Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

An industrial implementation of a generic NMPC controller with application to a batch process

B. Pluymers * J. Ludlage ** L. Ariaans ** W. Van Brempt *

* IPCOS Belgium, Technologielaan 11-0101, 3001 Leuven
** IPCOS Netherlands, Bosscheweg 135b, 5282 WV Boxtel

Abstract: In the last decade a lot of attention was given to non-linear model predictive control. On one hand, in many applications linear MPC does not suffice to achieve the control goals over a wide range of operating conditions, while on the other hand many academic challenges remained in the area of NMPC, such as stability, computational complexity, etc... This paper discusses the industrial implementation of an NMPC controller at IPCOS and the different trade-offs made during the design, with the aim of clarifying the different criteria that are used in an industrial context. Results are illustrated on a chemical batch reactor.

Keywords: Model-based control, Nonlinear models, Chemical industry, Batch control

1. INTRODUCTION

Both academically and industrially NMPC has received increasing amounts of interest in the last decade. However, many differences exist between what is used in industrial practice and what is being researched in academia. Only some of these differences can be attributed by a delayed adoption of new technologies in industry. This paper addresses some of these differences as they are perceived by the authors based on the NMPC technology developed at IPCOS for the control of – among others – batch processes. The aim is not so much to introduce academically novel techniques, but rather to discuss existing techniques from a different point of view and to point out those aspects that are either not frequently treated in academic papers or that result in the fact that industrial MPC techniques fall outside of the existing theoretical frameworks developed within academia.

REWRITE: The next section first gives a brief introduction on batch processes, after which Section 3 explains the modeling costs of the corresponding approaches. Section 4 explains the main characteristics of the NMPC controller developed at IPCOS, after which Section 5 highlights several features that illustrate some specific design choices that were made and compares them to current academic practice. Finally Sections 6 and 7 discuss the results of the developed NMPC controller and give the conclusions of this paper.

2. BUSINESS CASE

In order to understand some of the decisions discussed in this paper, it is important to understand the business case of APC[1] projects from both the vendor and the customer point of view.

The most common customer benefits of APC projects largely consist of increased production capacity in capacity-constrained situations (batch-time reduction, throughput maximization, . . .), decreased operational costs (less off-spec product due to decreased process variance, decreased energy consumption, . . .). These (recurrent) benefits have to be compared to the implementation cost of the APC project, which is predominantly determined by licenses and the cost of human resources. Typically, customers expect a ROI[2] within 6 to 12 months. Therefore, in the development of new technology, the main criterion should be the optimization of this ROI, either in terms of improving benefits for the customer, or improving the efficiency of APC project execution.

On the other hand, from a vendor point of view, developing new technology should also be in equilibrium with the size of the targeted market, the chances of penetrating that market and the expected benefits. Developing and implementing the new technologies into robust and reliable end products should therefore be doable with acceptable efforts.

Very often, in academic papers, only controller performance is investigated, which can be linked to customer benefits. The existence of a good trade-off between improved benefits on one hand and increased project execution cost (e.g., additional tuning parameters, cost of model construction) and complexity of implementation (e.g., drastically new optimization technologies) on the other hand is often not investigated. However, in industrial practice, this trade-off is the most essential part of every technological decision when developing new technologies.

3. BATCH PROCESSES

In this section some more technical background is given based on which the non-linear modeling and control technology of IPCOS has been developed. This helps to

[1] **A**dvanced **P**rocess **C**ontrol

[2] **R**eturn **O**n **I**nvestment

understand several key decisions in the development of the technology.

In recent years batch processes have regained popularity due to the possibility they offer to industry to produce relatively small quantities of a variety of products with a large added value (Bonvin [1998]). Examples of such products are fine chemicals, pharmaceutical products and certain classes of polymers.

In many situations up to 100 or more different products are produced in a single reactor. With only limited restrictions in terms of product type continuity between subsequent batch runs, these reactors offer larger production flexibility compared to continuous processes. This added flexibility represents the main advantage of batch processes and forms a significant competitive advantage in quickly fluctuating markets.

However, this flexibility directly translates into a higher level of complexity for the modeling and control of such processes. Classic (linear) MPC does not suffice because of changing gains and time constants:

- with fed-batch processes, the content of a reactor can vary up to about a factor of 10 within one batch run, which has similar effects on the gain and/or time constants of the temperature dynamics.
- heat conductivity between the reactor wall and the reactor contents can also vary by a factor of 5 (sometimes non-monotonically) within a batch run
- in flow-controlled cooling coils and jackets, the relation between coolant flow rate and heat extraction can be very non-linear, with saturation effects occurring for large coolant flows.

These effects necessitate the application of a non-linear control law. On top of that, the need to take certain restrictions into account, such as cooling constraints, maximal allowable adiabatic temperature, etc... necessitates the use of non-linear MPC controllers.

The increasing (batch) market, together with other potential applications such as e.g. crystallization, led to the development of a non-linear MPC control architecture at IPCOS. The next section first describes the non-linear models used in the controller

4. NONLINEAR MODELING

Before being able to discuss the NMPC implementation, the control models and their identification are discussed.

In academic research, two main strategies can be found: full black-box modeling (neural networks, (Wiener)- Hammerstein, Volterra kernels, ...) on one hand and full rigorous modeling on the other hand.

Full *non-linear black box modeling* has some important disadvantages from a practical point of view:

- Black-box models are typically **not reliable for extrapolation** and therefore tests over a large range of operating conditions should be performed. It can be expected that this would have a profound impact on production and hence is not acceptable in a production environment.
- Even if testing over a large operating range is acceptable, this would, in typical situations, still require an **excessive amount of tests**, using up to 100 or more batch runs (different temperature profiles, feeding patterns, cooling patterns, ... and all combinations thereof). Given the necessary supervision during these tests, this is not acceptable from a project cost point of view.
- Another important disadvantage of black-box models is that they are **incapable of predicting variables that are not measurable**, such as excess (unreacted product present in reactor) and adiabatic temperature.

On the other hand, *rigorous modeling* is often not possible due **lack of fundamental understanding of certain process details**. In batch reactors, for example, reaction kinetics are often not understood in detail. For these reasons, going through a full rigorous modeling process is also prohibitively expensive from an ROI point of view.

Therefore, for processes requiring nonlinear control, a *hybrid modeling* approach (Vandecraen et al. [2007]) is chosen, where rigorous modeling is chosen for those model parts that are well understood and a suitably parameterized black-box modeling approach is chosen for the remaining parts. For batch processes, all reactor peripherals (condensors, heat exchangers, cooling coils, ...) are modeled rigorously, whereas the reaction kinetics are modeled by means of an application-specifically parameterized black-box model.This *hybrid* modeling approach (not to be confused with hybrid models having both continuous and discrete states) leads to discrete-time nonlinear state space models, consisting of an interconnection of rigorous, semi-rigorous and black box submodels. For details, we refer to Vandecraen et al. [2007]. The main message here is that in order to obtain sufficiently accurate models with limited effort (cfr. Section 2), a mix between rigorous and black-box modeling is required. This approach is rarely (e.g., Potočnik et al. [2004]) found in academic literature.

However, the NMPC framework discussed here is also able to cope with other model structures. These can include rigorous models (e.g. gProms or other modeling packages) or any other type of non-linear models (neural network, Wiener, Hammerstein, Volterra kernels, ...) as long as predictions and linearized models can be generated with sufficient efficiency. Due to the availability of a MATLAB interface, new model types can be tested with much flexibility.

5. CONTROLLER DEVELOPMENT

In this section the extension of the classic linear INCA [3] controller towards nonlinear models is explained. Figure 1 illustrates the block structure of the controller.

The classic INCA controller is set up in a block structure, where every block is triggered in a fixed order. In the classical linear INCA controller the following block sequence is executed:

- **Prediction:** Compute the values of the outputs over the prediction horizon based on an initial guess of the

[3] **I**PCOS **N**ovel **C**ontrol **A**rchitecture

optimal solution (e.g., a shifted sequence of optimal values of the previous sample instant).

- **Static Optimization:** Compute the optimal steady state value for the process relative to the current steady state value (i.e., the values at the end of the horizon).
- **Dynamic Optimization:** Perform a full dynamic optimization to compute the optimal input sequence that brings the system from the current state to the computed optimal steady state (at the end of the horizon).

Although performing the optimization step in two phases (static and dynamic) is not often considered in academic research, it is a standard technique in industrial implementations. This technique allows the smaller-scale static optimization to be computed more rigorously, e.g. using prioritized constraints (see Section 6.2).

This block structure makes sure that a block of the control engine can be replaced or modified without drastically affecting the other blocks. This makes it possible to easily test new control algorithms or model structures without losing maintainability of the code. As a first step towards an NMPC implementation this block execution order has been made configurable. In this way it is possible to e.g. eliminate or replace specific blocks for certain control problem areas (e.g. to skip the static optimization for chemical batch reactors) or perform multiple optimization steps. In subsequent steps the different model blocks that are affected by the type of control model that is used, have been updated to tackle nonlinear control problems. These blocks are highlighted in Figure 1.

The most straightforward way to incorporate nonlinear models is to update the prediction block. Since the control models are discrete-time this step is extremely straightforward and boils down to a sequential evaluation of a nonlinear function over the prediction horizon. The main extension of this block is the incorporation of linearization functionality. This functionality computes linear state space models along the predicted input and state trajectory and stores these to be used in the steady state and dynamic optimization.

In the steady-state and dynamic optimization steps a QP based optimization step is performed. Also here, the block structure allowed an evolutionary approach to the extension of the controller to nonlinear models. The following extensions were implemented:

(1) extension of the steady-state optimization to incorporate a linearized model, linearized at the end of the prediction horizon.
(2) extension of the dynamic optimization to incorporate a linearized model, linearized at the beginning of the control horizon.
(3) extension of the dynamic optimization to incorporate multiple linear models, linearized at several (possibly all) samples within the control horizon.

The resulting extensions effectively allow the execution of a single SQP iteration to solve the NMPC optimization problem. Due to the configurable nature of the block execution order, multiple SQP iterations can be performed within one sampling period. Performing only a single SQP iteration can be compared to so called *real-time* variants of optimization algorithms such as that used in Diehl et al. [2002].

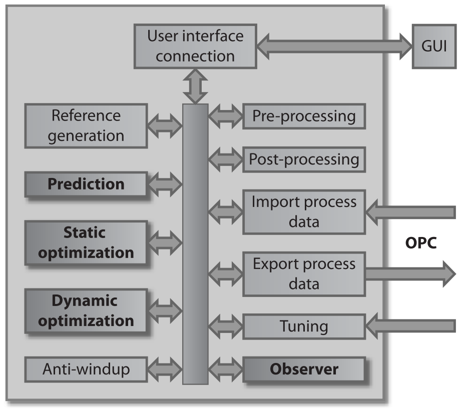

Fig. 1. Block structure of the INCA NMPC controller. Blocks affected by the use of nonlinear models are highlighted.

By choosing the above strategy towards an NMPC implementation, all intermediate implementations could already be validated industrially before moving on to the next phase. Another benefit of evolutionary development is market adoption. Small, incremental product improvements often have much faster market adoption, since the (perceived) risk is much smaller. Therefore, the above development approach can also be considered a strategic decision.

6. FEATURES

In this section several key features of the INCA NMPC controller are highlighted and compared to current academic results and interests in order to point out some interesting research opportunities.

6.1 Sequential optimization

A relatively straightforward sequential optimization technique is employed for the dynamic optimization. Input trajectories are reparameterized using move blocking, with user-configurable move times. Mostly quadratically or exponentially spaced move times are employed. The linearized models are used to construct a linear equality constraint matrix expressing the behavior between input corrections and the resulting output corrections. A quadratic cost function is used with weights on inputs, outputs and their derivatives.

This approach does not follow the recent trend towards simultaneous approaches (e.g. Diehl et al. [2002], Gattu and Zafiriou [1992]), due to the evolutionary approach explained above. Allowing validation of intermediate development steps was a key factor to decide for this evolutionary approach above a revolutionary approach using e.g. multiple shooting or other, more invasive approach. It is not excluded, however, that in future implementations more advanced optimization techniques will be evaluated.

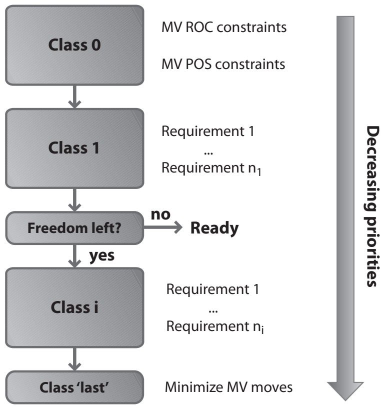

Fig. 2. Illustration of the concept of prioritized constraints.

6.2 Prioritized constraints and ideals

In the optimization stages of the INCA control architecture the concept of prioritized constraints has been built in (see Fig. 2) Constraints are handled starting with the highest priority class and proceeding to constraints and ideals with lower priority until no more degrees of freedom are left. Within each class constraint violation and ideal deviation trade-offs are defined by means of L_2 penalties.

As depicted in Figure 2, ROC (**R**ate **O**f **C**hange) constraints on the MVs (**M**anipulated **V**ariables) have the highest priority, followed by constraints on the absolute values of MV constraints (MV POS constraints) and CV (**C**ontrolled **V**ariables) constraints and ideals. The lowest priority class tries to minimize the MV moves.

The above mechanism of different priority classes and L_2 penalty functions within each priority class gives the user the ability to specify to the controller whether to make a trade-off between requirements or to give one requirement absolute priority above the other. Using either L_1 or L_2 penalty functions without the notion of priorities, the end-user would only be able to achieve one of the above effects. The notion of priorities is especially useful in non-linear control because, without priority classes, the different trade-offs between requirements would be different from one sample to another due to changing model gains, which would make tuning the controller especially cumbersome.

Despite the fact that using priority classes has significant practical benefits and is used by several APC vendors (Qin and Badgewell [2003]), only a limited number of publications (e.g., Tyler and Morari [1997], Vada et al. [1999, 2002]) can be found on the subject. The extension of existing theoretical results towards MPC controllers with prioritized constraints has not been published up to the authors' knowledge.

6.3 Stability measures

An aspect that has been intensely studied in academia is that of stability of MPC, which is by now very well understood (see e.g. Mayne et al. [2000]). However, these techniques have only been picked up by industry to a very limited extent.

The stability measures built into the INCA NMPC controller essentially boil down to imposing an end-point equality constraint (if the steady-state optimization block is used) and employing a prediction horizon that is significantly longer than the control horizon. Persisting stability problems – if ever encountered – can most often be avoided by decreasing the bandwidth of the controller by means of increasing move penalties.

The reasons why this practical approach works in most cases, although several assumptions, on which typical stability theories are based, do not hold (time-invariance of models, constraints and setpoints, state-feedback assumption, ...) can be understood as follows:

- stability theorems based on end-point inequality constraints (see Mayne et al. [2000]) make sure the controller drives the system inside an operating region at the end of the control horizon (imposed by means of a terminal constraint) for which it can be proven that the system can be further stabilized without violating constraints. By doing this in a provable way some conservativity is introduced: the system most often can be stabilized for a much larger region but not in a easily provable way.
- on the other hand, by simply omitting the terminal constraint, one potentially allows states for which it is known that the system cannot be stabilized without violating constraints.

The latter approach, however, seems to be working in many cases due to the fact that prioritized constraints are used and therefore the freedom introduced by slight violations in the lower-priority constraints in al observed cases allow to avoid infeasibilities. On top of that, the above situation can already be mostly avoided by choosing a sufficiently large control and prediction horizon.

The underlying reason why terminal constraints or terminal costs (see Mayne et al. [2000] for an overview) are not preferred is the fact that they depend on the controller model and/or the imposed constraints. This fact causes several impracticalities in an industrial context:

- In practice model gains and imposed constraints need to be modifiable on-the-fly. Especially during commissioning this happens frequently and therefore these modifications should not implicate heavy computations. For large-scale models (many MVs and CVs) computation of terminal constraints can be very tedious (see Pluymers [2006]).
- For non-linear models no generic techniques exist for efficiently computing these terminal regions.

Therefore, all stability measures that are either based on the controller model or the imposed constraints are mostly avoided in practical settings. Research efforts to decrease computation times for these stability measures could help to bridge this gap.

6.4 Adaptive control

The INCA control architecture can be set up in various ways to include adaptivity. Adaptivity is built in in three main areas: model adaptation, constraint adaptation and adaptive tuning. All three mechanisms lead to adaptive control behavior, which allows new information to be taken into account as it comes in. Essentially, all of the mechanisms presented in this section violate the assumptions underlying current established MPC stability frameworks and many other theoretical considerations.

Model adaptation One of the main reasons for long-term failures of MPC controllers is the degrading quality of the controller model, due to wear, upgrades, ... in the real process. In order to cope with such problems one can employ model adaptation techniques. In the case of batch control two different mechanisms exist: an *inter-batch observer* and an *intra-batch observer*.

The intra-batch observer is based on an EKF[4] state estimator, where uncertain model parameters are treated as additional model states. Model parameters adapted by this mechanism are those variables that can change significantly within one batch run and that are easily observable using an EKF.

The inter-batch observer adapts certain model parameters in between batch runs. Model parameters adapted by this mechanism are variables that change relatively slowly and for which relatively complex computations are needed to compute an updated value. An example in the framework of batch applications is cooling coil efficiency due to fouling or catalyst deactivation.

Other possible mechanisms that can be fitted in the INCA controller architecture are update mechanisms for remodeling black box (sub)models based on new process data (e.g. refitting static nonlinearities in (Wiener-)Hammerstein systems, ...), or e.g. more complex observer algorithms, such as moving horizon estimation.

Constraint adaptation In many applications where some form of *constraint pushing* is performed, the exact position of the constraint that should be reached is not exactly known and time-varying. Examples of this are cooling constraints in batch processes, where the MPC controller steers a setpoint of a slave PID controller, who in turn controls a valve that controls the cooling fluid flow. The constraint on the setpoint of the slave PID controller is determined by the valve position saturation. However, the relationship between setpoint and steady-state valve position might be unknown and time-varying due to changing cooling conditions (cooling fluid pressure, temperature, ...). Mechanisms are built into the INCA NMPC controller for updating the constraint position based on incoming measurements to avoid unexpected *constraint bumping*, which often leads to unwanted control behavior.

Adaptive tuning One final very powerful mechanism to achieve adaptive control is adaptive controller tuning. The INCA controller can be configured to read certain tuning variables from an OPC server every sample time. These variables can in turn be computed by means of calculation blocks, either in the DCS[5] or in the INCA software. In this way the tuning of the controller can be adjusted as a function of time, operating point or e.g. model fidelity (measured by e.g. the prediction error).

[4] **E**xtended **K**alman **F**ilter
[5] **D**istributed **C**ontrol **S**ystem

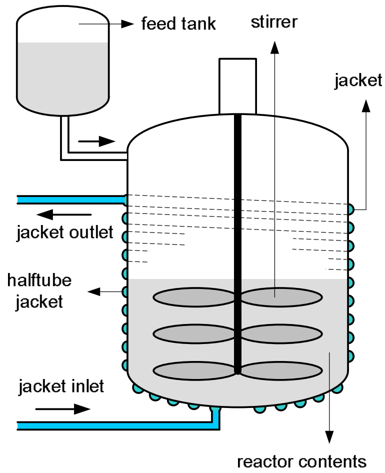

Fig. 3. Chemical batch reactor model used to illustrate controller performance.

7. RESULTS

In this section real-life results of an initial trial of the developed INCA NMPC controller on a chemical batch process are given. The reactor (see Figure 3) consists of the reactor vessel and a premix tank from which the main vessel is fed. The main vessel is cooled by means of a half-tube cooling jacket. The MVs of the control problem are the amount of cooling (jacket inlet temperature) and the feed flow. The main CV is the reactor temperature, which is related to product quality. The process operated in this reactor is an exothermic fed-batch process.

In the classical DSC control recipe a fixed feed flow pattern is used to make sure no cooling constraints are hit. Because the cooling capacity is season-dependent this approach leads to suboptimal batch cycle times.

The INCA NMPC controller was applied to the process – using the hybrid modeling technology – to achieve shorter batch cycle teams by means of maximized feeding rates. Figure 3 shows a comparison between both approaches. It is clear the the controller achieves a significant batch time reduction, while controlling the reactor temperature more tightly. One can see that different requirements are met at different instants during the batch. After the initial startup transient the desired feed flow rate is reached (around sample 400) after which the maximal cooling limit is reached (around sample 600). The disturbance noticed around sample 700 (INCA control) and sample 850 (DCS control) represents a manual operator intervention (modification of the stirrer speed) which is not known to the controller.

Several of the features of the INCA controller described in this paper are used in this example:

- due to the time-varying and nonlinear dynamics, the SQP-approach using multiple linear models over the horizon is needed to achieve stable control performance.
- cooling constraints (in terms of lowest obtainable jacket inlet temperature) are adapted at every sample instant in order to be able to achieve the highest feed flows subject to the real cooling constraints.

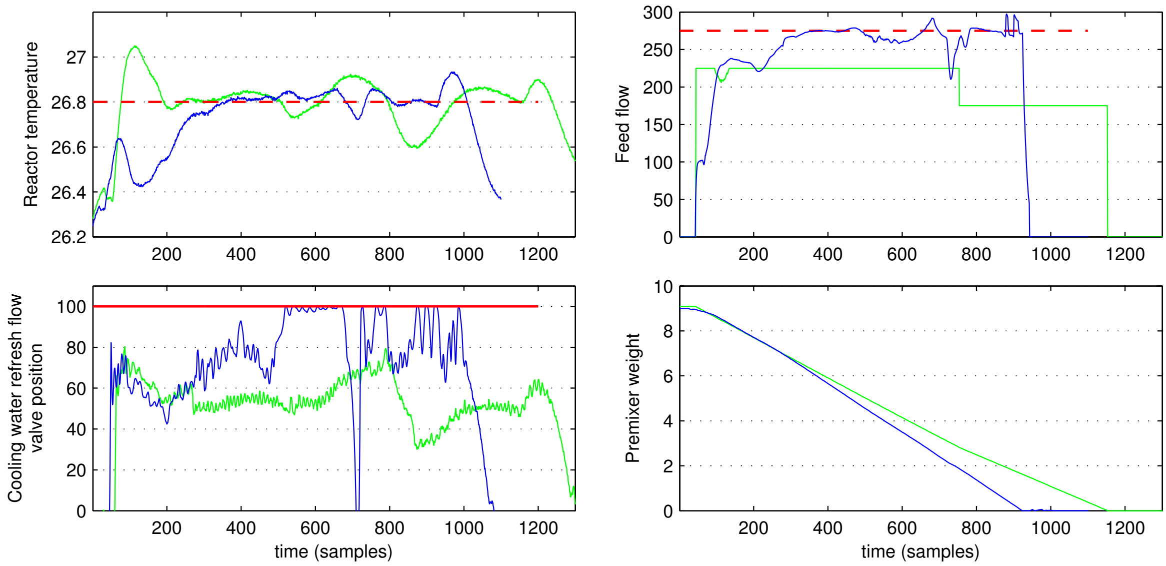

Fig. 4. Result of an industrial trial on a fed-batch reactor. Green lines depict result of DCS control using the classical batch recipe, while blue lines depict result of the INCA NMPC controller. Ideals and constraints are resp. depicted as dashed and solid red lines. Units are rescaled for confidentiality reasons.

- adaptive tuning is used to achieve robust performance during the initial rampup phase and tight temperature control afterwards.

It is clear that in this complex setting no stability results based on invariant sets can be used to guarantee stable control behavior and stability. On the other hand, while a relatively simple, sequential, one-step SQP based approach is used for the optimization, good control performance is achieved, while mainly still relying on large parts of the proven infrastructure of the linear INCA controller.

8. CONCLUSION

This paper discusses an industrial implementation of an NMPC controller. The controller can generically be coupled with any nonlinear discrete-time model that provides prediction and linearization functionality. The controller employs prioritized constraint handling, SQP-type optimization and adaptive control mechanisms that make it well-suited for a wide range of processes.

Part of this paper highlights those features where industrial and academic state-of-the-art differ substantially. Several interesting research opportunities are pointed out.

Finally, results are shown on a chemical batch reactor indicating good performance of the NMPC implementation and its successful industrial validation.

REFERENCES

R. Berber. *Control of Batch reactors: a review*, volume Methods of Model Based Process Control, pages 459–494. Kluwer Academic Publishers, 1995.

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

M. Diehl, H. G. Bock, J. P. Schlöder, R. Findeisen, Z. Nagy, and F. Allgöwer. Real-time optimization and nonlinear model predictive control of processes governed by differential-algebraic equations. *Journal of Process Control*, 12:577–585, 2002.

G. Gattu and E. Zafiriou. Nonlinear quadratic dynamic matrix control with state estimation. *Industrial & engineering chemistry research*, 31, 1992.

A. Marchetti, B. Chachuat, and D. Bonvin. Real-time optimization of continuous processes via constraints adaptation. *Proceedings of DYCOPS'07, Cancun, Mexico*, 2007.

D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert. Constrained model predictive control: Stability and optimality. *Automatica*, 36:789–814, 2000.

B. Pluymers. *Robust Model Based Predictive Control – An Invariant Set Approach*. PhD thesis, Katholieke Universiteit Leuven, 2006.

B. Potočnik, A. Bemporad, F. D. Torrisi, G. Mušič, and B. Zupančič. Hybrid modelling and optimal control of a multiproduct batch plant. *Control Engineering Practice*, 12:1127–1137, 2004.

S. J. Qin and T. Badgewell. A survey of industrial model predictive control technology. *Control Engineering Practice*, 11:733–764, 2003.

M. L. Tyler and M. Morari. Propositional logic in control and monitoring problems. *Proceedings of European Control Conference '97, Brussels, Belgium*, pages 623–628, 1997.

J. Vada, O. Slupphaug, and B. A. Foss. Infeasibility handling in linear mpc subject to prioritized constraints. *Proceedings of the 14th IFAC World Congress, Beijing, China*, 1999.

J. Vada, O. Slupphaug, and T. A. Johansen. Efficient infeasibility handling in linear mpc subject to prioritized constraints. *Proceedings of the American Control Conference, Achorage, Alaska*, 2002.

B. Vandecraen, J. Espinosa, B. Pluymers, D. R. Vinson, J. Ludlage, and W. Van Brempt. An industrial approach for efficient modeling and advanced control of chemical batch processes. *Proceedings of DYCOPS'07, Cancun, Mexico*, 2007.