UNIVERSITÀ DEGLI STUDI DI PADOVA

DIPARTIMENTO DI INGEGNERIA INDUSTRIALE
CORSO DI LAUREA IN INGEGNERIA CHIMICA E DEI PROCESSI INDUSTRIALI

Tesi di Laurea Magistrale in
Ingegneria Chimica e dei Processi Industriali

ADAPTIVE MODEL PREDICTIVE CONTROL OF BATCH SYSTEMS

Relatore: ***Prof. Massimiliano Barolo***
Correlatore: ***Prof. Dr.-Ing. Kai Sundmacher***

Laureanda: GIULIA SLAVIERO

ANNO ACCADEMICO 2016 - 2017

"Life can only be understood going backwards,
but it must be lived going forwards."

-S.A. KIERKEGAARD

Ringraziamenti

Voglio ringraziare di cuore il mio supervisore Erdal Aydin. La sua guida attraverso tutto il lavoro è stata di fondamentale importanza. Grazie per tutta la pazienza, il supporto e la conoscenza che mi ha trasmesso.

Un grazie va al Professor Sundmacher che mi ha permesso di fare la tesi nel suo gruppo di ricerca e che mi ha accolta al Max Planck Institute facendomi sentire parte integrante dell'istututo.

Non di meno vorrei ringraziare il Professor Barolo, per la passione che mette nell'insegnare. E' merito suo se mi sono appassionata alla dinamica di controllo e se ho fatto la tesi su questo argomento. Grazie, inoltre, per avermi convinta a svolgere la tesi all'estero e per essere sempre stato disponibile in caso di bisogno.

Un grazie doveroso va a tutte le persone che ho conosciuto alla Otto von Guericke Universität e al Max Plank Instutute. Tutti hanno contribuito in qualche modo alla realizzazione della tesi e ancor più a rendere la mia permaneza a Magdeburg speciale.

Un grazie particolare va a Karsten e alla sua famiglia. Grazie per l'amore, il supporto e per avermi fatta sentire a casa.

Voglio ringraziare tutti i miei compagni di corso, in particolare Marco, Francesca, Eugenio, Benedetta e Beatrice. E' stato un piacere condividere con loro questi anni di università.

Il mio più grande ringraziamento va alla mia famiglia. Alle mie sorelle, Aurora e Maria Chiara. Nonostante la differenza in età e carattere, non saprei esprimere quanto vuota sarebbe la mia vita senza di loro.
Infine, ringrazio i miei genitori: Manuela e Costantino. Superfluo dire che hanno reso possibili i miei studi e per questo hanno la mia più profonda gratitudine. Inoltre, voglio ringraziarli per essere stati ed essere sempre presenti ogni qualvolta mi serva aiuto, un consiglio o qualsiasi altra cosa. Loro sono le mie radici, salde e fondamentali nel terreno. Il traguardo che raggiungo oggi è per metà anche loro. Grazie.

Grazie di cuore a tutti.

Acknowledgement

First of all, I would like to thank my advisor Erdal Aidyn, that mentored me through the entire work. His guidance was of fundamental importance. I am grateful for his patience, his support and for all the knowledge he shared with me.

My gratefulness goes to Professor Sundmacher, who gave me the great opportunity to join his research group and the dynamic and stimulating environment of the Max Planck Institute.

I would like to sincerely thank Professor Barolo for the passion shown teaching. It is thank to him if I got keen on the world of process control. Additionally, I want to thank him for encouraging me to go abroad and to have always been available in case of any need.

I am grateful to all the people that I've met during the work of thesis. Everyone contributed somehow to it and to make the experience in Magdeburg special.
A particular thanks go to Karsten and his family. For the love, support and to make me feel home.

I want to thank my fellows and friends of university, particularly Marco, Francesca, Eugenio, Benedetta and Beatrice. It has been a great pleasure to share these years of university with them.

Finally, I am infinitely grateful to my family. My sisters Aurora and Maria Chiara. Even though we have different ages and temperament, I cannot express how plain my life would be without them.
Lastly, I thank my parents: Manuela and Costantino. Useless to say that they made my instruction possible and for that I'm more than grateful. Moreover, they have always been there when I needed help, advice or anything. They are my roots, steady and fundamental in the ground. Today this achievement is also theirs. Honestly thanks.

Thank you all.

Abstract

Batch and semi-batch processes are widely used in the chemical industry, due to their flexibility and capability to deal with fine chemicals and high-added value products. Batch processes are unsteady-state operations; therefore they are difficult to be controlled. Model predictive control (MPC) was proved [7] [6] to be a valid choice to regulate transient operations. Thus, it is capable of control and at the same time optimize the processes. On the other hand, linear and nonlinear MPC present some drawbacks such as inaccuracy of the model for LMPC and large computational time and the presence of different local solutions for NMPC. In order to overcome these issues, Adaptive Model Predictive Control (AMPC) can be employed. AMPC put together features of linear and nonlinear MPC. Particularly, a nonlinear model for the process is successively linearized at each time step. Therefore, the recursive linearization of the model ensures the tracking of the nonlinear dynamic, with a reduced computational effort and a global solution.

In this work, Adaptive Model Predictive Control (AMPC) was applied on a semi-batch reactor and a batch distillation column. The results obtained were compared with the performances achieved utilizing linear and non-linear model predictive control. It was shown that AMPC is capable of regulating batch operations with product performances similar to that of NMPC and it requires a computational time significantly lower than the one required with NMPC. Furthermore, the robustness of AMPC was tested introducing some process mismatch in the process model. Therefore, it was shown that some recursive tuning might be necessary to increase robustness.

Contents

List of Figures

List of Tables

Introduction

As the scope of this thesis, Adaptive Model Predictive Control was applied to batch processes and its performance was compared with the one obtained using linear and nonlinear Model Predictive Control.

In this years batch and semi-batch operations are gaining lots of importance, due to the high competitiveness on the market and to strict safety regulations [6]. Batch processes ensure more flexibility in the operations, since it is possible to change completely feed to the equipment and obtain the desired output just adjusting the operating conditions [11]. Thus, the scheduling of different operations and the optimization of the processes between a batch time and the following is allowed. Furthermore, small amount of materials can be handled, matching safety requirements. These kinds of operations are widely used in the pharmaceutical, biological and polymers industry [32].
Batch and semi-batch operations are characterized by an unsteady-state nature, which results in highly nonlinear state equations. Thus, the transient nature makes batch equipment difficult to control.

Due to the unsteady-state behavior, the common controllers such as PI or PID may not be sufficient to regulate batch operations. Model Predictive Control (MPC) was proved [7] [6] to be a valid choice for control transient systems. MPC technology has the unique feature of solving an optimization problem at each time step taking into account constraints on both input and output variables. Particularly, at each sampling time, the controller predicts the behavior of the process until the final time and optimize a certain number of control moves to minimize a cost function. After that, the first control move is implemented, and the hole optimization problem solved again at the next time step.

The solution of an optimization problem is a powerful tool that permits to obtain the maximum performance from the process. Optimization programming requires the knowledge of the process model which is implemented inside the controller. Based on the kind of model, it is possible to differentiate two type of MPC: linear and nonlinear. Linear Model Predictive Control (LMPC) is characterized by a linear model, which is usually a linearization of a more complex nonlinear model. This method requires the solution of a linear optimization problem, which is fast, easy to use and ensures the identification of the global minimum. On the other hand, the requirement of a linear model is a significant limitation, since the accuracy of the prediction is often approximate [37]. Thus, LMPC is effective when dealing with steady-state processes in which the operating conditions do not change much with respect to the one considered for the linearization of the model.
Non-linear Model Predictive Control (NMPC) uses nonlinear model. This results in accurate predictions and a wide range of operating conditions in which the controller can

be applied. NMPC requires the solution of a nonlinear programming (NLP), that requires long computational times. Furthermore, due to the nonconvexity of the problem the identification of a global optimum is not ensured. Therfore, it is pretty likely that the algorithm converges to local minima [10].

In order to avoid all these problems, Adaptive Model Predictive Control (AMPC) was introduced. AMPC combines the feature of LMPC and NMPC. In particular, the dynamic behavior is tracked by successively linearizing the model. In this way, linear programming with a quadratic objective function has to be solved. Therefore, the problem is convex and the convergence to a global minimum is ensured [23]. Besides, the solution of a quadratic programming (QP) requires usually short time.

In Chapter 1 it is explained how MPC operates, the theory behind it and the action of the main tuning parameters. Furthermore, a brief history on the introduction and development of MPC through the years is reported.

In Chapter 2, the focus is posed on dynamic optimization and particularly on the methods involved in the solution of non-linear problems.

In Chapter 3, batch and semi-batch operations are reviewed. Particular attention is addressed on semi-batch reactors and batch distillation columns. Moreover, problems affecting linear and non-linear MPC are presented. Thus, the characteristics of AMPC are introduced.

In Chapter 4, AMPC is applied on a semi-batch reactor and a batch distillation column. Thus, its performance is compared with the one obtained applying LMPC and NMPC. Moreover, AMPC robustness is tested in the presence of parameter uncertainties.

1

Model Predictive Control

1.1 Introduction

Model Predictive Control is a technique of advanced control introduced for regulating difficult multi-variable systems. It is common in the chemical industry to find complex plants in which multiple inputs and outputs are present and interact in a nested way. More than that, limitations expressed in terms of inequality constraints, are to be set on the variables in order to satisfy physical, mechanical or safety limitations. Model Predictive Control can regulate these complex systems and it does it in an optimal manner. Therefore, at each time step, an optimization problem is solved and the computed manipulated variable allows the process to maximize or minimize a particular cost function.

In MPC no explicit pairing of input/output variables is requested and the constraints are integrated directly in the controller structure [34]. What is required is the availability of the dynamic model of the process and current outputs measurements. Once this data is available, it is possible to predict the future behavior of the system. In the same way, the value of the manipulated variables is calculated based on the prediction forecast.

Model predictive control is an important tool because it allows capturing the dynamic interactions between input, output and disturbance variables. It considers constraints at each time and it combines control calculation with estimation of an optimal set point. If an accurate model of the process is present, it is possible to determine in advance potential problems [30]. The general objectives of an MPC controller are [15]:

1. prevent violation of input and output constraints,

2. drive the control variables (CV's) to their steady-state optimal values,

3. drive the manipulated variables (MV's) during time through an optimal path,

4. prevent excessive movements of the MV's,

5. when signals and actuators fail, control as much of the plant as possible.

1.2 Structure and Operation

Figure 1.1 shows a schematic representation of the MPC layout. The outputs from the plant are measured and sent to the controller. Sometimes a state estimator such as a Kalman Filter is used to predict the value of missing measurements. Inside the controller, the model of the process is used for predicting the future directions of the outputs. Thus, an optimization problem is solved in order to minimize a cost function. Simultaneously, a number of MVs variables is computed to lead the system to the computed optimum.

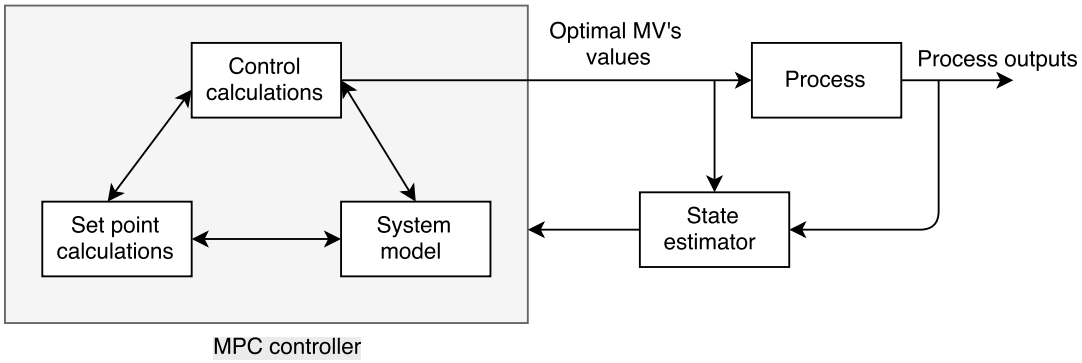


Fig. 1.1: Schematic block diagram for MPC.

In figure 1.2 it is reported how the controller operates. The duty of the controller is to calculate a set of optimal control moves that allows the predicted output sequence to reach the set point in an optimal way. Therefore, the controller calculates, from the resolution of an optimization problem, a set of M optimal control moves *u(k+1, k+2,..,k+M-1)* and a set of future outputs *y(k+1, k+2,..,k+P)*. M is defined as *control horizon* and P as *pre-*

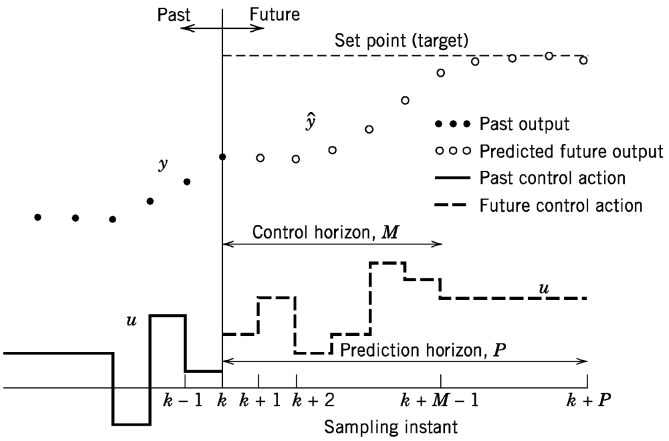


Fig. 1.2: Concepts of prediction and control horizon [36].

diction horizon. Both are tuning parameters for the controller, with $P \geq M$. At each time step, only the first control move is implemented; thus, this approach is known as *receding horizon approach*. Particularly, at each step the controller forecasts which direction the system will take and applies the first move of its optimized prediction. Then the plant is affected by the input and yields a set of output variables that are measured and send back to the controller. At this point, the controller forecasts the behavior of the system and the set of M control moves again. Thus, the procedure is repeated at each time step. It is important to notice that equality and inequality constraints are integrated into all the calculations. Usually, two kinds of constraints are set: hard and soft constraints. The former are bounds that cannot be exceeded. The second one, on the other side, can be violated for small amounts of time. This differentiation helps the resolution of the optimization problem, that could result in some decision step infeasible.

The MPC predictions are calculated based on a model of the process. In fact, it is necessary to know how the variables interact to forecast future behavior of the plant. There are three main categories of models:

- First order: these models are derived from physical consideration of the system, such as material and energy balances. They are an accurate representation of the reality and they can be used on a broad range of operating conditions. Tough, complexity is the main disadvantage, at the point that not always is possible to derive them.
- Empirical: these models are obtained starting from measurements, usually by the fitting of experimental data. Empirical models are typically simpler than the physical one, but they can be used only under limited operating conditions.
- Semi-empirical models: these models are a combination of first principle knowledge and empirical modeling.

Developing an adequate model is not a trivial task and is of fundamental importance. The model should be as much accurate as possible with respect to the reality to avoid process mismatch. At the same time the model should be not too complex for lower the computational effort. Therefore, the step involving the identification of the model is, usually, the one that requires the major amount of time.

Based on the type of model used inside the controller it is possible to differentiate between linear (LMPC) and non-linear (NMPC) controller. Both controllers work following the same scheme, but linear MPC uses a linear model while nonlinear implements a nonlinear one. LMPC is usually fast, but it can yield poor performances dealing with highly non linear processes [37]. Contrarily, NMPC allows performing a more precise calculation,

but it requires to solve a non-linear optimization, that introduces lots of issues. In fact, the computational time can be very high and there is no assurance that a global solution will be found. Additionally, another form of MPC is available: adaptive model predictive control (AMPC). This technique combines both linear and nonlinear MPC and permits to put together the nonlinearity of the models and the speed of the calculations. Hence the nonlinear model is successively linearized at each time step, yielding a linear optimization problem (§3.4).

1.2.1 Application to Step Response Models

In order to clarify how the prediction of future outputs works, an example of a step response model on *single input single output* (SISO) system is reported. The general form for expressing a step response model is reported below (1.1).

$$y(k+1) = y_0 + \sum_{j=1}^{N-1} S_j \, \Delta u(k-j+1) + S_N \, u(k-N+1) \tag{1.1}$$

Here S_j are the step-model parameters, N is the number of step-response coefficient and is an integer number. Usually $30 \leqslant N \leqslant 120$ and y_0 is the initial value of the output at time $t = 0$. Assuming that $y_0 = 0$ at the initial time it is possible to use equation 1.1 for calculate one step ahead prediction $\hat{y}(k+1)$. The formula can be written as:

$$\hat{y}(k+1) = S_1 \, \Delta u(k) + \sum_{j=2}^{N-1} S_j \, \Delta u(k-j+1) + S_N \, u(k-N+1). \tag{1.2}$$

In which is possible to define two contribution. The first term on the right hand side of the equation accounts for the effect of current control actions, while the second and third term account for the past control actions. In the same way for a generic i-th step ahead prediction it becomes:

$$\hat{y}(k+i) = \sum_{j=1}^{i} S_j \, \Delta u(k-j+i) + \sum_{j=j+1}^{N-1} S_j \, \Delta u(k-j+i) + S_N \, u(k-N+i). \tag{1.3}$$

The second and the third term of the equation 1.3 are often put together and renominated as predicted unforced response $\hat{y^o}(k+i)$. This therm takes into account all the previous values of the outputs [36].

The value of i can be used as a tuning parameter. Thus, it is possible to see that, when i is smaller, the response of the controller is aggressive. Besides, when i is increased the profile of y become more sluggish and the one of u smoother. The equation 1.3 can already be used as control law based on a single prediction done for i step ahead. Moreover, when

the predictions performed are more than one, the formulation will be the same, but it will be repeated every time step. In this case, it is possible to write the control law in a compact form using vectorial and matrix notation (equation 1.6). The vector of predicted responses over the prediction horizon and the one of the control actions over the control horizon are reported in 1.4. Also the step response coefficient can be put together in a matrix of dimension $P \times M$ called *dynamic matrix* (1.5) [36].

$$\hat{\mathbf{Y}} = \begin{bmatrix} \hat{\text{y}}(\text{k+1}) \\ \hat{\text{y}}(\text{k+2}) \\ ... \\ \hat{\text{y}}(\text{k+P}) \end{bmatrix} \qquad \Delta U = \begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \\ ... \\ \Delta u(k+M-1) \end{bmatrix} \tag{1.4}$$

$$\mathbf{S} = \begin{bmatrix} S_1 & 0 & ... & 0 \\ S_1 & S_2 & 0 & \vdots \\ S_M & S_{M-1} & \cdots & S_1 \\ S_{M+1} & S_M & \cdots & S_2 \\ \vdots & \vdots & \vdots & \vdots \\ S_P & S_{P-1} & \cdots & S_{P-M+1} \end{bmatrix} \tag{1.5}$$

In general:

$$\hat{\mathbf{Y}} = \mathbf{S}\hat{\mathbf{Y}} + \hat{\mathbf{Y}}^o. \tag{1.6}$$

Model inaccuracy or disturbances can give rise to imprecise predictions, so usually, a correction term called *bias correction* is added [3]. This correction is simply the error between the real values of the outputs measured and the one calculated with the process model (1.7). The term so obtained is add to the predicted output and allows to have a *corrected prediction* (1.8) [36].

$$b(k+i) = y(k) - \hat{y}(k), \tag{1.7}$$

$$\tilde{y}(k+i) = \hat{y}(k+i) + b(k+i). \tag{1.8}$$

The bias correction obtained can be incorporated in 1.6 as a P-dimensional vector. The value of the correction will be held constant over the whole prediction horizon and updated at each time step [3].

At each time step, control calculations are performed in order to minimize a cost function that can be expressed as minimization of [3]:

1. the predicted errors over the prediction horizon,
2. the deviation over the control horizon of $\Delta\mathbf{U}$,

3. the deviation of $\mathbf{x}(k+j)$ from its desire steady state value over M.

Furthermore, assuming that a reference trajectory to the optimal set point is available, it is possible to define a *predicted error* as:

$$\hat{\mathbf{E}}(k+1) = \mathbf{Y}_{referencetrajectory}(k+1) - \hat{\mathbf{Y}}(k+1). \tag{1.9}$$

Where $\hat{\mathbf{E}}(k+1)$ is a P-dimensional vector. Thus, considering the first two types of minimization and a quadratic programming objective function, the minimization problem is expressed as

$$\min_{\Delta\mathbf{U}(k)} \mathbf{J} = \hat{\mathbf{E}}(k+1)^T \mathbf{Q} \hat{\mathbf{E}}(k+1) + \Delta\mathbf{U}(k)^T \mathbf{R} \Delta\mathbf{U}(k). \tag{1.10}$$

Where $\mathbf{Q}$ and $\mathbf{R}$ are respectively a positive-definite and a positive semi-definite matrix and are called *weighting matrix*. $\mathbf{Q}$ and $\mathbf{R}$ are tuning parameters and are used to give more importance to the more important variables.

All the calculation reported so far are intended for a SISO system, but they can easily be extended to a *multiple input-multiple putput* (MIMO) system using the superposition principle. Hence, a system with g outputs and h inputs will yield gP-dimensional output vectors and hM-dimensional input vectors.

A schematic block of the MPC procedure is reported in figure 1.3. The first step is to measure the value of the output variables. If not all the measurements are available, it is possible to use a state estimator or not to transmit them. Indeed, MPC can control the system with a limited number of measured outputs. After that, the model predictions are updated and the ill-conditioning is checked. In particular, ill-conditioning is a problem that arises when excessive input movements are required to control the outputs independently. These phenomena takes place when inputs have almost the same, but opposite effect, on two or more outputs. Thus, a nearly singular process gain matrix is obtained. A large condition number for the process gain matrix means that big moves in the manipulated variable are made with little changes in the controller error [34]. Even though the ill-conditioning is checked at the beginning of the controller design, it is not possible to know if there will be some conditioning in a particular process configuration during the operations. Nevertheless, some methods for removing it are present. The most common are the *singular value tresholding* and the *input move suppression* [34].
In case of steady-state operations, the next step is to perform a local steady-state optimization to compute the optimal target. Thus, the optimal targets can change at any moment due to disturbances or to manual operator changes. Therefore, this local optimization uses a steady-state version of the dynamic model to re-compute the optimal target. In case of transient operations, such as batch processes, this step is avoided.

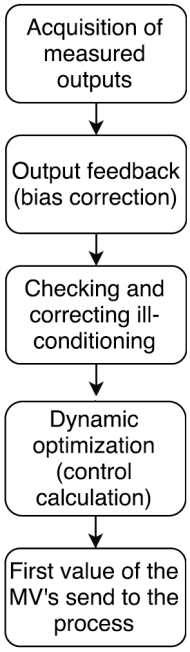


Fig. 1.3: Working procedure for MPC.

Successively, the control calculations are performed. Hence, a dynamic optimization problem is solved to determine the optimal path for the MV's. According to the *receding horizon approach*, once the optimal solution is found the first value of the manipulated variable is sent to the plant.

Typically, the optimization problem includes inequality constraints, which are determined by physical, safety or production requirements. The bounds on the input variable have, usually, to be fulfilled, mainly because of mechanical limitations. Therefore, these bounds are set as hard constraints and can never be exceeded. On the other hand, the introduction of constraints increases the complexity of the optimization that can sometimes become infeasible. Thus, to give some more flexibility to the optimizer, the constraints on the output variables are set as soft constraints. Therefore, these boundaries can be exceeded for some iteration and the violation is typically minimized using a quadratic penalty term in the optimization function [34]. Moreover, the imprecision introduced by (soft) constraints violation is acceptable, since only the first term of the manipulated variable is implemented.

1.2.2 Tuning Parameters

To ensure a proper performance of the controller, of great importance is an adequate tuning. The parameters used for tuning the controller are M, P, Δt, Q and R. It is not

possible to know a priori how the parameters affect the solution, but in general [34]:

- Δt: small Δt defines a better performance and more accurate solution, but requires then a bigger P to capture the dynamic of the process adequately. When the model is unstable or affected by errors a small sampling time is necessary. Moreover, large sampling time corresponds to more extensive computational time. It is, therefore, necessary to find a trade-off between accuracy and speed.

- $\boldsymbol{M}$: a small value of the control horizon yields a sluggish output response and a conservative input profile. If M is increased the solution is more aggressive, but the computational effort is greater.

- $\boldsymbol{P}$: a large value of the prediction horizon implies a more aggressive response and a larger computational bargain, but a more precise prediction.

- $\boldsymbol{Q}$ and $\boldsymbol{R}$: the weights define the scaling of the problem and the importance of the different variables.

1.3 History

MPC was introduced during 1970' in the industry for controlling difficult multivariable systems subjected to hard constraints. It was first born as linear controller and then expanded to the nonlinear version. Nowadays the theory behind linear MPC is largely developed and understood. The nonlinear version, on the other hand, propose still unsolved issues such as the feasibility of the online optimization, stability and performance. Besides, non-linear MPC requires the online solution of an optimization problem that in most cases requires a non-negligible computational time.

1.3.1 LQG

Model Predictive Control technology was introduced for facing the necessity of controlling complex systems subjected to hard constraints on input and output variables. The development of the idea behind MPC can be tracked in the work of Kalman in the 1960s [27] with the introduction of *LQG* (*Linear Quadratic Gaussian*). In this formulation, the process considered was a discrete-time, linear state-space model. A cost function needed to be minimized and had the following form [34]

$$\Phi = \xi(J)\,;\, J = \sum_{j=1}^{\infty} (\|\mathbf{x}_{k+j}\|^2_{\mathbf{Q}} + \|\mathbf{u}_{k+j}\|^2_{\mathbf{R}}) \tag{1.11}$$

Here the objective function penalized the input values and the deviation of the states from the original value. Two matrices Q and R were used as weighting and tuning parameters. The solution of this problem involved two steps. First, the output measurement was used to obtain an optimal state estimate and then an optimal input value was computed through an optimal proportional state controller. LQG algorithm involved a theoretically infinite prediction horizon which led to high stabilization properties as far as Q and R are positive semidefinite and positive definite, respectively [31]. This controller did not result in a big success in the chemical industry mainly because the constraints were not taken into account and nonlinear models not considered [34].

1.3.2 IDCOM

It is well known that the economic operating point of a typical process unit often lies at the intersection of constraints [18]. Thus, in order to try to keep the system closed to the boundaries without violating them, different controllers were developed. After the first development of LQG, a new structure for controlling the processes was introduced. In particular, a model of the process was introduced and a dynamic optimization problem solved at each time step. Furthermore, constraints were included directly in the model formulation in order to avoid possible future violations. Moreover, new methods for estimation of empirical models from test data were introduced, amplifying the capability of controlling different types of equipment.

A former kind of MPC was introduced in 1976 when *Richalet et al.* presented a method called Model Predictive Heuristic Control (MPHC) [25], which commercial software was known as IDCOM. Its main features were the use of impulse model and a quadratic performance objective over a finite horizon. Furthermore, a reference trajectory was set as a guide for the future output behavior, and constraints were included in the formulation. Moreover, the optimal manipulated variables profile was computed using a heuristic iterative algorithm. With this method, it was possible to drive the output variables as close as possible to the reference trajectory.
Additionally, it was pointed out by the same author that there is a hierarchy of control in the plant. First of all, time and space scheduling of the production are considered, after that, an optimization of the set points ensures that a certain goal is achieved. Only successively dynamic multi-variable control and actual control of the equipment are implemented. Thus, from this hierarchy is underlined how the optimization of the set points can yield a much greater benefit than just improving the dynamic multi-variable control. Therefore, it is possible to move the reference trajectory close to the constraints without violating them, reaching the economic operating points.

1.3.3 DMC

MPC was independently developed in the 1970's from engineers at the Shell Oil and the first application appeared in 1973 [34]. From a theoretical point of view, *Cutler and Ramaker* presented at the National AIChE meeting in 1979 a multivariable unconstrained control algorithm, which was called dynamic matrix control (DMC) [12]. *Prett and Gillette* in 1980 described an application of DMC technology able to handle non-linearities and constraints [18]. The main features of this algorithm were [34]

- the model of the plant was characterized by a linear step response procedure
- the objective with a quadratic performance was applied to a finite prediction horizon
- the future behavior of the outputs was determined following the reference trajectory
- the optimal input profiles were determined by solving a least-squared problem.

Using a step response model allowed to define the value of the outputs depending on the value of the past manipulated variables. Defining s_i as step response coefficient, it was possible to define the control law as [3]

$$y_{k+j} = \sum_{i=1}^{N-1} s_i \Delta u_{k+j-i} + s_N u_{k+j-N} \tag{1.12}$$

In other words, it was possible to define the outputs as a linear combination of the inputs. The matrix that relates I/O was called *Dynamic Matrix* and allowed the optimal vector to be computed solving a least-squares problem. Moreover, the output was driven as close as possible to the reference trajectory with a penalty term on the input. This forced the manipulated variable to take little step and ensured robustness to model errors.
IDCOM and DMC are usually defined as *first order MPC*.

1.3.4 QDMC

The tow method described so far did not take into account inequality constraints. For this reason, the DMC algorithm was modified into a quadratic programming in which constraints appeared explicitly [13]. This method was referred as QDMC and in it, the concepts of hard and soft constraints were introduced. QDMC and other algorithms able to take systematically into account constraints are known as *second generation* MPC. The incorporation of the constraints into the problem was possible formulating the problem as a QP.

Additionally, with the spreading of the MPC technology, certain problems had to be solved. Thus, in the first and second version of MPC [34]

- soft constraints were not well posed giving each the same importance,
- the case in which infeasibility occurred was not developed,
- the only models used were step response models, preventing the possibility of using more detailed models,
- the controller was not able to face the eventuality of lack of measured data,
- a single objective function was sometimes not enough for correctly describe the performance requirements.

1.3.5 SMOC and IDCOM-M

Later, two different versions of MPC were introduced: IDCOM-M by Setpoint company and SMOC by Shell Oil. The main features of the IDCOM-M comprehended the capacity to find out ill-conditioned scenarios, multi-objective function and definition of hard and soft constraints with priority [26]. The algorithm developed by Shell Oil engineers was very similar to the modern MPC, in particular [3]:

- full state estimation was used for the output feedback,
- the model used was in the state space form,
- QP formulation was used for accounting input and output constraints,
- a distinction between output variables used in the objective function and used for state estimation was introduced.

This controllers are referred as *third generation* MPC. There were also other similar controllers in this generation and they were in general characterized by differentiation of constraints that could be hard, soft and ranked, ability to recover from infeasible solutions, and a wide range of process dynamics and controller specification [34].

1.3.6 Forth and Fifth Generation

From 1990 to 2000 a *forth generation* of MPC was present. In this years MPC knew a great spreading and a high competition between the producing industries. The main leaders of the market in this field were AspenTech and Honeywell. The principal improvements introduced in this generation were [29]

1. windows-based interface,
2. introduction of uncertainties and robust control,
3. different level of optimization to address different priority to the variables,
4. steady state optimization characterized by more flexibility.

From 2000 to nowadays a *fifth generation* of controller is present on the market. At this stage, the research is mainly addressed to find adequate fast and reliable methods for solving nonlinear optimization problems. Furthermore, another issue is the dealing of uncertainties and the design of controllers in which robustness and performance are well balanced. Future directions for MPC seems to be in the field of artificial intelligence, thus, in the capability of the controllers to auto-tune and auto-regulate.

2

Dynamic Optimization

The core of Model Predictive Control is the capacity to define an optimal input sequence that minimizes a cost function. Typically, the objective is to minimize costs or maximize production. Dynamic optimization operates at more levels, involving the cost function, the dynamic of the process and the constraints. The dynamics of the process is expressed through the use of mathematical models, which are usually defined in the form of state space models.
Particularly, when the vector of the past states is available, together with the model of the process, the future direction of the system can be predicted. Therefore, it is possible to drive the process to the desired point.
In this chapter, a brief overview of the main categories and methods of optimization is reported.

2.1 Continuity

To perform optimization calculations, a model of the process has to be available. It frequently happens, that models derived from experimental data or first order model at initial conditions have discontinuity points. On the other hand in both numerical or analytical optimization it is always preferable to work with continuous functions. Continuity is defined as

$$f(x_0)\ exists \tag{2.1}$$

$$\lim_{x \to x_0} f(x)\ \ exists \tag{2.2}$$

$$\lim_{x \to x_0} f(x) = f(x_0) \tag{2.3}$$

A discontinuity in a function may or may not be a problem during the optimization procedure [41]. For instance, if a minimum occurs far from the discontinuity, this does not affect the solution. Also having cusp point may not be important, but methods can, in that case, fail using derivative [41]. Objective functions are often made up of discrete values and not continuous. Therefore, it is usually possible to interpolate the discrete values in order to obtain a continuous function.

2.2 Problem Formulation

The optimization problem can be stated as follow

$$\min_u J = f(\mathbf{x}) \tag{2.4}$$

$$s.t.\ \ g_i(x) \leq 0, \qquad i = 1,..,m \tag{2.5}$$

$$h_i = 0, \qquad i = 1,..,l \tag{2.6}$$

$$l_i \geq x_i(x) \geq u_i, \qquad i = 1,..,n. \tag{2.7}$$

Where x is a vector of n manipulated variables, J is the objective function, g represents the inequality and h the equality path constraints. The problem is said to be unbounded if the lower and upper bound are ∞ and $-\infty$. If one or more of the g and h functions are nonlinear, then the problem is also nonlinear. When x is not bounded and the functions h are not subjected to constraints, the problem is said to be *unconstrained*. In *linearly constrained* problems the cost function is nonlinear and the constraints are linear.
The same formulation can also used in case of integral costs, in that case
$J = f(x) + \int_0^{t_f} L(x,u)dt$.

2.2.1 Convexity Requirement

The existence of optimal points is not always guaranteed as well as the identification of global and not local optima. Nevertheless, it is present a condition that, when satisfied, always ensure a global optimum. This requirement is that the space of the equations is convex. A set is said to be convex when a straight segment joining two points in the convex set is also in the set. Comes naturally that all the linear combinations of points inside the set are also entirely in the convex set [41]. In the same way, it is possible to define a convex function f, as

$$f[\gamma x_1 + (1-\gamma)x_2] \leq \gamma f(x_1) + (1-\gamma) f(x_2), \tag{2.8}$$

where γ is a scalar value between $0 \leq \gamma \leq 1$ and x_1 and x_2 are two point inside the convex set. Similarly, it is possible to define a concave function as the negative of a convex function.
When both $f(x)$ and the inequality constraints are convex functions, it is possible to define a convex programming problem. The peculiarity of this formulation is that each local minimum of $f(x)$ are also the global minimum. This property introduces an important grade of simplification in optimization problems. Though, when the constraints are nonlinear, the assumption of convexity does not hold anymore. Thus, it is possible to check for convexity, but this operation is not always easy to perform. Nonetheless, there are

solutions for nonlinear programming problems, but it is harder to guarantee the absence of local optima.
Convex problems are important because [41]:

- when convexity is assumed, many significant mathematical results have been derived in the field of mathematical programming;

- often results obtained under assumptions of convexity can give insight into the properties of more general problems. Sometimes, such results may even be carried over to non-convex problems, but in a weaker form.

Particularly, certain requirements have to be satisfied in order to ensure convexity. Therefore, a set of point $\mathbf{x}$ is convex if

$$\mathbf{x}^T\mathbf{H}(x)\mathbf{x} \leq 1. \tag{2.9}$$

Where $\mathbf{H}$ is a real symmetric positive semi-definite Hessian matrix. From the study of the Hessian matrix, it is possible to understand the form of the function. For example, $f(x)$ is concave if $\mathbf{H}$ is negative semidefinite and strictly concave only when the Hessian is negative-definite. The condition of strict convexity is H(x) to be positive definite [41].

Of particular importance is when the function is quadratic or approximated by a quadratic function. Thus, for quadratic objective functions, the contours always form a convex region.
A requirement for a successful optimum technique is the ability to move rapidly in a local region along narrow valleys. A valley lies in the direction of the eigenvector associated with a small eigenvalue of the Hessian matrix of the objective function [41].

2.3 Linear Programming

When the objective function and the equality constraints are linear, it is possible to talk about linear optimization. An example of how a linear optimization problem looks like is reported in figure 2.1.

It is possible to see that the feasible region is the one that lies inside the boundaries. Thus, the solution of the optimization problem will be in that space and particularly it will be on a vertex. The main property of linear programming is that the optimal points are always situated on a vertex independently from the number of variables [41]. Moreover, this property is at the base of the simplex method of resolution.

$$
\begin{aligned}
& \max f(x,y) = 2x - y \\
s.t. \quad & 2x - y - 8 \leq 0 \\
& x + 2y - 14 \leq 0 \\
& -x + y - 4 \leq 0 \\
& -x \leq 0 \\
& -y \leq 0
\end{aligned}
$$

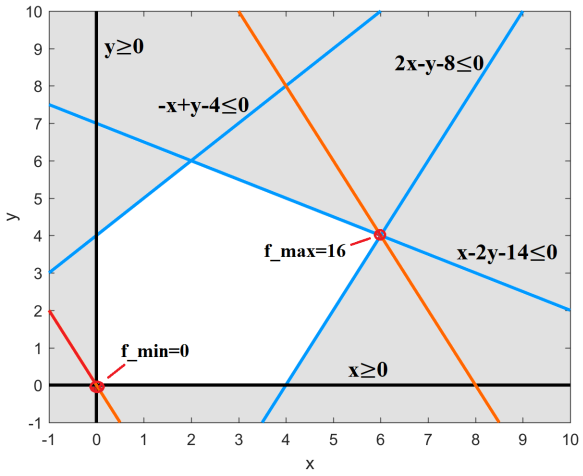


Fig. 2.1: Graphical example of a linear programming.

In matrix form a linear optimization problem can be written as

$$\min \quad f = \mathbf{cx}, \tag{2.10}$$

$$s.t. \quad \mathbf{Ax} = \mathbf{b}, \tag{2.11}$$

$$\mathbf{l} \leq \mathbf{x} \leq \mathbf{u}. \tag{2.12}$$

Where $\mathbf{A}$ is a matrix containing constraint coefficients, $\mathbf{b}, \mathbf{c}, \mathbf{l}, \mathbf{u}$ are vectors. Equations 2.11 are a set of linear combinations of the states and is possible to treat them using the common algebra. Therefore, the desired situation for optimization is when the number of variables is greater than the number of equations, yielding a potentially infinite number of solutions. The procedure of linear programming is then first to check whether solutions exist and then finding the one minimizing f.

In 2.11 all the equations are equality constraints, but in real problems also inequality constraints are present. One advantage of linear ptogramming is that it is easily possible to transform inequality constraints in equality constraints by adding some terms called *slack variables* [41]. For example adding $\mathbf{y} \geq 0$ to 2.13 it becomes

$$\mathbf{Ax} \leq \mathbf{b} \tag{2.13}$$

$$\mathbf{Ax} + \mathbf{y} = \mathbf{b} \tag{2.14}$$

The commonly used algorithm for solving LP is the simplex method. A simplex is an equilateral triangle in two dimensions, a regular tetrahedron in three and so on for higher dimensionality. This algorithm evaluates $f(x)$ at each vertex of the simplex. Once the minimum value is found another step is performed. Thus, the new search direction is opposite to the current vertex passing through the centroid of the simplex. Usually, the step size is fixed and then decreased when the solution is approached.

The main advantage of LP is that the feasible region is a convex polytope. A convex polytope can be defined as a set of convex point in the space. As already stated in section § 2.2.1 the characteristic of convexity ensures the presence of a global optimum. Comes naturally that in this case, the value of the initial guess does not affect the existence and the quality of the solution.

2.4 Direct Methods

When the set of equations reported in § 2.2 is nonlinear, a nonlinear programming has to be solved. There are two main approaches to do that:

- direct optimization methods, through which the optimization is computed directly;
- indirect methods, where the original problem is reformulated.

In direct methods the first thing performed is the discretization of the inputs. It is necessary to define a finite amount of parameters for the input in order to apply a numerical method. The peculiarity of the direct methodology is that the solution is performed directly on the original problem without reformulating it. Based on how the states are calculated, direct methods are divided in simultaneous and sequential.

2.4.1 Sequential Approach

In this method, the optimization is performed only by discretizing the input variables. Thus, at each time step, the differential equations are integrated using an integration algorithm and the cost function minimized. The optimal sequence of input is calculated using an NLP solver. This method is also named *feasible-path method* since time integration is used and the states computed are always the actual ones without approximation. The basic procedure for applying it is the following [7]:

1. the input, u, is parametrized using a finite number of decision variables,
2. an initial guess for u is chosen
3. integration to the final time is performed and the objective function computed. At each time step a set of input moves is defined to minimize the cost while satisfying equality and inequality constraints,
4. the procedure is repeated until convergence to a minimum value.

Sequential method usually yields precise solutions, but it can be slow particularly when dealing with path constraints [7]. The finer the discretization is, the more computational time is required, so a trade-off between performance and accuracy has to be found.

2.4.2 Simultaneous Approach

The most expensive part of the sequential approach is the accurate integration of the differential equations. In the simultaneous approach, this problem is skipped by discretizing the entire problem with respect to time. In this way, a large system of algebraic equations is obtained. Generally, the differential equations are satisfied only at the solution of the optimization problem and for this reason the method is often called *infeasible path approach*. The main steps used are the following [7]:

1. parametrization of both the inputs and the states using a finite number of decision variables (typically piecewise polynomials);
2. discretization of the differential equations. Therefore, the dynamic optimization problem is changed into a standard nonlinear program (NLP);
3. an initial guess for the input is chosen;
4. the procedure is iteratively repeated optimizing the problem using a NLP solver.

Solving simultaneous problems requires efficient numerical methods. Although this method is faster with respect to sequential method, it is less precise because of the approximation introduced with the integration. Simultaneous technique allows the efficient solution of large-scale optimization problems, anyway a balance between approximation and optimization must be considered carefully [7].

2.4.3 Necessary Conditions of Optimality

When solving nonlinear optimization problems, some conditions have always to be satisfied to ensure optimality.

In case of unconstrained systems, the necessary and sufficient conditions for guaranteeing that the point x_{opt} is a minimum for $\mathbf{f}(\mathbf{x})$ are

1. $\mathbf{f}$ has zero gradient at x_{opt}

$$\nabla f(x_{opt}) = 0 \tag{2.15}$$

2. the Hessian of $\mathbf{f}$ at the correspondent input point u_{opt} is negative semi-definite.

When inequality constraints are present the so called *Karush-Kuhn-Tucker* (KKT) conditions have to be satisfied for optimality. In particular when minimizing a function $\mathbf{f}(\mathbf{x})$ subjected to the inequality constraints $\mathbf{g}(\mathbf{x})$ and equality $\mathbf{h}(\mathbf{x})$ it is possible to define the Lagrangian as [35]

$$\mathcal{L}(\mathbf{x}, \lambda, \mu) = \mathbf{f}(\mathbf{x}) + \lambda^T \mathbf{g}(\mathbf{x}) + \mu^T \mathbf{h}(\mathbf{x}). \tag{2.16}$$

Where λ and μ are scalar vectors called Lagrangian multipliers. In this case a point x_{opt} is a minimum if and only if exist a pair of λ^* and μ^* for which

- $\mathcal{L}\ (\mathbf{x}_{opt}, \lambda^*, \mu^*) = 0,$
- $\mathbf{g}(x_{opt}) \leq 0,$
- $\mathbf{h}(x_{opt}) = 0,$
- $\lambda^* \geq 0,$
- $\lambda^*\ \mathbf{g}(x_{opt}) = 0.$

2.5 Indirect Methods

Another approach to solving optimization problems is to use indirect methods. The peculiarity of this approach is that the problem solved is not the same one stated in eq 2.4 - 2.7. Hence, it is reformulated in a new way that already takes into account the optimization problem. For this reason, this approach is usually referred as "first optimize and then discretize".

Considering the system stated below the aim is to find a set of input variables u_i that minimizes a cost function. For a generic free discrete problem [33]

$$x_{i+1} = f_i(x_i, u_i), \qquad i = 0, 1, ..., N-1, \qquad x(0) = x_0 \tag{2.17}$$

the cost function is

$$J = \phi(x_N) + \sum_{i=0}^{N-1} L_i(x_i, u_i). \tag{2.18}$$

Here N is given and determines the number of discretization intervals, J, ϕ and L are scalars. The problem is to minimize 2.18 with the set of equality constraints 2.17. In this formulation at each set of equality constraint a vector λ of Lagrange multipliers (or costates or adjoint states) is associated. The vector of λ_{i+1} is coupled with $x_{i+1} = f_i(x_i, u_i)$.

The necessary Euler-Lagrange conditions of optimality are then introduced [33].

$$x_{i+1} = f_i(x_i, u_i) \qquad state\,equations \tag{2.19}$$

$$\lambda_i^T = \frac{\partial}{\partial x} L_i(x_i, u_i) + \lambda_i^T \frac{\partial}{\partial x} f_i(x_i, u_i) \qquad costate\,equations \tag{2.20}$$

$$0^T = \frac{\partial}{\partial u} L_i(\mathbf{x}, \mathbf{u}) + \lambda_{i+1}^T \frac{\partial}{\partial u} f_i(x_i, u_i) \qquad stationary\,conditions \tag{2.21}$$

subjected to the boundary conditions

$$x(0) = x_0 \qquad \lambda_N^T = \frac{\partial}{\partial x}\phi(x_N). \tag{2.22}$$

The Hamiltonian function, which is a scalar function, is then defined as

$$H_i(x_i, u_i, \lambda_{i+1}) = L_i(x_i, u_i) + \lambda_{i+1}^T f_i(x_i, u_i) \tag{2.23}$$

Writing everything in a more compact way:

$$x_{i+1}^T = \frac{\partial}{\partial \lambda} H_i; \qquad \lambda_i^T = \frac{\partial}{\partial x} H_i; \qquad 0^T = \frac{\partial}{\partial u} H_i. \tag{2.24}$$

Subjected to the boundary condition of 2.22. It is possible to notice that the state equations act forward in time, the costate equations backward and the stationary conditions link the two [33]. The solution of the problem allows finding the value of u_i, x_i, λ_i. This resolution is valid for discrete systems, but it is possible to extend it also to the continuous case.

2.5.1 LQR

In the same contest it is possible to introduce the *linear quadratic regulator*. LQR is defined when the dynamic of the system is described by a set of linear differential equations and the cost function is described by a quadratic function. For a discrete system

$$x_{i+1} = Ax_i + Bu_i, \qquad J = \frac{1}{2} x_N^T P x_N + \frac{1}{2} \sum_{i=0}^{N-1} (x_i^T Q x_i + u_i^T R u_i) \tag{2.25}$$

It is possible in this case to define a the necessary condition for optimality which is expressed as [33]

$$u_i = -k_i x_i \tag{2.26}$$

$$k_i = [R + B^T S_{i+1} B]^{-1} B^T S_{i+1} A \tag{2.27}$$

$$S_i = A^T Si + 1A - A^T Si + 1B(B^T Si + 1B + R)^{-1} BSi + 1A + Q, \qquad S_N = P \tag{2.28}$$

where k is a gain given by the equation 2.27 and A, B, Q, R are known real coefficient matrices. S is a matrix found by the backwards recursion 2.28 which is also known as *Riccati equation*. This result can be extended also to the continuous case.

2.5.2 Optimization with Terminal Constraints

It is not uncommon that terminal constraints are present on all or some of the equality constraints. In general for a state equation ψ

$$\psi_N(x_N) = 0 \qquad discrete\,time \tag{2.29}$$

$$\psi_T(x_T) = 0 \qquad continuous\,time \tag{2.30}$$

Consider now a discrete time system described by the function 2.31 and subjected to 2.32, with a single terminal constraint 2.33

$$x_{i+1} = f_i(x_i, u_i), \qquad i = 0, 1, .., N, \qquad x(0) = x_0 \tag{2.31}$$

$$J = \phi(x_N) + \sum_{i=0}^{N-1} L_i(x_i, u_i), \tag{2.32}$$

$$x_N = \bar{x}_N \tag{2.33}$$

where $\bar{x}_N$ is a scalar. In this case the aim of the optimization is that to minimize the objective function while bringing the system from x_0 to x_N. N represents the length of the discretization vector and at each of the equality constraints a vector of Lagrange multiplier λ_{i+1} is associated. In the same way a vector of Lagrange multiplier is associated also to the terminal constraints equations. This scalar vector is denoted with ν. As before the Hamiltonian vector is defined as

$$H_i(x_i, u_i, \lambda_{i+1}) = L_i(x_i, u_i) + \lambda_{i+1}^T f_i(x_i, u_i) \tag{2.34}$$

The Euler-Lagrange necessary condition are then defined as [33]

$$x_{i+1} = f_i(x_i, u_i) \qquad state\,equations \tag{2.35}$$

$$\lambda_i^T = \frac{\partial}{\partial x_i} H_i \qquad costate\,equations \tag{2.36}$$

$$0^T = \frac{\partial}{\partial u} H_i \qquad stationary\,conditions \tag{2.37}$$

$$x(0) = x_0, \qquad x(N) = x_N \qquad boundary\,conditions \tag{2.38}$$

$$\lambda_N^T = \nu^T + \frac{\partial}{\partial x_N}\phi \qquad equality\,constraint\,on\,\nu \tag{2.39}$$

2.5.3 Pontryagins maximum principle

In PMP method the problem 2.31 - 2.32 is reformulated and solved in order to minimize the Hamiltonian function $H(t)$. The aim of the method is that to find a sequence of control moves u_i that takes the system from the initial to the final state through a path that

minimizes the cost index J. Considering a continue problem, it is possible to reformulate the equations as [33]

$$\min_{u(t)} = \mu_t^T L_t(x_t, u_t) + \lambda_t^T f_t(x_t, u_t) \tag{2.40}$$

$$s.t. \qquad \dot{x} = f_t(x_t, u_t) \qquad x(0) = x_0 \tag{2.41}$$

$$\dot{\lambda}_t^T = -\frac{\partial H}{\partial x_t} \tag{2.42}$$

$$\dot{\lambda}_{t_f}^T = -\frac{\partial \phi_{t_f}}{\partial x}\bigg|_{t_f} + \nu^T \frac{\partial \psi}{\partial x}\bigg|_{t_f} \tag{2.43}$$

$$\mu^T L_t = 0, \qquad \nu^T \psi_t = 0. \tag{2.44}$$

Here μ is the vector of Lagrangian multiplier for the path constraints and ν the one for the terminal constraints. The two terms in 2.44 are different than zero when the constraints are active and equal to zero otherwise. The necessary condition of optimality in the PMP principles requires the minimization of the Hamiltonian [38], therefore

$$\frac{\partial H_t}{\partial \mathbf{u}_t} = 0. \tag{2.45}$$

Which implies

$$\frac{\partial H_t}{\partial \mathbf{u}_t} = \lambda^T \frac{\partial f_t}{\partial \mathbf{u}_t} + \mu^T \frac{\partial \psi_t}{\partial \mathbf{u}_t}. \tag{2.46}$$

Equation 2.46 is the necessary condition of optimality for indirect methods. One of the advantage of PMP is that the complexity increases with the parametrization. On the other hand it is often ill-conditioned [38].

With the use of indirect methods, it is possible to solve analytically simple problems. More complex and constrained problems require a higher computational bargain and are usually resolved using shooting methods. The drawback of this approach is that the integration of the co-state equations forward in time may introduce instabilities that prevent fast convergence if no good initial guess is available [19]. Nowadays no fast-convergence solution strategy exists for constrained problem besides shooting method. Recently an alternative approach has been proposed by *Aydin et al. 2017* [19]. The idea is to parametrize the inputs and integrate the state equations forward in time and co-stated backward in time. Then, a gradient-based algorithm is used for optimization, for which good initial guess is useful but not necessary for convergence. This method was tested on batch systems giving promising results.

Another interesting topic in optimization is the one with free ending time. In this way the optimizer gains flexibility and is more likely to find the optimal solution satisfying path and mainly terminal constraints.

A big problem that affects the world of model identification is that of uncertainties. Always when dealing with situations that describe reality some variables can not be predicted precisely and for this reason it is necessary to deal with distribution and when possible density functions [8]. Stochastic dynamic programming and robust optimization are the two main branches of research that investigates uncertainties. Lots results, especially in the last years, have been achieved in this fields, but lots have not been investigated and solved yet.

3

MPC Applied to Batch and Semibatch Processes

Chemical processes can be classified in **continuous**, **batch** and **semibatch**.

1. **Continuous** processes are characterized by a constant flow of material that enters and exits the system.

2. In **Batch processes**, all the reagents are charged in the reactor at the beginning. The system is then run and only at the end of the operations the products are taken out.

3. **Semibatch processes** have a configuration in between continuous and batch systems. It is present a single material stream that enters or exits the equipment during the operations.

Additionally, chemical processes can be run in two different ways: **steady-state** or **dynamically**. The first condition shows when there is no energy or mass accumulation in the system. Steady-state operations are typical of productions and of operations which are easy to control. Dynamic condition presents when the process variables change with respect to time. This situation is typical for batch processes but also presents in the start-up and shut-down phase of continuous plants.

Batch processes are mainly used to produce small amounts of products, usually with high commercial value. Batch and semibatch reactors play a fundamental role in the production of fine chemicals, pharmaceutical, bioprocess and polymer productions [32]. Batch operations are particularly appreciated for their flexibility. It is, in fact, possible to use the same batch equipment to treat various substances. By adjusting the operating condition, a wide range of different processes can be performed. Moreover, safety is an easier issue to handle, since the quantity of dangerous materials treated is usually lower than the one handled in continuous operations. Additionally, temperature and pressure are easier to be controlled, the equipments can be frequently cleaned and it is possible to treat all gas, liquid and solid phases.

It is possible to define three main steps in batch operations [38]. First of all the process is designed and the operating conditions specified to achieve a certain product purity and

quantity. The design phase may differ every time. It changes based on the components considered and on the result desired.
Secondly, the operations need to be optimized. The batch campaigns have to be carefully scheduled in order to maximize productivity. Additionally, the dead-time required to charge and discharge the equipment is lowered and the duration of the batch operation optimized. Finally, the transient nature of batch operations has to be taken into account. An adequate system of control needs to be designed to ensure productivity and safety requirements.

3.1 Batch and Semibatch Reactors

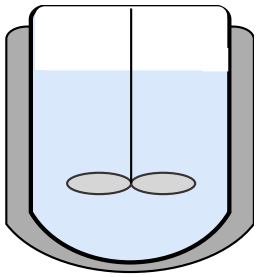


Fig. 3.1: Schematic representation of a batch reactor.

Figure 3.1 shows the structure of a batch reactor. It is composed of a tank and an impeller. No input or output flows are present and perfect mixing is assumed. The reactants are charged into the reactor at the beginning of the operation and discarded only at the end of it. A cooling/heating system is always present and is typically used as optimization variable.

The batch reactor balance in terms of moles for the generic species i is reported in equation 3.2.

$$ACCUMULATION = INPUT - OUTOUTPUT + PRODUCTION - CONSUME \tag{3.1}$$

$$\frac{dN_i}{dt} = n_{in} - n_{out} + r_i V \tag{3.2}$$

Since in batch operations there is no mass entering and leaving the system and taking into account all the reactions (NR) in which the species j take part, equation 3.3 is obtained. The same balance can be written in terms of concentrations assuming constant volume

(equation 3.4).

$$\frac{dN_i}{dt} = V \sum_j^{NR} \nu_j R_j, \tag{3.3}$$

$$\frac{dc_i}{dt} = r_i. \tag{3.4}$$

Semibatch reactors differ from batch because some material enters or leaves the equipment during the operations. A schematic figure of a semibatch reactor is reported in fig 3.2

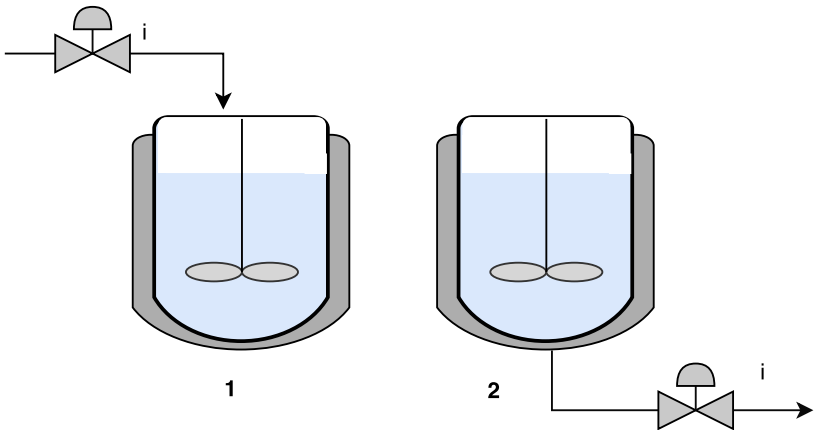


Fig. 3.2: Semibatch reactors with input feed (1) and output flow (2).

The mass balance for a fed-batch reactor (figure 3.2 (1)) can be derived from 3.2.

$$\frac{dN_i}{dt} = n_{inlet} + r_i V \tag{3.5}$$

Since $N_i = c_i V$ and $n_{inlet} = c_{i,inlet}\dot{V}$ and assuming that the concentration of species i in the inlet is constant, then

$$\frac{dN_i}{dt} = \frac{d(c_i V)}{dt} = V\frac{dc_i}{dt} + c_i\frac{dV}{dt} = V\frac{dc_i}{dt} + c_i\dot{V}, \tag{3.6}$$

$$\frac{dc_i}{dt} = \frac{\dot{V}}{V}(c_{i,inlet} - c_i) + r_i. \tag{3.7}$$

Batch reactors are usually non-isothermal. Thus it is appropriate to consider also the energy balance

$$\rho c_P \frac{dT}{dt} = \sum_j^{NR}(-\Delta H_R)_j r_j - \frac{\dot{Q(t)}}{V}. \tag{3.8}$$

Semibatch reactors present two main manipulated variables. The first one is the quantity of reactants feed or taken out from the reactor. For example, considering the case of a sequential reactions (equation 3.9) with D as by-product, the way in which the reactants are dosed during the operations can determine the quantity of sub-product produced. Furthermore, an appropriate dosing allows to control the heat generated during the reaction and the volume level.

$$A + B \rightarrow C, \qquad C \rightarrow D. \tag{3.9}$$

The other variable is temperature. Through an appropriate regulation of the temperature, it is possible to adjust the reaction kinetics. Besides temperature control is very important to match safety requirements. Especially in case of exothermic reactions, the cooling system acts to prevent runaway reactions that could lead to dangerous accidents or equipment breakages. In case of gas phase reactions pressure is an important control parameter especially to guarantee safety conditions.

3.2 Batch Distillation Columns

Distillation is the most common way to purify, separate or removing wastes from mixtures in the chemical industry. There are two main kinds of distillations: continuous and batch. In continuous distillation columns, there is a constant feed, usually placed in the middle of the column. Additionally, a lighter product is constantly retrieved from the top (distillate) and a heavier one from the bottom (residue).
Batch distillation is mainly used for fine chemicals, especially in pharmaceutical industry and biological applications. In batch distillation columns (fig 3.3) the mixture to separate is charged at the beginning in the reboiler. The operation is then started heating the blend. The lighter component is first evaporated and continuously accumulated on the top, while the heavier elements are concentrated at the bottom. Since the amount and concentration of distillate and residue changes over time, the process is unsteady-state. Batch distillation is preferred when small amounts of mixtures are treated or when there are big changes in the feed composition. Furthermore, it requires a lower capital cost [9]. Batch distillation is also the standard choice when there are seasonal operations or different production campaigns. Thus it is possible to treat a diverse range of components by adjusting only the operating conditions.

Commonly batch distillation columns are operated in three different ways. It is possible to keep the reflux ratio constant, yielding a variable product composition. Moreover, it is viable to vary the reflux ratio in order to maintain a precise composition for the desired component. The other possibility is to work with optimal reflux ratio. This case

is in between the first two and allows to obtain the most profitable operation for the distillation [17].

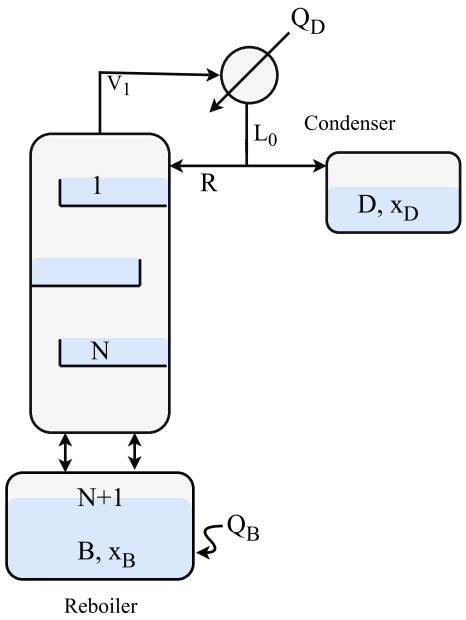


Fig. 3.3: Schematic representation of a batch distillation column.

A rigorous mathematical model for batch distillation was proposed in 1968' by Distefano [16]. It was pointed out that is much more difficult to solve the system of equations for a batch distillation column than for continuous one. Furthermore, it is pretty common to have big changes in the operating conditions and in the dynamic of the column [17].

The model proposed by Distefano takes into account the dynamic of each plate (figure 3.4). Moreover, certain assumptions have to be made [16]:

- negligible vapour hold up;

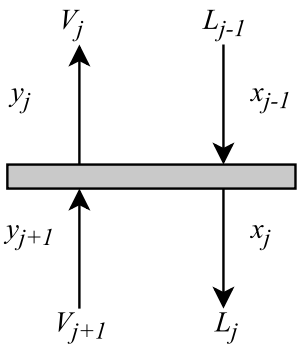


Fig. 3.4: Schematic representation of a batch distillation tray.

- adiabatic operations;
- theoretical plates;
- constant molar hold up.

For the generic species i in a mixture, it is possible to obtain the equations 3.10-3.16 [16]. Here H_j is the molar hold up in the plate, N is the number of plates in the column, $\delta_t I_i$ is the enthalpy change of the liquid stream leaving plate j and $\delta_t J_j$ is the enthalpy change of the vapour stream leaving plate j.

$$\frac{dx_D^{(i)}}{dt} = \frac{V_1}{D}(y_1^{(i)} - x_D^{(i)}), \qquad condenser\,dynamic; \tag{3.10}$$

$$\frac{x_j^{(i)}}{dt} = \frac{V_j}{H_j}[V_{j+1}y_{j+1}^{(i)} + L_{j-1}x_{j-1}^{(i)} - V_i y_i^{(i)} - L_j x_j^{(i)}],\ j = 1, 2, .., N \qquad plate\,dynamic; \tag{3.11}$$

$$\frac{dx_B^{(i)}}{dt} = \frac{1}{B}[L_N(x_N^{(i)} - x_B^{(i)}) - V_B(y_B^{(i)} - x_B^{(i)})], \qquad reboiler\,dynamic; \tag{3.12}$$

$$\frac{dD}{dt} = \frac{V_1}{(R+1)}, \qquad distillate\,flow\,rate; \tag{3.13}$$

$$\frac{dB}{dt} = V_{N+1} + L_{N-1} - V_N - V_B, \qquad bottom\,flow\,rate; \tag{3.14}$$

$$Q_D = V(J_1 - I_D) - H_D\delta_t I_D, \qquad condenser\,duty; \tag{3.15}$$

$$Q_B = V_B(J_B - I_B) - L_N(I_N - I_B) + B\delta_t I_B \qquad reboiler\,duty. \tag{3.16}$$

Batch distillation presents some drawbacks. First of all, a high amount of energy is required for operations, resulting in high costs. Secondly, high temperatures are applied to the feed vessel, that combined with the long residence time can lead to a degradation of the components [43]. To make batch distillation operations profitable a control system is required. Even though this is not a trivial task, in fact, batch columns are subjected to [39]

- high non-linearity and strong control loop interactions,
- time delays in the measurements (especially composition),
- noise and disturbances.

3.3 LMPC and NMPC - Pro and Cons

As explained in the previews sections batch processes are characterized by highly nonlinear dynamics. Thus, the application of conventional controller systems such as PI or PID could be insufficient. Applying MPC permits to control the process and in the meantime to ensure the optimality of the operations. The most commonly used MPC are the linear and the nonlinear one. Nevertheless, both present some drawbacks.

Linear model predictive control uses linear models to predict the output moves and optimize the inputs (§1.2). Linear models have the advantage of being simple to formulate and to handle. Frequently these models are linearizations of more complex nonlinear models. Furthermore, the optimization problem requires the solution of linear optimal control problems. Then, the solution space is convex; therefore, the identification of a global solution is ensured and the time needed for the calculations is reduced as opposed to a non-linear problem. On the other hand, some drawbacks are present. As a matter of fact the accuracy yield by a linear model is frequently low. This leads to poor solution performance, especially when complex processes are considered [37].
For continuous complex nonlinear systems, it is possible to perform a model linearization and use LMPC. Thus the operating conditions do not change much with respect to the linearization point and the approximation introduced with linearization can be acceptable. Furthermore, since continuous processes have steady states, the process can be pushed to that steady-state only by some time delay. On the other hand, when dealing with intrinsic unsteady-state processes, as batch operations, linear MPC is not able to adequately control the operations. Nevertheless, due to its relative simplicity, MPC is widely used in the industry, and many of companies implement it in control systems [34].

NMPC, as described in section §1.2, employs nonlinear models. This kind of models are harder to derive but they ensure a higher amount of information. The main drawback is that to optimize the inputs an NLP problem needs to be solved on-line at each time step. Hence, the equations are nonlinear and the problem results to be generally nonconvex [23]. Therefore, the existence of a unique optimal point is not ensured. Moreover, if a feasible solution is identified, there is no guarantee that it is a global minimum and

not a local one [23]. Thus the solution of the NLP at each time step is the limiting stage of NMPC. Frequently the time required for the computations and the sampling time is comparable, introducing a feedback delay in the control of the process. This implies that the input signals are implemented in a process that is already some step ahead.
Due to the comparable advantages of both linear and nonlinear MPC's, it is suggested to use adaptive MPC for the control of semi-batch processes in this theses.

3.4 Adaptive Model Predictive Control

In order to avoid the problems related to the solutions of NLP, present in NMPC, AMPC was introduced. The idea behind that is to perform, throughout the computed process trajectory, linearization using Taylor expansion at the current nominal operating conditions and then to discretize the resulting linear model [40]. If the objective function is quadratic, resulting structure is a LMPC [34]. Thus, the problem can be then solved by using simple quadratic programming (QP) algorithms. Additionally, the accuracy of the model obtained can be increased by linearizing the model equations several times along the computed system trajectory [44]. This procedure allows maintaining the information of the nonlinear model since its behavior is tracked. At the same time the convexity of the problem is guaranteed due to linearized system equations and constraints. For this reason the solution found is the global one and the calculation are performed in a fast way.

The linearization procedure can be summarized as follow. A generic function $f(x)$ (3.17) for the states is approximated by a Taylor series (3.18) around a steady-state operating point x_S and input u_S.

$$\dot{x} = \frac{dx}{dt} = f(x,u) \tag{3.17}$$

$$\begin{aligned}\dot{x} = f(x_S,u_S) + \left.\frac{\partial f}{\partial x}\right|_{x_S,u_S}(x-x_S) + \left.\frac{\partial f}{\partial u}\right|_{x_S,u_S}(u-u_S) + \frac{1}{2}\left.\frac{\partial^2 f}{\partial x^2}\right|_{x_S,u_S}(x-x_S)^2 + \\ \left.\frac{\partial^2 f}{\partial x \partial u}\right|_{x_S,u_S}(x-x_S)(u-u_S)\frac{1}{2}\left.\frac{\partial^2 f}{\partial u^2}\right|_{x_S,u_S}(u-u_S)^2 + higher\,order\,terms\end{aligned} \tag{3.18}$$

At steady-state $f(x_S,u_S) = 0$ and neglecting the higher order terms it becomes

$$\frac{d(x-x_S)}{dt} \approx \left.\frac{\partial f}{\partial x}\right|_{x_S,u_S} (x-x_S) + \left.\frac{\partial f}{\partial u}\right|_{x_S,u_S} (u-u_S). \tag{3.19}$$

Defining $\hat{x} = x - x_S$

$$\frac{d\hat{x}}{dt} \approx \left.\frac{\partial f}{\partial x}\right|_{x_S,u_S} \hat{x} + \left.\frac{\partial f}{\partial u}\right|_{x_S,u_S} \hat{u}. \tag{3.20}$$

In state space form

$$\frac{d\hat{x}}{dt} \approx \mathbf{A}\hat{x} + \mathbf{B}\hat{u} \tag{3.21}$$

The same procedure is performed differentiating output equations with respect to the inputs and the states. This yields

$$y = g(x,u), \tag{3.22}$$

$$g(x,u) \approx g(x_S,u_S) + \left.\frac{\partial g}{\partial x}\right|_{x_S,u_S} (x-x_S) + \left.\frac{\partial g}{\partial u}\right|_{x_S,u_S} (u-u_S), \tag{3.23}$$

$$(y-y_S) \approx \mathbf{C}\,\hat{x} + \mathbf{D}\,\hat{u}. \tag{3.24}$$

In general the linearization procedure results in equations 3.25 and 3.26. Here $\hat{\dot{x}}$ and $\hat{y}$ represent the linearized state and output equations; $\hat{x} = x - x_S$ and $\hat{u} = u - u_S$, with x_S and u_S values of the state and the input around the steady-state operating point.

$$\hat{\dot{x}} = \mathbf{A}\,\hat{x} + \mathbf{B}\,\hat{u}, \tag{3.25}$$

$$\hat{y} = \mathbf{C}\,\hat{x} + \mathbf{D}\,\hat{u}. \tag{3.26}$$

It is possible to graphically represent the successive linearization as in figure 3.5. At each time step, the model is linearized, and the nonlinear behavior is tracked. It comes naturally that the smaller Δt is, the more the linearization is precise.

During the years lots of attention has been posed on linear MPC, leading to many implementations in the industry. Lately, many studies have also been carried out to better understand NMPC and solve the main issues such as feasibility, stability and robustness. As a matter of fact, few studies and application of AMPC have been exploited so far

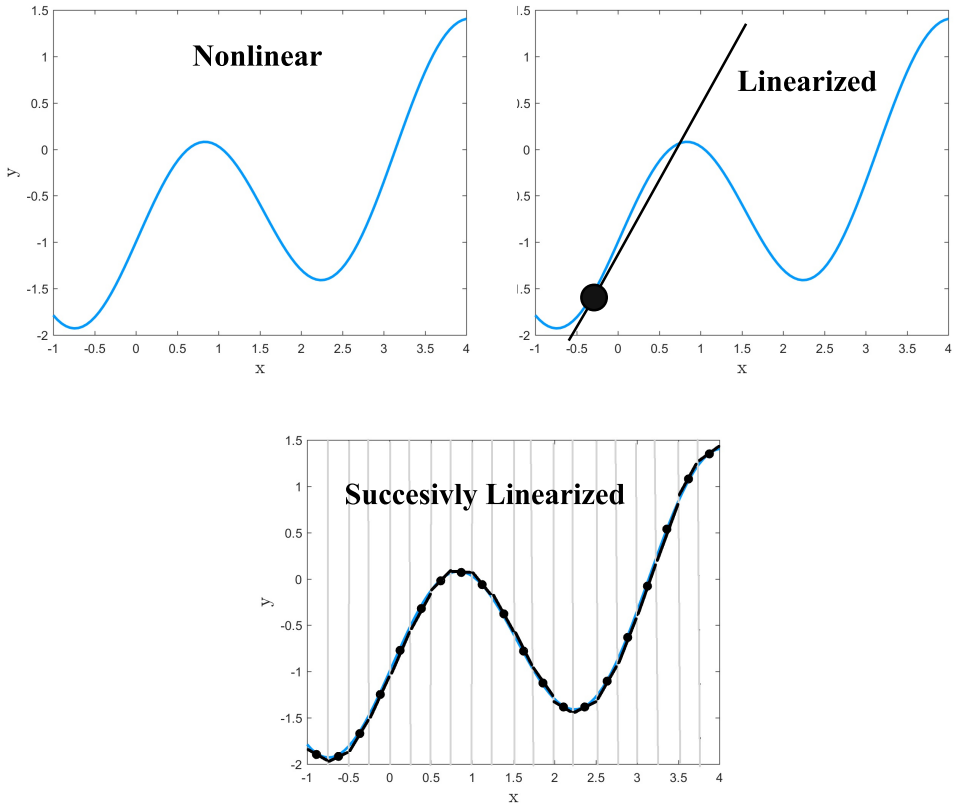


Fig. 3.5: Comparison between nonlinear, linearized and successively linearized model.

in the chemical industry. In particular for what concerns batch processes. Thus, just a few studies on AMPC applied on batch reactor can be found in the literature (such as [24]).

To overcome speed and feasibility problems of NMPC and solution accuracy problems of LMPC, in this work AMPC was used. In particular, in the next chapter, a comparison between different controller performances is made on a batch reactor and on a batch distillation column.

4

Application of AMPC to a Semibatch Reactor and to a Batch Distillation Column

In section §3.3 the advantages and disadvantages of linear and nonlinear model predictive control were discussed. In order to deal with highly nonlinear systems, adaptive model predictive control was proposed as control technique. In this chapter, AMPC is applied to two examples and the performances compared to the one obtainable using LMPC and NMPC. The first application is a semibatch reactor and the second one a batch binary distillation column. Additionally, some parametric uncertainty realizations are considered to test the behavior of AMPC in the presence of process mismatches.

4.1 Non-isothermal Semibatch Reactor

Batch and semibatch reactors play a fundamental role in fine chemicals, pharmaceutical, bioprocess and polymer productions [32]. In particular the use of semibatch reactors improves the selectivity of the reactions and ensures a better temperature control. In the last years these operations have gained popularity due to the strict environmental regulations and industrial competitiveness [19]. In order to match production and economic goals, MPC techniques seem the most appealing way to control the processes. In the following sections MPC methods are applied on a semibatch reactor (example taken from [7]).

Figure 4.2 shows the system studied. Particularly, a non-isothermal fed-batch reactor is considered, where the temperature is controlled through a cooling jacket. Inside a consecutive reaction in the form of equation 4.1 takes place.

$$A + B \xrightarrow{k_1} C \xrightarrow{k_2} D \tag{4.1}$$

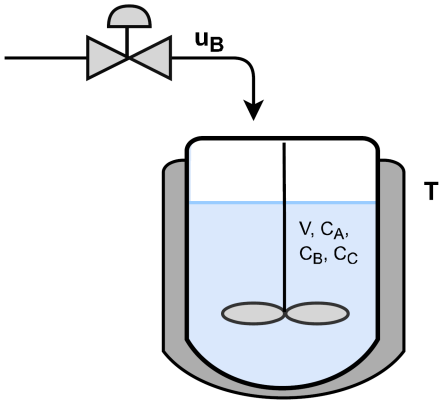


Fig. 4.1: Non-isothermal semibatch reactor with feed of species B and temperature control.

Here the desired product is C and D is a side product. There are two manipulated variables: the temperature and the feed flow rate of reactant B. The objective is to maximize the final production of C. The process presents constraints on the two inputs $T(t)$ and $u(t)$ and on the outputs. Particularly, the volume of the reactor cannot exceed $1.1\,l$ to avoid overflows. Furthermore, the heat produced by the reaction should not be greater than q_{max}, to fulfil safety requirements.
The control problem is stated in 4.2 - 4.11. In table 4.1 the values of model parameters, operating bounds and initial conditions are reported.

$$Cost function: \quad \max_{u(t),T(t)} J = c_{\mathrm{C}} V(t_f) \tag{4.2}$$

$$State\,equations: \quad \dot{c}_{\mathrm{A}} = -k_1 c_{\mathrm{A}} c_{\mathrm{B}} + \frac{u}{V} c_{\mathrm{A}};\ c_{\mathrm{A}}(0) = c_{\mathrm{A0}}; \tag{4.3}$$

$$\dot{c}_{\mathrm{B}} = -k_1 c_{\mathrm{A}} c_{\mathrm{B}} + \frac{u}{V}(c_{\mathrm{B,in}} - c_{\mathrm{B}});\ c_{\mathrm{B}}(0) = c_{\mathrm{B0}} \tag{4.4}$$

$$\dot{c}_{\mathrm{C}} = k_1 c_{\mathrm{A}} c_{\mathrm{B}} + \frac{u}{V} c_{\mathrm{A}};\ c_{\mathrm{C}}(0) = c_{\mathrm{C0}}; \tag{4.5}$$

$$\dot{V} = u;\ V(0) = V_0; \tag{4.6}$$

$$k_1 = k_{10} e^{\frac{-E_1}{RT}}; \quad k_2 = k_{20} e^{\frac{-E_2}{RT}}; \tag{4.7}$$

$$Path\,constraints: \quad T_{\min} \le T \le T_{\max}; \tag{4.8}$$

$$u_{\min} \le u \le u_{\max}; \tag{4.9}$$

$$Output\,constraints: \quad (-\Delta H_1) k_1 c_{\mathrm{A}}(t) c_{\mathrm{B}}(t) V(t) + (-\Delta H_2) k_2 c_{\mathrm{C}}(t) V(t) \le q_{\mathrm{rx,max}}; \tag{4.10}$$

$$V(t) \le V_{\max} \tag{4.11}$$

If the manipulated variables are fixed as $T = 308\,K$ and $u = 0.5\,l/h$, and the process is run at open loop, the results in figure 4.2 are obtained. Even though a high amount of

Tab. 4.1: Values of model parameters, constraints and initial conditions.

Variable	Value	Unit of Measurement
k_{1o}	4	l/mol h
k_{2o}	800	l/h
E_1	6×10^3	J/mol
E_2	20×10^3	J/mol
R	8.31	J/mol K
ΔH_1	-3×10^4	J/mol
ΔH_2	-10^4	J/mol
$u_{\min}$	0	l/h
$u_{\max}$	1	l/h
$V_{\max}$	1.1	l
$T_{\min}$	20	°C
$T_{\max}$	50	°C
$q_{\text{rx,max}}$	1.5×10^5	J/h
c_{Aom}	10	mol/l
c_{Bo}	1.1685	mol/l
c_{Co}	0	mol/l
V_{o}	1	l
$c_{\text{B,in}}$	20	mol/l
t_{f}	0.5	h

species C is produced, the constraints on the volume and the maximum heat of reaction are violated. It is then clear that a control system is required to regulate the process.

4.1.1 Controller Tuning

In order to apply MPC to the system, some assumptions are made. First of all, full feedback information is assumed. Thus, all the signals out from the process are collected and send back to the controller (figure 4.2). In case of slow measurements (e.g. composition or temperature), it is not possible to have real-time values. In this case, a state-space estimator, such as a Kalman filter, should be added to predict the value of the missing measurements [21].

In section §1.2.2 the importance of the sampling time Δt as tuning parameter was discussed. Small sampling times ensure a better tracking of the nonlinear system. Mainly, it permits to have more accurate solutions and allows to deal with model errors. On the other hand, small sampling time result in high computational bargain. Equation 4.12

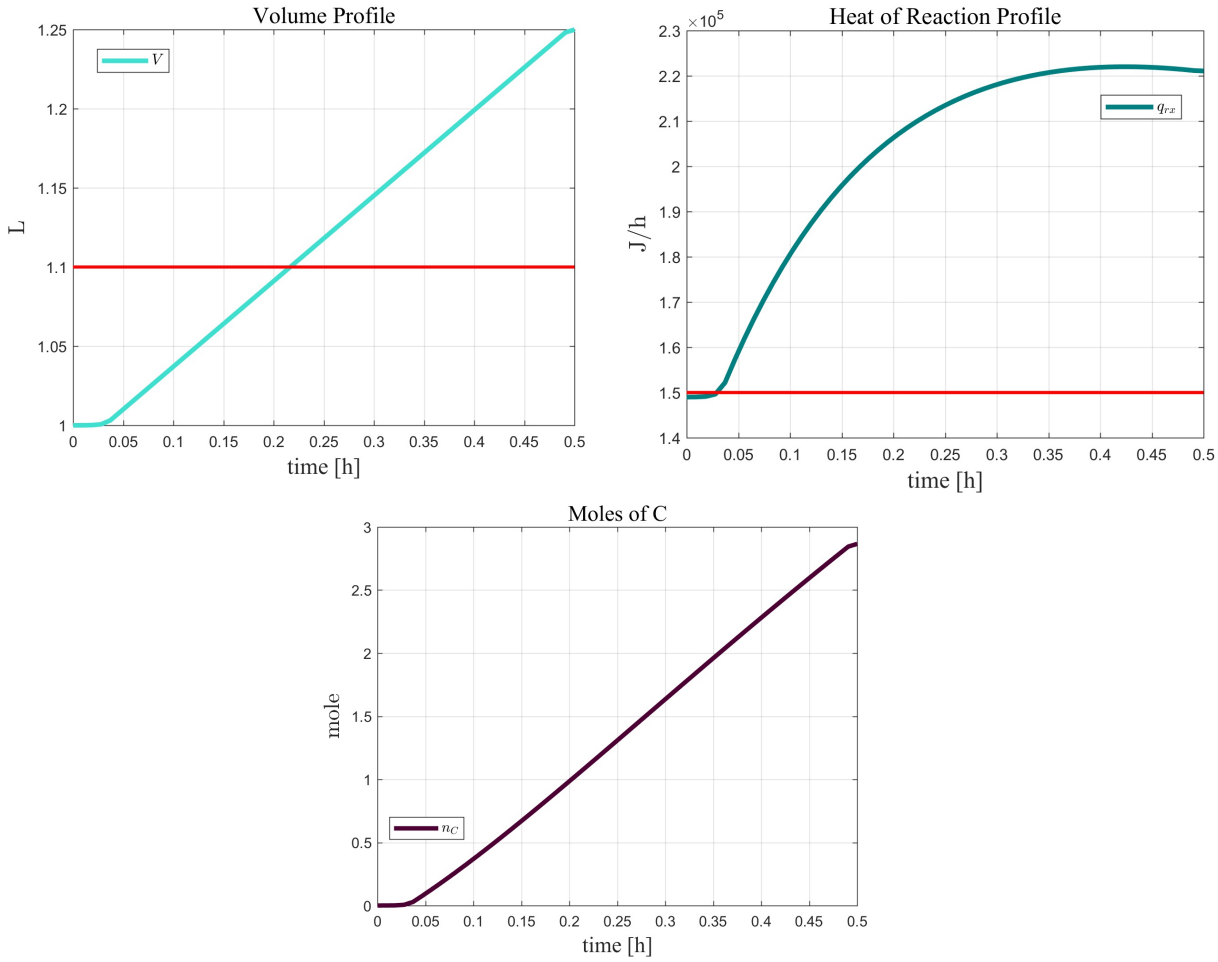


Fig. 4.2: Profiles of heat of reaction, volume and mount of C with respect to time at open loop with constant value of $T = 308$ K and $u = 0.5$ l/h.

indicates the length of the time step used.

$$\Delta t = 0.001\,h = 3.6\,s \tag{4.12}$$

$$prediction\,horizon = 500 \tag{4.13}$$

$$control\,horizion = 20 \tag{4.14}$$

In dynamic optimization problems, there are two kinds of prediction and control horizons: moving and shrinking. In moving horizon the time/step length considered is always the same and is shifted ahead of Δt at each time step. On the contrary, in shrinking horizon, the time length is kept fixed at the final time and shrunk of Δt at each time step (figure 4.3).

As a consequence of fixed final time, shrinking horizon is usually applied to batch operations. In this problem, the constraints on the outputs are path constraints and have to be satisfied at each time step. Therefore, computing prediction and control moves until

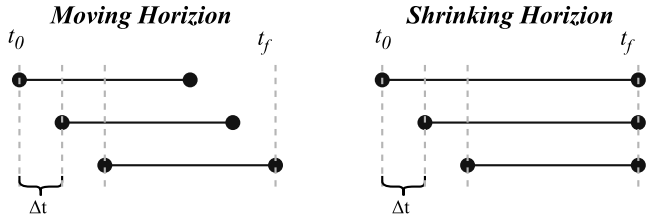


Fig. 4.3: Comparison between moving and shrinking horizons.

the final time may not be computationally favorable. Thus, a moving horizon approach may turn out to be sufficient to predict and optimize the dynamic behavior adequately. Specifically, it was chosen to predict the moves until the final time but to optimize them only for a time length equivalent to 1.2 minutes.

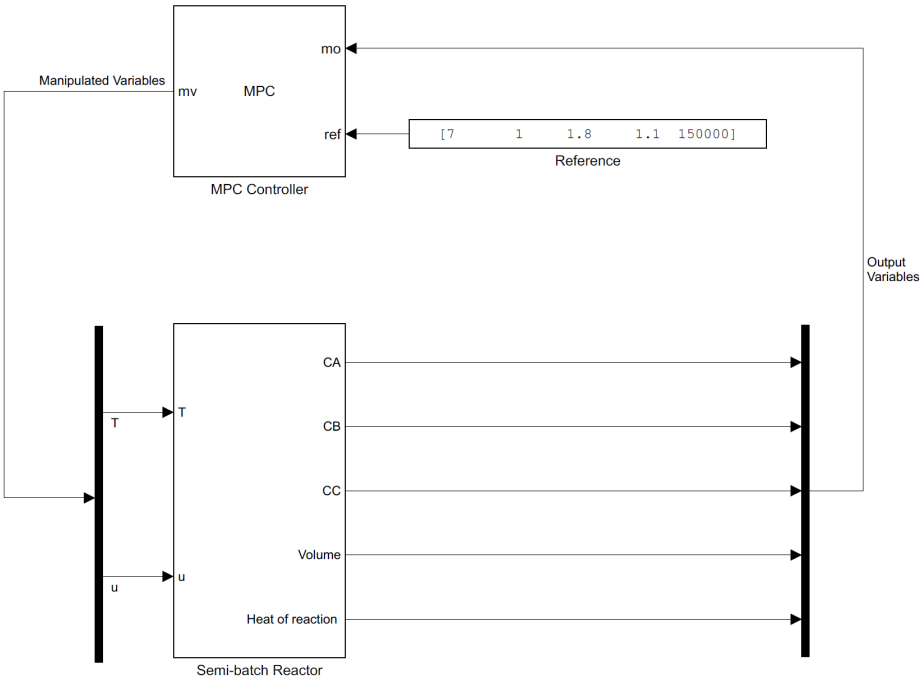


Fig. 4.4: Schematic of the control system. All the outputs from the process are measured and send back to the controller.

4.1.2 LMPC

To perform this simulation *Simulink* MPC toolbox was used. The controller model was linearized at the initial conditions. Figure 4.5 shows the variable profiles obtained from the simulation. All the figures with output profiles are reported in Appendix A.1.
It is possible to notice that at the beginning the temperature is kept at the maximum value to push the desired reaction. After some time T is lowered to inhibit the production of the side product. In the same time, the feed flow rate gradually decreases to satisfy the volume and heat of reaction constraints. Thus, the constraints V_{max} and $q_{rx,max}$ are never violated.

Even though the process is correctly controlled the final value of product C is particularly suboptimal (table 4.2).

The solution of a linear optimal control, as already discussed in chapter §2, is usually fast. As a consequence, only around 6 seconds are necessary to complete the simulation (table 4.3).

Tab. 4.2: Value of the objective function obtained applying LMPC to the system.

Objective function	
$n_C(t_f)$	1.467 mole

Tab. 4.3: Computational time required to perform the simulation applying LMPC.

Simulation Time	Time per Sampling Time
6.053 sec	0.012 sec

4.1.3 NMPC

To perform the simulation using NMPC an NLP problem has to be solved. Therefore, *IPOPT*, an open source software package for large-scale nonlinear optimization problems,

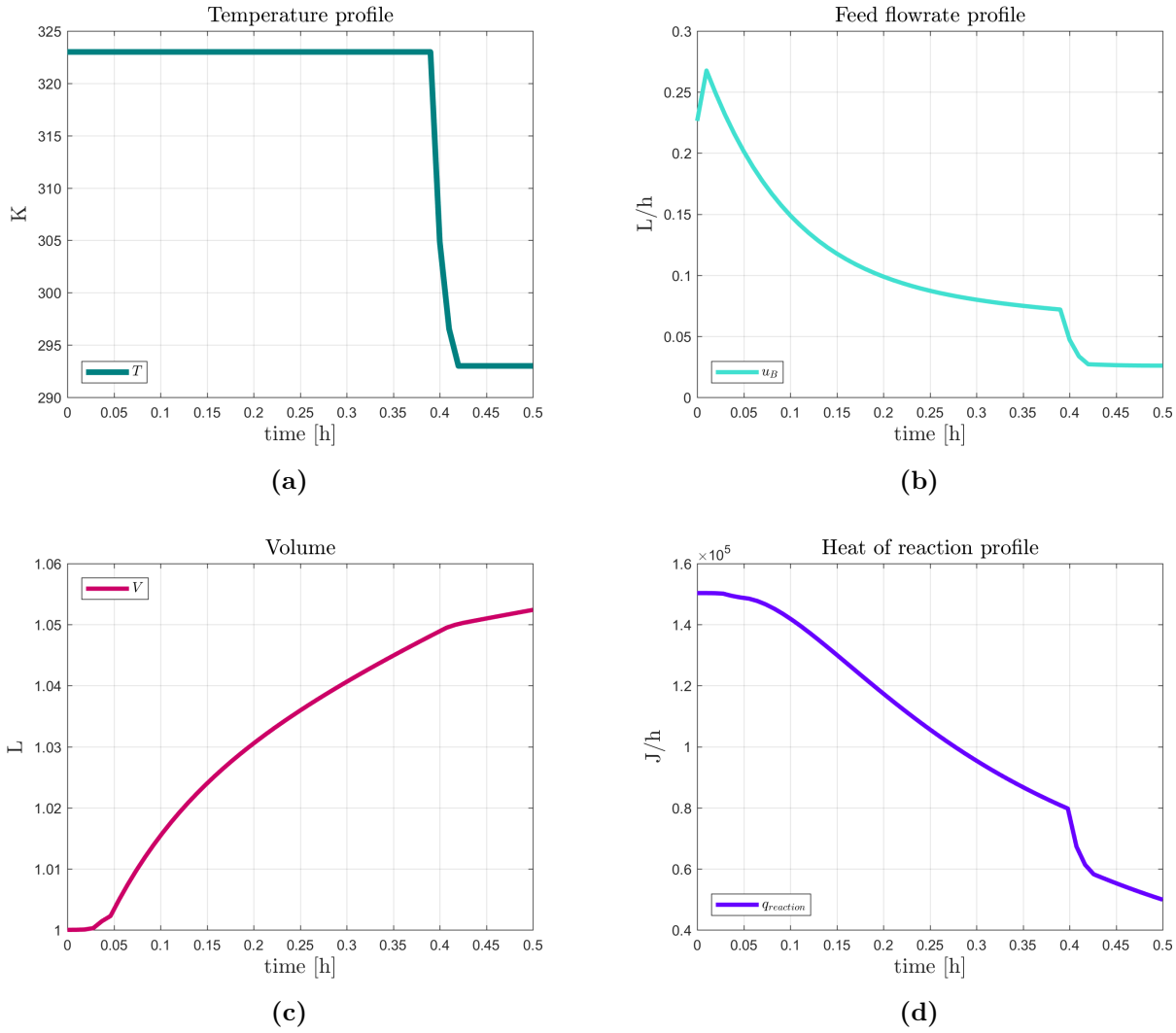


Fig. 4.5: Profiles of temperature (**a**), feed flow rate (**b**), volume (**c**) and heat of reaction (**d**) obtained applying linear MPC to the semi-batch reactor system and linearizing the model at the initial conditions.

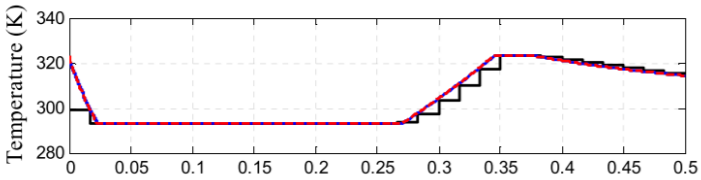
is employed using a single shooting method (more information can be found on www.coin-or.org/Ipopt [1]). To check if the solution found is good, the results obtained at closed loop are compared with the open loop results reported in the literature (figure 4.6).

Figure 4.7 shows the results obtained from the simulation (Appendix A.2).

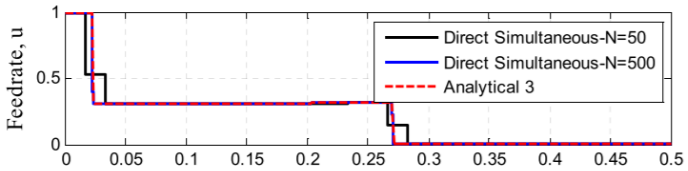
It is possible to notice that the input profiles of T and u (figures 4.7a and 4.7b) are similar to the one reported in the literature fig 4.6.

The feed flow rate profile is high at the beginning. Thus, the volume is pushed closed to its upper bound and the heat of reaction constraint is kept active (fig 4.7d). When the volume reaches V_{max} (fig 4.7c) no more material is send to the reactor.

Similarly, the temperature profile is kept for the first part at the lower bound and then set at the maximum value. Therefore, a compromise between main and side reaction is



(a) *Temperature profile*



(b) *Feed flow rate profile*

Fig. 4.6: Open loop MVs profiles (from *Aydin et al., 2017)*

Tab. 4.4: Objective function value obtained applying NMPC to the system.

Objective function	
$n_C(t_f)$	2.039 mole

found. Furthermore, figure 4.7d indicates a slight violation of the constraint.

The application of NMPC yields the results in table 4.15 and table 4.5. The amount of species C obtained is the optimal one, but it is slightly different than the results of $n_C = 2.05$ reported in the literature (e.g., [19] and [7]). Indeed, the nonlinear nature of the solution method does not exclude different local solutions [10].

Table 4.5 shows that the time required to perform the simulation using NMPC is two orders of magnitude greater than the one required for linear MPC. Even though it is still lower than the sampling time, it introduces feedback delay. This means that between the measurement of the outputs and the actual implementation of the control moves, a certain amount of time passes. Therefore, the input moves are implemented on a system which is already forward in time with respect to the one for which the inputs have been computed. It is clear that this behavior is undesired, especially during transient operations such as batch operations, and may lead to suboptimality, infeasibility or instability.

Tab. 4.5: Computational time required to perform the simulation applying NMPC.

Simulation Time	Time per Sampling Time
7.36 minutes	0.8814 sec

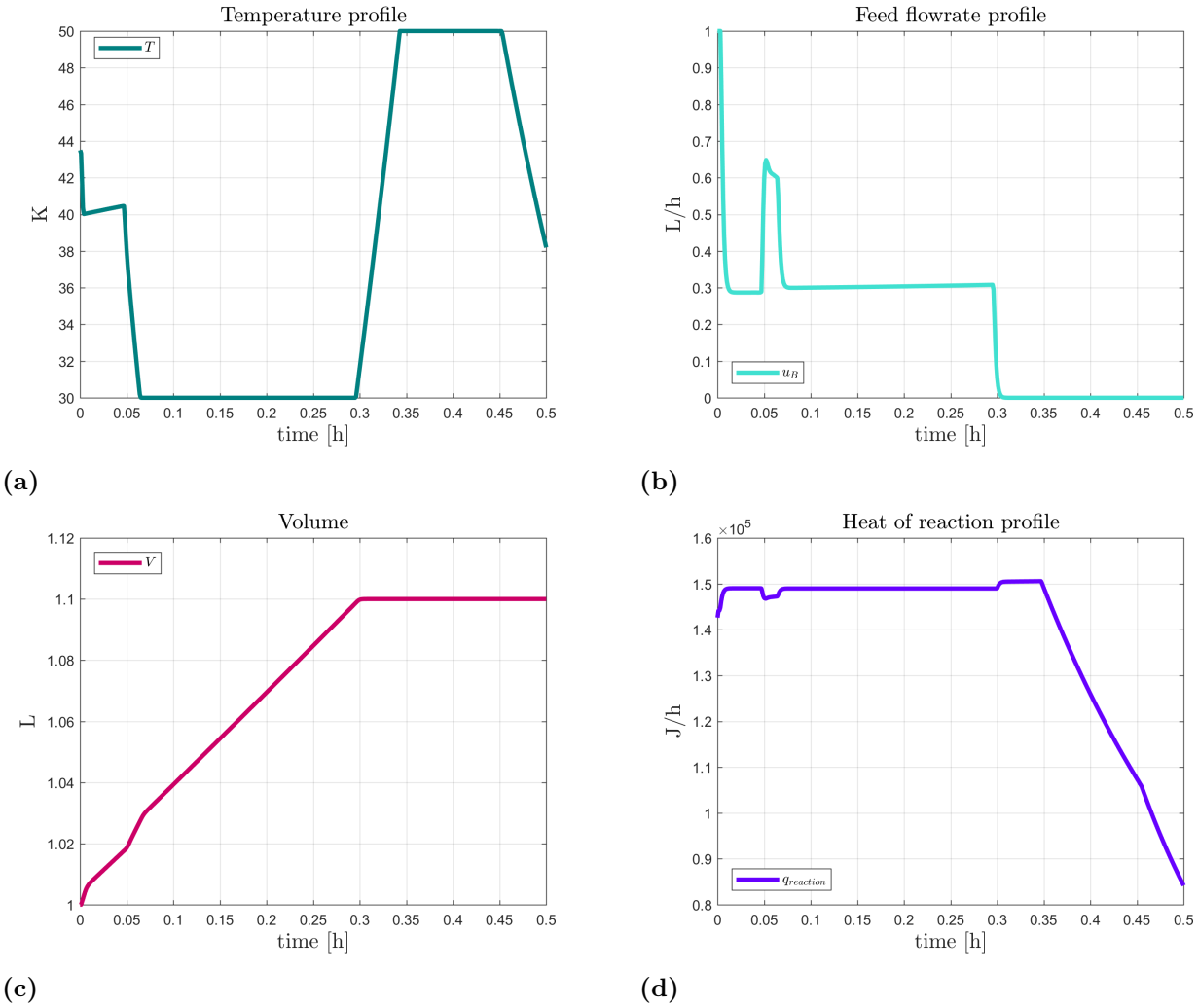


Fig. 4.7: Profiles of temperature (**a**), feed flow rate (**b**), volume (**c**) and heat of reaction (**d**) obtained applying non-linear MPC to the semi-batch reactor system.

4.1.4 AMPC

In order to avoid long computational times, but still requiring good performances, adaptive model predictive control is tested on the system. This simulation is performed using *Simulink* Adaptive Model Predictive Control Toolbox.

First of all, successive linearization is performed at open-loop and the profiles compared with the non-linearized one to check for possible discontinuities or anomalous behaviors 4.8. It is possible to notice from the plot that the nonlinear patterns are well tracked from the linearized ones. On the other hand, the heat of reaction shows a different behavior due to its algebraic nature (equation 4.10).

In the simulation ±2% of white noise was added to the output variables. The results obtained are reported in figure 4.9 (Appendix A.3).

It is possible to notice that the output variables do not violate the constraints, even though

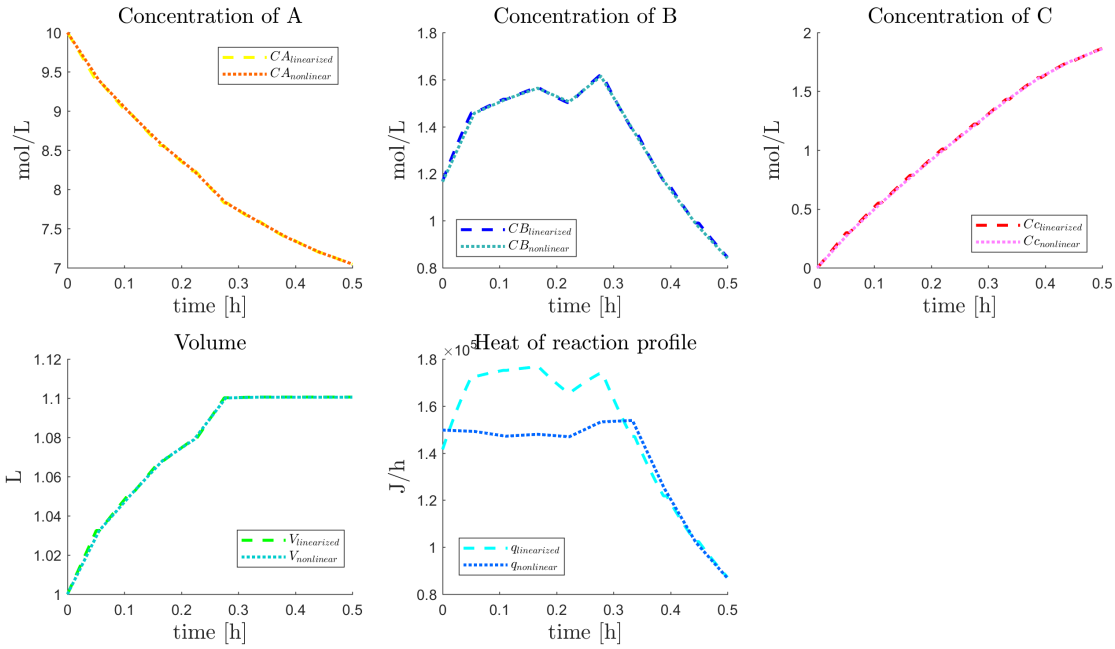


Fig. 4.8: Comparison between nonlinear and successively linearized output profiles.

they are still close to the bound (figure 4.9c and 4.9d). The profiles of the manipulated variables show a path different than the one obtained applying NMPC. In particular, the temperature grows linearly and sets near to the upper bound to favor the production of C. Besides, the feed flow rate of B is at the beginning set to zero to inhibit the side reactions and then it is started to match the volume requirements.

Table 4.6 indicates the amount of product obtained at the final time. While table 4.7 shows the time required to complete the simulation.

Tab. 4.6: Objective function value obtained applying AMPC to the system.

Objective function	
$n_C(t_f)$	1.808 mole

Tab. 4.7: Computational time required to perform the simulation applying AMPC.

Simulation Time	Time per Sampling Time
1.737 minutes	0.208 sec

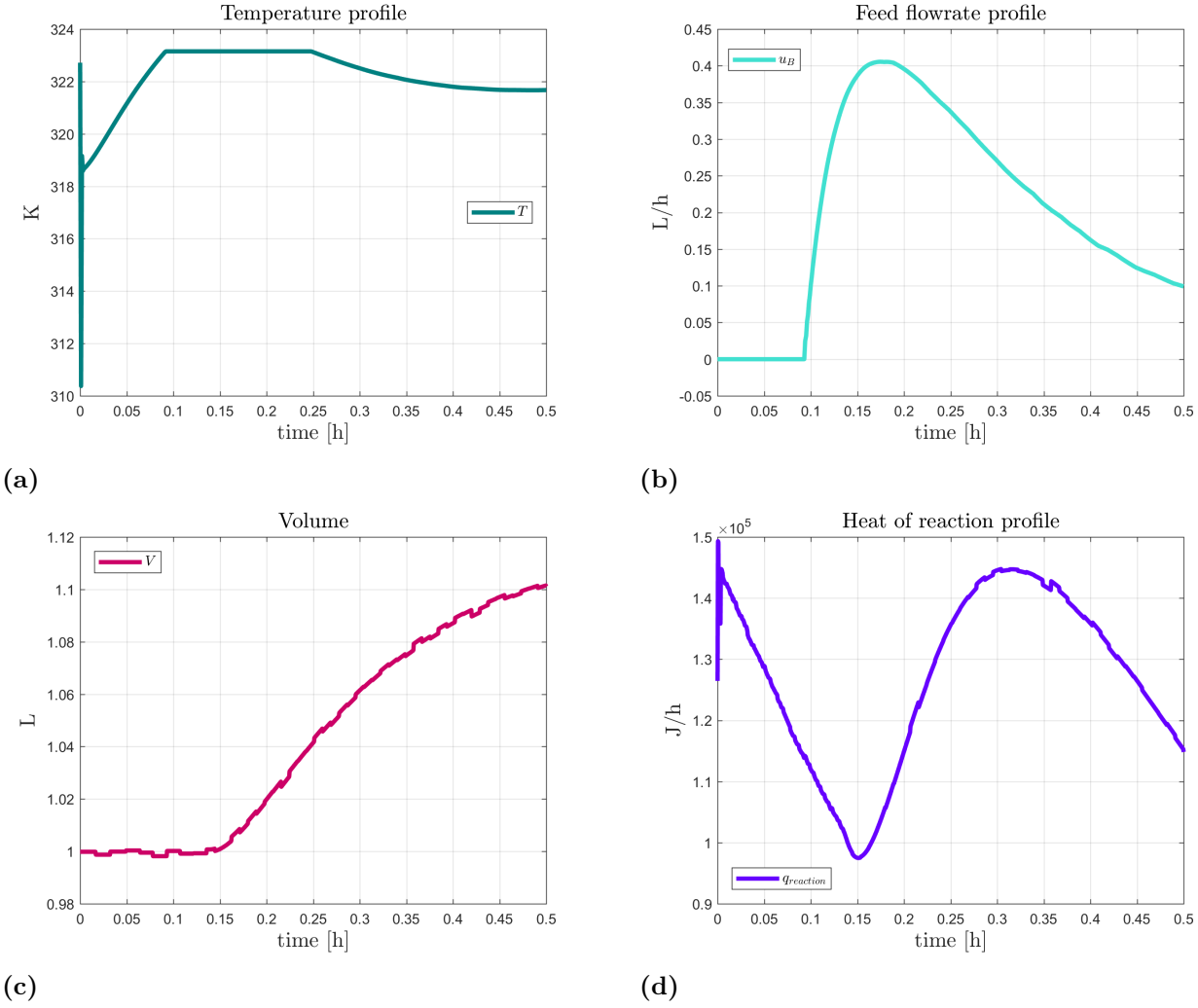


Fig. 4.9: Profiles of temperature (**a**), feed flow rate (**b**), volume (**c**) and heat of reaction (**d**) obtained applying adaprive model predictive control (AMPC) to the semi-batch reactor system.

4.1.5 Results

All the results obtained so far are summarized in table 4.8.

Tab. 4.8: Performance comparison between LMPC, NMPC and AMPC for the semi-batch reactor system.

Controller Comparison			
	Amount of Product [mole]	Simulation Time [min]	Average Simulation time per Sampling Time [sec]
LMPC	1.467	0.101	0.012
NMPC	2.039	7.360	0.8814
AMPC	1.808	1.737	0.208

Tab 4.8 shows that the product performances of AMPC are better than the one of LMPC and comparable with the one of NMPC. Of great interest is the comparison between simulation times. As a matter of fact, the time required to perform the simulation using AMPC is larger than the one necessary for the linear simulation, but it is about 74% lower than the one used for NMPC. This result is of great importance since a fast response is a critical issue when dealing with unsteady-state systems, particularly with the ones involving dangerous reactions. Thus, this is a promising result for application of AMPC to even more complex batch reactor systems.

4.1.6 Process Mismatch

Once AMPC is identified as a promising method to control unsteady-state highly nonlinear processes, some process mismatches are introduced to test the robustness of the controller.

A sensitivity analysis is performed in order to identify the parameters that more affect the outputs when subjected to a fixed input change. In particular k_{01}, k_{02}, E_1, E_2 are identified as "key" parameters for the process model. Successively a $\pm 20\%$ of uncertainty is added to these values. Through a Monte Carlo simulation, 20 realizations with different values of uncertain parameters are implemented in the process model and the system is run in closed-loop. The results are then put together (figure 4.10) and the worst scenario is identified.

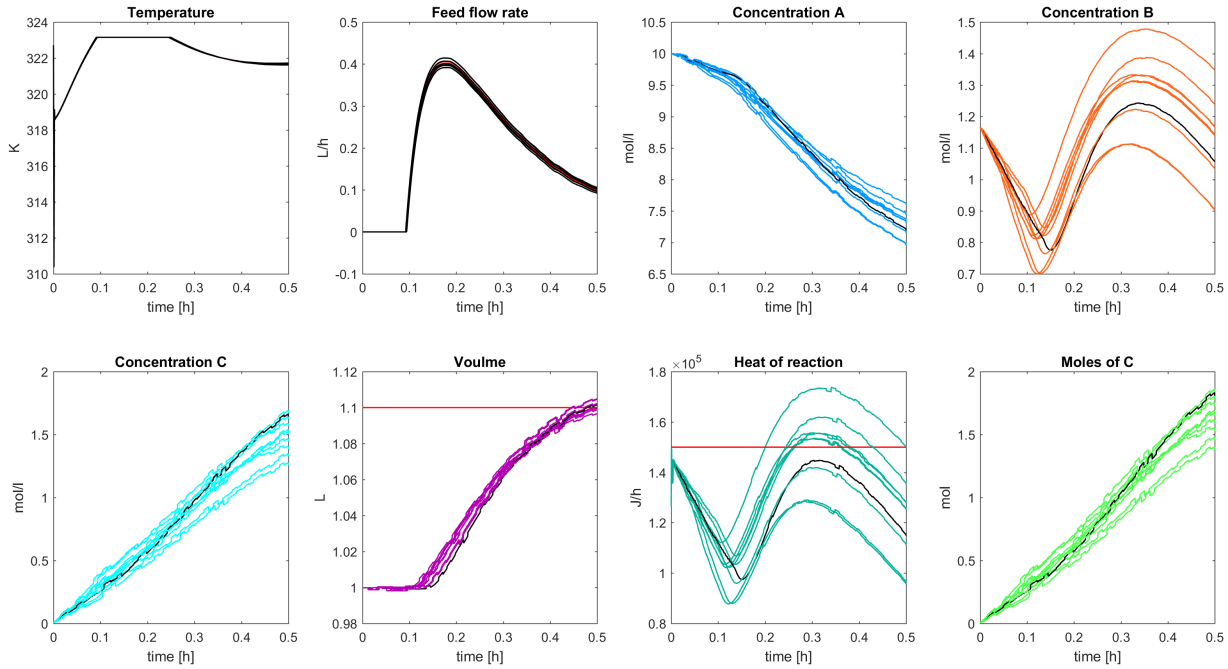


Fig. 4.10: Closed-loop Monte Carlo simulations for 20 uncertainties realizations on the semi-batch reactor system.

In table 4.9 are reported the values of the parameters yielding the worst case scenario, which corresponds to the most severe constraint violations.

The controller is then re-tuned in order to face the worst scenario without violating the constraints. The results found are then reported in table 4.10. It is possible to notice that the new controller gives a lower value of the cost function but the computational time remains faster than the one required for NMPC.

Tab. 4.9: Values of the parameters yielding the worst possible scenario.

Parameter	Value	
k_{01}	3.257	$\mathrm{l/(mol\,h)}$
k_{02}	696.27	$\mathrm{l/h}$
E_1	6.53×10^3	$\mathrm{J/mol}$
E_2	19.78×10^3	$\mathrm{J/mol}$

Tab. 4.10: Values of cost function and simulation times for the re-tuned controller in the case of process mismatch.

Retuned Controller	
Amount of Product [mole]	1.684
Simulation Time [min]	1.954
Average Simulation time per Sampling Time [sec]	0.2345

4.2 Batch Distillation Column

Batch distillation is widely used when small amounts of material are processed at scheduled time frames. The main advantage is flexibility and the possibility to change the composition of the feed. AMPC is applied to a binary batch distillation column (from [6]) and the results compared with the one obtained applying LMPC and NMPC.

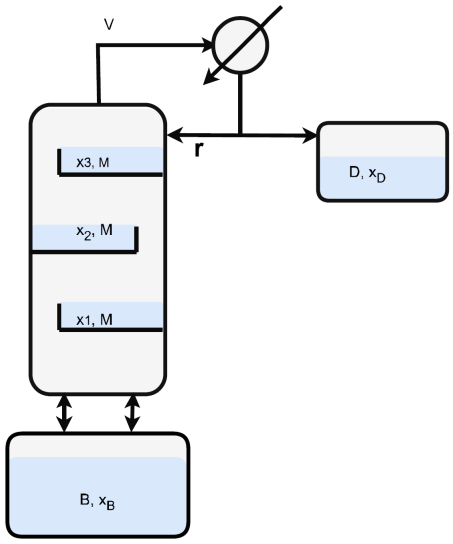


Fig. 4.11: Schematic of a batch distillation column with three trays and reflux ratio as manipulated variable.

No reaction takes place inside the column and a binary mixture is separated. Two constraints are set on the molar fraction of the light component in the distillate and the residue. The only manipulated variable is represented by the reflux ratio, which is bounded between $0 \leq r \leq 1$. The goal of the optimization is to maximize the amount of distillate.
The control problem is described from equations 4.15 - 4.23. In table 4.11 the values of the main parameters are reported.

Tab. 4.11: Values of model parameters, constraints and initial conditions for the batch distillation process.

Vapor flow rate, V	50	kmol/h
Relative volatility, α	2.25	-
Initial charge, B_0	115	kmol
Molar fraction of B in the charge, x_{B0}	0.4	-
Molar hold-up per plate, M	5	kmol
Final time, t_f	3	h

$$Cost\,function: \quad \min_{r(t)} J = D(t_f) \tag{4.15}$$

$$State\,equations: \quad \dot{D} = V(1-r);\ D(0) = 0; \tag{4.16}$$

$$\dot{B} = V(r-1);\ B(0) = B_0; \tag{4.17}$$

$$\dot{x_{\mathrm{B}}} = \frac{V}{B}(x_{\mathrm{B}} - y_{\mathrm{B}} + r(x_1 - x_{\mathrm{B}});\ x_{\mathrm{B}}(0) = x_{\mathrm{B0}}; \tag{4.18}$$

$$\dot{x_{\mathrm{k}}} = \frac{V}{M}(y_{\mathrm{k-1}} - y_{\mathrm{k}} + r(x_{\mathrm{k+1}} - x_{\mathrm{k}}); x_{\mathrm{k}}(0) = x_{\mathrm{B0}}; k = 1,2,3; \tag{4.19}$$

$$\dot{x_{\mathrm{D}}} = \frac{V(1-r)}{D}(y_3 - x_{\mathrm{D}});\ x_{\mathrm{D}}(0) = x_{\mathrm{B0}}; \tag{4.20}$$

$$y_0 = y_{\mathrm{B}};\ x_4 = y_3;\ y_{\mathrm{k}} = \frac{\alpha x_{\mathrm{k}}}{1 + (\alpha - 1)x_{\mathrm{k}}}; k = B,1,..,3; \tag{4.21}$$

$$Output\,costraints: \quad x_{\mathrm{D}}(t_f) \geq 0.8; x_{\mathrm{B}}(t_f) \leq 0.2; \tag{4.22}$$

$$Path\,constraints: \quad 0 \leq r(t) \leq 1 \tag{4.23}$$

The system is tested at open loop keeping the value of the manipulated variable constant at $r = 0.6$. The results show that one constraint is not satisfied. Therefore, a control system is required in order to satisfy the constraints and in the meantime increase the distillate production.

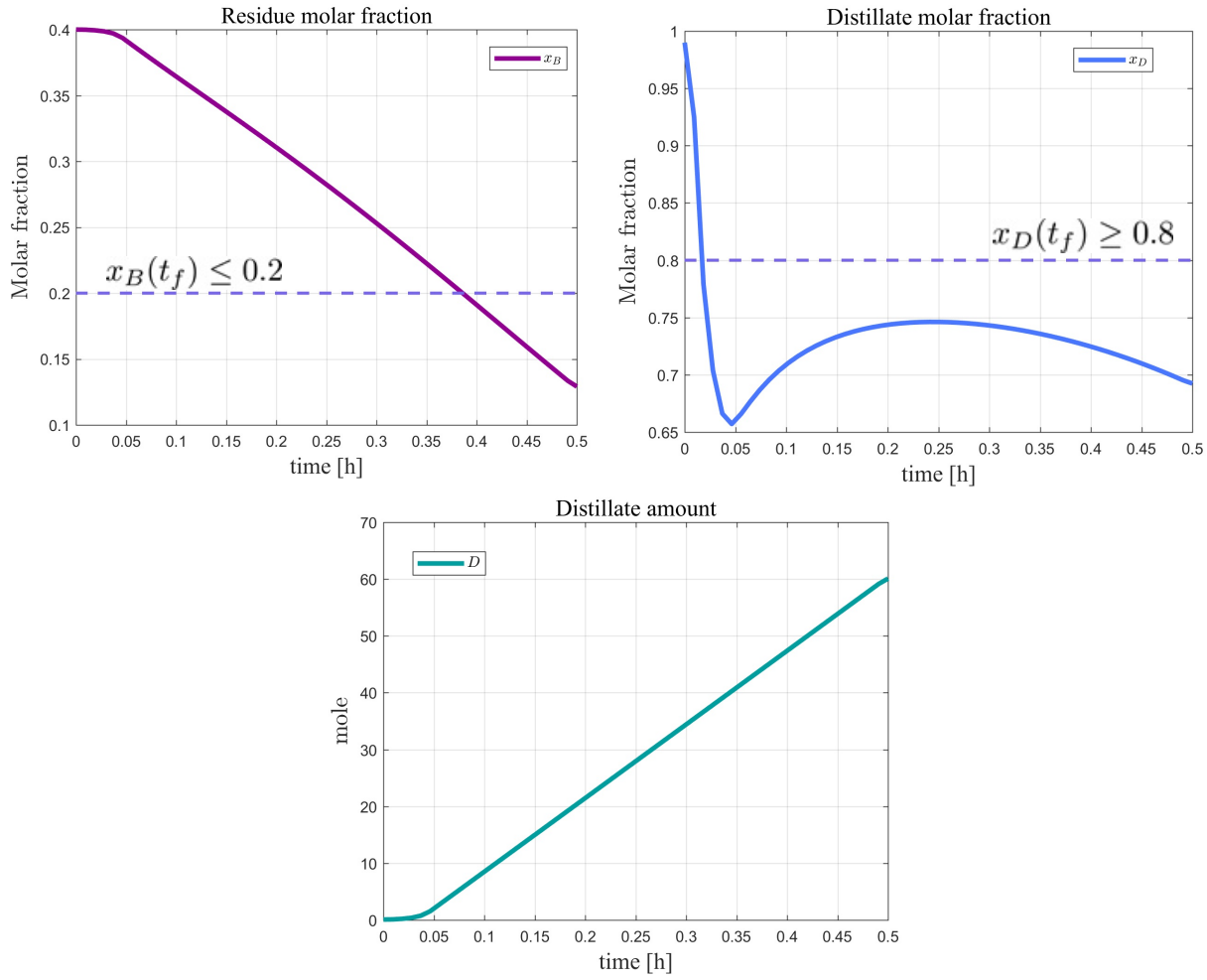


Fig. 4.12: Open-loop behaviour of the process with fixed $r = 0.6$.

4.2.1 Tuning Parameters

As already stated for the batch reactor (§ 4.1.1), full feedback information is assumed. Distillations are generally slow processes. For this reason the sampling time is set as

$$\Delta t = 0.01\,h = 36\,sec. \tag{4.24}$$

From equation 4.22 it is possible to see that the bounds on the output variables are final constraints. Hence, a shrinking horizon approach is implemented. In this way, the whole time horizon is taken into account at each time step, for both prediction and control actions. Even though this approach increases the computational effort, it guarantees a more accurate control and feasibility.

$$prediction\,horizon = 300, \tag{4.25}$$

$$control\,horizon = 300. \tag{4.26}$$

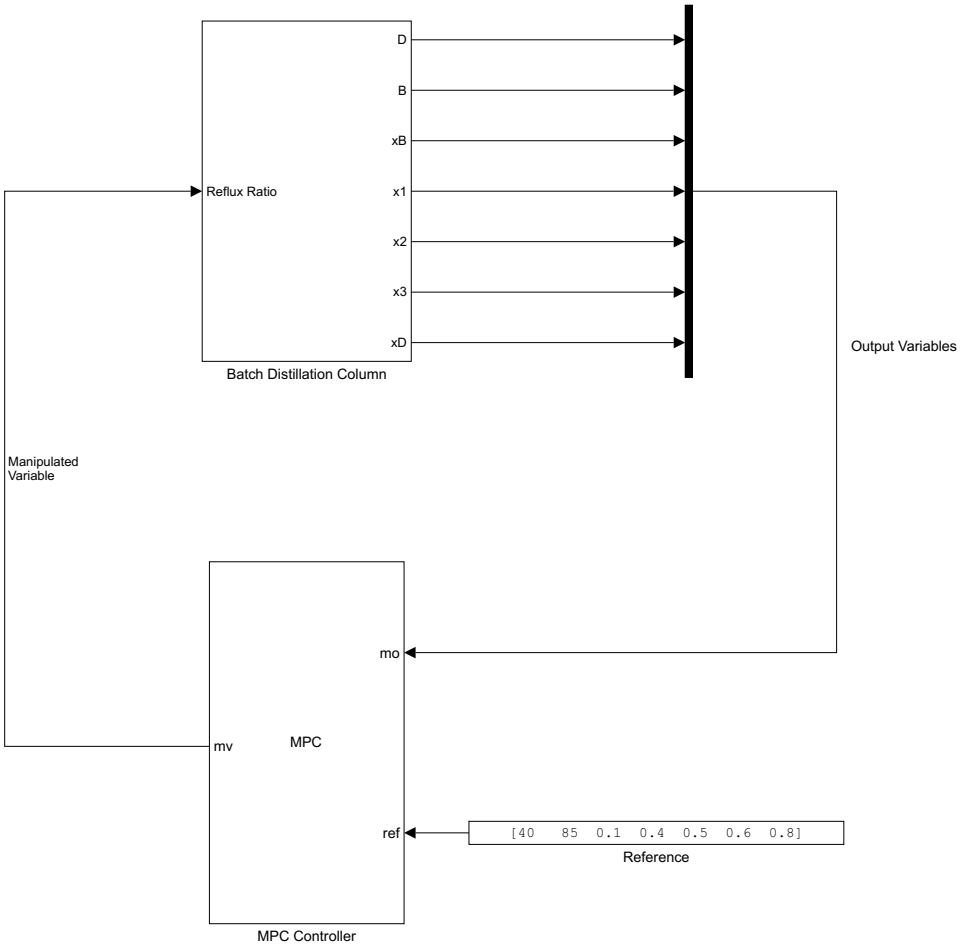


Fig. 4.13: Schematic of the control system.

4.2.2 LMPC

First of all, linear model predictive control is applied to the system. The process model(equations 4.16 - 4.21) is linearized at the initial conditions. Even though the controller is properly tuned it is not possible to achieve the desired specification for the outputs (figure 4.14 and Appendix B.1). It is clear, that, due to the high nonlinearity of the system, a single linearization point is not enough to capture the dynamic behavior of the process.

Since the constraints are not respected the objective value cannot be taken into account for considerations on productivity performances. On the other hand, the time required for the simulation is still of interest for a later comparison(table 4.12).

Tab. 4.12: Computational time required to perform the simulation applying LMPC to the batch distillation column, lnearizing the process model at the initial conditions.

Simulation Time	Time per Sampling Time
7.36 minutes	0.8814 sec

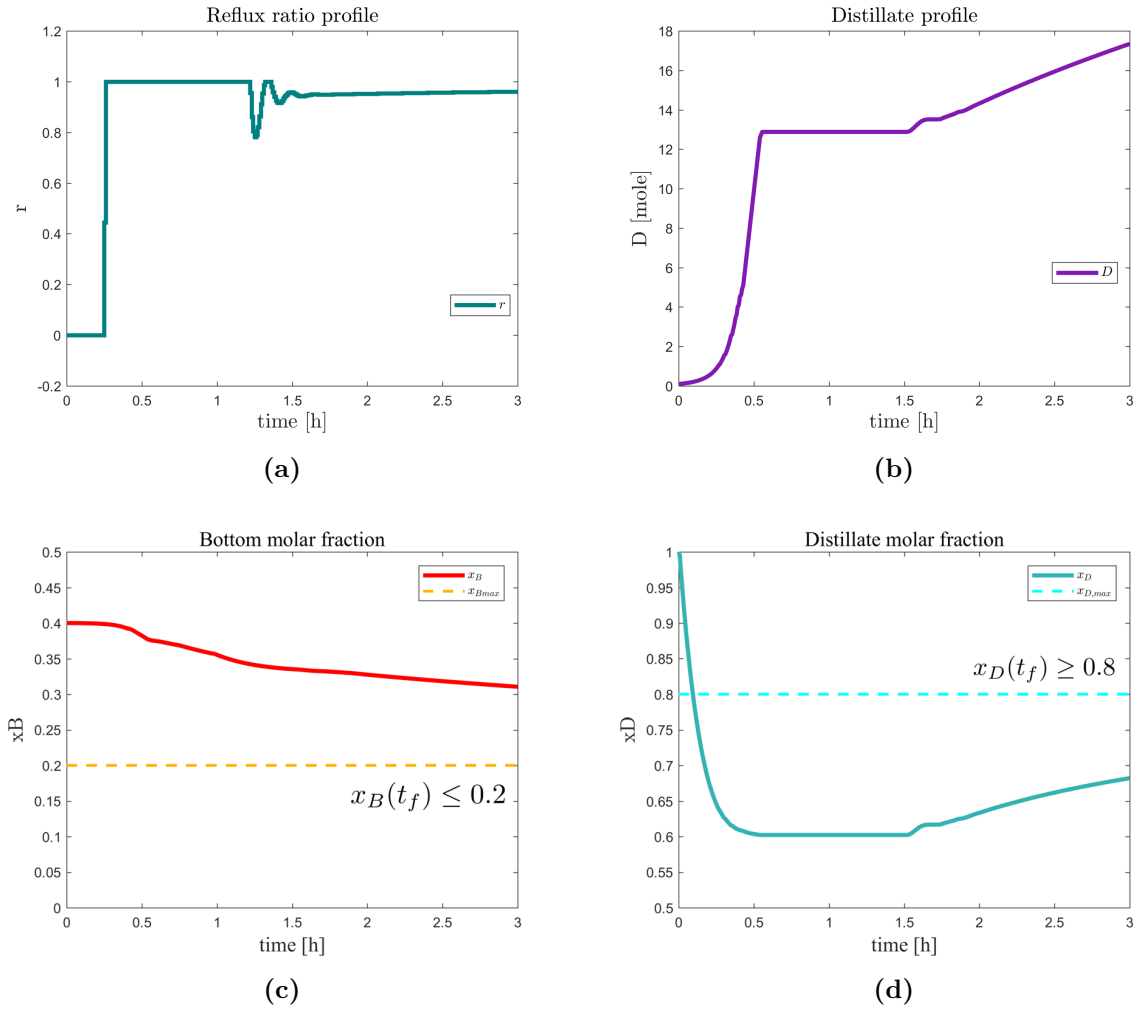


Fig. 4.14: Profiles of reflux ratio (**a**), distillate amount (**b**), bottom composition (**c**) and distillate composition (**d**) obtained applying linear MPC to the batch distillation system and lineazizing the model at the initial conditions.

4.2.3 NMPC

The control problem is solved using nonlinear model predictive control.

Figure 4.16 shows the results obtained from the simulation (Appendix B.2). It is possible to notice that the final constraint requirements are satisfied. In fact, the distillate composition is greater than $x_D \geq 0.8$ and the bottom one lower than $x_B \leq 0.2$.
At the beginning, the reflux ratio (figure 4.16a) is the maximum possible. All the condensate is sent back in the column in order to increase the purity of the distillate. Therefore, in this time frame, the concentration of the light component in the bottom continuously decrease because it evaporates. After 0.5 hours the value of the reflux ratio is lowered and then kept almost constant to maximize the value of the cost function. Due to the shrinking horizon approach, the behavior of the controller becomes more aggressive and the profile shows some oscillations approaching the final time. Decreasing the sampling time could avoid some oscillations but then the computational time would increase.
The closed-loop profile of r obtained is comparable with the open-loop behavior reported in the literature (fig 4.15).

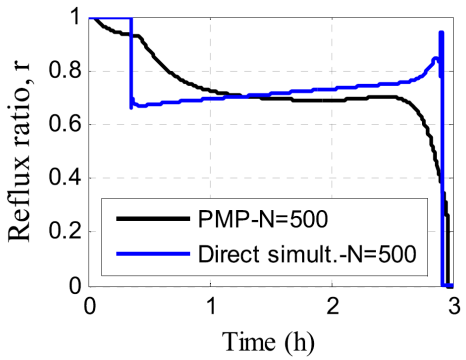


Fig. 4.15: Open-loop reflux ratio profile obtained applying PMP method and direct simultaneous method (From *Aydin et al., 2017)*

The optimal distillation amount produced is 39.02 moles and is lower than the one obtained at open-loop. Thus, at each time step, the optimization problem is started at different conditions, sometimes leading to suboptimal solutions.
The time required to complete the simulation is of 12 minutes (table 4.14), which corresponds to an average of 2.39 seconds each time step. This time is still lower than the sampling time Δt but it introduces a feedback delay which cannot be neglected.

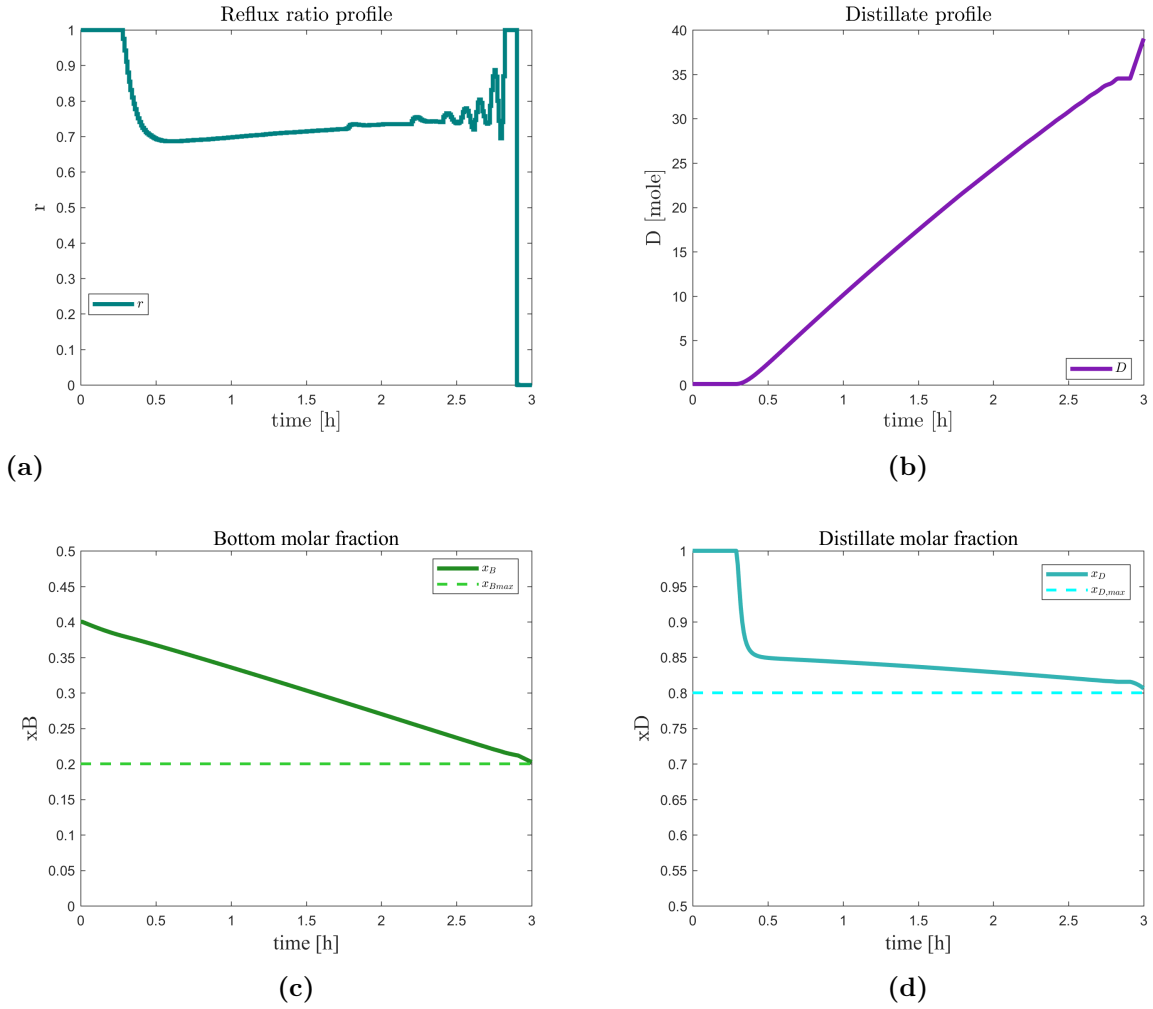


Fig. 4.16: Profiles of reflux ratio (**a**), distillate amount (**b**), bottom composition (**c**) and distillate composition (**d**) obtained applying nonlinear MPC to the batch distillation system.

Tab. 4.13: Optimal cost function value obtained applying NMPC to the batch distillation system.

Objective function	
D	39.02 mole

Tab. 4.14: Computational time required to perform the simulation applying NMPC to the batch distillation system.

Simulation Time	Time per Sampling Time
12 minutes	2.39 sec

4.2.4 AMPC

The core of adaptive model predictive control is the successive linearization of the nonlinear model. Therefore an open loop successive linearization is performed using the optimal input profile found applying NMPC(fig. 4.16a). In this way, it is possible to check how the nonlinear behavior is tracked.

Figure 4.17 illustrates that the linearized model tracks the nonlinear one, but it also shows that some discrepancies are present concurrently with the oscillations of the input variable.

The adaptive scheme is applied to the system introducing a $\pm 2\%$ of noise to the output variables. The profile of the reflux ratio obtained (fig 4.18b) shows a path similar to the one coming from NMPC. In the beginning, the reflux is almost total to maintain the purity of the distillate. After 15 minutes it is lowered to increase the distillate production. At the end of the batch time, the specifications on the bottom and top compositions are satisfied.

At the final time, the amount of distillate produced is of $D = 38.117$ mole and the time required for the simulation is $t_{simulation} = 2.40$ minutes (table 4.16).

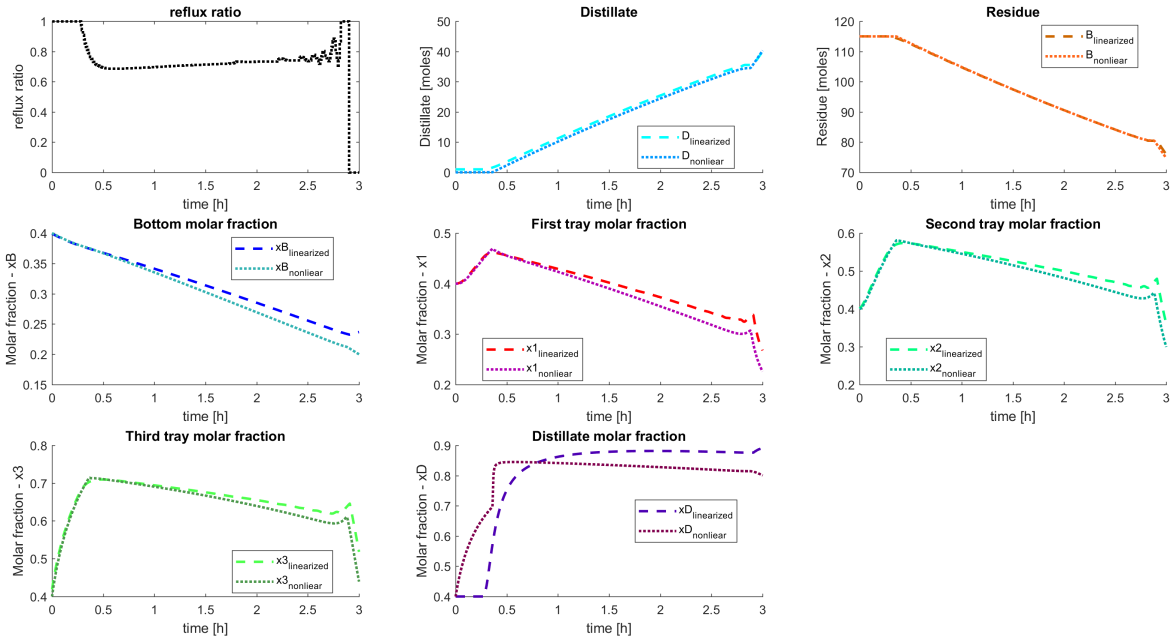


Fig. 4.17: Successive linearization of the nonlinear model profiles coming from the closed loop simulation applying NMPC to the batch distillation system.

Tab. 4.15: Objective function value obtained applying AMPC to the batch distillation system.

Objective function	
D	38.117 mole

4.2.5 Results

In table 4.17 all the performances of the controllers are reported. The linear model predictive controller shows to be unable to regulate the system. The process nonlinearity is too high to be accurately described by a model linearized in a single point. The maximum value of distillate is obtained applying NMPC. Nevertheless, the distillate amount retrieved with AMPC is comparable to the maximum one, since it is only 2.3% less than the one resulting from NMPC application.

Linear MPC remains as a faster method to solve the problem, but AMPC shows to be competitive. The adaptive time to perform the simulation is -88% less than the one required for NMPC.

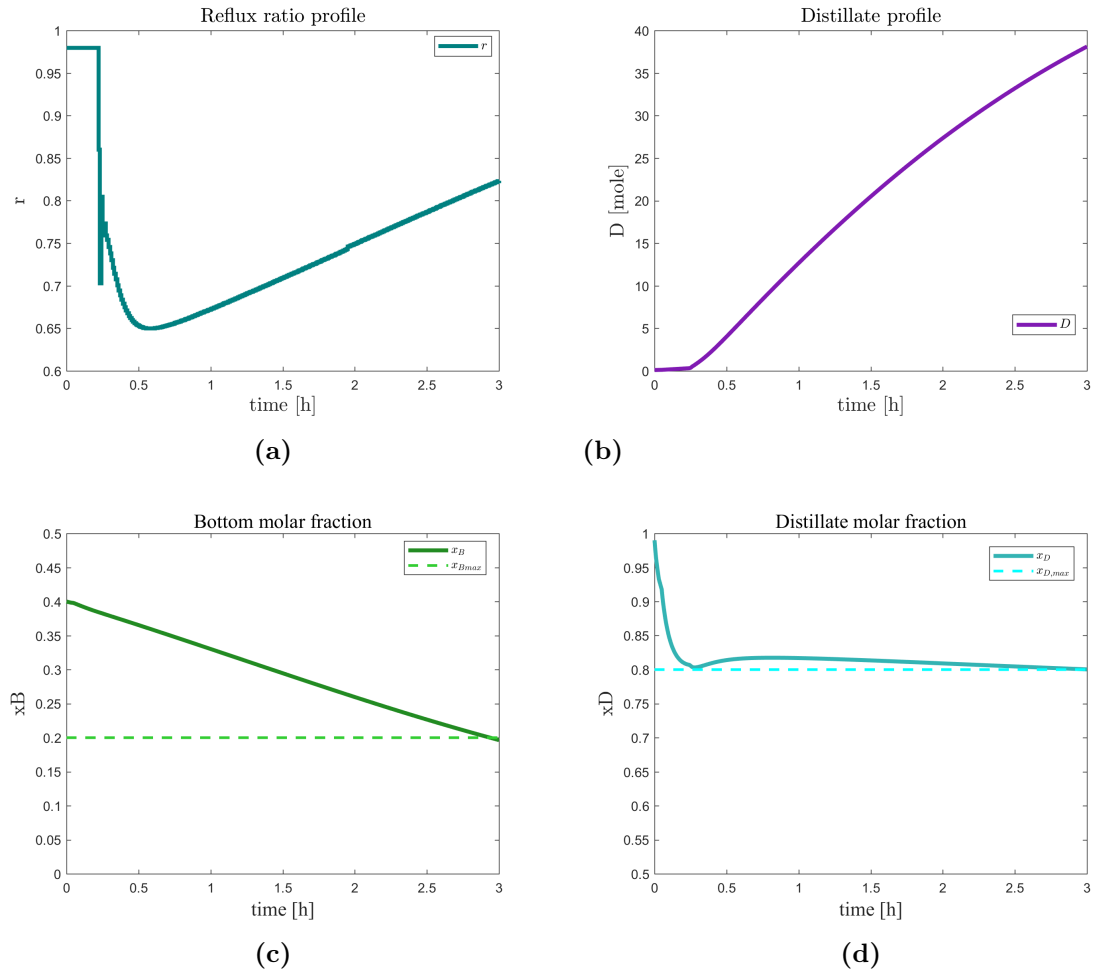


Fig. 4.18: Profiles of reflux ratio (**a**), distillate amount (**b**), bottom composition (**c**) and distillate composition (**d**) obtained applying AMPC to the batch distillation system.

Tab. 4.16: Computational time required to perform the simulation applying AMPC to the batch distillation system.

Simulation Time	Time per Sampling Time
2.40 minutes	0.5316 sec

Tab. 4.17: Value of the optimal cost, simulation time and average simulation time per sampling time for the different controllers applied to the batch distillation system.

Controller Comparison			
	Amount of Product [mole]	Simulation Time [min]	Average Simulation time per Sampling Time [sec]
LMPC	Infeasible	1.35	0.269
NMPC	39.02	12	2.392
AMPC	38.12	2.40	0.532

4.2.6 Process Mismatch

To test the robustness of the controller some process mismatch is applied to the process. In particular, an uncertainty of $\pm 5\%$ is added to the volatility parameter α. A Monte Carlo simulation is performed at open-loop in order to identify the worst possible scenario (fig 4.19). As expected, lowering the volatility of -5% yields the biggest final constraints violation. In this case, the controller is not re-tuned but the final time is let free. When the time is not fixed, more flexibility is given to the system, allowing the controller to regulate the system better. The simulation is then stopped when the constraints are satisfied. Table 4.18 reports the value of the new final time. Figure 4.20 indicates the profiles of the input and the outputs with final time free to change. It appears that, even though more flexibility is given to the system, one constraint is slightly violated in some cases. This result implies that some recursive tuning would be necessary in order to increase robustness.

Tab. 4.18: Final time at which the simulation is stopped in case of process mismatch.

Final Time	
t_f	3.36 h

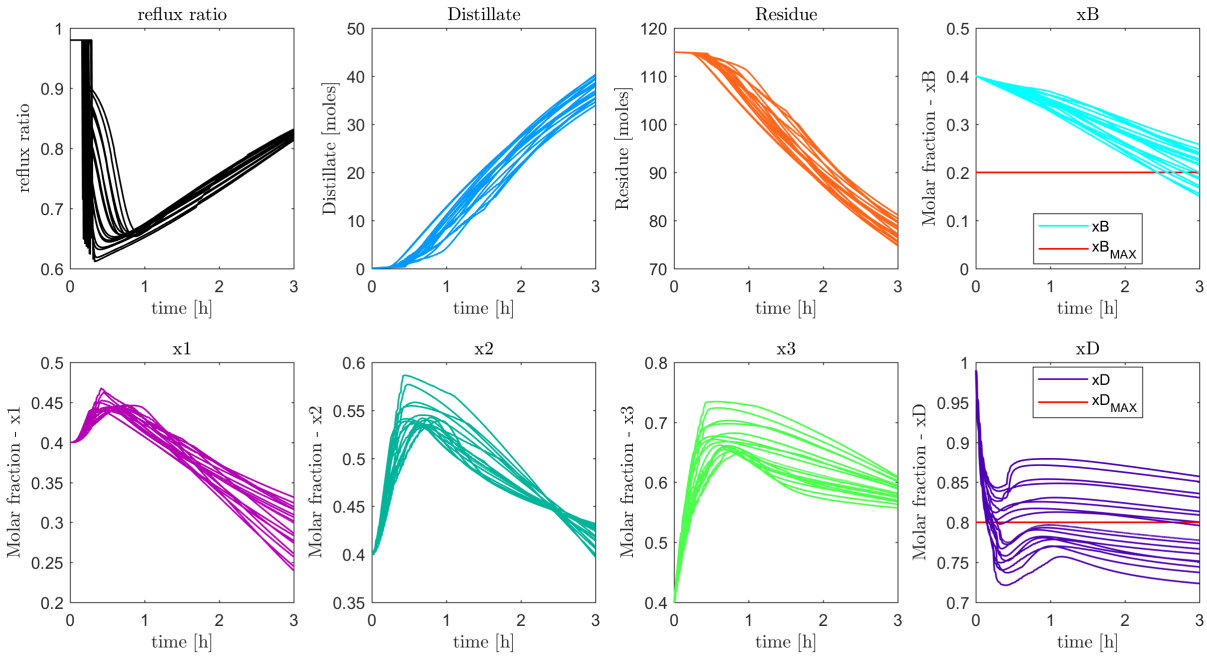


Fig. 4.19: Closed-loop Monte Carlo simulations for 20 uncertainties realizations.

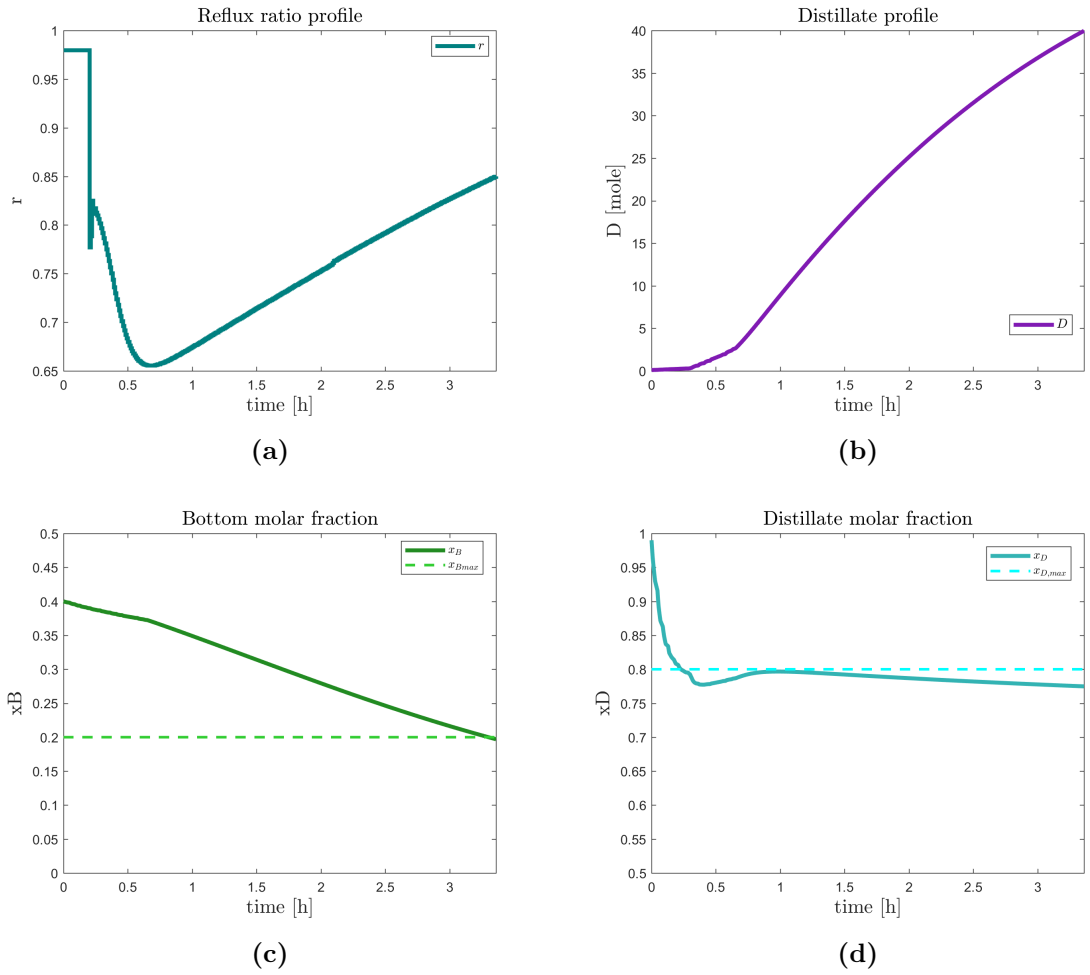


Fig. 4.20: Profiles of reflux ratio (**a**), distillate amount (**b**), bottom composition (**c**) and distillate composition (**d**) obtained letting the final time free.

Conclusions

As the scope of this thesis, Adaptive Model Predictive Control was applied to batch processes and its performance is compared with the one obtained using linear and nonlinear Model Predictive Control.

Due to the high competitiveness on the market and to strict safety regulations, batch and semi-batch operations are gaining lots of importance. Based on their unsteady-state nature, this processes are characterized by highly nonlinear state equations. Thus, they are difficult to control. Nevertheless, Model Predictive Control was introduced as a valid choice for control this transient systems and at the same time optimize them.

In this work, AMPC method was applied to two case studies: a semi-batch reactor and a batch binary distillation column. In both cases, AMPC has shown promising results. The value of the optimal cost was in both cases higher than the one coming from LMPC but lower that the one from NMPC. Though, the costs were comparable and the difference lower than the 11%. The more promising result is the one concerning the computational time. In fact, it was shown that AMPC requires a simulation time up to 88% lower than the one required from NMPC. This result is of great importance since the high computational time is one of the biggest issue.

To test the robustness of AMPC parametric uncertainties were added to the process and multiple realizations performed through a Monte Carlo simulation to identify the worst possible scenario. It turned out that, in the presence of process mismatch higher than about 5%, the constraints are not satisfied anymore. For this reason, recursive tuning might be necessary to increase robustness.

After this study, the benefits of the application of AMPC on batch operations has been found significant. The computational time is significantly reduced, while maintaining a relatively closer optimal cost. Finally, this work has pointed out that future investigations are required to increase the capability of AMPC to deal with parametric uncertainties.

APPENDIX

A

Semi-batch Reactor

A.1 LMPC

Tab. A.1: Weights applied on output and manipulated variables using LMPC.

Weights on Output Variables		Weights on Manipulated Variables	
C_A	0	T	0
C_B	0	u	0
C_C	50		
V	100		
q_{rx}	500		

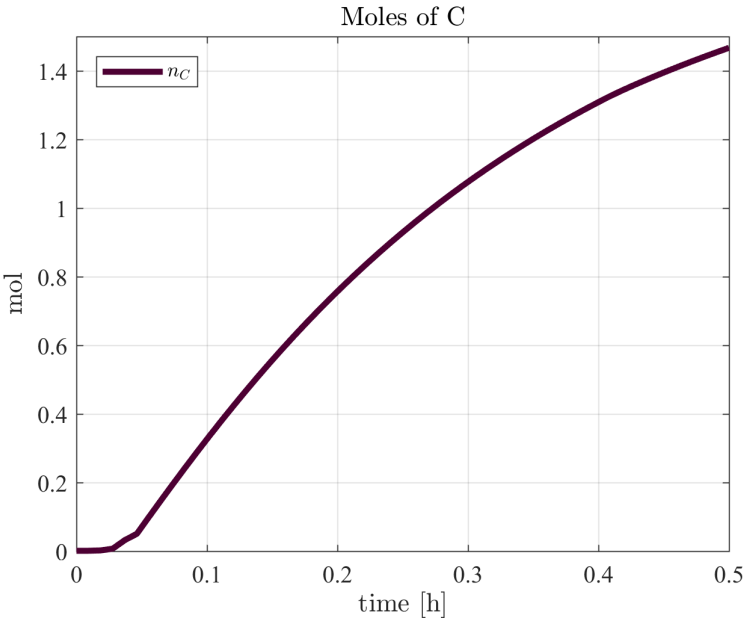


Fig. A.1: Moles of C producted with respect to time, obtained applying LMPC to the batch reactor system.

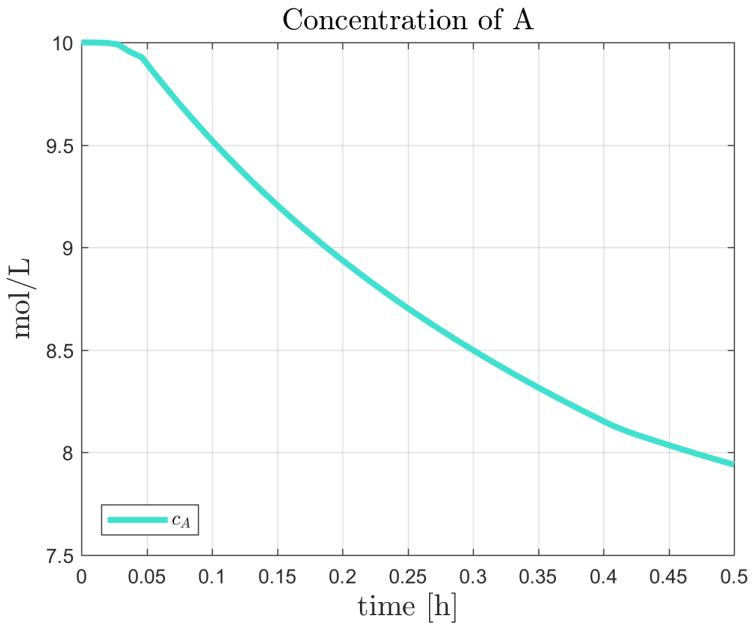


Fig. A.2: Concentration of specie A profile with respect to time, obtained applying LMPC to the batch reactor system.

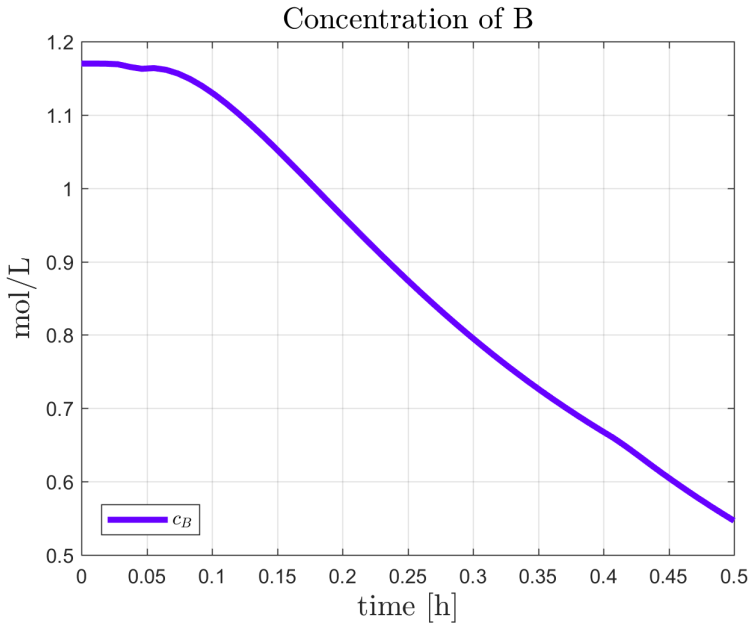


Fig. A.3: Concentration of specie B profile with respect to time, obtained applying LMPC to the batch reactor system.

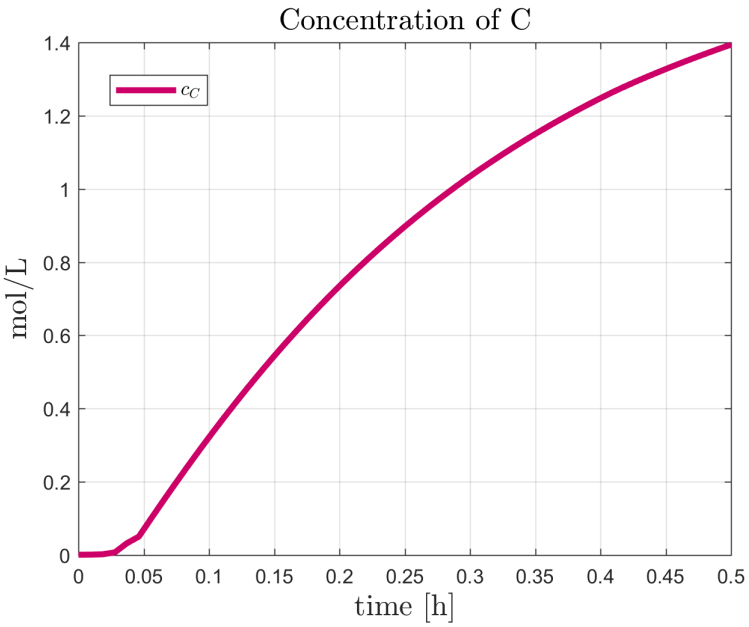


Fig. A.4: Concentration of specie C profile with respect to time, obtained applying LMPC to the batch reactor system.

A.2 NMPC

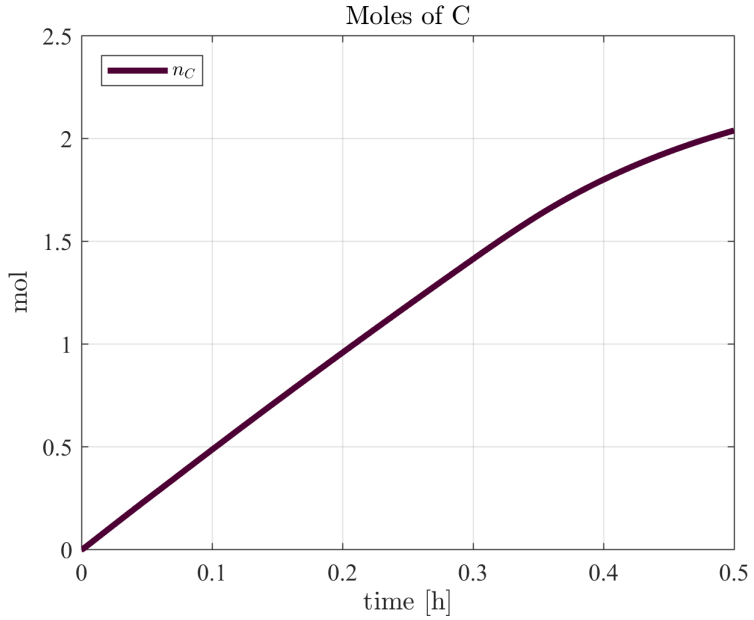


Fig. A.5: Moles of C producted with respect to time, obtained applying NMPC to the batch reactor system.

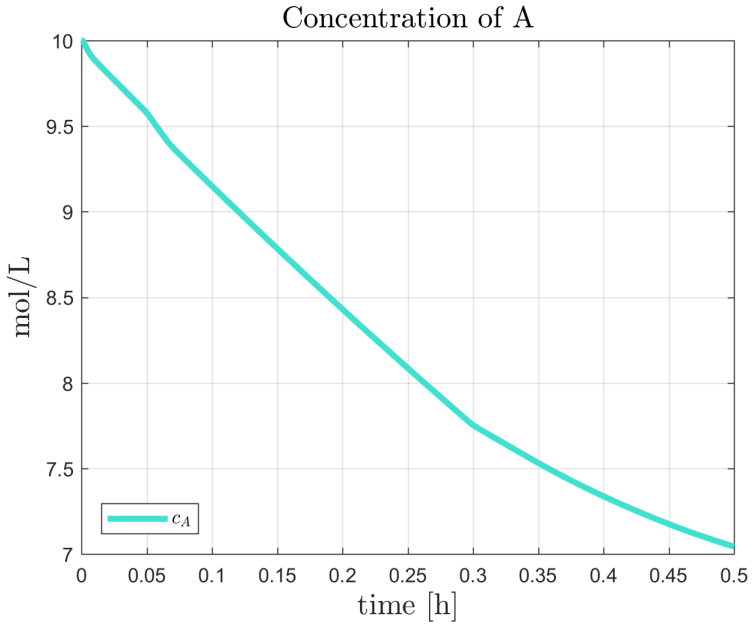


Fig. A.6: Concentration of specie A profile with respect to time, obtained applying NMPC to the batch reactor system.

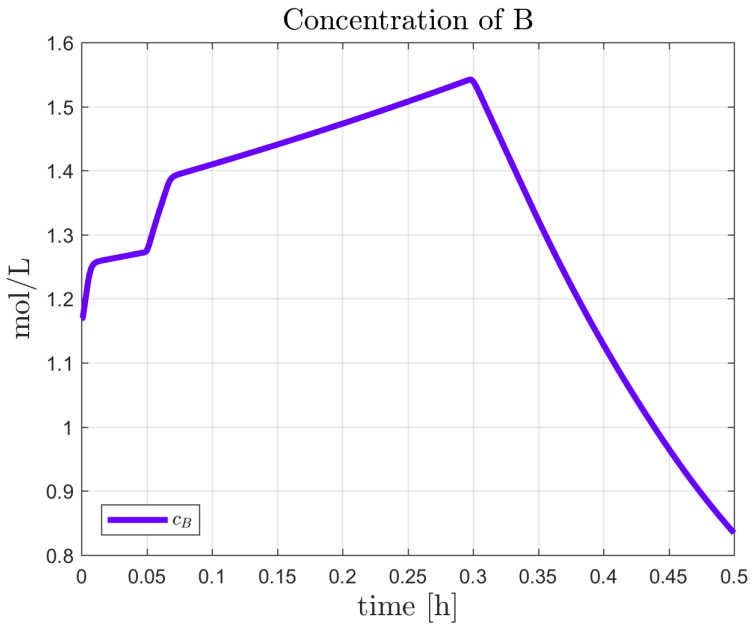


Fig. A.7: Concentration of specie B profile with respect to time, obtained applying NMPC to the batch reactor system.

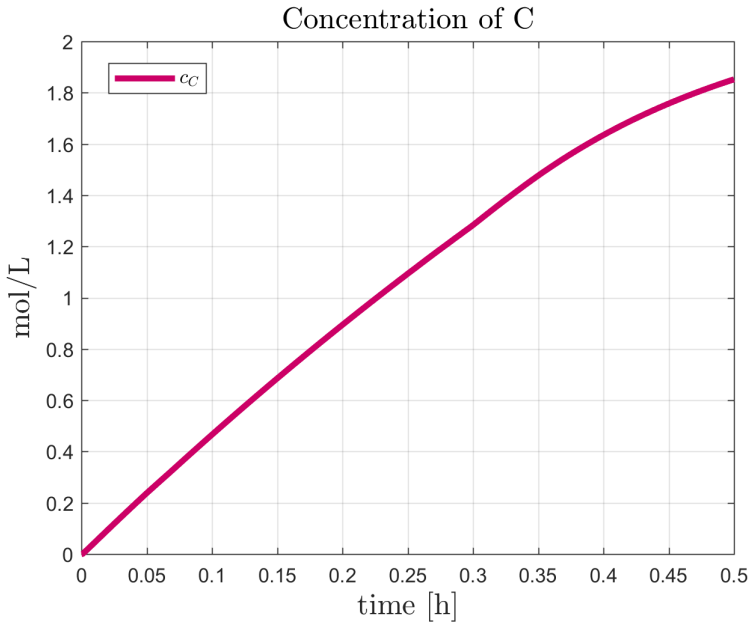


Fig. A.8: Concentration of specie C profile with respect to time, obtained applying NMPC to the batch reactor system.

A.3 AMPC

Tab. A.2: Weights applied on output and manipulated variables using AMPC.

Weights on Output Variables		Weights on Manipulated Variables	
C_A	0	T	50
C_B	0	u	50
C_C	0		
V	91		
q_{rx}	800		

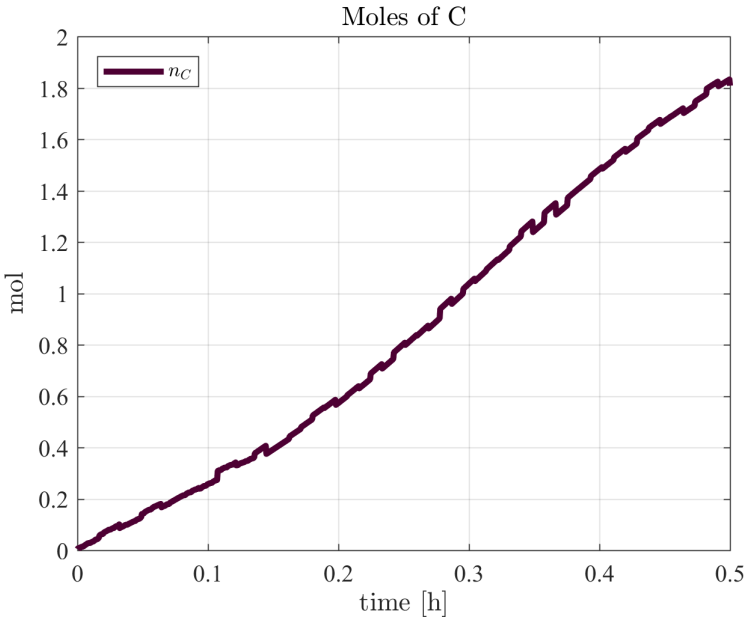


Fig. A.9: Moles of C producted with respect to time, obtained applying AMPC to the batch reactor system.

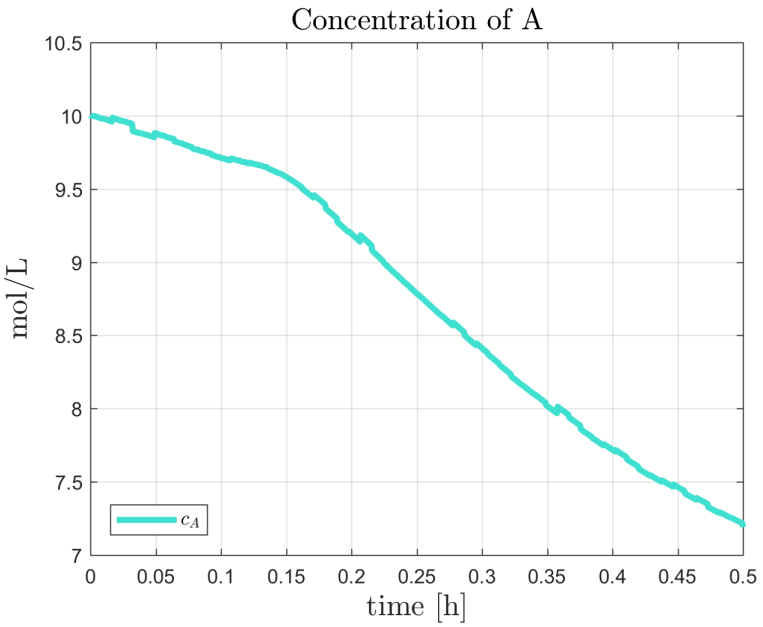
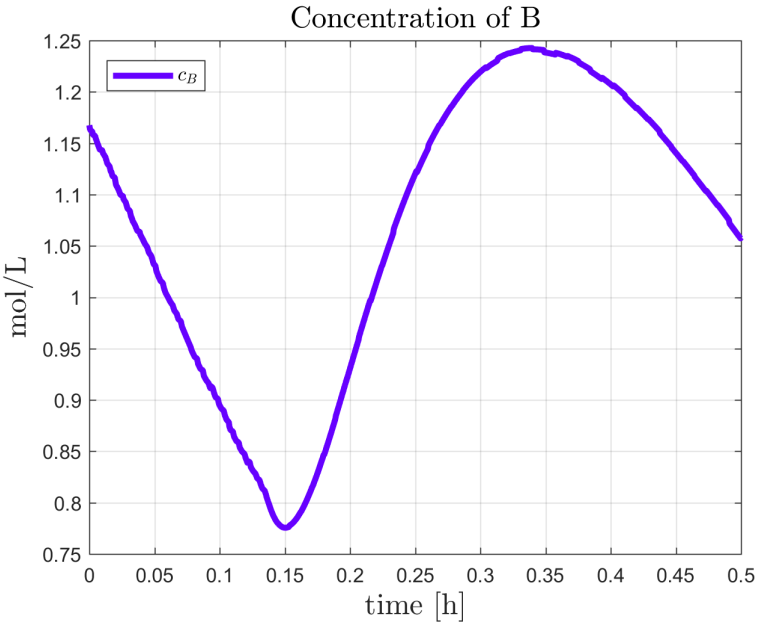


Fig. A.10: Concentration of specie A profile with respect to time, obtained applying AMPC to the batch reactor system.



]

Fig. A.11: Concentration of specie B profile with respect to time, obtained applying AMPC to the batch reactor system.

A.4 AMPC - Process Mismatch

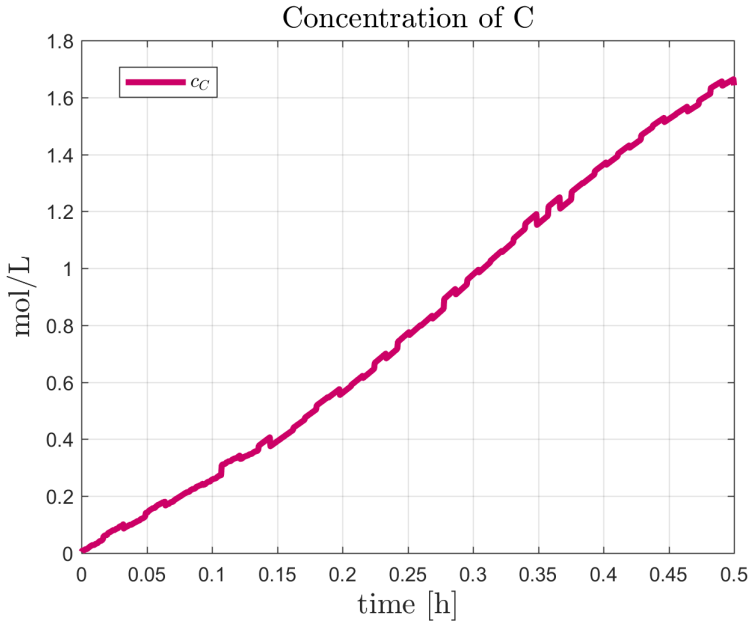


Fig. A.12: Concentration of specie C profile with respect to time, obtained applying AMPC to the batch reactor system.

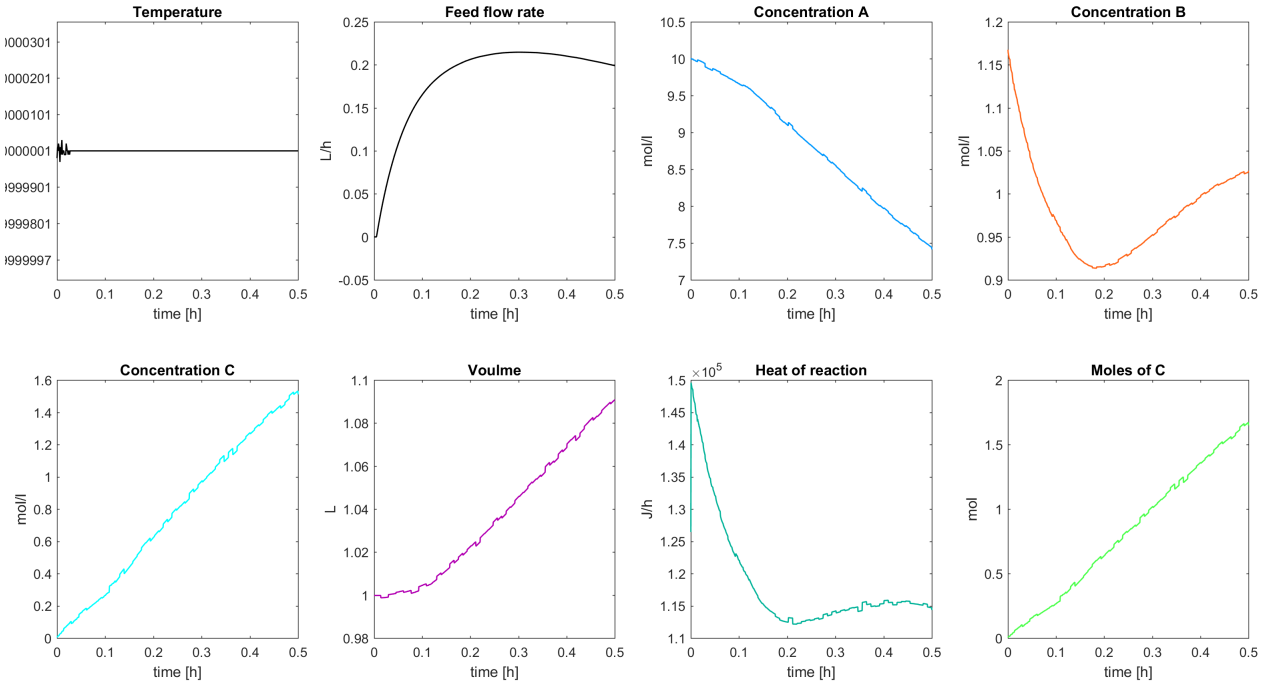


Fig. A.13: Manipulated and output variables profile obtained re-tuning the controller in presence of process mismatch.

B

Batch Distillation Column

B.1 LMPC

Tab. B.1: Weights applied on output and manipulated variables using LMPC.

Weights on Output Variables		Weights on Manipulated Variables	
D	500	r	50
B	0		
x_B	1		
x_1	1		
x_2	1		
x_3	50		
x_D	1		

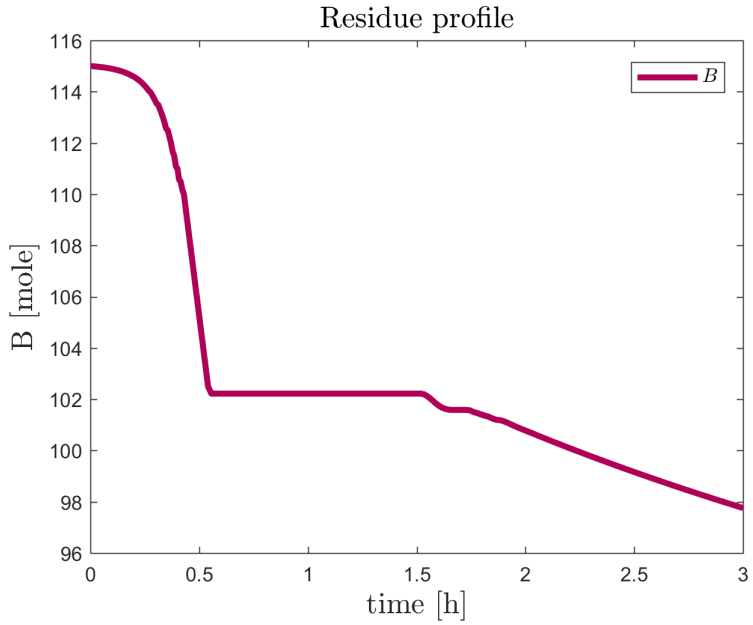


Fig. B.1: Profile of the residue amount B with respect to time, obtained applying LMPC to the batch distillation column.

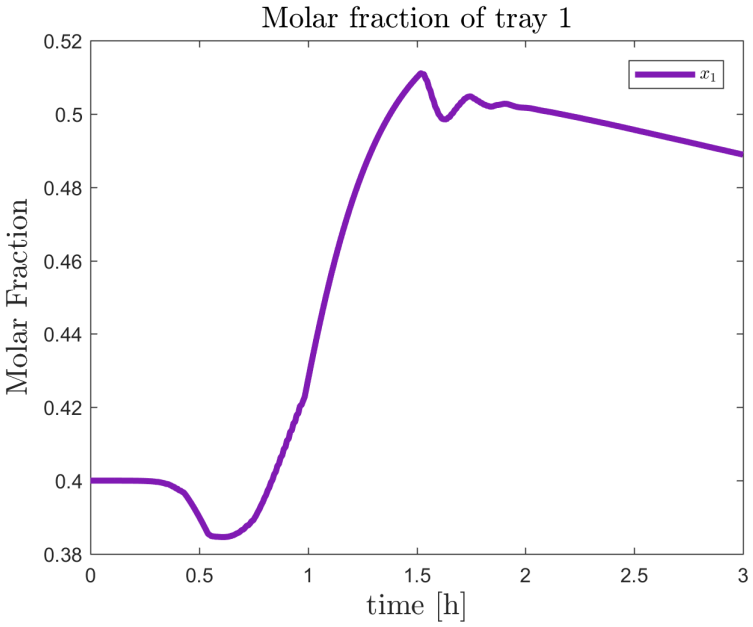


Fig. B.2: Molar fraction of light component on the first tray obtained applying LMPC to the batch distillation column.

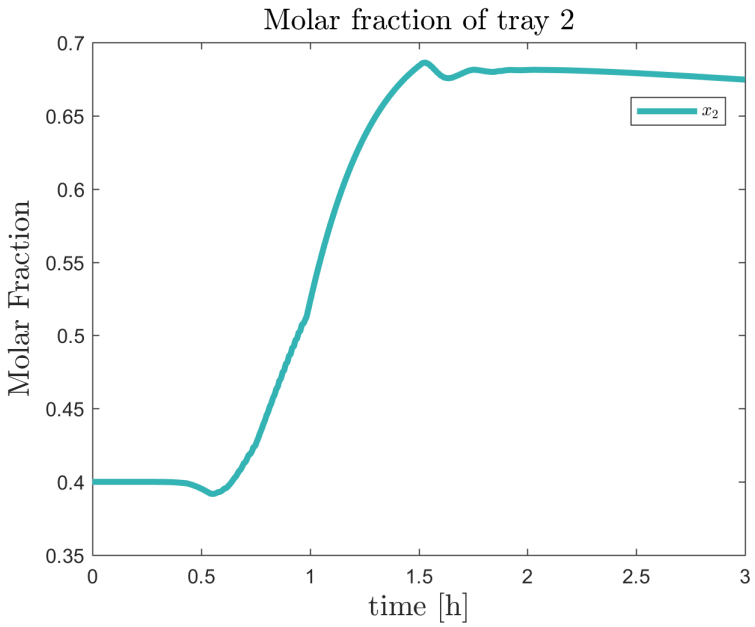


Fig. B.3: Molar fraction of light component on the second tray obtained applying LMPC to the batch distillation column.

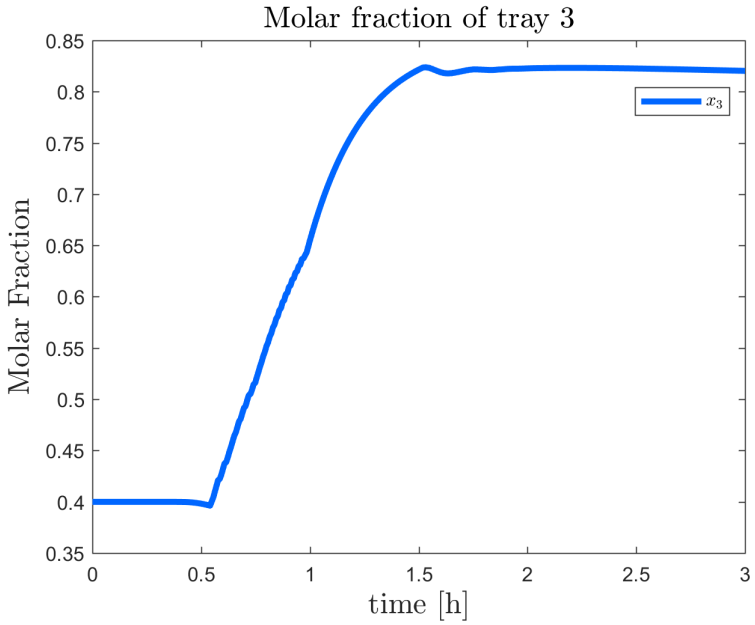


Fig. B.4: Molar fraction of light component on the third tray obtained applying LMPC to the batch distillation column.

B.2 NMPC

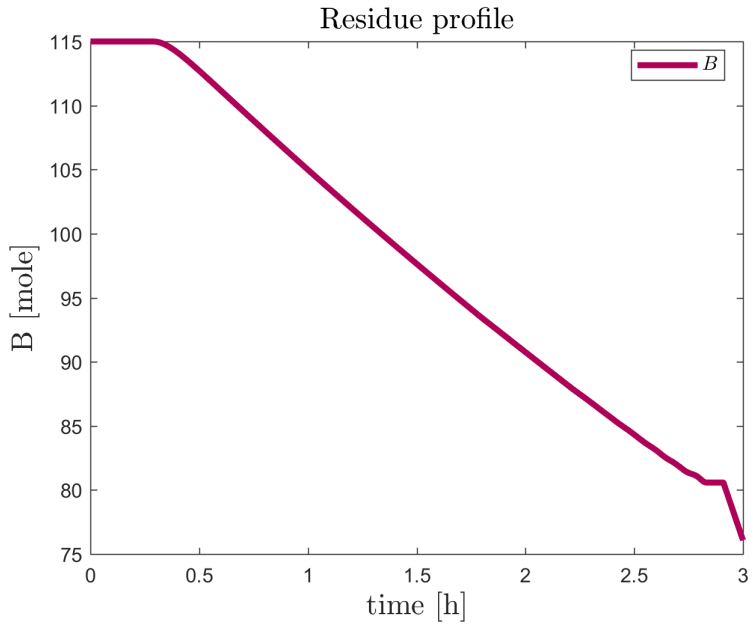


Fig. B.5: Profile of the residue amount B with respect to time, obtained applying NMPC to the batch distillation column.

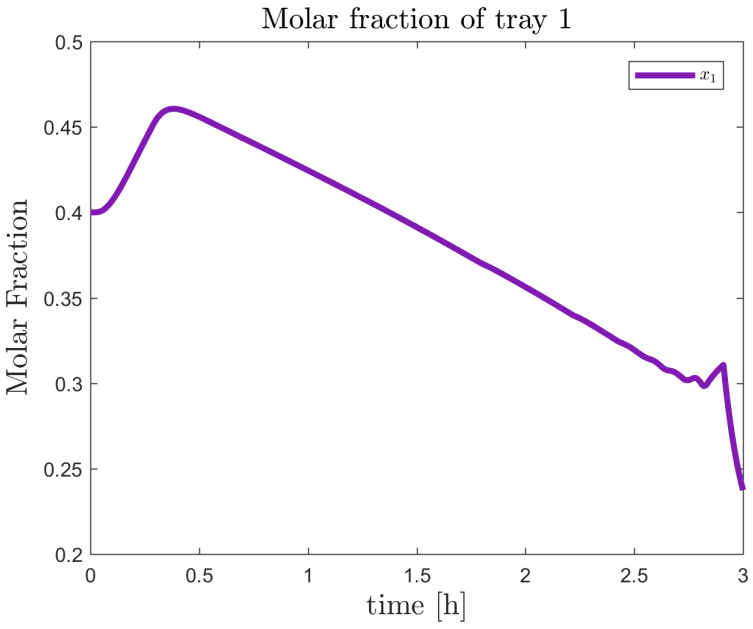


Fig. B.6: Molar fraction of light component on the first tray obtained applying NMPC to the batch distillation column.

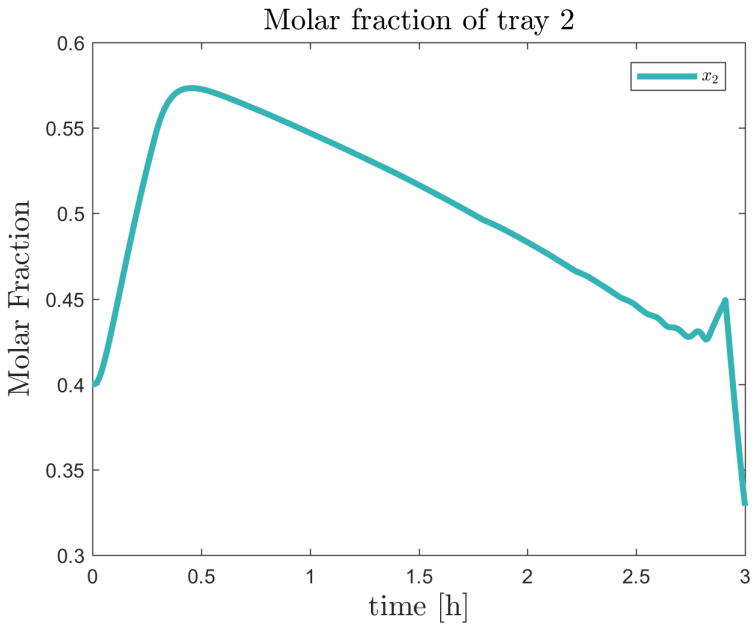


Fig. B.7: Molar fraction of light component on the second tray obtained applying NMPC to the batch distillation column.

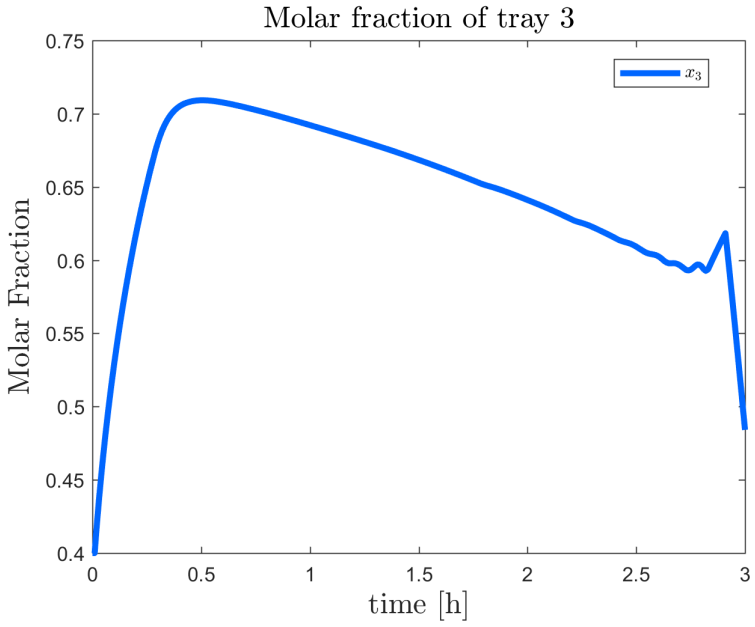


Fig. B.8: Molar fraction of light component on the third tray obtained applying NMPC to the batch distillation column.

B.3 AMPC

Tab. B.2: Weights applied on output and manipulated variables using LMPC.

Weights on Output Variables		Weights on Manipulated Variables	
D	40	r	50
B	0		
x_B	1		
x_1	200		
x_2	200		
x_3	200		
x_D	500		

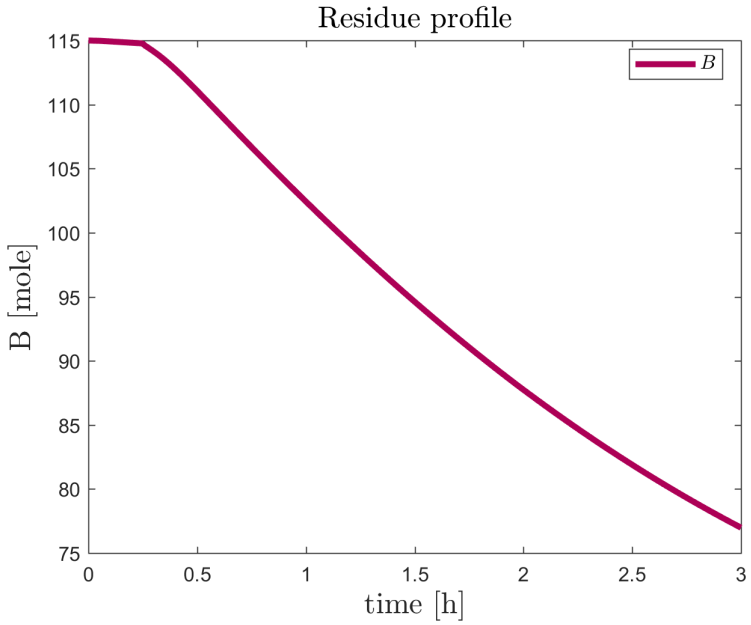


Fig. B.9: Profile of the residue amount B with respect to time, obtained applying AMPC to the batch distillation column.

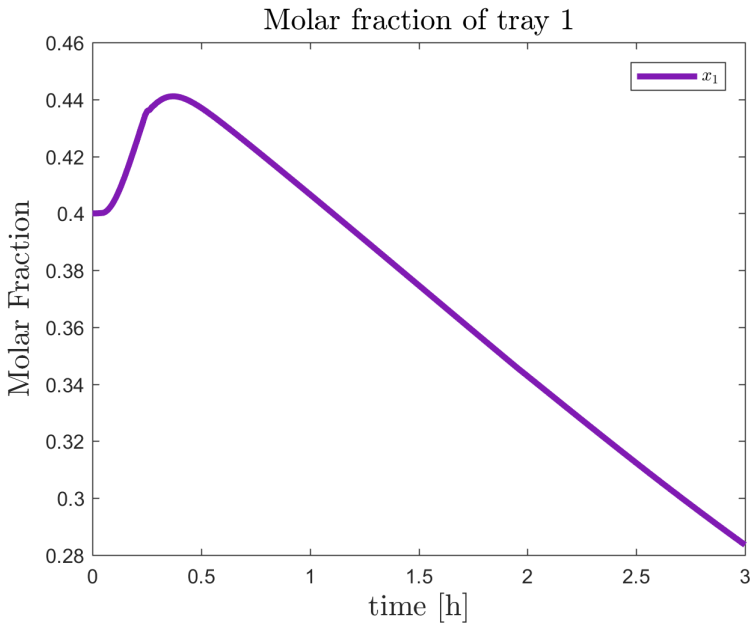


Fig. B.10: Molar fraction of light component on the first tray obtained applying AMPC to the batch distillation column.

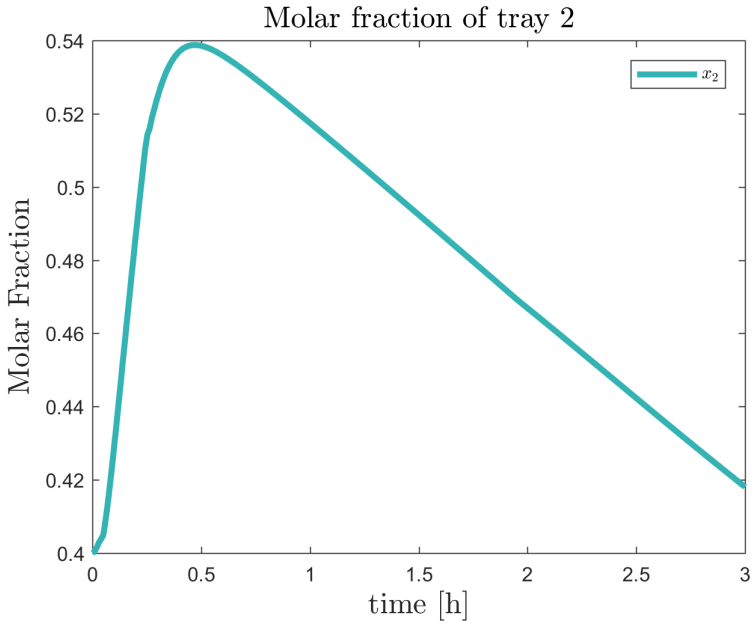


Fig. B.11: Molar fraction of light component on the second tray obtained applying AMPC to the batch distillation column.

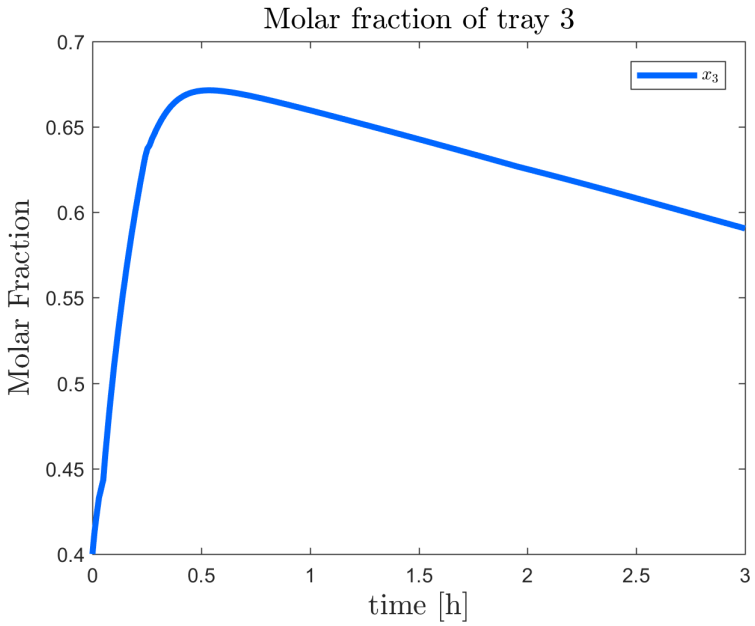


Fig. B.12: Molar fraction of light component on the third tray obtained applying AMPC to the batch distillation column.

Bibliography

[1] URL `www.coin-or.org/Ipopt`.

[2] Alberto Bemporad, Manfred Morari, N. Lawrence Ricker. Model Predictive Control Toolbox: For Use with MATLAB ®.

[3] F. Allgöwer, T. A. Badgwell, J. S. Qin, J. B. Rawlings, and S. J. Wright. Nonlinear Predictive Control and Moving Horizon Estimation — An Introductory Overview. pages 391–449.

[4] E. Aydin. Fast NMPC of Semi-Batch Processes via Simplified Solution Models: The Parsimonious Shrinking-Horizon NMPC.

[5] E. Aydin. NMPC USING PONTRYAGIN'S MINIMUM PRINCIPLE – APPLICATION TO A TWO-PHASE SEMI-BATCH HYDROFORMYLATION REACTOR UNDER UNCERTAINTY. *Computers & Chemical Engineering*, 108:47–56, 2017. ISSN 00981354.

[6] E. Aydin, D. Bonvin, and K. Sundmacher. Dynamic optimization of constrained semi-batch processes using Pontryagin's minimum principle—An effective quasi-Newton approach. *Computers & Chemical Engineering*, 99:135–144, 2017. ISSN 00981354. doi: 10.1016/j.compchemeng.2017.01.019.

[7] B. Srinivasan. Dynamic optimization of batch processes I. Characterization of the nominal solution. *Computers & Chemical Engineering*, 27:1–26, 2003. ISSN 00981354.

[8] D. B. E. V. S. P. B. Srinivasan. Dynamic Optimization of Batch Processes: II. Handling Uncertainty Using Measurements.

[9] M. Barolo and F. Berto. Composition Control in Batch Distillation: Binary and Multicomponent Mixtures. *Industrial & Engineering Chemistry Research*, 37(12): 4689–4698, 1998. ISSN 0888-5885. doi: 10.1021/ie980439o.

[10] J. C. Binette, B. Srinivasan, and D. Bonvin. On the various local solutions to a two-input dynamic optimization problem. *Computers & Chemical Engineering*, 95: 71–74, 2016. ISSN 00981354. doi: 10.1016/j.compchemeng.2016.09.003.

[11] D. Bonvin. Optimal operation of batch reactors - a personal view.

[12] C.R. Cutler and B.L. Ramaker. Dinamic Matrix Control - A computer control algorithm. *AIChE National Meeting, Huston*, 1979.

[13] C.R. Cutler and B.L. Ramaker. Dinamic Matrix Control - A computer control algorithm. *Proc. Joint Automatic Control Conference*, 1(WP5-B), 1980.

[14] W. F. S. H. I. H. F. M. D. C. KENDALL and B. G. LIPTÁK. Chemical Reactors: Basic Control Strategies. *Process Control and Optimization*, 2.

[15] M. Diehl, H. J. Ferreau, and N. Haverbeke. Efficient Numerical Methods for Nonlinear MPC and Moving Horizon Estimation. In M. Morari, M. Thoma, L. Magni, D. M. Raimondo, and F. Allgöwer, editors, *Nonlinear Model Predictive Control*, volume 384 of *Lecture Notes in Control and Information Sciences*, pages 391–417. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009. ISBN 978-3-642-01093-4. doi: 10.1007/978-3-642-01094-1_32.

[16] G. P. Distefano. Mathematical modeling and numerical integration of multicomponent batch distillation equations. *AIChE Journal*, 14(1):190–199, 1968. ISSN 00011541. doi: 10.1002/aic.690140132.

[17] U. Diwekar and K.-J. Kim. Batch Distillation. In E. Korovessi and A. Linninger, editors, *Batch Processes*, volume 106 of *Chemical Industries*, pages 107–150. CRC Press, 2005. ISBN 978-0-8247-2522-8. doi: 10.1201/9781420028164.ch5.

[18] D.M. Prett and R.D. Gillette. Optimization and constrained multivariable control of a catalytic cracking unit. *Proceedings of the Joint Automatic Control Conference*, 1(WP5-C), 1980.

[19] Erdal Aydin, Dominique Bonvin, Kai Sundmacher. Dynamic Optimization of Constrained Semi-Batch Processes using Pontryagin's Minimum Principle and Parsimonious Parameterization. *Symposium on Computer Aided Process Engineering – ESCAPE 27.*

[20] Eva Nagypal. Dynamic Optimization user's guide. *Economics 212.*

[21] D. Gillet. State-space observer.

[22] I. E. Grossmann, R. M. Apap, B. A. Calfa, P. García-Herreros, and Q. Zhang. Recent advances in mathematical programming techniques for the optimization of process systems under uncertainty. *Computers & Chemical Engineering*, 91:3–14, 2016. ISSN 00981354. doi: 10.1016/j.compchemeng.2016.03.002.

[23] M. A. Henson. Nonlinear model predictive control: Current status and future directions. *Computers and Chemical Engineering*, 23:187–202, 1998.

[24] Hyung-JunRho. Application of adaptive model-predictive control to a batch MMA polymerization reactor. *Chemical Engineering Science*, 53(21):3729–3739, 1998. ISSN 00092509. URL `https://doi.org/10.1016/S0009-2509(98)00166-3`.

[25] J. Richalet. Algorithmic control of industrial processes. *Proc. 4th Symposium on Identification and System Parameter Estimation*, pages 1119–1167, 1976.

[26] J.B. Froisy and T. Matsko. IDCOM-M application to the Shell fundamental control problem. *AIChE annual meeting*, 1990.

[27] R. E. Kalman. A New Approach to Linear Filtering and Prediction Problems. *Journal of Basic Engineering*, 82(1):35, 1960. ISSN 00219223. doi: 10.1115/1.3662552.

[28] E. Korovessi and A. Linninger, editors. *Batch Processes*. Chemical Industries. CRC Press, 2005. ISBN 978-0-8247-2522-8. doi: 10.1201/CRCCHEMINDUS.

[29] S. K. Lahiri. *Multivariable predictive control: Applications in industry / Sandip K. Lahiri*. John Wiley & Sons, Hoboken, New Jersey, 1st edition, 2017. ISBN 978-1-119-24360-1.

[30] Manfred Morari. Model predictive control: past, present and future. *Computers and Chemical Engineering*, 23:667–682, 1999.

[31] Michael A. Henson. Nonlinear model predictive control: current status and future directions. *Computers and Chemical Engineering*, 23:187–202, 1998.

[32] E. Pahija, F. Manenti, I. M. Mujtaba, and F. Rossi. Assessment of control techniques for the dynamic optimization of (semi-)batch reactors. *Computers & Chemical Engineering*, 66:269–275, 2014. ISSN 00981354. doi: 10.1016/j.compchemeng.2014.02.022.

[33] N. K. Poulsen. *Dynamic Optimization: Optimal Control*. 2012.

[34] S. Qin and T. A. Badgwell. A survey of industrial model predictive control technology. *Control Engineering Practice*, 11(7):733–764, 2003. ISSN 09670661. doi: 10.1016/S0967-0661(02)00186-7.

[35] S. S. Rao. Engineering Optimization: Theory and Practice, Fourth Edition.

[36] D. E. Seborg. *Process dynamics and control*. John Wiley & Sons Inc, Hoboken N.J., 3rd ed. edition, 2011. ISBN 0470128674.

[37] SEKI, Hiroya and OOYAMA, Satoshi and OGAWA, Morimasa. Nonlinear Model Predictive Control Using Successive Linearization: Application to Chemical Reactors. *Transactions of the Society of Instrument and Control Engineers*, 38:61–66, 2002.

[38] B. Srinivasan. Dynamic optimization of batch processes I. Characterization of the nominal solution. *Computers & Chemical Engineering*, 27:1–26, 2003. ISSN 00981354.

[39] M. Stojkovic, V. Gerbaud, and N. Shcherbakova. Cyclic operation as optimal control reflux policy of binary mixture batch distillation. *Computers & Chemical Engineering*, 108:98–111, 2018. ISSN 00981354. doi: 10.1016/j.compchemeng.2017.09.004.

[40] T. Peterson, E. Hernandez, Y. Arkun. A nonlinear DMC algorithm and its applcation to a semibatch polymerization reactor. *Chemical Engng. Sci.*, 47:737–753, 1992.

[41] D. M. H. L. S. L. Thomas F. Edgar. Optimization of chemical processes / Thomas F. Edgar, David M. Himmelblau, Leon S. Lasdon.-2nd ed. *McGraw-Hill Chemical Engineering Series*, 2001.

[42] S. Vajda. Linear programming. In S. Vajda, editor, *Linear Programming*, pages 1–15. Springer Netherlands, Dordrecht, 1981. ISBN 978-0-412-16430-9.

[43] M. Warter, D. Demicoli, and J. Stichlmair. Operation of a batch distillation column with a middle vessel: Experimental results for the separation of zeotropic and azeotropic mixtures. *Chemical Engineering and Processing: Process Intensification*, 43(3):263–272, 2004. ISSN 02552701. doi: 10.1016/S0255-2701(03)00122-3.

[44] L. B. W.C. Li. Multistep, Newton-type control strategies for constrained nonlinear systems. *Chem. Engng. Res. Dev.*, 67:652–577, 1997.