ denote the latent variables corresponding to the $y_{hd}[k] \in \mathbb{R}^{n_{hd} \times 1}$ denoting the set of high dimensional output variables and $P \in \mathbb{R}^{n_{hd} \times n_{lv}}$ is the loading matrix obtained from PCA (see Equation (6.2)). n_{lv} is the no. of latent variables. For the rest of the manuscript, these latent variables are referred to as the 'measured' outputs y of the batch process.

6.3.1.2 Subspace Identification for Batch Processes

The proposed modelling approach builds an LTI state space model of the following structure:

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k, \tag{6.4a}$$

$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{D}\mathbf{u}_k, \tag{6.4b}$$

where $\mathbf{A} \in \mathbb{R}^{n \times n}$, $\mathbf{B} \in \mathbb{R}^{n \times m}$, $\mathbf{C} \in \mathbb{R}^{l \times n}$, $\mathbf{D} \in \mathbb{R}^{l \times m}$ are the associated system matrices and $\mathbf{x}_k$ denotes the state vector.

In this manuscript, the LTI SS model between the inputs and the measured process outputs (the latent variables) is identified using a deterministic subspace identification routine (Moonen et al., 1989) adapted for batch processes (Corbett and Mhaskar, 2016). The order n of this state space model is determined by evaluating the fit performance on the testing data.

Past work utilized matrix algebra concepts like decomposition and projection techniques to develop variations of the subspace algorithm (Larimore, 1990; Van Overschee and De Moor, 1995; Verhagen and Dewilde, 1992). However, these developments were tailored for continuous processes (Wang and Qin, 2002; Pour et al., 2010; Qin et al., 2005). Batch processes differ fundamentally from continuous processes, as the latter often operate around a 'process mean,' a concept not applicable to batch operations that traverse a wide range of conditions. Additionally, batch data comprises multiple experiments that are not contiguous to each other, unlike continuous process data. Consequently, the existing subspace algorithm needs modification to accommodate such data forms. The batch nature of the data while using the subspace algorithm is accounted for (Mhaskar et al., 2019) by initially forming a pseudo-Hankel matrix of inputs and outputs for individual batches and then concatenating them horizontally to obtain the cumulative pseudo-Hankel matrix for all batch data. The equations for forming these matrices for the inputs are provided by Equations 6.5 and 6.6 below:

$$\mathbf{U}_{1|i}^{(b)} = \begin{bmatrix} \mathbf{u}^{(b)}[1] & \mathbf{u}^{(b)}[2] & \cdots & \mathbf{u}^{(b)}\left[j^{(b)}\right] \\ \vdots & \vdots & & \vdots \\ \mathbf{u}^{(b)}[i] & \mathbf{u}^{(b)}[i+1] & \cdots & \mathbf{u}^{(b)}\left[i+j^{(b)}-1\right] \end{bmatrix} \tag{6.5}$$

$$\mathbf{U}_{1|i} = \begin{bmatrix} \mathbf{U}_{1|i}^{(1)} & \mathbf{U}_{1|i}^{(2)} & \cdots & \mathbf{U}_{1|i}^{(nb)} \end{bmatrix} \tag{6.6}$$

, where U is the Input-Hankel matrix, $u[k]$, is the input at time step k, (b) denotes the particular batch, nb is the number of batches used for training, i is the number of Hankel rows, $j^{(b)}$ is the number of Hankel columns which changes for each batch since each batch can be of different length.

Similarly, the equations mentioned above are utilized to construct the output pseudo-Hankel matrices. The deterministic subspace technique (Moonen et al., 1989) is applied using these input and output Hankel matrices to derive the state sequence and the model matrices. A system order of 2 was found to be adequate for the LTI SS model after using the identified latent variable trajectory as the output and more importantly, after achieving a reasonable prediction performance with the quality model as well.

Due to the batch nature of the data and reiterating a point made in Section 6.2.1, there is negligible effect of noise on the process data. If we recall the deterministic-stochastic algorithm (Qin, 2006), the term containing the correlation of the noise and the inputs can be ignored due to the argument above, reducing the problem to a deterministic identification algorithm. Although the results are not presented in the manuscript, both algorithms were tested and produced very similar results.

6.3.1.3 Terminal Quality Model

Once the LTI SS model is identified using the subspace approach mentioned above, this LTI SS model is used to generate the terminal states of each batch. These terminal/final states are then linked with the quality measurements using Partial Least Squares (PLS) regression. Such a modelling strategy involving two models is necessary because the state space model alone cannot predict the quality variables since measurements for these variables are only available after the end of a batch and not at every time step. The PLS model for the quality variables is as follows:

$$Q_f = \hat{R}_m + \hat{P}_m x_f \tag{6.7}$$

, where Q_f denotes quality measurements at the termination of batch at time t_f, $\hat{P}_m$ and $\hat{R}_m$, are the linear quality model parameters, x_f are final states as predicted by the state-space model.

In summary, the dynamic model predicts the latent variable behaviour for a given input sequence, and the PLS model, utilizing the terminal predicted states, predicts the product qualities. The structure of the modelling approach is illustrated in the bottom half of Figure 6.3, which essentially employs two models to capture both the dynamics of the process variable (in this case, the latent variable obtained from filtering the image through various techniques as explained in the previous sections) and the quality variables.

Remark 47. *It is important to compare the proposed approach with possible alternatives. As discussed in the introduction section, a PLS model can be attempted to directly be fitted to relate the quality variables to the input-output trajectories. However, in this case, batches need to be aligned before fitting a model since all batches might not be of the same length. Subspace techniques adapted for batch processes do not require batches to be of the same length. They provide a compact model using the notion of state space representation, with the 'states' storing information about the process dynamics. Hence, it is intuitive to connect the final 'states' of a batch sequence to the product quality using a PLS model and is an efficient way to utilize the dynamic relation between the input and the output and relate it to the quality variables.*

Remark 48. *There are some simulation examples (Narasingam and Kwon, 2017; Lu and Zavala, 2021) where Dynamic Mode decomposition (DMDc) is applied on high-fidelity simulation data consisting of partial differential equations (PDEs). Data at each time step at each spatial point is available and a specific setpoint for the outputs is also available. Moreover, all of the outputs are assumed to be the states of the*

system. In theory, this method could be applied but generally speaking, one cannot assume that the states of the system are entirely contained in a high-dimensional output. Nevertheless, even if all the states of the system are assumed to be present in the high dimensional output, or even consider the output as it is and directly apply subspace identification, there might not be enough input-output data to satisfy the rank requirements/assumptions for the Hankel matrices (Equations (6.5), (6.6)) formed as part of the subspace identification algorithm (Moonen et al., 1989). Essentially the assumption about the formed Hankel matrices is that the number of data points (corresponding to the number of Hankel columns) should be very much larger than the number of input and output variables multiplied by the number of Hankel rows (corresponding to the rows of the Hankel matrix). For example, in this work, around 8 *batches of data are available, each consisting of around* 100 *time steps of data, totalling approximately* 800 *time instances of input-output data. Each time step contains* 1 *thermal image output (i.e.,* $90 \times 120 = 10800$ *variables) and* 1 *heater input. For* $i = 3$ *Hankel rows there would be* 32403 *rows and around only* 500 *columns in the final Hankel matrix. This clearly violates the assumption and this is similar to an under-determined system of equations, where the model obtained would be clearly overfitting to the data. The proposed method is developed with this in mind and uses the latent features of the thermal image as the outputs, instead of states of the state space model.*

Other examples are also mentioned in the manuscript with a more complex architecture involving different types of neural networks (Tan et al., 2024; Masti and Bemporad, 2021). However, these require a high amount of input-output data to fit a decent model, which is obtained using a set of Partial Differential equations (PDEs). Generating huge amounts of data might not be this easy when dealing with real systems. Moreover, only a state space model relating the output images and the inputs is finally obtained but the problem of control of quality variables is not addressed.

6.3.1.4 Predictive Model for a new batch

While testing the model, or using the model for control on a new batch, the first few data points are needed to estimate the states of the dynamic LTI SS model. A Luenberger observer (Equation (6.8)) is chosen for this work and is deployed to continuously update the states as the new output measurement (in this case, the latent variable, obtained when a new measurement of the high dimensional output is processed using the CNN and PCA models at every time step) is available. Here we change the notation slightly, to differentiate between the states of the system $z[k]$ that have been calculated and corrected using the feedback and the states $x[k]$ that are predicted in the MPC iterations/model predictions. The state estimates are obtained using the Luenberger observer as follows:

$$\hat{\mathbf{z}}[k+1] = \mathbf{A}\hat{\mathbf{z}}[k] + \mathbf{B}\mathbf{u}[k] + \mathbf{L}(\mathbf{y}[k] - \hat{\mathbf{y}}[k]), \tag{6.8a}$$

$$\hat{\mathbf{y}}[k]) = \mathbf{C}\hat{\mathbf{z}}[k] + \mathbf{D}\mathbf{u}[k], \tag{6.8b}$$

where A, B, C and D are LTI SS model matrices, obtained from the approach detailed in Section 6.3.1.2. The observer, L is chosen in such a way that the error in the states decays to zero as time progresses. In particular, the poles of the matrix, $A - LC$ are distinctly chosen to be within the unit circle to ensure the desired convergence. The convergence of the output predictions implies that the states have been estimated, and at this point, the model is ready to start predicting the rest of the trajectory using the most recent state estimate on its own without an observer.

6.3.2 MPC Design

An MPC is an optimization-based controller that determines the sequence of inputs to drive the process in a desired fashion. Naturally, the success of the MPC largely depends on the ability of the model to precisely capture and predict the evolution of the process based on an input sequence. A traditional MPC is designed to track the setpoints of the variables of interest in a process. In batch processes, however, there are seldom any setpoints that need to be tracked, instead, there are some product quality requirements that need to be achieved. Choosing an objective function that captures the 'goodness' of the product and/or the production costs and determining the set of control moves that maximize the objective function while meeting certain quality constraints is the idea behind the controller design. This naturally results in Economic MPC (EMPC) formulations and one example of an EMPC is where the objective function minimizes the total input consumption while meeting user-specified constraints on the product qualities is given below:

A PI controller is initially run while collecting output measurements and estimating the states using a Luenberger observer (Equation (6.8)) on the data collected during the PI run. Once the states have converged, the states at the most recent time step are used to initialize the MPC with Equation (6.12a). At successive MPC iterations, the current state estimate determined using the Luenberger observer is fed as the initial condition through Equation (6.12a).

At a sampling instance k, the optimal input trajectory until the end of the batch is obtained by solving the following optimization problem:

$$U_k = argmin(J) \tag{6.9a}$$

$$where \;\; J = \sum_{j=0}^{j=l_f-k} \bar{u}[j] \tag{6.9b}$$

With input constraints,

$$U_{min} \leq \bar{u}[j] \leq U_{max}, \;\; \forall \;\; 0 \leq j \leq l_f - k \tag{6.10a}$$

input rate constraints,

$$|\bar{u}[0] - \bar{u}[k-1]| \leq \delta, \tag{6.11a}$$

$$|\bar{u}[j] - \bar{u}[j-1]| \leq \delta, \; \forall \;\; 1 \leq j \leq l_f - k \tag{6.11b}$$

subject to the model constraints and initial condition:

$$\hat{x}[0] = \hat{z}[k] \tag{6.12a}$$

$$\hat{x}[j+1] = A\hat{x}[j] + B\bar{u}[j] \tag{6.12b}$$

$$\hat{y}[j] = C\hat{x}[j] + D\bar{u}[j] \;\; \forall \;\; 0 \leq j \leq l_f - k \tag{6.12c}$$

$$\hat{Q}_{t_f} = R_m + P_m\hat{x}[l_f] \tag{6.12d}$$

and finally, quality constraints (requiring product qualities to be above or below desired thresholds, as appropriate)

$$\Lambda\hat{Q}_f \leq \Gamma \tag{6.13a}$$

Remark 49. *The main motivation for this work is to develop a framework to utilize*

high dimensional outputs for closed-loop control that can achieve quality specifications and the output for this motivating example is in the form of an image. The important thing to note here is that these quality variables are not measured during the batch and are only available once the batch is completed. However, there has been some work with PLS on developing image-based monitoring and control for applications such as froth flotation (Duchesne et al., 2012; Liu and MacGregor, 2008). However, multivariate analysis is used to directly extract physical features from the images, that are linked to the quality variables of interest. For example, the images would contain the bubbles, and the variable that would be controlled would be the size of the bubble itself. This can be thought of as unsupervised soft-sensing, where a target variable is not available when a soft-sensor model is built, in contrast to a more common approach where one has access to relevant historical sensor data. Most importantly, in these applications, a target image setpoint (with a desired bubble size distribution) is provided and is used for setpoint tracking. For the present manuscript (and many other situations), the thermal image does not have a direct, static relation to the quality variable. Additionally, there might be a case where there could be two batches whose final image could be very similar, but possibly with entirely different process trajectories. In that case, the product qualities might be completely different even though the final images are similar. Hence, simply using a final image of a "good" batch that produced a good product as a target setpoint may not always result in a good product. However, in this work, the quality variables that need to be tracked are not visible through the high dimensional data (thermal images of the mold in this case) itself and therefore the final image of the process cannot be attributed to the corresponding product quality of that batch since the product quality depends on the entire batch trajectory rather than the final output value. In the proposed approach, the high dimensional output is used to build a dynamic model and then link the states of this dynamic model to the quality variables, an approach that has been used in many other batch applications as well (Corbett and Mhaskar, 2016) most image-based MPCs

in the literature have been implemented in simulation case studies and not on a real system.

6.4 Implementation to the motivation example

In this section, the proposed approach elaborated in Section 6.3 will be further explained in detail in the context of the Rotational Molding process.

6.4.1 Data Gathering and Pre-Processing

Before running the rotomolding experiments with the polymer, a couple of dry runs were done, i.e., the experiment was carried out without the material, to obtain images and train the auxiliary models like the CNN models and the PCA model. These models do not require any information about the quality of the product but require a large amount of images to be trained sufficiently hence the need for dry runs. Around $10,000$ images were produced during the dry runs, across a wide temperature range that the actual experiment operates within. Furthermore, 4 batches were run in PI mode for $14-18$ mins, where the PI was set to track the oven temperature to 300°C. Once a sufficient amount of data was collected, 4 more batches were produced using the control strategy described in previous results (Chandrasekar et al., 2022b,a). The input and output data is collected (and also the control action is implemented) around every 7.5 seconds, which translates to 8 RPM which is the rotation speed of the rotomolder. An Economic Model Predictive Controller (EMPC) has been developed, with the primary objective of minimizing input energy consumption while satisfying the constraints imposed by product quality requirements. These quality requirements are embedded as constraints within the optimization problem, alongside model and input constraints. To validate the model, the database was partitioned in a 50:50

ratio for training and testing, respectively.

The implementation specific to the motivating example is illustrated in Figure 6.4. Due to the additional complexities present in the motivation example as described in Section 6.2.2, two CNN models will be used to pre-process the images before the proposed method is applied. Specifically, a CNN-based classifier is utilized to determine whether an image contains the mold, which was always cooler than the oven walls due to the heating of the polymer within. The classifier assesses the images and enables other models to utilize only those images where the mold is present. For instance, in Figure 6.1, the classifier permits only the leftmost image to undergo further processing by subsequent models like PCA. This CNN has 2 sets of convolution layers, a batch normalization layer, a ReLU layer and a maxpool layer in series, 1 fully connected layer, and finally 1 softmax layer (Conv->BatchNorm->ReLu->MaxPool->Conv->BatchNorm->ReLu->MaxPool->FC->SoftMax). Around 9000 images were used to train the CNN classifier with a 97.3% accuracy.

The second CNN (Yolov3, (Redmon and Farhadi, 2018)) outputs the approximate coordinates of the centre of the mold, and this only gets activated when the first CNN detects a mold in the image. This particular model also outputs the approximate height and width of the object that it detects, but we only take the centre coordinates. This model was pre-developed in the community and is available as an open-source package, which we simply installed and used. The details of this model are available in their paper (Redmon and Farhadi, 2018). From the centre coordinates, a 40x40 frame is drawn and this sub-image is taken and reduced by the PCA model further. The second CNN is however trained to detect the object (mold in our case) and hence the edges are taken into account while training. However, we recognize that the object detection CNN would not guarantee a fixed frame of the image. Consequently, we adopted the approach discussed above. In doing so, sometimes, this $40x40$ image might slightly contain the background as well, which is subject to model accuracy.

The functioning of the CNN classifier along with the CNN-based object detection model is illustrated in Figure 6.5 on the left.

In the present manuscript, PCA is employed to condense the high-dimensional thermal image, which consists of $40 \times 40 = 1600$ temperature measurements (1 measurement at each pixel location). Ultimately, only one principal component (or latent variable) per image is selected and utilized as the output variable, representing the dynamics of the system. In the current implementation, a single latent variable was chosen because it effectively captured the most variance in the data.

Furthermore, this 40x40 image is subjected to PCA. Around 3000 images that contained more than 80% of the mold are used to train the PCA model. The first latent variable explains 85% of the accuracy and is found to be sufficient in the overall modelling context. The latent variable is meant to capture the key information available in the thermal image, which in turn is expected to contain 'more' information than a point value of a temperature. Therefore, the behaviour of the latent variable would be expected to align with that of a representative temperature of the mold. It is known from previous experience (Chandrasekar et al., 2022b) that the internal mold temperature should monotonically increase, so the latent variable is also expected to have a monotonic trend (either increasing or decreasing). This is corroborated through Figure 6.5 below. Hence, the first latent variable is extracted from the image and then used as an output variable along with the heater input while constructing an LTI SS model (Equation (6.4)) of order 2 Finally, a PLS-based quality model (Equation (6.7)) is developed to establish a connection between the product qualities and the dynamics of the process.

Remark 50. *One thing to note is that in the batch system considered for this process, the sampling interval for the inputs and the outputs is around* 7.5 *seconds, but the thermal camera is able to capture and send information almost continuously and not just every* 7.5 *seconds. So there is actually a huge number of thermal images. However,*

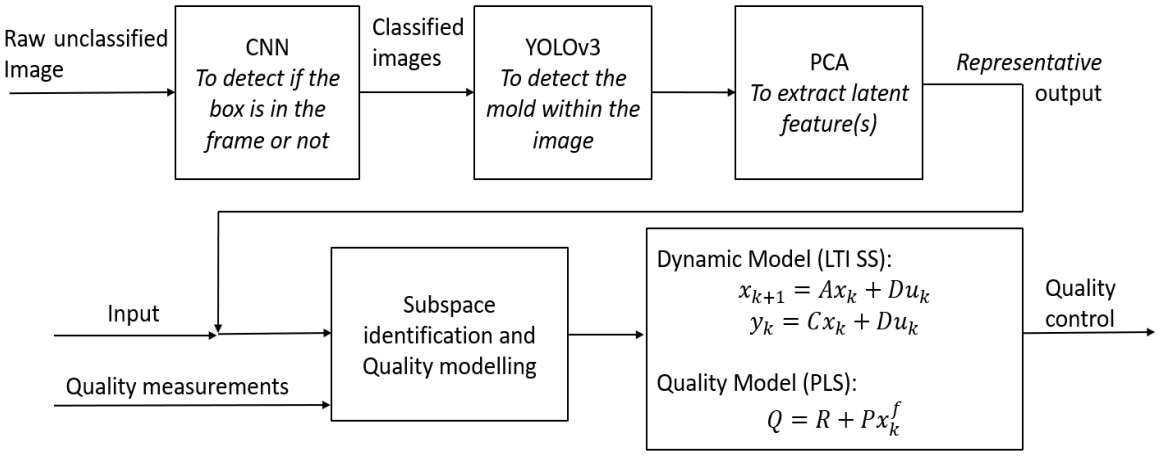

Figure 6.4: Application of the proposed approach with modifications to accommodate the specific motivating example.

because of the control interval, when the input-output pairs are considered, the thermal image set accessible for building a dynamic model is limited. That said, the exhaustive set of these thermal images is nevertheless used to build the CNNs and the PCA models as they don't require the input information whatsoever and the same dimensionality problem doesn't occur in these cases.

Remark 51. *In this work, the focus was to develop a general modelling framework that allows for images to be used under feedback control, especially for batch processes. One can potentially improve the modelling by including more latent variables or even changing PCA to autoencoder for better accuracy. It is worth mentioning that an exhaustive search over the tuning parameters of all the models that are working in tandem wasn't conducted to find the 'best' model. However, it is evident that each of their accuracy affects the overall prediction of the final product quality. A satisfactory open-loop quality variable predictions can be observed (shown in Figure 6.6) for the set of model parameters mentioned in this section and hence closed-loop control was implemented using models with this set of parameters. It could be the case where changing the CNN structure or changing the number of states in the state space model could change the scenario and perhaps having more latent features could benefit the quality model predictions. The focus of the present implementation was to achieve a*

successful demonstration of the image-based controller on a batch process.

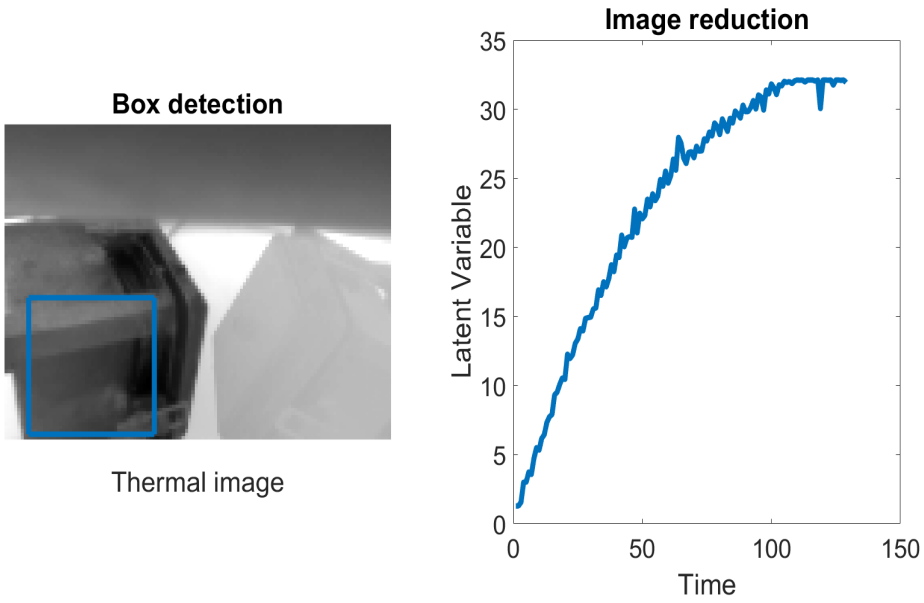

Figure 6.5: (Left) shows the working of the CNN-based classifier in detecting the image with the box and another CNN-based model (YOLO) in detecting the mold within the image. (Right) shows the evolution of the latent variable obtained by applying PCA on the highlighted portion on the left image, throughout the batch time.

Remark 52. *Techniques involving CNNs with first principles equations (Pahari et al., 2024) can be used to describe the dynamics of the process provided there is access to even some of the first principles models associated with the quality variables or the process variables of the process. However, the present manuscript handles the situation where, as in many instances, a first principles model that relates the quality variables to the input variables is very difficult to obtain and therefore not available. The proposed modelling approach therefore does not need and use the kind of first principles knowledge about the process that the suggested references need. The present approach does use some knowledge about the batch process in 1) deciding the inputs that are to be manipulated, 2) using thermal images as an output feedback mechanism, 3) recognizing that the batch duration is a decision variable in the MPC formulation since batch time also affects the product quality along with input profile.*

The efficacy of the proposed modelling approach is demonstrated on a single batch by evaluating its performance in predicting the latent variable trajectory and the quality variables, as illustrated in Figure 6.6. For this purpose, three different times are chosen $t_1, t_2 and t_3$. A Luenberger observer is used along with the state space model up to these time instances first to let the states converge. Then the model alone without the observer is used to predict the latent variable trajectory from these times till the end of the batch and also predict the final quality variables using the quality (PLS) model with the final states predicted from these times. It is apparent that while the predictions might be far from exact at the initial stages of the process, they progressively improve as the batch advances.

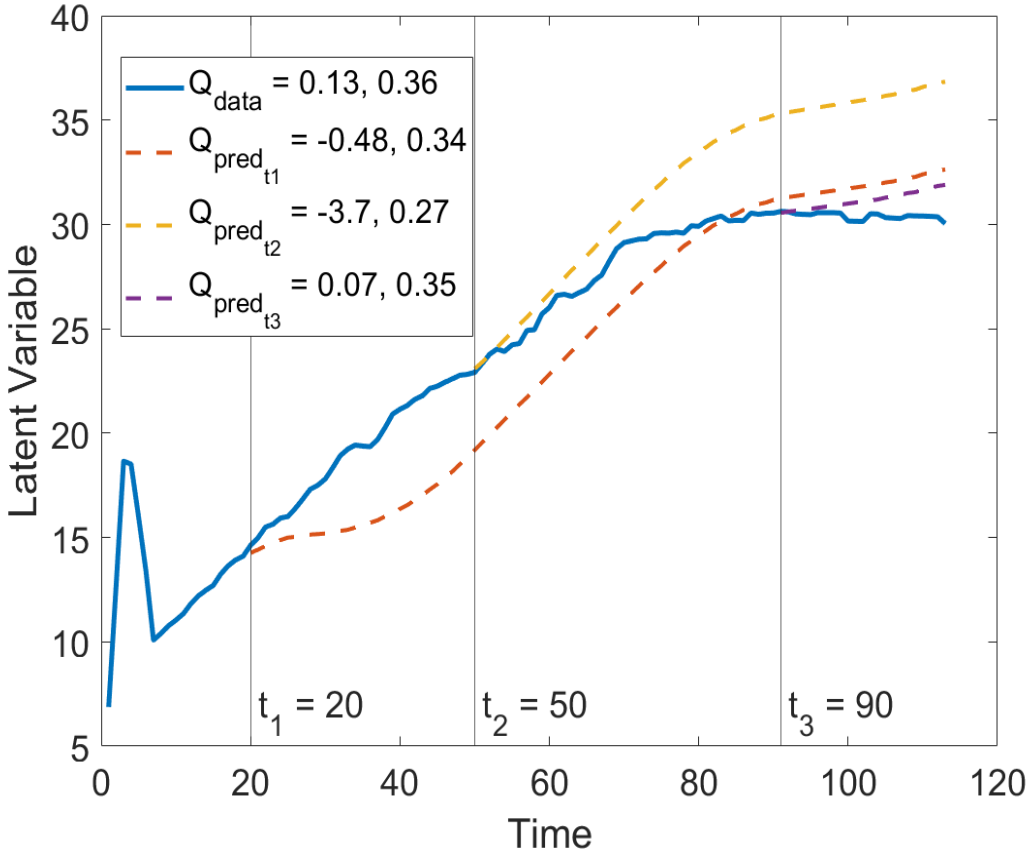

Figure 6.6: Figure shows the evolution of the latent variable in a batch, and the open loop predictions of the same using the state space model, with a Luenberger observer run till times t_1, t_2 and t_3. The legend shows the final quality variable predictions as seen from times $t_1 = 20$, $t_2 = 50$ and $t_3 = 90$ compared with the quality measurement for this particular batch.

A detailed summary of the quality predictions is provided in Table 6.1. Each error (RMSE) in the table has been calculated over all eight available batches of data. It must be noted that the quality variables do not exist during the batch, and hence we compare the predictions of the quality variables as seen from particular times during a batch. Three equally spaced times are chosen for this purpose and the respective RMSEs are tabulated in Table 6.1.

Table 6.1: Summary of Quality modelling results

Open loop predictions from time t_i	$t_1 = 30$	$t_2 = 60$	$t_3 = 90$
Quality Error (across all batches) RMSE	1.84	1.29	1.24

6.4.2 Control implementation

The closed-loop implementation presented in this section showcases the performance of the implemented Economic Model Predictive Controller (EMPC) based on the proposed modelling approach. It's essential to note that among the multiple models involved in the layered modelling strategy (depicted in Figure 6.4), only the Linear Time Invariant State Space (LTI SS) and the Partial Least Squares (PLS) models actively contribute to the MPC calculations. The other models play a crucial role in processing the feedback images, acting as an observer/filter to enhance the estimation of the system's states from the high-dimensional output. Specifically, the CNNs only tell which image contains the mold, and if a mold is detected, another CNN detects the approximate centre of the mold. A 40x40 frame is created around this centre and this reduced image is taken, on which PCA is performed. The latent feature is extracted and is used as feedback along with a Luenberger observer to correct for the states of the state space model. The non-linearity of the CNNs, or even PCA (if one decides to replace PCA with an autoencoder) does not matter once the latent features are extracted. The CNNs and PCA can be thought of as a static observer/filter.

The Model Predictive Controller (MPC), formulated in Section 6.3.2, is implemented on the Bi-axial Rotational molding setup (Section 6.2.2). The performance of the MPC is compared with an open-loop operation strategy, where a Proportional-Integral (PI) controller is designed to regulate the oven temperature at 300°C, and the only

decision variable in the operation is the batch time, which is to be set by the user before the start of the experiment. The approximate batch time was selected based on prior experience running the experiments under PI control (the current state of the art), with variation allowed a few minutes. Moreover, the MPC formulation accommodates different batch times so that the operator would not have to choose manually, as is the case with the PI control. It's important to note that the duration of the heating cycle in the rotomolding process is a variable to be decided, and this must be appropriately considered during the design of the optimization scheme. The MPC formulation allows for three possible batch times, which end up being around 0, 2, and 4 mins less than the maximum batch time.

To implement the proposed modelling and control strategy on the laboratory-scale rotomolding setup, MATLAB is connected to LabView, which, in turn, interfaces with sensors and actuators, functioning as a data acquisition system. First, the image is captured by the infrared camera, which is then processed by the MATLAB code. The MPC iteration is activated only if the image contains the mold as distinguished by the CNN models. Once the image with the mold is detected, the image is further reduced using the YOLOv3 and the PCA models as discussed in the previous sections, to a single latent variable. Since this is done at every iteration, the latent variable (which acts as the output for the LTI SS model) is recorded at every time step. A Moving Average (MA) filter of window size 3 is applied on the latest latent variable recorded during an experiment, to filter out noise/disturbance that usually occurs due to imperfect capturing of the images by the CNN models. Every new output measurement (in this case, the latent variable, obtained when a new image is processed using the CNN and PCA models at every time step) is available approximately every 7.5 seconds, and subsequently, every control action is also calculated in this interval through the MATLAB-LabView interface.

At each sampling instance l, the optimal input trajectory until the end of the batch is

obtained by solving the following optimization problem, employing Equation (6.14e) to compute the state estimate at the l^{th} instant:

$$\min_{U_f, l_f} \sum_{k=0}^{k=l_f-l} u_f[k] \tag{6.14a}$$

$$s.t.\ U_{j,min} \leq u_f[k] \leq U_{j,max},\ \ \forall\ \ 0 \leq k \leq l_f - l \tag{6.14b}$$

$$|u_f[0] - u[l-1]| \leq \delta, \tag{6.14c}$$

$$|u_f[k] - u[k-1]| \leq \delta,\ \forall\ \ 1 \leq k \leq l_f - l \tag{6.14d}$$

$$\hat{x}[0] = \hat{z}[l] \tag{6.14e}$$

$$l_f \in \{k_{switch} + 85, k_{switch} + 100, k_{switch} + 115\} \tag{6.14f}$$

$$\Lambda \hat{Q}_{t_f} \leq \Gamma \tag{6.14g}$$

$$\hat{x}[k+1] = A\hat{x}[k] + Bu_f[k] \tag{6.14h}$$

$$\hat{y}[k] = C\hat{x}[k] + Du_f[k]\ \ \forall\ \ 0 \leq k \leq l_f - l \tag{6.14i}$$

$$\hat{Q}_{t_f} = R_m + P_m \hat{x}[l_f] \tag{6.14j}$$

with,

$$\hat{Q} = \begin{bmatrix} \hat{Q}_1 & \hat{Q}_2 \end{bmatrix}^T \tag{6.15a}$$

$$\Lambda = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \tag{6.15b}$$

$$\Gamma = \begin{bmatrix} 1 \\ -0.3 \end{bmatrix} \tag{6.15c}$$

where, $U_f = [u_f[0], u_f[1], ...u_f[l_f - l]]$ are the set of decision variables consisting of the heater input at each sampling instance over the control horizon and l_f denotes the heating cycle termination time. The set of possible batch termination times,

l_f, is specified in Equation (6.14f) based on experience. The MPC is solved three times with each of the batch times, and the MPC with the best objective function is implemented at each iteration. $\delta = 0.5$ is the allowed rate of input change specified through Equations (6.14c) and (6.14d), and $U_{j,\min}$ and $U_{j,\max}$ are the lower and upper bounds on the input variable (Equation (6.14b)) with $U_{j,\min}$ and $U_{j,\max}$ being 0 and 5 respectively. In addition, k_{switch} denotes the time instance at which the controller switches to MPC from the PI controller. This is set to 30, which is roughly 30% of the batch time.

Equation (6.14g) outlines the user-specified constraints on the quality variables. Finally, Equation (6.14j) is the quality model that the MPC employs to predict the terminal product quality using the terminal states (of the heating cycle), which, in turn, are determined by the dynamic state-space model as specified in Equations (6.14h)-(6.14i). In Equation (6.15a), $\hat{Q}_1$ and $\hat{Q}_2$ denote the estimated values of the two quality variables, namely, sinkhole area percentage and impact strength, respectively.

The summary of the control implementation can be seen in Table 6.2. In contrast with the PI controller, which is purely an open-loop implementation with respect to the product qualities, the results tabulated for the MPC are obtained via a closed-loop implementation of the same. The experiments thus demonstrate a successful implementation of image-based closed-loop control.

In Table 6.2, the numbers denote the batch number, for example, PI 1 is the first batch run with the PI controller. Additionally, the numbers also denote the chronological sequence in which the batches were conducted. Experiments PI 1 & 2 and MPC 1 & 2 were conducted on the same day, whereas PI 3 and MPC 3 were carried out much later, after including the data from the prior runs and refining the models further. It was concluded that the model refinement was necessary after comparing the results and observing sub-par MPC performance as mentioned in the next paragraph.

PI 1 & 2 and MPC 1 & 2 were one of the first few experiments conducted. The total input power under MPC 1 is much less than PI 1. Moreover, we can observe that the MPC tries to meet the quality constraints but fails to do so for the second quality constraint when the batch time is decreased and the quality constraint on the first quality variable was tightened in MPC 2. This can be attributed to the low availability of data to build reliable dynamic (LTI SS) and quality (PLS) models initially. Later these batches were added to the database and the models were refined. The next set of experiments, namely PI 3, PI 4, and MPC 3, were conducted after further decreasing the maximum possible batch time to 14 minutes. It is observed that the quality constraints were now successfully met by the MPC even with the reduced batch time, while the same was not achievable by the PI. In almost all cases, the MPC strives to minimize input consumption, adhering to the quality constraints. The improved performance of the MPC in terms of the input consumption in 3^{rd} experiment over the 2^{nd} can be attributed to a better model. The same cannot be inferred while comparing the 1^{st} with 2^{nd} and the 3^{rd} ones since the quality constraints are different, hence the comparison is not valid in the first place.

Additionally, another 4^{th} MPC experiment separately mentioned in Table 6.2 shows an abnormally high input consumption. At the beginning of the experiment, as outlined in the preceding sections, the oven undergoes preheating, followed by the insertion of the mold. Opening the oven during this process results in some heat loss, necessitating a quick operation. In this specific instance, the oven wasn't closed properly initially, and although it was eventually shut, a significant amount of heat was lost in the process. Remarkably, the MPC adapted to this disturbance and successfully adhered to the quality constraints, demonstrating its capability to handle disturbances in an ongoing experiment.

Remark 53. *The ability of the MPC to adapt to disturbance should be compared to that of a PI controller. However, there is little to no scope for such an experiment in this case. The reason why the oven wasn't closed for this experiment was due to*

a mishap in the placement of the camera. However, for PI control, a camera is not required hence this problem didn't occur and consequently the oven was closed properly all the time. However, if such an incident were to happen hypothetically, even then the PI would perform worse than MPC. If we look at Table 6.2, we can see that the PI doesn't meet the quality constraints even under normal conditions for a batch time of 14 minutes. If such a disturbance were to happen, where a significant portion of heat was lost, then the PI which is designed to only track a specific oven temperature, would take longer to get back to the set point and as a result, could give a worse product.

Table 6.2: Summary of Closed loop results

Batch	Batch time (min)	MPC Quality Constraints $Q_1 \leq$, $Q_2 \geq$	Product Qualities Q_1 , Q_2	Total input power consumed
PI 1	18	$\sim$	$0.27, 0.31$	250.3
PI 2	16	$\sim$	$0.16, 0.36$	214.8
PI 3	***14***	$\sim$	***9.15***, 0.28	***182.8***
PI 4	14	$\sim$	***5.23, 0.51***	***199.65***
MPC 1	18	$2, 0.3$	$0.49, 0.34$	189.9
MPC 2	***17***	***1, 0.3***	***0.12, 0.27***	***279.6***
MPC 3	14	$1, 0.3$	$0.85, 0.64$	230.3
MPC 4	**14**	**1, 0.3**	**0.07, 0.53**	**420.9**

6.5 Conclusions

In this manuscript, we introduce a layered modelling strategy for modelling and terminal quality control in batch processes with high dimensional output feedback. The key is to model the process dynamics with high-dimensional output data and connect it to the final quality variables effectively. The proposed strategy employs a CNN classifier and an object detection model to identify the mold's location in the images. A PCA model is then applied to reduce the feedback thermal image to a representative output, serving as an output variable for the subspace-based state space model along with the heater input as the input variable. Finally, a PLS model connects the product qualities to the final states of a batch obtained from the state space model. The open-loop predictions for product quality demonstrate the efficacy of the proposed modelling approach, indicating a reliable capture of system dynamics. The closed-loop operation of the Model Predictive Controller (MPC) demonstrates the capability of the approach to steer product qualities toward user-specified quality requirements. The potential applications of the proposed method are enormous. From metal refining to biopharmaceuticals to food production, many such instances of batch processes exist where the only online sensor available is a thermal/regular image with the product qualities of interest available at the end of the heat/batch.

Bibliography

Abdi, H. and Williams, L. J. (2010). Principal component analysis. *Wiley interdisciplinary reviews: computational statistics*, 2(4):433–459.

Chandrasekar, A., Abdulhussain, H. A., Gritsichine, V., Thompson, M. R., and Mhaskar, P. (2022a). Adaptive predictive control algorithm for batch processes: Application to a rotational molding process. *Industrial & Engineering Chemistry*

Research, 61(48):17572–17581.

Chandrasekar, A., Garg, A., Abdulhussain, H. A., Gritsichine, V., Thompson, M. R., and Mhaskar, P. (2022b). Design and application of data driven economic model predictive control for a rotational molding process. *Computers & Chemical Engineering*, 161:107713.

Cho, J.-H., Lee, J.-M., Choi, S. W., Lee, D., and Lee, I.-B. (2005). Fault identification for process monitoring using kernel principal component analysis. *Chemical engineering science*, 60(1):279–288.

Corbett, B. and Mhaskar, P. (2016). Subspace identification for data-driven modeling and quality control of batch processes. *AIChE Journal*, 62(5):1581–1601.

Duchesne, C., Liu, J., and MacGregor, J. (2012). Multivariate image analysis in the process industries: A review. *Chemometrics and Intelligent Laboratory Systems*, 117:116–128.

Dunia, R., Qin, S. J., Edgar, T. F., and McAvoy, T. J. (1996). Identification of faulty sensors using principal component analysis. *AIChE Journal*, 42(10):2797–2812.

Gopaluni, R. B., Tulsyan, A., Chachuat, B., Huang, B., Lee, J. M., Amjad, F., Damarla, S. K., Kim, J. W., and Lawrence, N. P. (2020). Modern machine learning tools for monitoring and control of industrial processes: A survey. *IFAC-PapersOnLine*, 53(2):218–229. 21st IFAC World Congress.

Hijazi, S., Kumar, R., Rowen, C., et al. (2015). Using convolutional neural networks for image recognition. *Cadence Design Systems Inc.: San Jose, CA, USA*, 9(1).

Larimore, W. (1990). Canonical variate analysis in identification, filtering, and adaptive control. In *29th IEEE Conference on Decision and Control*, pages 596–604 vol.2. IEEE.

Le Cun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., and Jackel, L. D. (1989). Handwritten digit recognition with a back-propagation network. In *Proceedings of the 2nd International Conference on Neural Information Processing Systems*, NIPS'89, page 396–404, Cambridge, MA, USA. MIT Press.

Liu, J. J. and MacGregor, J. F. (2008). Froth-based modeling and control of flotation processes. *Minerals Engineering*, 21(9):642–651.

Lu, Q. and Zavala, V. M. (2021). Image-based model predictive control via dynamic mode decomposition. *Journal of Process Control*, 104:146–157.

Masti, D. and Bemporad, A. (2021). Learning nonlinear state–space models using autoencoders. *Automatica*, 129:109666.

Mhaskar, P., Garg, A., and Corbett, B. (2019). *Modeling and Control of Batch Processes*. Advances in Industrial Control. Springer International Publishing, Cham.

Moonen, M., De Moor, B., Vandenberghe, L., and Vandewalle, J. (1989). On- And Off-Line Identification Of Linear State Space Models. *International Journal of Control*, 49(1):219–232.

Narasingam, A. and Kwon, J. S.-I. (2017). Development of local dynamic mode decomposition with control: Application to model predictive control of hydraulic fracturing. *Computers & Chemical Engineering*, 106:501–511.

Oyama, H., Leonard, A. F., Rahman, M., Gjonaj, G., Williamson, M., and Durand, H. (2022). On-line process physics tests via lyapunov-based economic model predictive control and simulation-based testing of image-based process control. In *2022 American Control Conference (ACC)*, pages 2479–2484.

Pahari, S., Shah, P., and Kwon, J. S.-I. (2024). Achieving robustness in hybrid models: A physics-informed regularization approach for spatiotemporal parameter estimation in pdes. *Chemical Engineering Research and Design*.

Paul, L. C. and Al Sumam, A. (2012). Face recognition using principal component analysis method. *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)*, 1(9):135–139.

Pour, N. D., Huang, B., and Shah, S. L. (2010). Subspace Approach to Identification of Step-Response Model from Closed-Loop Data. *Industrial & Engineering Chemistry Research*, 49(18):8558–8567.

Qin, S. J. (2006). An overview of subspace identification. *Computers and Chemical Engineering*, 30(10-12):1502–1513.

Qin, S. J., Lin, W., and Ljung, L. (2005). A novel subspace identification approach with enforced causal models. *Automatica*, 41(12):2043–2053.

Rawat, W. and Wang, Z. (2017). Deep convolutional neural networks for image classification: A comprehensive review. *Neural computation*, 29(9):2352–2449.

Redmon, J. and Farhadi, A. (2018). Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*.

Sanguansat, P. (2012). *Principal component analysis: engineering applications*. BoD–Books on Demand.

Tan, W. G. Y., Xiao, M., and Wu, Z. (2024). Robust reduced-order machine learning modeling of high-dimensional nonlinear processes using noisy data. *Digital Chemical Engineering*, page 100145.

Van Overschee, P. and De Moor, B. (1995). A unifying theorem for three subspace system identification algorithms. *Automatica*, 31(12):1853–1864.

Verhagen, M. and Dewilde, P. (1992). Subspace model identification Part 1. The output-error state-space model identification class of algorithms. *International Journal of Control*, 56(5):1187–1210.

Wang, J. and Qin, S. J. (2002). A new subspace identification approach based on principal component analysis. *Journal of Process Control*, 12:841–855.

Wold, S., Esbensen, K., and Geladi, P. (1987). Principal component analysis. *Chemometrics and Intelligent Laboratory Systems*, 2(1):37–52. Proceedings of the Multivariate Statistical Workshop for Geologists and Geochemists.

Yu, J., Mo, Y., and Qin, S. J. (2023). Latent dynamic networked system identification with high-dimensional networked data. In *2023 62nd IEEE Conference on Decision and Control (CDC)*, pages 461–466.

Zhiqiang, W. and Jun, L. (2017). A review of object detection based on convolutional neural network. In *2017 36th Chinese control conference (CCC)*, pages 11104–11109. IEEE.

Chapter 7

Conclusions and recommendations for future work

The main focus of research in this thesis has been addressing different challenges while modelling batch processes. The techniques proposed in this thesis are applicable to any batch process with a similar input-output-quality variable structure. The thesis demonstrates the effectiveness of these approaches through experiments on example batch processes. This chapter summarizes the main contributions and key findings of this thesis and proposes directions for future research.

7.1 Conclusions

In Chapter 2, a data-driven modelling and control framework is developed to facilitate quality control in batch processes lacking first principles or physics-based equations that connect quality variables with process variables. This approach employs a subspace-based dynamic and quality modelling technique, which is then integrated into an EMPC and applied to an experimental batch process to achieve products with desired qualities. The experimental results underscore the framework's capacity to: a) attain the highest possible product qualities, and b) produce on-spec products while respecting constraints and minimizing operating costs.

Chapter 3 introduces an adaptive modelling strategy tailored for quality control in non-linear batch processes. The proposed strategy leverages the batch nature of process data to selectively use relevant portions of past data for re-identifying subspace-based dynamic and quality models during an ongoing batch. The superior performance of the proposed method under closed-loop control with existing re-identification methods and a re-identification-less strategy demonstrates the efficacy of the proposed approach.

Chapter 4 proposes a non-linear modelling approach to directly address the non-linearities in the process. Although RNNs are a popular choice, the proposed approach

seeks to enhance the robustness of neural network identification by incorporating subspace identification algorithms to mitigate overfitting. Specifically, subspace identification is first applied to the batch data to obtain Kalman-filtered state trajectories, which are then used to construct a NARX network. The output equation is derived from the previously implemented subspace algorithm, while the quality model follows the same construction method as in Chapters 2. This modelling approach not only outperforms a regular RNN case but also results in a less complex network.

Chapter 5 explores the possibility of refining both the dynamic and quality models within a combined optimization framework. Until now, these models were identified sequentially: the dynamic model first, followed by the quality model to relate qualities with states derived from the dynamic model. The proposed approach maintains the overall modelling structure but identifies both models simultaneously using a prediction error minimization-based optimization, resulting in an NLP problem. Initial models from the previous approach serve as initial guesses for further refinement through this NLP formulation. This method, tested on the same batch process from earlier chapters, demonstrates superior performance in predicting process outputs and, critically, quality variables.

In Chapter 6, the challenge of incorporating high-dimensional outputs directly into the dynamic and quality modelling framework is addressed. Building on the core strategy from Chapter 2, extra model layers are added that preprocess high-dimensional outputs, ultimately reducing them to latent variables representing the process's dynamic outputs. This enhanced modelling strategy is tested within an EMPC framework on a lab-scale Biaxial Rotational Molding batch process, where the high-dimensional output is a thermal image. The proposed approach's effectiveness is evidenced by its quality variable prediction performance and its ability to achieve robust closed-loop quality control in an EMPC framework, particularly with high-dimensional outputs.

There have also been many contributions from our group in the past in the context

of quality modelling and control. For example, Dr. Brandon Corbett has presented the state space-based quality modelling framework (Corbett and Mhaskar, 2016) to relate the process variables and the quality variables, whose structure is majorly used in all the chapters of this thesis. Dr. Masoud Kheradmandi proposed a subspace-based reidentification algorithm (Kheradmandi and Mhaskar, 2018) for the setpoint tracking problem, which although applied in batch processes, does not address the end quality tracking problem. Dr. Debanjan Ghosh (Ghosh et al., 2019) and Dr. Hesam Hassanpour (Hassanpour et al., 2020) have explored hybrid modelling in the context of integrating first principles models with data-driven approaches. However, the contents of this thesis are distinct from these previous contributions. The work in Chapter 3 considers the case of reidentification and modifies it thoroughly so as to accommodate and utilize the batch nature of the process data. In particular, only the relevant portions of the batches in the database are chosen for reidentification, and this is done using a variable that tracks the progress of the batch. The hybrid modelling approach proposed in Chapter 4 is a hybrid of two data-driven approaches and does not rely on first principles models. Specifically, subspace identification is first used to identify the filtered state sequences for all the batches, and then a neural network is fit on the input-state dynamics. In Chapter 5, Dr. Corbett's modelling structure using two models: a dynamic (LTI SS) and a linear quality model, is preserved but the identification is improved upon using an optimization framework that minimizes the errors from both the models together.

7.2 Future work

The results obtained for research during this thesis show a few possibilities for future work:

- While investigating Neural Networks in Chapter 4, it was noticed that RNNs

have not been implemented in the community in the correct way. In some cases, although the claim would be that an RNN is fit to process data but in reality, a NARX structure is imposed on the inputs and outputs of the RNN. Upon further investigation, RNNs in most software packages like MATLAB and Pytorch (Python) have a structure that does not respect the state-space form and hence it simply cannot be used for Process Systems Engineering (PSE) applications. Furthermore, the treatment the hidden states get in neural networks is completely different from the states of a state space model, in the sense that the state of the neural network does not get identified as part of the model identification procedure. Future work can focus on fixing these issues of RNNs to match the state space description which can further enhance the quality modelling aspect as well.

- Following the results from Chapter 5 and considering the above-discussed issues get resolved, one direction for future work could consider setting up a framework to fit a nonlinear version of this model using an RNN that takes in the input sequence and not only gives the output but also predicts the qualities. With the existing tools for creating RNN models, this approach can be useful to accurately capture any sort of non-linearity in the outputs as well as the qualities.

- The success of the combined dynamic and quality modelling approach proposed in 5 calls for implementing closed-loop control with this model on both, Uniaxial and Biaxial Rotational Molding batch process.

- The proposed model identification framework in 5, along with the resultant control application, has the potential to benefit a wide range of applications, including bioreactors and batch crystallization processes, where quality variables such as detailed assays or particle size distribution (PSD) are only available at the end of the batch. In such scenarios, the enhanced model achieved through the integrated model identification approach is anticipated to offer substantial

advantages. Specifically, the challenge of lack of observability is a common issue in crystallization control. By developing a synergistic model that incorporates online measurements with terminal PSD, it may be possible to enable effective PSD control of crystallization processes, which remains a promising direction for future research.

- With an increasing interest in recent Machine Learning tools like Transformers, Diffusion models and Generative networks to name a few, one can investigate these further in the context of quality modelling and control. Reinforcement learning (RL) is also picking up in the PSE community. It can be worthwhile to explore RL techniques and examine if they are compatible with batch processes. Recalling that in most batch processes, there are no setpoints and the control objective is to simply achieve either a product with the best qualities, or a product that meets certain quality constraints, while the batch process uses the least energy-intensive input profile. Whereas most RL applications as of now focus on setpoint tracking.

Bibliography

Corbett, B. and Mhaskar, P. (2016). Subspace identification for data-driven modeling and quality control of batch processes. *AIChE Journal*, 62(5):1581–1601.

Ghosh, D., Hermonat, E., Mhaskar, P., Snowling, S., and Goel, R. (2019). Hybrid modeling approach integrating first-principles models with subspace identification. *Industrial & Engineering Chemistry Research*, 58(30):13533–13543.

Hassanpour, H., Mhaskar, P., House, J. M., and Salsbury, T. I. (2020). A hybrid modeling approach integrating first-principles knowledge with statistical methods for fault detection in hvac systems. *Computers & Chemical Engineering*, 142:107022.

Kheradmandi, M. and Mhaskar, P. (2018). Model predictive control with closed-loop re-identification. *Computers & Chemical Engineering*, 109:249–260.