



Contents lists available at ScienceDirect

Computers and Chemical Engineering

journal homepage: www.elsevier.com/locate/compchemeng

Closed-loop integration of planning, scheduling and multi-parametric nonlinear control

Vassilis M. Charitopoulos, Lazaros G. Papageorgiou, Vivek Dua*

Department of Chemical Engineering, Centre for Process Systems Engineering, University College London, Torrington Place, London WC1E 7JE, United Kingdom

ARTICLE INFO

Article history:

Received 10 January 2018

Revised 15 June 2018

Accepted 26 June 2018

Available online xxx

Keywords:

Multi-parametric programming

Enterprise wide optimization

Integration

Process planning

Rescheduling

Closed-loop optimization

ABSTRACT

In this article, motivated by the need for efficient closed-loop implementation of the control objectives set within the integrated planning, scheduling and control (iPSC) problem we introduce a novel framework that enables its online solution under dynamic disturbances. We introduce the concept of multi-setpoint explicit controllers through the use of a new multi-parametric nonlinear programming algorithm and develop a rigorous rescheduling mechanism that mitigates the impact of the dynamic disruptions on the operational decisions of planning and scheduling. The overall closed-loop problem is formulated as mixed integer linear program with the control problem integrated via an outer loop. The benefits of the proposed framework are highlighted through two case studies and the results indicate the necessity of considering dynamic disruptions within the scope of the integrated problem.

© 2018 The Author(s). Published by Elsevier Ltd.

This is an open access article under the CC BY license. (<http://creativecommons.org/licenses/by/4.0/>)

1. Introduction

Volatile global market environment, increasing competition and the need for reduction in cost and environmental impact are only a few of the reasons that have led the process industries to seek more responsive and integrated operations. Enterprise Wide Optimisation (EWO) aims to address the aforementioned challenges and provide the industries with tools that can serve as means for enhanced profitability and more sustainable operations (Grossmann, 2012). Within the EWO scope one seeks for more integrated decision making via the coordinated optimisation of the supply chain functionalities so as to holistically guarantee the efficient information sharing and optimal operations among the different levels of decision making. A conceptual representation of the EWO scope is given in Fig. 1, where the different levels of decision making, the key decisions and timescales involved are summarised.

Some of the most important operational functionalities of the process industries comprise of production planning, scheduling, real time optimisation and control. To this end, the process systems engineering (PSE) community has focused on the development of methods for their integration so as to exploit the inherent synergies and prevent suboptimal operations due to negligence of their underlying interdependence

(Chu and You, 2015; Grossmann, 2012; Harjunkski et al., 2014). While traditionally, the problems of planning, scheduling and control have been modelled and solved in a decoupled and sequential fashion due to more favorable computational requirements. Recently a number of research works have been devoted to their integration (Charitopoulos et al., 2017b; Gutierrez-Limon et al., 2014; Shi et al., 2015). Integration of planning and scheduling has been studied extensively in the past and their simultaneous optimisation has proven to result in improved profitability since decisions such as inventory calculation and production targets from the planning problem are highly interconnected with the optimal resource allocation which stems from the scheduling problem (Castro et al., 2004; Maravelias and Sung, 2009). Another significant trend is towards the integration of scheduling and control and a considerable amount of research work has been reported on that problem (Chu and You, 2015). While scheduling deals with the optimal allocation of limited resources and the sequencing of tasks, the underlying dynamics of the systems which are mostly dealt by the control functionality can highly affect the duration of changeovers and the quality of products manufactured during the production periods. It follows naturally that the integration of planning, scheduling and control (iPSC) can result in more optimal operations since the underlying synergies among the individual problems can enhance process operations. Unfortunately, if enhanced operations is the gift of integration, its price is quite high as it results in large-scale, typically non-convex, optimisation problems and extensive computational times that prohibit its application to large scale

* Corresponding author.

E-mail address: v.dua@ucl.ac.uk (V. Dua).<https://doi.org/10.1016/j.compchemeng.2018.06.021>0098-1354/© 2018 The Author(s). Published by Elsevier Ltd. This is an open access article under the CC BY license. (<http://creativecommons.org/licenses/by/4.0/>)

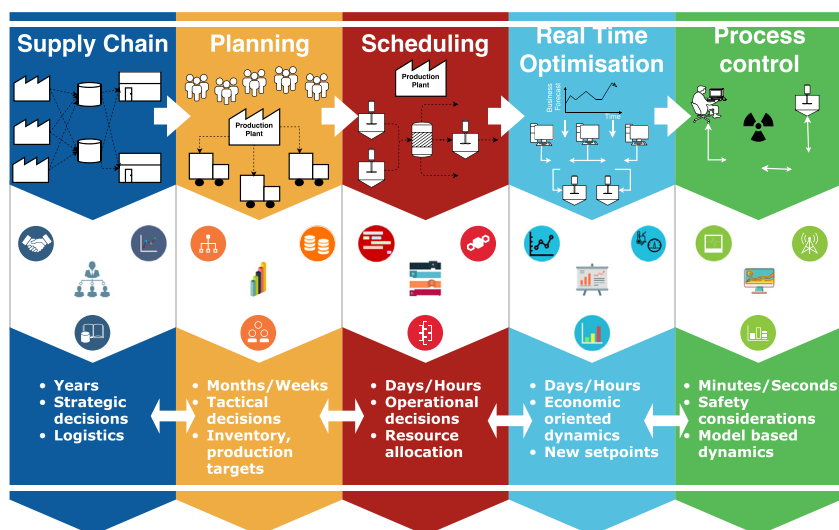


Fig. 1. Enterprise wide optimisation scope.

Index of abbreviations.

Abbreviation	Meaning
CAD	Cylindrical Algebraic Decomposition
CR	Critical Region
EWO	Enterprise Wide Optimisation
iPSC	integrated Planning Scheduling and Control
iSC	integrated Scheduling and Control
KKT	Karush–Kuhn–Tucker
MILP	Mixed Integer Linear Programming
MINLP	Mixed Integer Nonlinear Programming
MMA	methyl-methacrylate
MPC	Model Predictive Control
mp-P	Multi-parametric Programming
mp-MPC	Multi-setpoint explicit Model Predictive Control
NLP	Nonlinear Programming
NLMPC	Nonlinear Model Predictive Control
OFC	Objective Function's Coefficient
PSE	Process Systems Engineering
RHS	right-hand side
RTO	Real Time Optimisation
SISO	Single Input Single Output
TSP	Travelling Salesman Problem

systems. To this end, decomposition and simplifications of the integrated problem have been proposed in the literature so as to allow for the study of large-scale systems (Pistikopoulos and Diangelakis, 2016; Zhuge and Ierapetritou, 2016).

The efficient online computation of the decisions involved in the iPSC under uncertain conditions remains an open challenge (Dias and Ierapetritou, 2016). The main goal of the present work is to propose a framework for the closed-loop iPSC under dynamic disturbances and illustrate how the consideration of uncertain operating conditions accentuates the need for integration among the different levels of decision making. In this article, we build on the developments previously presented by our group (Charitopoulos et al., 2017b) for the open-loop case and with the use of multi-parametric programming a novel framework for the closed-loop implementation of the iPSC is proposed. The key elements of the proposed framework involve: (i) linear metamodelling that correlate transition times and costs based on closed-loop simulations of the underlying dynamic systems, (ii) the implementation of novel multi-parametric nonlinear model predictive controllers and (iii) an optimisation based algorithm for the efficient rescheduling that mitigates the impact of disturbances on the online implementation of the integrated problem. The remainder of the article is structured as follows: first a literature review is presented on the topic of integrating control with operations and the

need for a closed-loop framework for iPSC is underlined. Next, the key elements of the proposed framework are introduced; we briefly summarise the model employed for the open-loop iPSC and then the role of multi-parametric programming in the closed-loop implementation of the iPSC is presented. Subsequently, the overall closed-loop framework is presented in detail with its necessity and efficiency been shown through two case studies. Finally, concluding remarks and future research directions are discussed.

2. Literature review

Integrating control and operations has attracted significant amount of attention from the research community because of the potential benefits that result from the exploitation of their underlying synergies (Chu and You, 2015; Dias and Ierapetritou, 2016). Control relevant decisions provide an important set of data, such as transition times and production rates, which are crucial for modelling and solving in an optimal manner the scheduling problem. On the other hand, sequencing decisions are needed by the control decision layer so as to proceed with the manipulation of the dynamics of the production system.

The aforementioned interactions between cyclic scheduling and control were examined by Flores-Tlacuahuac and Grossmann (2006) and the authors showed how their open-loop integration can yield better results when compared to the conventional sequential solution of the problems. The closed-loop integration of cyclic scheduling and control (iSC) for continuous processes was studied by Zhuge and Ierapetritou (2012) and a model predictive control inspired mechanism was proposed so as to mitigate the effect that disturbances had on the execution of the schedule. Through a number of case studies the authors demonstrated how the closed-loop iSC can cope with disturbance rejection during production and transition periods. An alternative methodology for the closed-loop iSC has been reported in Chu and You (2012). The authors, in an offline step, generated a number of PI-controllers for each possible transition and studied the integrated problem as the optimal simultaneous scheduling and controller selection. In order to achieve fast computational times, the resulting MINLP with fractional objective function was solved using Dinkelbach's algorithm. Aiming to reduce the time needed for the solution Zhuge and Ierapetritou (2014) suggested the use of multi-parametric model predictive control within the context of iSC. First, the original nonlinear dynamics of the underlying production system were linearised and then the explicit controller was designed through the

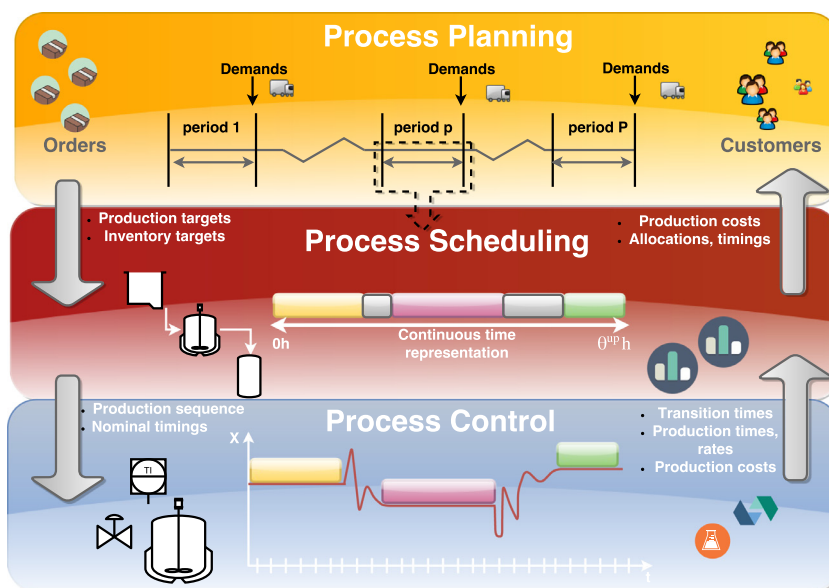


Fig. 2. Conceptual representation of the integrated planning, scheduling and control problem along with the related interdependent decision.

solution of the corresponding multi-parametric program. The explicit solutions were then incorporated in the scheduling model as a set of big-M constraints and the overall iSC was modelled as an MILP. Later on, the same authors proposed the use of fast MPC (Zhuge and Ierapetritou, 2015) and in that work piecewise affine approximations of the nonlinear dynamics were employed.

One of the main bottlenecks in the integrated problem is the time-scale separation among the different layers of decision making. To this end, Baldea et al. (2015) proposed the use of lower-dimensional dynamic models and embedded them in the scheduling formulation as a set of soft constraints. In their time scale-bridging approach, a scheduling oriented MPC was also employed so as to synchronise the calculations between MPC and scheduling.

However, when the demand is not assumed to follow a periodic pattern, the integration of planning along with scheduling and control becomes necessary. The iPSC is an inherently multi-scale problem for which one aims to optimise simultaneously the decisions involved in the levels of planning, scheduling and control so as to improve process operations and take explicitly into account their interdependence. As shown in Fig. 2, the different problems communicate via a number of interconnected decisions and there is information flow throughout. Gutierrez-Limon et al. (2014) extended the work of Flores-Tlacuahuac and Grossmann (2006) and formulated the problem as a large-scale monolithic MINLP along with a nonlinear model predictive control (NLMPC); a number of case studies were presented that resulted in large computational times for the solution of the integrated problem while disturbance rejection was not considered. Recently, in Gutierrez-Limon et al. (2016) a preliminary study on the effect of rush orders on the optimal solution of the iPSC was conducted and a number of heuristics were proposed in a reactive strategy sense. Shi et al. (2015) motivated by the need for faster computational times, proposed a decomposition framework based on flexible recipes involving all the potential transition between products. The overall flexible recipe iPSC was modelled as an MILP and the bilevel decomposition method by Dogan and Grossmann (2006) was employed to further enhance the computational behavior of the proposed framework. In our previous work (Charitopoulos et al., 2017b) we studied the iPSC of continuous processes using a traveling salesman problem (TSP) based model that proved to allow for significant computational savings when

compared to the time slot based formulations. We proposed the use of linear metamodels that correlate transition time and cost and under deterministic assumptions solved the integrated problem as an MILP whose optimal solution was equivalent to the one computed by the monolithic nonconvex MINLP.

Even though the problem of integrating control with operations has received considerable attention from the research community, no previous work has considered the iPSC under dynamic disturbances, i.e. closed-loop iPSC. For the online implementation of the iPSC to be effective and realistic one would have to account for dynamic disruptions at the level of control and develop an uncertainty-aware framework so as to secure optimal operations and real time execution. Due to the integrated nature of the problem it is reasonable to expect an immediate effect on the scheduling and planning decisions whenever the dynamics of the system are significantly disturbed.

The need for efficient reactive policies in the context of process scheduling has been long underlined in the open literature (Aytug et al., 2005). A least impact heuristic proposed by Kanakamedala et al. (1994) was among the first works published that considered the problem of reactive scheduling in multi-product batch plants. The reactive scheduling of batch processes was also studied by Vin and Ierapetritou (2000) and the authors considered two different kinds of disturbances, rush orders and machine breakdowns. In their work, the rescheduling mechanism is developed by the means of a repetitive solution of a reduced MILP problem every time new information about disruptions becomes available to the plant. The degree of deviation from the original schedule is also controlled through the use of a penalty in the objective function. Scheduling disruptions and reactive policies were studied by Mendez and Jaime (2004), where the authors used continuous time representation for multistage batch facilities and formulated the rescheduling problem as an MILP. The impact of rescheduling penalties in the objective function on the quality of the reschedule solution was also studied by Kopanos et al. (2008). Novas and Henning (2010) proposed a reactive scheduling framework based on a combination of constraint programming and explicit object oriented domain model that resulted in nearly optimal solutions at relatively low computational times. The use of multi-parametric programming has also been reported in the literature as a way of developing a reactive policy in scheduling problems by

several researchers (Kopanos and Pistikopoulos, 2014; Li and Ierapetritou, 2008). Recently, Maravelias and co workers (Gupta et al., 2016; Subramanian et al., 2012) in a series of papers based on the state-space interpretation of the scheduling of batch processes studied the effect of different factors on the derivation of periodic and reactive schedules.

In the following section, we review the mathematical developments proposed in the present work as a means for the real-time closed-loop implementation of the iPSC decisions along with an MIP-based rescheduling mechanism.

3. Mathematical formulations

In this section the main methodological components of the proposed framework are presented. First, we briefly review on the open-loop problem and its related modelling aspects. Next, the concept of multi-setpoint explicit controllers is presented and lastly the overall framework along with the corresponding algorithmic steps are outlined.

3.1. Modelling the integrated planning, scheduling and control problem

3.1.1. Open-loop integration of planning, scheduling and control

The open-loop case of iPSC was treated in a work recently presented by our group where a TSP based model was proposed for the integrated problem along with the use of linear metamodels (Charitopoulos et al., 2017b). Compared to the time slot based formulations used in the literature (Gutierrez-Limon et al., 2014; Shi et al., 2015), in the TSP based formulation there is not a fixed number of time slots to be postulated a priori but rather implicit unique pairs of products/tasks that need to be sequenced. For sequencing purposes, the time slot based formulations require the introduction of binary variables to assign products to time slots for each planning period that tend to increase the solution times for large planning horizons. On the contrary, the TSP based formulations track the sequencing of products/tasks in a similar way to the classic TSP problem using binary variables to model the relevant changeovers and computational saving compared to the time slot based formulation are achieved (Aguirre et al., 2017; Charitopoulos et al., 2017a; 2017b; Liu et al., 2008).

We briefly review the main equations for ease of understanding while the interested reader is referred to our earlier works for detailed exposition on the computational behavior and analysis of the proposed model (Charitopoulos et al., 2017b; Liu et al., 2008). In the TSP-based model, a hybrid time formulation is employed with planning periods modelled as discrete time points whereas within each point, continuous time formulation of θ_p^{up} duration is considered. Within each period only one product (i) can be first (F_{ip}) and last (L_{ip}) as shown by Eqs. (1) and (2) while the assignment of the products in each period is done via Eqs. (3) and (4) with the use of binary variable E_{ip} .

$$\sum_i^N F_{ip} = 1, \quad \forall p \quad (1)$$

$$\sum_i^N L_{ip} = 1, \quad \forall p \quad (2)$$

$$F_{ip} \leq E_{ip}, \quad \forall i, p \quad (3)$$

$$L_{ip} \leq E_{ip}, \quad \forall i, p \quad (4)$$

Sequencing of different products (i, j) within the same planning period is tracked with the binary variable Z_{ijp} and Eqs.

(5) and (6) while between adjacent planning periods changeovers are tracked with the binary variable ZF_{ijp} and Eqs. (7) and (8).

$$\sum_{i \neq j}^N Z_{ijp} = E_{jp} - F_{jp}, \quad \forall j, p \quad (5)$$

$$\sum_{j \neq i}^N Z_{ijp} = E_{ip} - L_{ip}, \quad \forall i, p \quad (6)$$

$$\sum_i^N ZF_{ijp} = F_{jp}, \quad \forall j, p > 1 \quad (7)$$

$$\sum_j^N ZF_{ijp} = L_{i,p-1}, \quad \forall i, p > 1 \quad (8)$$

In order to avoid infeasible production subcycles and the enumeration of symmetric solutions, the integer variable O_{ip} along with Eqs. (9)–(11) are used.

$$O_{jp} - (O_{ip} + 1) \geq -M(1 - Z_{ijp}), \quad \forall i, j \in I, j \neq i, p \quad (9)$$

$$O_{ip} \leq M \cdot E_{ip}, \quad \forall i, p \quad (10)$$

$$F_{ip} \leq O_{ip} \leq \sum_i^N E_{ip}, \quad \forall i, p \quad (11)$$

where M is a big number which for the sake of tight relaxation is equated to the cardinality of the set of products. Within each planning period, the processing (T_{ip}) and transition time (T_{ijp}^{trans} , TF_{ijp}^{trans}) are modelled as continuous variables. More specifically, production times are bounded between minimum and maximum times, θ_p^{lo} and θ_p^{up} respectively as shown by Eq. (12). Moreover, the changeover time between adjacent periods are allowed to split into two parts ($CT1_p$ and $CT2_p$) as shown by Eq. (13) while the overall time balance for each period is given by Eq. (14). The variables $CT1_p$ and $CT2_p$ are employed to allow for instances where a transition time can be modelled to be split between two adjacent periods and thus result in more efficient utilisation of resources (Kopanos et al., 2010).

$$\theta_p^{lo} E_{ip} \leq T_{ip} \leq \theta_p^{up} E_{ip}, \quad \forall i, p \quad (12)$$

$$CT1_p + CT2_{p-1} = \sum_i \sum_j TF_{ijp}^{trans}, \quad \forall p > 1 \quad (13)$$

$$\sum_i^N T_{ip} + \sum_i^N \sum_{j \neq i}^N T_{ijp}^{trans} + CT1_{p|p>1} + CT2_{p|p<|P|} = \theta_p^{up}, \quad \forall p \quad (14)$$

Notice that Eq. (14) accounts for idle production time by considering a relevant dummy product. The amount of product i produced during period p (Pr_{ip}) is calculated based on Eq. (15), by assuming constant production rate (r_i). Given product demand per customer (D_{cip}), backlog (B_{cip}), sales (S_{cip}) and inventory (V_{ip}) calculations are based on Eqs. (16) and (17) respectively. Also minimum (V_{ip}^{min}) and maximum (V_{ip}^{max}) inventory levels can be specified by Eq. (18).

$$Pr_{ip} = r_i T_{ip}, \quad \forall i, p \quad (15)$$

$$B_{cip} = B_{ci,p-1} + D_{cip} - S_{cip}, \quad \forall c, i, p \quad (16)$$

$$V_{ip} = V_{i,p-1} + Pr_{ip} - \sum_c S_{cip}, \quad \forall i, p \quad (17)$$

$$V_i^{\min} \leq V_{ip} \leq V_i^{\max}, \quad \forall i, p \quad (18)$$

Transitions times are allowed to be decision variables but in order to avoid the resulting bilinear terms, Glover linearisation is employed as shown in Eqs. (19)–(26). For the purpose of the linearisation two more artificial positive variables are defined, τ_{ijp} and τ_{ijp}^F .

$$T_{ijp}^{\text{trans}} \geq \tau_{ijp} + \theta_p^{\text{up}}(Z_{ijp} - 1), \quad \forall i, j \neq i, p \quad (19)$$

$$T_{ijp}^{\text{trans}} \leq \tau_{ijp}, \quad \forall i, j \neq i, p \quad (20)$$

$$T_{ijp}^{\text{trans}} \leq \theta_p^{\text{up}} Z_{ijp}, \quad \forall i, j \neq i, p \quad (21)$$

$$T_{ijp}^{\text{trans}} \geq \tau_{ij}^{\min} Z_{ijp}, \quad \forall i, j \neq i, p \quad (22)$$

$$TF_{ijp}^{\text{trans}} \geq \tau_{ijp}^F + \theta_p^{\text{up}}(ZF_{ijp} - 1), \quad \forall i, j, p > 1 \quad (23)$$

$$TF_{ijp}^{\text{trans}} \leq \tau_{ijp}^F, \quad \forall i, j, p > 1 \quad (24)$$

$$TF_{ijp}^{\text{trans}} \leq \theta_p^{\text{up}} ZF_{ijp}, \quad \forall i, j, p > 1 \quad (25)$$

$$TF_{ijp}^{\text{trans}} \geq \tau_{ij}^{\min} ZF_{ijp}, \quad \forall i, j, p > 1 \quad (26)$$

The transition costs are correlated via linear meta-models with the transition times as explained in Charitopoulos et al. (2017b) and as shown by Eq. (27). In the literature of integrating control with operations (Chu and You, 2015) the calculation of transition costs based on the system's dynamics results in complex nonlinear calculations. As a trade-off between computational complexity and model accuracy the use of linear meta-models was proposed, where coefficients α_{ij} and β_{ij} refer to the slope and intercept of the different correlations. The interested reader is referred to our recent work (Charitopoulos et al., 2017b) where a thorough discussion on computational steps and accuracy of this approach is provided.

$$CT_{ij}^{\text{tran}} = \alpha_{ij} T_{ij}^{\text{trans}} + \beta_{ij} \quad \forall i, j \in I, i \neq j \quad (27)$$

The revenue from product sales (RV) is given by Eq. (28), the operational cost (OC) is given by Eq. (29), the inventory (IC) and backlog (BC) cost are given by Eqs. (30) and (31), respectively while production (PC) and transition costs (PC) are calculated based on Eqs. (32) and (33). Given product prices (P_i), unit operational (C_i^{oper}), inventory (C_i^{inv}), backlog (CB_{ic}) and raw material cost (C_m^{raw}) the profit (PROF) over the planning period is computed as shown below.

$$RV = \sum_c \sum_i \sum_p P_i S_{cip} \quad (28)$$

$$OC = \sum_i \sum_p C_i^{\text{oper}} Pr_{ip} \quad (29)$$

$$IC = \sum_i \sum_p C_i^{\text{inv}} V_{ip} \quad (30)$$

$$BC = \sum_c \sum_i \sum_p CB_{ic} B_{cip} \quad (31)$$

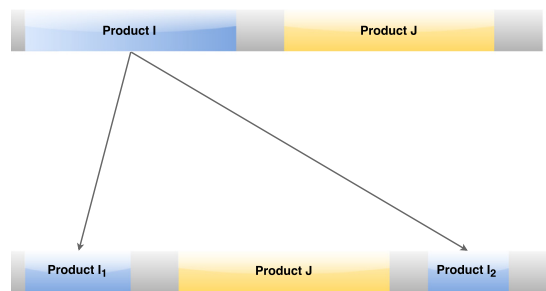


Fig. 3. Conceptual instance of product duplication due to production disruption.

$$PC = \sum_m \sum_i \sum_p C_m^{\text{raw}} u_{mi}^{\text{ss}} T_{ip} \quad (32)$$

$$TC = \sum_m C_m^{\text{raw}} \sum_p \sum_i \sum_{j \neq i} \alpha_{ij} (T_{ijp}^{\text{trans}} + TF_{ijp|p>1}^{\text{trans}}) + \beta_{ij} (Z_{ijp} + ZF_{ijp|p>1}) \quad (33)$$

Overall, the decomposed iPSC model is an MILP and is formulated as follows:

$$\begin{aligned} \text{Open-loop iPSC: } \max \text{ PROF} &= RV - OC - IC - BC - PC - TC \\ \text{Subject to: } &\text{Eqs. (1)–(18)} \quad (\text{Planning - Scheduling}) \\ &\text{Eqs. (19)–(27)} \quad (\text{Control considerations}) \end{aligned}$$

3.1.2. Open-loop iPSC with rescheduling considerations

The decomposition of the iPSC through the use of linear meta-models and the offline derivation of the minimum transition times is valid under a number of deterministic assumptions throughout the three levels of integration. However, when dynamic disruptions are considered the need to account for possible reschedulings has to be addressed. To this end, the model presented in the previous section is modified accordingly. When dynamic disruptions are identified during the production of a product, it should be allowed to resume the production so as to attempt to fulfill the remaining demand. Product duplication is employed so as to facilitate this issue as shown in Fig. 3.

Through product duplication, an identical product is created and inherits all the relevant information from the original one. Next a dynamic set is created, $I_1(i, j, p)$ which denotes the set of products that are considered for duplication on a specific planning period, during which the disruption occurred. Moreover, the following sets are considered: $I_R(i)$ which is the set of only the original products and $I_D(i, p)$ which is the set of the dummy products that represent the disturbance occurrence during period p . Eq. (34) is the modified version of Eq. (16) for the calculation of backlog, while for inventory calculations Eq. (17) is replaced by Eq. (35).

$$B_{cip} = B_{ci,p-1} + D_{cip} - S_{cip} - \sum_{j \in I_1} S_{cjp}, \quad \forall c, i \in I_R, p \quad (34)$$

$$V_{ip} = V_{i,p-1} + Pr_{ip} - \sum_c S_{cip} + \sum_{j \in I_1} Pr_{jp} - \sum_{j \in I_1} S_{cjp}, \quad \forall i \in I_R, p \quad (35)$$

The minimum and maximum production times are relaxed for the case of disturbance modelling and thus Eq. (36) arises instead of Eq. (12). Disturbances do not result production of products thus Eq. (37) is only employed for any product except for the ones that belong in I_D

$$\theta_p^{\text{lo}} E_{ip} \leq T_{ip} \leq \theta_p^{\text{up}} E_{ip} \quad \forall i \notin I_D, p \quad (36)$$

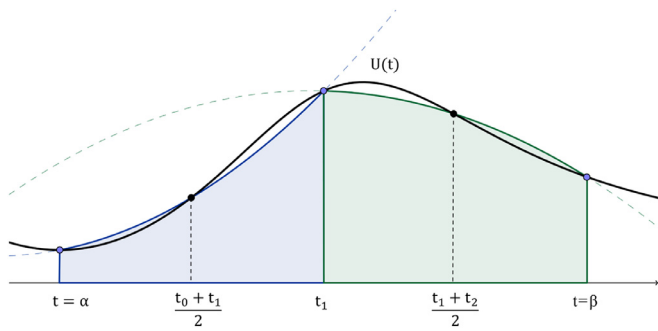


Fig. 4. Numerical integration of arbitrary transition curve via Simpson's rule.

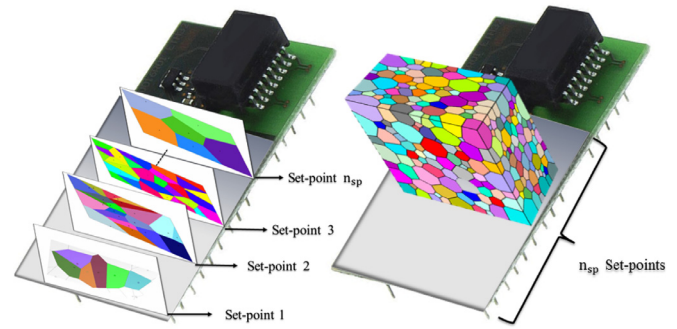


Fig. 5. Conceptual representation of a multi-setpoint explicit controller. On the left hand side a conventional multi-parametric controller for various set-points is depicted while on the right hand side a multi-setpoint explicit controller can be visualised with the third dimension being the continuous set-point space.

$$Pr_{ip} = r_i T_{ip} \quad \forall i \in I_D, p \quad (37)$$

As will be discussed in the next section, the closed-loop implementation of the iPSC is enabled via the use of novel multi-parametric controllers which communicate with the open-loop iPSC via a number of ways. One of them is via the information the controllers offer back to the integrated problem about the approximate transition cost between products in the case of rescheduling. In general the transition cost is the integral of the control actions ($u(t)$) over the transition period ($[0, T_{ij}^{trans}]$), i.e., $\int_0^{T_{ij}^{trans}} u(t)dt$. In order to allow for fast calculations the controller is programmed to compute the numerical integral of the transition based on Simpson's rule which provides a good trade-off between numerical accuracy and computational expense as shown in Fig. 4. Thus, the quantity $CT_{ij} \approx \int_0^{T_{ij}^{trans}} u(t)dt$ represents the numerical approximate value of the transition cost as computed by the multi-parametric controller.

This results in the modification of Eq. (33) about the calculation of the cumulative transition cost as shown in Eq. (38).

$$TC = \sum_p \sum_{i \in I_D} \sum_{j \neq i} \alpha_{ij} (T_{ijp}^{trans} + T_{ijp|p>1}^{trans}) + \beta_{ij} (Z_{ijp} + ZF_{ijp|p>1}) + \sum_{i \in I_D} \sum_{j \neq i} CT_{ij} (Z_{ijp} + ZF_{ijp|p>1}) \quad (38)$$

3.2. Closing the loop via multi-setpoint explicit nonlinear controllers

3.2.1. Multi-parametric model predictive control

In this section we present the idea behind the design of multi-setpoint explicit controllers. Multi-parametric programming (mp-P) as an optimisation based methodology has found numerous applications in the field of process systems engineering with the design of explicit controllers probably being the most popular (Charitopoulos and Dua, 2016; Dua et al., 2008; Oberdieck et al., 2016). Within the context of explicit MPC, one considers as uncertain parameters the initial states of the system at each sampling instance and thus an mp-P problem with uncertainty in the right hand side (RHS) of the constraints is formulated (Bemporad et al., 2002). The solution of this mp-P problem leads to the computation of the control law, i.e. the optimal control input as explicit function of the state of the system together with the regions where each expression holds. Despite the fact that explicit MPC is one of the most well studied areas of mp-P theory, the design of explicit controllers for set-point tracking remains a rather difficult task as one would have to design a controller for each set-point target given the methodologies that have been presented in the literature until now, especially for the nonlinear case (Pistikopoulos et al., 2015).

A generic mathematical formulation of the explicit MPC problem is given by Eq. (39),

$$\begin{cases} \Phi(\mathbf{x}(t_k)) = \min_{\mathbf{u}} \sum_{t=0}^{N-1} \mathcal{L}(\mathbf{x}_t, \mathbf{u}_t) + E(\mathbf{x}_N) \\ \text{Subject to: } \mathbf{x}_{t|t=0} = \mathbf{x}(t_k) \\ \mathbf{x}_{t+1} = \mathbf{f}(\mathbf{x}_t, \mathbf{u}_t) \quad t = 0, 1, \dots, N-1 \\ \mathbf{y}_{t+1} = \mathbf{h}(\mathbf{x}_t) \quad t = 0, 1, \dots, N-1 \\ \mathbf{g}(\mathbf{x}_t, \mathbf{u}_t, \mathbf{y}_t) \leq \mathbf{0} \quad t = 0, 1, \dots, N \end{cases} \quad (39)$$

where $\mathbf{x}_t, \mathbf{u}_t, \mathbf{y}_t$ are the state, control input and system output vectors respectively at every sampling instance, t , and are n_x, n_u, n_y dimensional. Inequality constraints for the state, output and control inputs are represented without loss of generality by the vector function $\mathbf{g} \in \mathbb{R}^{n_g}$, the mappings $\mathbf{h} : \mathbb{R}^{n_x} \rightarrow \mathbb{R}^{n_y}$ and $\mathbf{f} : \mathbb{R}^{n_x+n_u} \rightarrow \mathbb{R}^{n_x}$ correlate the output with the state and dictate the state evolution of the system respectively. $\mathcal{L} : \mathbb{R}^{n_x+n_u} \rightarrow \mathbb{R}$ is a stage cost and $E : \mathbb{R}^{n_x} \rightarrow \mathbb{R}$ is a terminal cost function over the prediction horizon N . The repetitive solution of problem (39) provides the optimal cost $\Phi(\mathbf{x}(t_k))$ and the optimisation vector, which in this case is the sequence of optimal control inputs $\mathbf{u}^* = [\mathbf{u}_1^*, \mathbf{u}_2^*, \dots, \mathbf{u}_{N-1}^*]$ over the finite prediction horizon N . While normally, the online repetitive solution of the receding horizon control problem is required, via the means of mp-P one can solve problem (39) for all possible realisations of the system's measurements and thus compute offline once and for all the optimal control law as a function of the system's measurements, $\mathbf{u} = \delta(\mathbf{x}_{t|t=0})$ along with the corresponding critical regions (CRs), i.e. the parametric ranges over which each explicit expression is optimal.

The class of problems described in (39) involves uncertain parameters on the right hand side (RHS) of the constraints. When multiple set-points need to be considered then there are two ways of designing the explicit controller(s). The first one, is to solve n_{sp} mp-P problems, where n_{sp} is the number of set-points considered and thus design n_{sp} explicit controllers. The second alternative, that we propose in the present work, is to design one multi-setpoint explicit controller (mp-MPC). The idea is to solve only one mp-P problem for n_{sp} set-points and create a universal "multi-layer" controller as shown conceptually in Fig. 5.

Designing a multi-setpoint explicit controller mathematically can be expressed by Eq. (40).

$$\begin{cases} \Upsilon(\mathbf{x}(t_k), \mathbf{x}_{sp}) = \min_{\mathbf{u}} \sum_{t=0}^{N-1} \mathcal{L}(\mathbf{x}_t, \mathbf{u}_t, \mathbf{x}_{sp}) + E(\mathbf{x}_N, \mathbf{x}_{sp}) \\ \text{Subject to: } \mathbf{x}_{t|t=0} = \mathbf{x}(t_k) \\ \mathbf{x}_{t+1} = \mathbf{f}(\mathbf{x}_t, \mathbf{u}_t) \quad t = 0, 1, \dots, N-1 \\ \mathbf{y}_{t+1} = \mathbf{h}(\mathbf{x}_t) \quad t = 0, 1, \dots, N-1 \\ \mathbf{g}(\mathbf{x}_t, \mathbf{u}_t, \mathbf{y}_t) \leq \mathbf{0} \quad t = 0, 1, \dots, N \\ \mathbf{x}_{sp} \in \mathcal{X}^{SP} \end{cases} \quad (40)$$

The difference between problem (39) and problem (40) lies on the treatment of the set-points as uncertain parameters, which results in an mp-P problem with both RHS and objective function's coefficients (OFC) uncertainty. As seen in (40), apart from the vector of the initial states ($\mathbf{x}(t_k)$) the various set-points (\mathbf{x}_{sp}) are considered as uncertain parameters as well. It is interesting to notice that, within the context of EWO, being able to design this kind of controllers is of great importance because the set-points targets are calculated dynamically by the decisions at the scheduling level.

3.2.2. A methodology for the design of multi-setpoint explicit controllers

As mentioned above we are interested in the following case: given a system that is required to operate at multiple set-points and the nonlinear terms involved in its model are non-transcendental we aim to design a single explicit controller that contains all the associated control laws. To do so, we employ concepts from computer algebra since the uncertain parameters are treated herein as symbolic expressions while the underlying optimisation problem is solved analytically using Gröbner bases theory (Charitopoulos et al., 2017c; Dua, 2015). Gröbner bases theory emerged from the Ph.D. thesis of Bruno Buchberger as a way to analytically solve systems of polynomial multi-variable equations (Buchberger, 2006). Briefly, Gröbner bases and the Buchberger algorithm can be seen as a generalisation of the Gaussian elimination for the case of linear systems. Before we proceed further it is important to provide some formal definitions that are crucial in Gröbner bases theory.

Let k be any field and let $k[\mathbf{x}] = k[x_1, \dots, x_t]$ be the ring of polynomials in t indeterminates. Any polynomial can be described as a sum of terms of the form: $\alpha x_1^{\beta_1} \dots x_t^{\beta_t}$ with $\alpha \in k$ and $\beta_i \in \mathbb{N}$, $i = 1, \dots, t$ and the term $x_1^{\beta_1} \dots x_t^{\beta_t}$ is called *power – product*.

Definition. (Gröbner basis (Buchberger, 2006)) A set of non-zero polynomials $G = \{g_1, \dots, g_t\}$ contained in an ideal I , is called a Gröbner basis for I if and only if for all $f \in I$ such that $f \neq 0$, there exists $i \in \{1, \dots, t\}$ such that $\text{lp}(g_i)$ divides $\text{lp}(f)$, where $\text{lp}(\cdot)$ stands for the leading power-product of a polynomial function.

In the definition given, an ideal is a set of polynomials of the form $\sum_{i=1}^t u_i g_i$ with g_i in G and arbitrary polynomials u_i . The existence of such ideal is guaranteed by the Hilbert Basis theorem (Bochnak et al., 2013), which also guarantees the termination of algorithms that are used for the computation of Gröbner bases. Roughly speaking, within Gröbner bases theory a set of polynomial V is transformed into an other set of polynomials G which is equivalent to the former but has certain favourable computational properties. At the core of Gröbner bases theory the Buchberger algorithm is found (Buchberger, 2006) which is employed for the computation of the Gröbner basis of a specific set of polynomials. Buchberger introduced within the algorithm the concept of S-polynomials as well as provided a theorem for the proposed algorithm which for the sake of space are not discussed in the present article; however, the interested reader can refer to the book of Bochnak et al. (2013). The implementation of the proposed methodology was done in Mathematica 11. The reason why computer algebra was chosen for the design of the multi-setpoint explicit controllers is because it provides us with the following degree of freedom. One can consider the various set-points as a single uncertain parameter bounded as shown by Eq. (41).

$$x_{sp}^{lo} \leq x_{sp} \leq x_{sp}^{up} \quad (41)$$

where x_{sp}^{lo} , x_{sp}^{up} represent the lower and upper bounds of the set-points set for the controller. However, within a computer algebra

environment, one can actually perform computations either in continuous or a discrete sets fashion as indicated by Eq. (42). Thus following the methodology followed in the present work, the uncertain parameters involved in the mp-P problem for the design of the multi-setpoint explicit controller can be treated in either way.

$$x_{sp} \in \{x_{sp}^1, x_{sp}^2, \dots, x_{sp}^{n_{sp}}\} \quad (42)$$

Solving multi-parametric nonlinear programming problems (mp-NLPs) still remains a challenging task despite the research effort put in the literature of multi-parametric programming. Problem (43) provides a generic mathematical formulation of mp-NLPs:

$$\begin{aligned} z(\theta) &= \min_{\mathbf{x}} f(\mathbf{x}, \theta) \\ \text{Subject to: } \mathbf{g}(\mathbf{x}, \theta) &\leq 0 \\ \mathbf{x} \in X \subseteq \mathbb{R}^{n_x}, \theta &\in \Theta \subseteq \mathbb{R}^{n_\theta} \end{aligned} \quad (43)$$

where \mathbf{x} is the n_x -vector of optimisation variables and belongs to the bounded set X , θ is the n_θ -vector of uncertain parameters and belongs to the set Θ which may be unbounded. The function f and is a scalar-valued function and the function \mathbf{g} is a vector-valued function of n_g dimensions denoting the constraints of the optimisation; note that both of the functionals mentioned can be linear or non-transcendental nonlinear. While a comprehensive review on the topic can be found in Domínguez et al. (2010) no previous work in the field has presented an algorithm that can facilitate mp-NLPs with simultaneous variations on the RHS and the OFC. A considerable amount of research work has been devoted to the analytical solution of mp-NLPs using computer algebra principles (Charitopoulos and Dua, 2016; Charitopoulos et al., 2017d; Dua, 2015; Fotiou et al., 2005) but only the case of RHS uncertainty has been treated. Recently, Charitopoulos et al. (2018) proposed an algorithm for the solution of mp-NLPs with non-transcendental nonlinear terms under the presence of simultaneous variations on RHS, left hand side (LHS) and OFC and this is the main machinery that is employed in the present work for the design of multi-setpoint explicit controllers.

They key idea of the aforementioned algorithm can be summarised as follows: given an mp-NLP, formulate the first order KKT conditions and solve the resulting system of nonlinear equations using Gröbner Bases while treating the uncertain parameters as symbols. This step results in a set of candidate solutions for the optimisers \mathbf{x} and the Lagrange multipliers λ which are parametric in θ and include: infeasible, local and global optima. For the candidate solutions computed, qualify with the primal and dual feasibility together with a constraint qualification and remove the infeasible explicit solutions. Finally, perform a comparison procedure and keep only the globally optimal solutions along with their corresponding CRs, by computing the corresponding Cylindrical Algebraic Decompositions (CAD). For a detailed exposition the interested reader is referred to Charitopoulos et al. (2018) while a comprehensive overview on topics of Gröbner Bases Theory and CAD can be found in Bochnak et al. (2013). It is worth to note that the proposed algorithm is dependent on Gröbner Bases calculations which have been proven to be doubly exponential with respect to the variables under determination in the worst case (Charitopoulos et al., 2018; Fotiou et al., 2005). In Fig. 6, an outline of the algorithm is presented.

3.3. The overall closed-loop integrated framework

Real processes are subject to a number of disturbances that endanger the feasibility and optimality of operations. For continuous manufacturing processes, fluctuations on the feed composition, feed temperature as well as the flow rate of the reactants can lead to significant deviations from the desired open-loop state

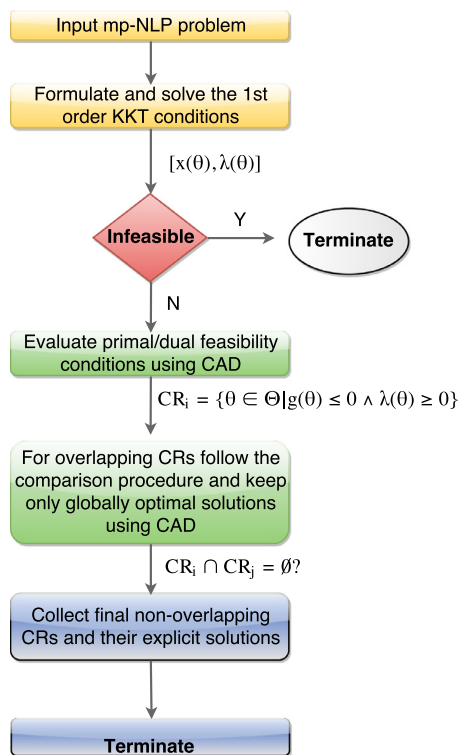


Fig. 6. Outline of the mp-NLP algorithm.

trajectory and result in the production of off-spec material. In the context of iPSC the different timescales involved, lead to the formulation of large scale MINLPs that pose an additional degree of complication that exacerbates the computational requirements and thus prohibit its online solution. In order to alleviate the computational complexity the proposed strategy involves an offline step where linear metamodells that correlate the transition time and cost are built, along with the related nominal minimum transition times. By doing so, this aspect of the interdependence between scheduling and control is exploited and at the same time the control timescale is de-dimensionalised thus reducing the computational complexity. Furthermore, it can happen that the data provided about the transition time and cost from the optimal control simulations can lead to inconsistencies when compared to the closed-loop behavior of the system due to potential model mis-

match or because of the different objectives considered. To this end, the data used for the derivation of the linear metamodells as well as the minimum transition times are computed via a number of closed-loop simulations of the underlying dynamic system through the use of the multi-setpoint explicit controller that was introduced earlier. This was also done in order to simulate what would happen in a real process where the mechanisms that dictate the transition costs and times may not be sufficiently captured by open-loop dynamic optimisation simulations but from historical data.

For the online part we consider the solution of an MILP (open-loop iPSC problem) and then the corresponding control problem that tracks online the open-loop solution. Conceptually, the proposed strategy for the closed loop solution is given in Fig. 7.

Firstly, the mp-MPC for the underlying dynamic system is designed. As shown in Fig. 7 in an offline step the linear metamodells that correlate transition time and cost are computed based on the closed-loop simulations of the system. The linear coefficient of the metamodells are then used in the (open-loop iPSC) problem which is an MILP. The two parts (open-loop iPSC and (mps)-MPC) are coupled through the decision variables E_{ip} , O_{ip} and Z_{ijp} , ZF_{ijp} . More specifically, the set of assigned products to the planning period is created and renewed in every planning period, i.e. $I_{active}(p) = \{i | E_{ip} = 1\}$. Notice that every time a rescheduling takes place the set of assigned products is revised accordingly. Next, the production sequence is made known to the controller through the integer variable O_{ip} and the binary variables that indicate the changeovers (Z_{ijp} and ZF_{ijp}) are employed for the derivation of the desired set-points. In Algorithm 1, an algorithmic chart for the integration of closed loop control is given. Initially a nominal production plan and the relevant schedules are computed by solving the open-loop iPSC (Algorithm 1, Step 1). The main computational loop iterates over all the planning periods under consideration (Algorithm 1, Step 2). The sequencing decisions and the set of assigned products are developed and then the schedule is supervised in a logic manner. A scalar k is used to track the order of the product that is currently being processed (Algorithm 1, Steps 3–6). Then following the proposed framework the information goes to the outer control loop where the mps-MPC is employed and the system is regulated around the desired set-point subject to the quality bound (\mathbf{x}_q). If the system stays within this limit (Algorithm 1, Step 10) then we sample the next instance normally; otherwise if the threshold/quality bound is violated that means that we have to fix the products current production time, set the current measurement as the state of the disturbance and go

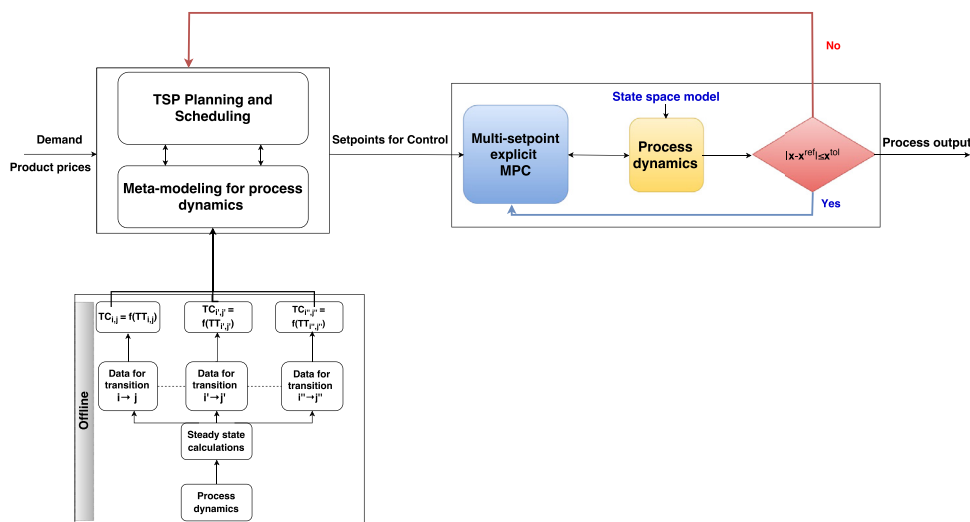


Fig. 7. Flowchart of the proposed closed loop implementation of the iPSC for continuous manufacturing processes.

Input: iPSC problem, ϵ, ω, x_q

Output: Closed loop iPSC solution

```

1: Solve open-loop iPSC model
2: For p = 1, ..., |P|:
3:   Create a set of assigned products  $I_{active}(i) = \{i|E_{ip} = 1\}$ 
4:   Get  $O_{ip}$  and  $Z_{ijp}, Z_{fijp}$ 
5:   Do  $i, j \in I_{active}$  While  $(k \leq |I_{active}| \wedge O_{ip} = k \wedge Z_{ijp}$  or  $Z_{fijp} = 1)$ :
6:     (mp-MPC regulatory)
7:     While  $(T_{mpc} \leq T_{ip})$ :
8:        $x_i^{ss} = x^{sp}$ ;
9:       Do mp-MPC regulatory steps;
10:      If  $(|x_0 - x^{sp}| \leq x_q)$ :
11:         $T_{mpc} = T_{mpc} + h$ ;
12:      Else:
13:         $x_{t+1} = x_d, u_t = u_d$ ;
14:         $T_{mpc} = T_{mpc} + h$ ;
15:        Go To ('Reschedule_PR')
16:      EndDo
17:    EndWhile
18:     $T_{ip} = T_{mpc}$ ;
19:     $T_{mpc} = 0$ ;
20:     (mp-MPC tracking)
21:      $x_0 = x_i^{ss}, x_j^{ss} = x^{sp}$ ;
22:     Do counter While  $(|x_0 - x^{sp}| \geq \omega)$ :
23:       Do mp-MPC tracking steps;
24:       If  $(|d| \leq \epsilon)$ :
25:          $T_{mpc} = h \cdot \text{counter}$ ;
26:          $x_{t+1} = x_t$ ;
27:         counter=counter + 1;
28:       Else:
29:          $x_{t+1} = x_d, u_t = u_d$ ;
30:         counter=counter + 1;
31:          $T_{mpc} = h \cdot \text{counter}$ ;
32:         Go To ('Reschedule_TR')
33:       EndDo
34:     EndWhile
35:      $T_{ijp}^{trans} = T_{mpc}$ ;
36:      $T_{mpc} = 0$ ;
37:   EndWhile
38:    $k=k+1$ 
39: EndDo
40:  $k=0$ 
41: EndFor

```

Algorithm 1. Closed-loop iPSC.

to Algorithm 2 to initiate the rescheduling (Algorithm 1, Steps 10–16). If the nominal production time has been accomplished without disruptions then we set the set-point of the mps-MPC as the steady state of the next product (Algorithm 1, Step 20) and start the changeover. Similar to the previous steps there has been set a threshold (ϵ) for which a disturbance during a transition is supposed to lead to negligible disruption and thus no rescheduling is triggered (Algorithm 1, Steps 21–33); otherwise a rescheduling needs to be initiated and the steps outlined in Algorithm 3 should be followed.

As shown in Algorithm 1 whether the disturbance is detected during a production or a transition period calls for different strategies. During the production period the role of control is to regulate the system around the desired steady state given a quality bound for allowable deviation (x_q), which in the context of iPSC reflects a specific product grade, while during the transition period there is a need for set-point tracking control. In the case that disturbance is detected during the production time of a product, it is assumed that its production is instantly interrupted as the disturbance exceeds the prespecified threshold (ϵ); note that in the present work we follow the convention of Zhuge and Ierapetritou (2012) and these parameters are assumed to be determined via heuristics based on process knowledge. Practically this happens because in real processes there will always exist some noise and model-mismatch and without those tolerances there would be excessive need for rescheduling. Once the disturbance is detected during the production period, it is reasonable to consider

Input: Implemented actions from iPSC

Output: Rescheduled iPSC solution

```

1: Define dummy product "d_name" with  $x_{d\_name}^{ss} = x_d$ 
2: For i = "d_name":
3:   For j = 1, ..., |J|:
4:      $x_0 = x_i^{ss}; x_j^{ss} = x^{sp}$ ;
5:     Do counter While  $(|x_0 - x^{sp}| \geq \omega)$ :
6:       mp-MPC tracking steps
7:        $T_{mpc} = \text{counter} \cdot h$ ;
8:       counter=counter + 1;
9:     EndWhile
10:     $T_{ij}^{trans} = T_{mpc}$ ;
11:     $CT_{ij} = \int_0^{T_{mpc}} u dt$ ;
12:     $T_{mpc} = 0$ ;
13:   EndFor
14: EndFor
15: Define  $I_{past} = \{i|O_{ip} \leq k\}$ 
16: For i, j  $\in I_{past}$ :
17:   Fix  $T_{ip}, Z_{ijp}, E_{ip}, T_{ijp}^{trans}$ 
18: EndFor
19: Find i  $|O_{ip} = k$ , create an alias element, e.g.  $i_d$ , and the corresponding dynamic
    set,  $I_d(p) = \{i, p| E_{ip} = 1 \text{ for } p=p_{current}; 0 \text{ otherwise}\}$ 
20: Duplicate the data for alias element
21: Solve reduced iPSC
22: Set  $k=k+1$ 
23: Go To (Step 3) of Algorithm 1

```

Algorithm 2. Rescheduling routine for production disruption.

Input: Implemented actions from iPSC

Output: Rescheduled iPSC solution

```

1: Define dummy product "d_name" with  $x_{d\_name}^{ss} = x_d$ 
2: For i = "d_name":
3:   For j = 1, ..., |J|:
4:      $x_0 = x_i^{ss}; x_j^{ss} = x^{sp}$ ;
5:     Do counter While  $(|x_0 - x^{sp}| \geq \omega)$ :
6:       mp-MPC tracking steps
7:        $T_{mpc} = \text{counter} \cdot h$ ;
8:       counter=counter + 1;
9:     EndWhile
10:     $T_{ij}^{trans} = T_{mpc}$ ;
11:     $CT_{ij} = \int_0^{T_{mpc}} u dt$ ;
12:     $T_{mpc} = 0$ ;
13:   EndFor
14: EndFor
15: Define  $I_{past} = \{i|O_{ip} \leq k\}$ 
16: For i, j  $\in I_{past}$ :
17:   Fix  $T_{ip}, Z_{ijp}, E_{ip}, T_{ijp}^{trans}$ 
18: EndFor
19: Solve reduced iPSC
20: Set  $k=k+1$ 
21: Go To (Step 3) of Algorithm 1

```

Algorithm 3. Rescheduling routine for transition disruption.

the need of re-assignment of that product within the same planning period and this is achieved via a product duplication. The steps outlined in Algorithm 2 are then followed in order to decide upon the rescheduled optimal sequence that allows for resume of the production. That is, the current measurement of the system is assumed to be the steady state of a dummy product ($x_{d_name}^{ss}$) and the mps-MPC is employed to simulate all the potential transitions to the other products and compute transition times and costs (Algorithm 2, Steps 1–14). Next, the set of products that have already been processed in the current period is constructed (I_{past}) and the relevant timing and sequencing decisions are fixed (Algorithm 2, Steps 15–17). Notice that by fixing the timing de-

cisions, production decisions are also fixed given the assumption about constant production rate. Since the disruption was detected during a production period, it should be allowed to the model to choose the resume of production and to do so we follow the product duplication concept and define an alias element in the products set that is set to be active only for the current planning period and then we proceed in solving the reduce iPSC (Algorithm 2, Steps 19–23). We employ the term “reduced iPSC” since the open-loop iPSC model is solved in reduced decision space, i.e. with the past decisions fixed.

Another case involves the disturbance detection and rejection during the transition time from one product to the other. In that case, there is no need to account for product duplication and we only introduce a new dummy product that represents the current state of the system after the disturbance is realised. Similar to Algorithm 2, we exploit the ability of mp-MPC to run simulations in very fast times and the potential transition times and costs are computed (Algorithm 3, Steps 1–14). The output of these computations are passed to the model and the past decisions are fixed (Algorithm 3, Steps 15–18) and the reduced iPSC is solved again to define the optimal rescheduling action (Algorithm 3, Steps 19–21). A summary of these steps is given in Algorithm 3.

Overall, the integrated problem is formulated and solved as an MILP and the closed-loop is achieved through the rescheduling mechanism outlined in this section via the use of the proposed mp-MPC.

4. Case studies

In this section the closed loop implementation of iPSC is illustrated through two case studies, where each planning period is assumed to be equal to one week. In all the case studies presented, it is assumed that every system exhibits multiple steady states at which a specific product is produced in a single CSTR while the occurrence of idle time results in the related start-up and shut-down requirements from a systems dynamics perspective. Moreover, a short discussion on the results is conducted at the end of each case study. All the optimisation problems, are formulated and solved using GAMS 24.7.4, on a Dell workstation with 3.70 GHz processor, 16GB RAM and Windows 7 64-bit operating system using CPLEX 12.6.1 for the solution of the MILPs and BARON 16.8.24 (Sahinidis, 1996) for the solution of NLPs. BARON was chosen for the comparison between the NLMPC and mp-MPC scheme because it is a global optimisation solver and the explicit solutions computed for the design of the mp-MPC controllers are globally optimal as well.

4.1. Single input single output CSTR

First, a case study involving a SISO multi-product CSTR is considered. Based on the concentration (C_R) and the volumetric flow of the reactant (Q_R) a number of products can be manufactured at different steady state operating conditions; the related data are given in Table 1. From the control perspective, the state variable of the system is the concentration of the reactant, while the control input is the volumetric flow of the liquid. A conceptual representation of the related system is given in Fig. 8. The reaction is 3rd order irreversible, i.e. $R \xrightarrow{k} 3P$, $-\mathcal{R}_R = kC_R^3$. The nonlinear dynamic model of the system is given by Eq. (44)

$$\frac{dC_R}{dt} = \frac{Q_R}{V} (C_0 - C_R) + \mathcal{R}_R \quad (44)$$

where C_0 denotes the concentration of the reactant in the feed stream, V is the reactor volume and k is the reaction's kinetic constant.

In order to design the multi-setpoint explicit controller the system's model is transformed into its algebraic equivalent. In this

Table 1
Data of SISO CSTR case study.

Cost data of SISO CSTR.		
Product	OC_i ($\frac{\text{mmu}}{\text{mol}}$)	P_i ($\frac{\text{mmu}}{\text{mol}}$)
A	0.13	200
B	0.22	150
C	0.35	130
D	0.29	125
E	0.25	120
F	0.18	180
G	0.27	124
H	0.29	140
Dynamic data of SISO CSTR.		
Product	x_i^{ss} ($\frac{\text{mol}}{\text{l}}$)	u_i^{ss} ($\frac{\text{l}}{\text{h}}$)
A	0.0967	10
B	0.2	100
C	0.3032	400
D	0.393	1000
E	0.5	2500
F	0.15	39.7
G	0.45	1656.8
H	0.247	200.1

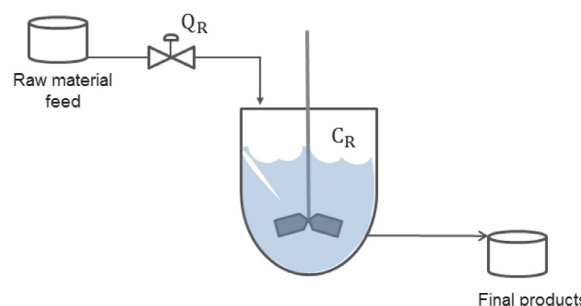


Fig. 8. SISO CSTR production scheme, the manipulated variable is the volumetric flowrate of the liquid (Q_R) while the state variable is the concentration of the reactant (C_R).

work, for the sake of simplicity an Euler integration scheme is followed and the problem is formulated as shown in Eq. (45).

$$\min_{u(t)} J(\theta) = \sum_{t=0}^{t_{PH}} \|x(t) - x_{ref}\|_2$$

Subject to:

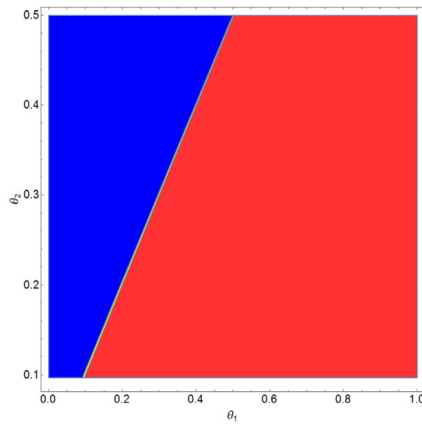
$$\begin{cases} \frac{dx}{dt} = \frac{u}{V} (C_0 - x) + \mathcal{R}_R \\ x(t|_{t=0}) = \theta_1 \\ x^{ref} = \theta_2 \\ 0 \leq x(t) \leq 1, \quad 0 \leq t \leq t_{PH} \\ 0 \leq u(t) \leq 3000, \quad 0 \leq t \leq t_{PH} \end{cases} \quad (45)$$

For the design of multi-setpoint explicit controller the algorithm outlined in Section 3.2 was employed; the interested reader is referred to Charitopoulos et al. (2018) for a more detailed exposition on the main algorithmic steps. The mp-MPC was designed for prediction horizons of unity, two and three; the final globally optimal explicit solutions along with their corresponding CRs are given in Table 2 while the final partition of the parametric space is given in Fig. 9. The corresponding explicit solutions are given in Table 3. With regards to the computational behavior of the proposed scheme solving the corresponding mp-NLP for $t_{PH} = 1$ takes 22.65 s, $t_{PH} = 2$ takes 262.43 s and $t_{PH} = 3$ takes 2540.65 s. As typically observed in the mp-P literature the computational effort grows rapidly with the number of variables and constraints (Oberdieck et al., 2016).

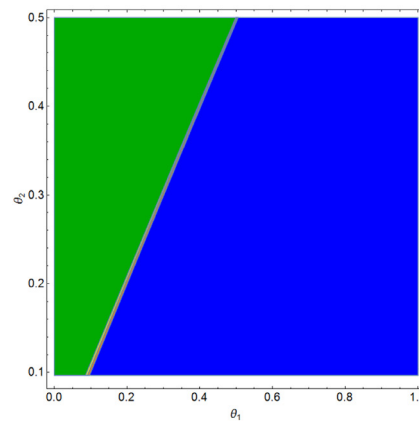
Once the multi-setpoint explicit controller is designed we investigate its performance in comparison to the use of conventional MPC within the context of iPSC. First we consider the case of no imposed disturbances and then at specific time an additive distur-

Table 2
Final CRs and explicit solutions (control law and state evolution) for $t_{PH} = 1$ for the SISO CSTR case study.

CRs	Mathematical expression	Explicit solution
$CR_1 :=$	$\theta_2 \geq 0.0967 \begin{cases} 0.096718 \leq \theta_1 \leq 0.503 \\ 0.02\theta_1^3 + \theta_2 \leq \theta_1 \\ 0.503 \leq \theta_1 \leq 1 \\ \theta_2 \leq 0.5 \end{cases}$	$\begin{cases} C_R(t=1) = \theta_1 - \frac{\theta_1^3}{50} \\ Q_R(t=0) = 0 \end{cases}$
$CR_2 :=$	$\theta_2 \leq 0.5 \begin{cases} \theta_2 \geq 0.0967 \begin{cases} \theta_1 = 1 \\ 0 \leq \theta_1 \leq 0.09126 \end{cases} \\ 0.09126 \leq \theta_1 \leq 0.4994 \\ 0.994\theta_1 + 0.006 \leq 0.02\theta_1^3 + \theta_2 \\ 0.0967 \leq \theta_2 \\ 0 \leq \theta_1 \leq 0.09126 \end{cases}$	$\begin{cases} C_R(t=1) = \frac{(3-10\theta_1^2+497\theta_1)}{500} \\ Q_R(t=0) = 3000 \end{cases}$
$CR_3 :=$	$\begin{cases} 0.0913 < \theta_1 \leq 0.0967 \\ 0.0967 \leq \theta_2 \leq 0.002(3 + 497\theta_1 - 10\theta_1^3) \\ 0.499 \leq \theta_1 \leq 0.503 \\ 0.02(50\theta_1 - \theta_1^3) \leq \theta_2 \leq 0.5 \\ 0.0967 \leq \theta_1 \leq 0.4994 \\ 0.02(50\theta_1 - \theta_1^3) \leq \theta_2 \leq 0.002(-10\theta_1^3 + 497\theta_1 + 3) \end{cases}$	$\begin{cases} C_R(t=1) = \theta_2 \\ Q_R(t=0) = \frac{10000(\theta_1^3 - 50\theta_1 + 50\theta_2)}{1 - \theta_1} \end{cases}$



(a) Final CRs of the SISO CSTR case study for $t_{PH} = 1$.



(b) Final CRs of the SISO CSTR case study for $t_{PH} = 2$.

Fig. 9. Optimal partition of the parametric space for the SISO CSTR case study.

Table 3
Final explicit optimal solutions for $t_{PH} = 2$ for the SISO CSTR case study.

if $(\theta_1, \theta_2) \in CR_1$ then	$\begin{cases} C_R(t=1) = \theta_1 - \frac{\theta_1^3}{50} \\ C_R(t=2) = -\frac{1}{50} \left(\theta_1 - \frac{\theta_1^3}{50} \right)^3 - \frac{\theta_1^3}{50} + \theta_1 \\ Q_R(t=0) = 0 \\ Q_R(t=1) = 0 \end{cases}$
if $(\theta_1, \theta_2) \in CR_2$ then	$\begin{cases} C_R(t=1) = \theta_1 - \frac{\theta_1^3}{50} \\ C_R(t=2) = \theta_2 \\ Q_R(t=0) = 0 \\ Q_R(t=1) = -\frac{4(\theta_1(\theta_1^2 - 50)(\theta_1^2(\theta_1^2 - 50)^2 - 125000) - 6250000\theta_2)}{\theta_1^3 - 50\theta_1 + 50} \end{cases}$
if $(\theta_1, \theta_2) \in CR_3$ then	$\begin{cases} C_R(t=1) = \frac{(3-10\theta_1^2+497\theta_1)}{500} \\ C_R(t=2) = \frac{(-2\theta_1^3 + \frac{(10\theta_1^2 - 497\theta_1 - 3)^3}{62500000} + \frac{3(10\theta_1^2 - 497\theta_1 + 497)}{2500} + 100\theta_1 - \frac{3(\theta_1 - 1)}{5})}{100} \\ Q_R(t=0) = 3000 \\ Q_R(t=1) = 3000 \end{cases}$
if $(\theta_1, \theta_2) \in CR_4$ then	$\begin{cases} C_R(t=1) = \frac{(3-10\theta_1^2+497\theta_1)}{500} \\ C_R(t=2) = \theta_2 \\ Q_R(t=0) = 3000 \\ Q_R(t=1) = \frac{-1000\theta_1^3 + 149100\theta_1^2 + 900\theta_1 - 7410270\theta_2^3 - 89460\theta_1^4}{25(10\theta_1^3 - 497\theta_1 + 497)} + \frac{247763203\theta_1^3 + 2223081\theta_1^2 - 6212486581\theta_1 + 6250000000\theta_2 - 37499973}{25(10\theta_1^3 - 497\theta_1 + 497)} \end{cases}$
if $(\theta_1, \theta_2) \in CR_5$ then	$\begin{cases} C_R(t=1) = \theta_2 \\ C_R(t=2) = \theta_2 \\ Q_R(t=0) = \frac{10000(\theta_1^3 - 50\theta_1 + 50\theta_2)}{1 - \theta_1} \\ Q_R(t=1) = \frac{10000\theta_2^2}{1 - \theta_2} \end{cases}$

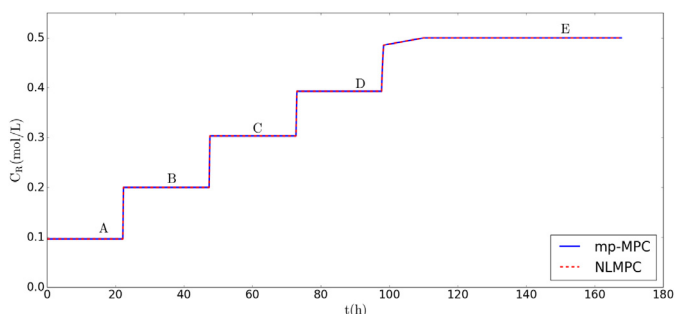


Fig. 10. Comparison of the closed loop behavior of the state of the SISO CSTR (C_R) using explicit MPC (blue continuous line) and conventional MPC (red dashed line) for prediction horizon of unity (1st planning period). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

bance is imposed and the dynamic behavior of the closed loop system is evaluated.

4.1.1. Case 1: No additive disturbance imposed

First the closed loop behavior for prediction horizon of unity and then two is evaluated for the mp-MPC, the threshold values are set to $\omega = 10^{-6}$, $\epsilon = \pm 5\%$ and no imposed additive disturbance is considered, while a planning horizon of two weeks is employed. When the mps-MPC is employed, it takes 0.1705 CPU s for the nominal iPSC solution to be computed. For the case of the conventional NLMPC the same results are computed at 291.92 CPU s, a rather considerable difference in terms of computational effort. It is important to note that based on Fig. 10, the dynamic response of the system as computed by the mp-MPC and the conventional NLMPC are identical.

The same instance of the case study was investigated, by employing prediction horizon of 2 for both the explicit and the conventional MPC schemes. Regarding the dynamic response and the stability of the underlying control system, the results indicate no difference when compared to the ones computed for prediction horizon of unity. As far as the online computational complexity is concerned, for the case of the multi-setpoint explicit MPC it takes 0.1707 s for the whole iPSC to be solved and validated in a closed loop manner while the same problem takes 831.24 s using the conventional MPC using BARON 16.3.4 and optimality tolerance of 10^{-5} .

As mentioned earlier in the article, real process systems are subject to a number of disturbances that may affect significantly the performance of the process. For the case that no disturbances are accounted for, the solution of the open loop and the closed loop iPSC are identical as demonstrated in case 1. However, under the effect of disturbances the need of feedback control mechanism becomes crucial. In the next two cases we investigate the effect of the implementation of the closed-loop strategy under the occurrence of disturbances that lead to state deviation.

4.1.2. Case 2: State deviation during the transition period

In this case we assume that the nominal iPSC has been solved and the optimal decisions begin to be applied to the plant. During

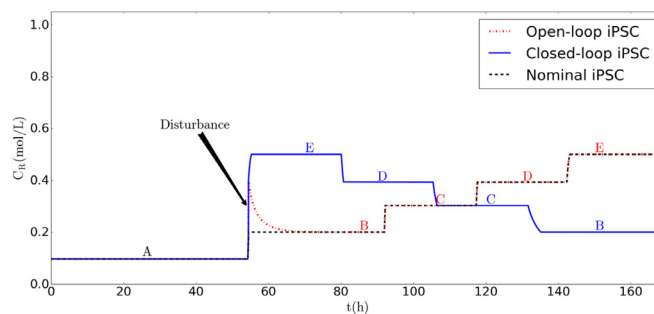


Fig. 11. Comparative plot of the state deviation during the transition period.

the first planning period, a disturbance is detected 12 minutes after the beginning of the transition from product A to product B and its magnitude exceeds the prespecified threshold. The rescheduling mechanism is triggered and first the current state of the system is passed to the mp-MPC according to the steps outlined in Algorithm 3. It takes 0.008s for the mp-MPC to compute the candidate transition times and costs which are subsequently passed to the iPSC rescheduling model in order to compute the next optimal step. In this instance, the rescheduling mechanism dictates the change in the nominal sequence and instead of B the system is driven to the production of E while the production of B is set to be the last of the planning period. A graphical representation of this instance is given in Fig. 11.

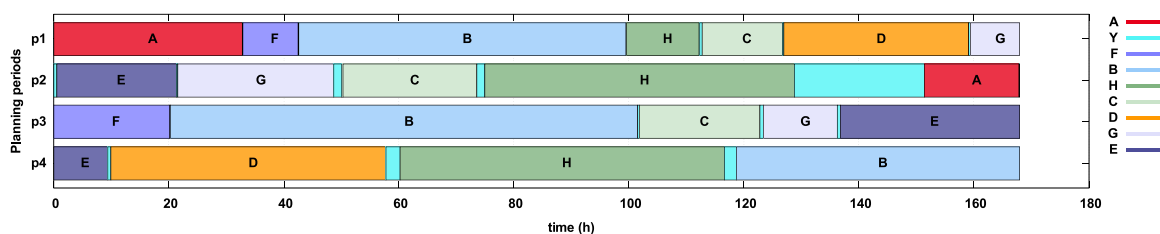
On the other hand, if the iPSC solution was applied without any feedback mechanism that would effectively close the loop, the pre-computed nominal control would have been applied to the system regardless of the disturbance occurrence. The impact of the disturbance on the open-loop framework was simulated by fixing all the relevant decisions and as shown in Fig. 11, it results in significantly extended transition time from product A to B which results in turn in considerable reduction of the production time and thus increase in the backlog of the corresponding unmet demand. A summary of the results is given in Table 4.

4.1.3. Case 3: Multiple disturbances over the planning horizon

In this case we consider 8 products and 4 planning periods. Solving the nominal problem, no disturbance is assumed and it takes 9.984 s for CPLEX 12.6.3 to compute the optimal solution. Within the first planning period, during the transition from H to C, 4.2 min after the transition has started (nominal transition time is 6.6 min), a state deviation is realised which exceeds the threshold and is equal to $0.04 \frac{\text{mol}}{\text{L}}$ resulting in a concentration of $0.329 \frac{\text{mol}}{\text{L}}$, after its realisation the proposed framework is employed and the need for rescheduling is examined. First, the explicit controller is used for a simulation between all the remaining products of the set $I_{\text{active}}(p)$ and then the reduced iPSC is solved (7.083 s) for the remainder of the planning horizon. In this case, the optimal solution dictates that it is preferable to keep on the prolonged transition rather than switching production to another product. The result of the extended transition is the decrease of production time of product G and its subsequent increase during planning period 2.

Table 4
Comparative results for the solutions computed by the closed loop iPSC and the open loop iPSC.

	Open loop no disturbance	Closed loop with disturbance	Open loop with disturbance
iPSC solution	A → B → C → D → E E → C → B → A	A → E → D → C → B B → A → C → E	A → B' → C → D → E E → C → B → A
Profit (rmu)	$9.18 \cdot 10^6$	$9.174 \cdot 10^6$	$8.65 \cdot 10^6$
Backlog cost (rmu)	62331.9	68729.1	338677.65
Total CPU (s)	0.152 (mp-MPC)/ 831.24 (NLMPC)	0.27 (mp-MPC)	0.152 (mp-MPC)



(a) Gantt chart for production schedule of the four planning periods nominally.

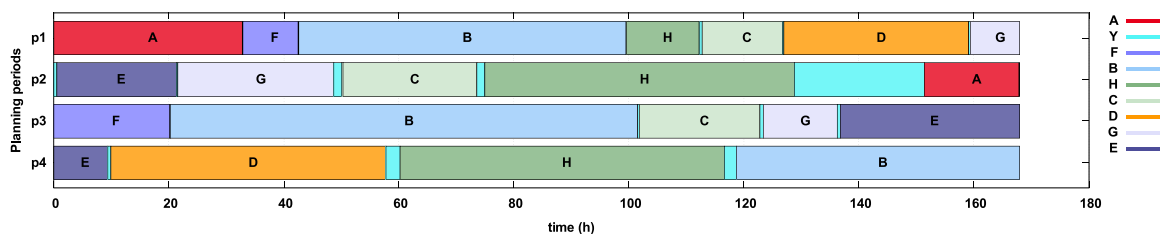
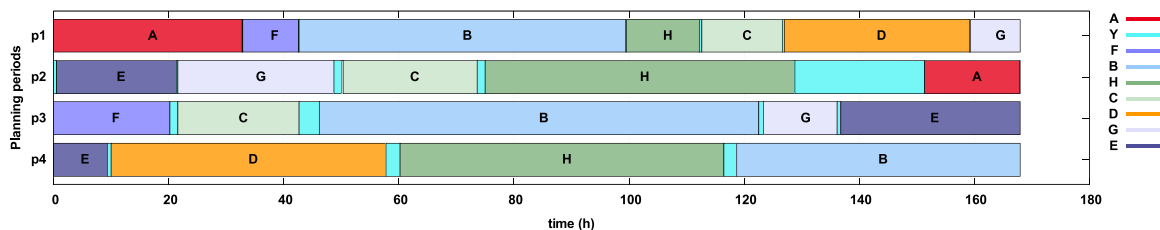
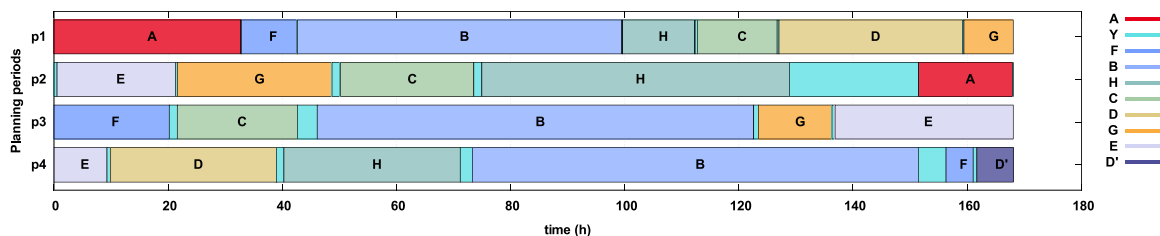
(b) Gantt chart for production schedule of the four planning periods after rescheduling in period p_1 .(c) Gantt chart for production schedule of the four planning periods after rescheduling in period p_2 .(d) Gantt chart for production schedule of the four planning periods after rescheduling in period p_4 .

Fig. 12. Gantt charts of the closed-loop iPSC for the SISO CSTR case study. The horizontal axis represent the time (h) and each week spans across 168 h with changeovers indicated using the “Y” label.

Moving on to period 2, another dynamic disruption is realised during the transition from G to C (after 1.4 h of its start) and the system is led to $x_{d_2} = 0.2306 \frac{\text{mol}}{\text{L}}$. Following Algorithm 1, this measurement is passed to mp-MPC for the calculation of the potential transition times and costs. This instance of the closed-loop integration is quite interesting as the rescheduling mechanism for the current planning period generates a completely different solution for the remaining iPSC and this can be visualised in Gantt chart that is given in Fig. 12.

As shown in Fig. 12 the disruption results in decrease of production of product C that in order to be rectified the sequencing decisions in planning period p_3 were revised. Finally, during period 4, the case of dynamic disruption during production period is examined. While product D is being produced there is a dis-

crepancy in the system's dynamics. Due to this discrepancy during the production period the steps outlined in Algorithm 2 are followed and a duplicate product of D is created and the reduced iPSC is solved again with fixed the past decisions. As shown also in Fig. 12, it was computed that the optimal corrective move would be to return in the production of D. A graphical interpretation of the system's dynamics throughout the 4 planning periods is given in Fig. 13, where also the last rescheduling instance is illustrated in more details.

4.2. Methyl-methacrylate polymerisation reactor

Next the closed-loop iPSC of an isothermal methyl-methacrylate (MMA) polymerisation CSTR is studied. The free radical polymeri-

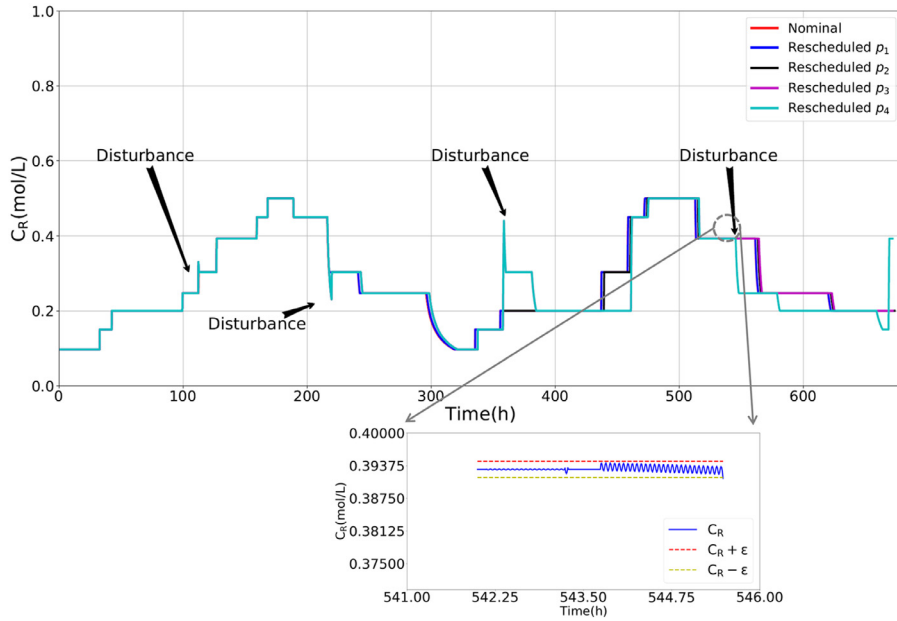


Fig. 13. $C_{R=f(t)}$ plot indicative of the system's dynamics over the entire planning horizon.

Table 5
Decision variables of the MMA kinetic model.

C_m ($\frac{\text{kmol}}{\text{m}^3}$)	State: concentration of monomer
C_I ($\frac{\text{kmol}}{\text{m}^3}$)	State: concentration of initiator
D_0 ($\frac{\text{kmol}}{\text{m}^3}$)	State: dead chains molar concentration
D_1 ($\frac{\text{kg}}{\text{m}^3}$)	State: dead chains mass concentration
F_I ($\frac{\text{m}^3}{\text{h}}$)	Input: flow rate of initiator
$y = D_1/D_0$ ($\frac{\text{kg}}{\text{kmol}}$)	Output: molecular weight

Table 6
Kinetic data for the MMA polymerisation reactor.

$F = 10.0 \text{ m}^3/\text{h}$	Monomer flowrate
$V = 10.0 \text{ m}^3$	Reactor volume
$f^* = 0.58$	Initiator efficiency
$k_p = 2.50 \times 10^6 \frac{\text{m}^3}{\text{kmol}\cdot\text{h}}$	Propagation rate constant
$k_{Td} = 1.09 \times 10^{11} \frac{\text{m}^3}{\text{kmol}\cdot\text{h}}$	Termination by disproportionation Rate constant
$k_{Tc} = 1.33 \times 10^{10} \frac{\text{m}^3}{\text{kmol}\cdot\text{h}}$	Termination by coupling Rate constant
$C_{Iin} = 8.00 \text{ kmol}/\text{m}^3$	Inlet initiator concentration
$C_{min} = 6.00 \text{ kmol}/\text{m}^3$	Inlet monomer concentration
$k_{fm} = 2.45 \times 10^3 \frac{\text{m}^3}{\text{kmol}\cdot\text{h}}$	Chain transfer to monomer rate constant
$k_i = 1.02 \times 10^{-1} \text{ h}^{-1}$	Initiation rate constant
$M_m = 100.12 \text{ kg}/\text{kmol}$	Molecular weight of monomer

sation reaction takes place in an isothermal CSTR at the temperature of 335K, where MMA is produced using azobis (isobutyronitrile) as initiator and toluene as solvent. The mathematical model is adopted from Chu and You (2012) and is given by Eqs. (46)–(50). The system involves 4 state variables, i.e. the concentration of the monomer (C_m), the concentration of the initiator (C_I), the molar concentration of the dead chains (D_0) and the mass concentration of the dead chains (D_1), one control input, i.e. the flowrate of the initiator (F_I) and one output, i.e. the molecular weight of the polymer produced (y). Based on different steady states that the system exhibits it is possible to produce different polymeric grades which correspond to different molecular weights and each grade forms a product within the iPSC framework. The notation used in the present case study is given in Table 5, while the values of kinetic parameters are given in Table 6.

$$\frac{dC_m}{dt} = -(k_p + k_{fm})\sqrt{\frac{2f^*k_iC_I}{k_{Td} + k_{Tc}}}C_m + \frac{F(C_{min} - C_m)}{V} \quad (46)$$

Table 7
Steady state information about the different polymer grades of the MMA CSTR case study.

Product	C_m^{ss}	C_I^{ss}	D_0^{ss}	D_1^{ss}	Y^{ss}	F_I^{ss}
A	3.2285	0.1216	0.0163	277.47	17,000	0.1675
B	3.0780	0.1487	0.0195	292.54	15,000	0.2049
C	3.3667	0.1009	0.0138	263.64	19,000	0.1391
D	3.3331	0.1056	0.0144	261.01	18,500	0.1455
E	3.4635	0.0885	0.0123	253.95	20,500	0.1219
F	3.5552	0.0780	0.0111	244.76	22,000	0.1075
G	3.7257	0.0615	0.0091	227.69	25,000	0.0847
H	3.8815	0.0491	0.0075	212.09	28,000	0.0677
I	3.9786	0.0426	0.0067	202.38	30,000	0.0587
J	4.1154	0.0346	0.0057	188.68	33,000	0.0476
K	4.2015	0.0302	0.0051	180.06	35,000	0.0416
L	4.3238	0.0248	0.0044	167.81	38,000	0.0341
M	4.4013	0.0217	0.0040	160.05	40,000	0.0299
N	4.4760	0.0191	0.0036	152.57	42,000	0.0263
O	4.5302	0.0173	0.0033	147.15	43,500	0.0239
P	4.5830	0.0157	0.0031	141.86	45,000	0.0217

$$\frac{dC_I}{dt} = \frac{F_I C_{Iin} - F C_I}{V} - k_i C_I \quad (47)$$

$$\frac{dD_0}{dt} = (0.5k_{Tc} + k_{Td})\frac{2f^*k_iC_I}{k_{Td} + k_{Tc}}C_m + k_{fm}\sqrt{\frac{2f^*k_iC_I}{k_{Td} + k_{Tc}}}C_m - \frac{F D_0}{V} \quad (48)$$

$$\frac{dD_1}{dt} = M_m(k_p + k_{fm})\sqrt{\frac{2f^*k_iC_I}{k_{Td} + k_{Tc}}}C_m - \frac{F D_1}{V} \quad (49)$$

$$y = \frac{D_1}{D_0} \quad (50)$$

In order to demonstrate the merits of the proposed framework we assume that there are 16 polymer grades that can be produced and their corresponding steady state information are given in Table 7; notice that for the sake of space the values are given with truncated decimal points while for the numerical calculation the precision was up to 10 decimal places.

Table 8

Economic data of the MMA case study.

Product	OC_i [$\frac{mmu}{kg}$]	P_i [$\frac{mmu}{kg}$]	r_i [$\frac{kg}{h}$]
A	263	388.5	100.529
B	188	252.8	122.937
C	163	247.8	83.457
D	226	293	87.325
E	220	330.5	73.165
F	210	260	64.504
G	190	290	50.833
H	240	350	40.635
I	230	395	35.214
J	290	325	28.607
K	205	310	24.997
L	210	316	20.500
M	183	220	17.996
N	155	260	15.814
O	149	300	14.359
P	134	324	13.040

Table 9

Computational study of CPU (s) for different prediction horizons using BARON with zero relative optimality gap.

H_p	CPU (s)
1	602.676
2	1905.328
5	3600 ^a
10	3600 ^a
20	3600 ^a

^a The solver failed to converge within 3600 s.**Table 10**

Uncertain parameters of mps-MPC for the MMA CSTR case study.

Uncertain parameter	Bounds	Correlation
θ_1	$\in [0, 5]$	$C_m _{t=0}$
θ_2	$\in [0, 0.5]$	$C_i _{t=0}$
θ_3	$\in [0, 0.05]$	$D_0 _{t=0}$
θ_4	$\in [0, 300]$	$D_i _{t=0}$
θ_5	$\in [0, 5]$	Set-points for C_m
θ_6	$\in [0, 0.5]$	Set-points for C_i
θ_7	$\in [0, 0.05]$	Set-points for D_0
θ_8	$\in [0, 300]$	Set-points for D_i

With regards to the economic data a summary of the selling prices and costs is given in Table 8 while inventory and backlog costs are calculated as 10% and 30% of the product's selling price, P_i .

Since none of the nonlinear terms in the system is transcendental, the proposed methodology for the design of the explicit controller can be employed. More specifically, the infinite dimensional optimal control problem is transformed into a finite one with the employment of a discretisation scheme. The present case study has been studied by a considerable number of researchers and it has been reported that its optimal control through numerical schemes is rather challenging due to numerical instabilities that may arise during the discretisation. To this effect, the most common discretisation scheme used is orthogonal collocation on finite elements because of its inherent properties of numerical stability. However, in the present work we employed forward Euler discretisation with a step size of $h_e = 0.01$ h as trade-off between computational com-

plexity and stability of the integration scheme. A number of simulations were conducted so as to decide on a step size that can be small enough so as to avoid oscillatory behavior but not too small so as to avoid an extenuating repetitive solution that would result in further computational effort. Moreover, as will be shown in the results the nature of proposed methodology, i.e. the analytical and not numerical solution of the mp-MPC problem, enhances the robustness of the solutions computed as numerical instabilities were circumvented.

Following the proposed methodology, first the set of differential equations involved in the problem is discretised and then the parametric optimisation problem is formulated and solved in Mathematica 11. At this point, the prediction horizon was set to be equal to unity as from offline simulations the enhancement of the controller's stability was not highly affected. On the contrary, following the proposed methodology the global optimality of the explicit solutions is guaranteed whereas using readily available numerical solvers for online implementation of globally optimal solutions results in rather exhaustive computational times given the need for rapid solutions within the context of iPSC. For illustration purposes the results of a comparative study using BARON 16.8.24 solver in GAMS with varying prediction horizons is provided in Table 9.

In order to design the explicit controller we consider 8 uncertain parameters, 4 for each state and 4 for each family of set-points. From previous experience within our group the number of uncertain parameters considered does not affect the computational complexity of the solution procedure as the uncertain parameters are treated in a symbolic way. In Table 10, the notation for the uncertain parameter introduced in the present example is given. Note that the bounds on the uncertain parameters regarding the set-points are continuous. It can be argued that having these parameters as continuous may lead to unnecessary computations but one could argue that in the context of enterprise wide optimisation where the supervisory controller receives data from the RTO layer the constant use of the same set-points is not guaranteed. In that case a conventional explicit controller would have to be re-designed from the beginning (solution of the corresponding mp-P problem, storage of the explicit solutions, possibly, in a micro-chip) whereas following the methodology proposed herein, there would be no need for that, under the assumption that the bounds used initially are the feasible range of the system.

The corresponding mp-NLP problem consists of one optimisation variable, the control input, ten constraints and eight uncertain parameters. Note that even though the optimisation variable is one, the variables for which we seek analytical solution are eleven, i.e. the optimisation variable and the Lagrange multipliers of the constraints. Solving the mp-NLP results in 5 candidate solutions which are shown in Table 11. An interesting observation is that the underlying control law is linear function function of the uncertain parameters and most specifically of the concentration of the initiator at each sampling instance and the set point for that state, even though the original optimisation problem is highly nonlinear.

Following the proposed mp-NLP algorithm, the final set of globally optimal explicit solutions are 3. The mathematical definition of CR_2 is given by Eq. (51) as an indicator of the non-convex nature of the underlying parametric programming problem.

Table 11

Candidate solutions for explicit controller of the MMA case study.

	F_1	λ_1	λ_2	λ_3	λ_4	λ_5	λ_6	λ_7	λ_8	λ_9	λ_{10}
1	$125\theta_6 - 124$	0	0	0	0	0	0	0	0	0	0
2	0.4	$-0.0158237\theta_2 + 0.016\theta_6 - 0.0000512$	0	0	0	0	0	0	0	0	0
3	0	0	$0.0158237\theta_2 - 0.016\theta_6$	0	0	0	0	0	0	0	0
4	$-124\theta_2$	0	0	0	$-2\theta_6$	0	0	0	0	0	0
5	$62.5 - 124\theta_2$	0	0	0	0	0	0	$2\theta_6 - 1$	0	0	0

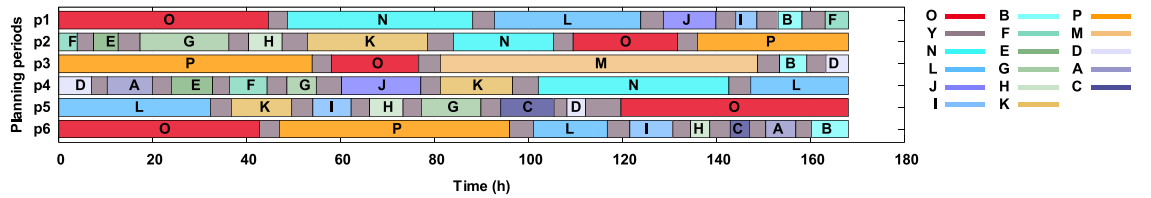
$$\text{CR}_2 = \left\{ \begin{array}{l} 0 \leq \theta_3 \\ 0 \leq \theta_4 \\ 0 \leq \theta_5 \leq 5 \\ 0.0032 + \\ 0.988\theta_2 \leq \theta_6 \\ \theta_6 \leq 0.5 \\ 0 \leq \theta_7 \leq 0.05 \\ 0 \leq \theta_8 \leq 300 \end{array} \right. \left\{ \begin{array}{l} \theta_1 = 1.8769 \left\{ \begin{array}{l} 0.42 \leq \theta_2 \leq 0.5 \\ 0.00113\theta_2 + 0.0000457\sqrt{\theta_2} + \theta_3 \leq 0.05 \\ 4.6722\sqrt{\theta_2} + \theta_4 \leq 303.03 \\ 0 \leq \theta_2 \leq 0.42 \\ \theta_3 \leq 0.05 \\ \theta_4 \leq 300 \end{array} \right. \\ 2.43 \cdot 10^{-5}\sqrt{\theta_1^2\theta_2} + \\ 0.00113\theta_2 + \theta_3 \leq 0.05 \left\{ \begin{array}{l} \theta_4 + 2.49\sqrt{\theta_2}\theta_1 \leq 303.03 \\ \theta_2 \leq 0.5 \end{array} \right. \\ \left\{ \begin{array}{l} 1.72 < \theta_1 \\ 2.319 \cdot 10^{-5}\theta_1^2 + 0.446 < \\ 8.85 \cdot 10^{-16}\sqrt{6.86 \cdot 10^{22}\theta_1^4 + 2.64 \cdot 10^{26}\theta_1^2} + \theta_2 \\ \theta_2 \leq \frac{1.48}{\theta_1^2} \end{array} \right. \left\{ \begin{array}{l} \frac{1.48}{\theta_1^2} < \theta_2 \\ 1.72 < \theta_1 < 1.877 \\ 1.87695 \leq \theta_1 \leq 5 \\ 2.319 \cdot 10^{-5}\theta_1^2 + 0.446 < \\ 8.85 \cdot 10^{-16}\sqrt{6.86 \cdot 10^{22}\theta_1^4 + 2.64 \cdot 10^{26}\theta_1^2} \\ + \theta_2 \\ 0 \leq \theta_1 \leq 1.72 \\ 2.319 \cdot 10^{-5}\theta_1^2 + 0.446 < \\ 8.85 \cdot 10^{-16}\sqrt{6.86 \cdot 10^{22}\theta_1^4 + 2.64 \cdot 10^{26}\theta_1^2} \\ + \theta_2 \\ \theta_4 \leq 300 \end{array} \right. \\ \left\{ \begin{array}{l} 1.877 < \theta_1 \leq 5 \\ \theta_2 > \frac{1.48}{\theta_1^2} \\ \theta_4 + 2.49\sqrt{\theta_2}\theta_1 \leq 303.03 \\ 0 \leq \theta_1 < 1.879 \\ \theta_4 \leq 300 \end{array} \right. \\ \left\{ \begin{array}{l} 1.87 < \theta_1 \leq 4.989 \\ \theta_4 \leq 300 \\ \theta_1 > 4.989 \\ \frac{40280.2 - 16144.7\theta_1}{\theta_1^2} + 1617.74 \leq \theta_2 \end{array} \right. \\ \theta_3 \leq 0.05 \left\{ \begin{array}{l} 8.85 \cdot 10^{-16}\sqrt{6.86 \cdot 10^{22}\theta_1^4 + 2.64 \cdot 10^{26}\theta_1^2} \\ + \theta_2 \leq 2.319 \cdot 10^{-5}\theta_1^2 + 0.446 \\ \theta_2 \leq \frac{1.48}{\theta_1^2} \end{array} \right. \end{array} \right. \quad (51)$$

After the explicit controller has been designed the explicit solutions and their CRs are coded in GAMS in the form of conditional statements where they are integrated with the open loop iPSC model so as to validate the computed solution and reject systematic disturbances; thus closing the loop in the iPSC.

4.2.1. Discussion of results

The overall closed-loop iPSC problem is formulated as an MILP with 15,334 equations, 10,376 continuous variables and 3485 binary variables while the solution time for the nominal case takes 26.12 s using CPLEX 12.6.3 in GAMS. It is worth to note that the optimisation dictates to run the plant in a continuous way through the planning period so as to avoid the related long start-up/shut-down times. In this case study we account for minimum production time of 4h, i.e. for a product to be assigned to a planning period at least 4h should be allocated for its production. As mentioned earlier in the article, the main requirement for the efficient online implementation of the closed-loop iPSC solution is that the computational time needed for the solution of the optimisation problem is less than sampling time of the underlying control system. For this case study, the sampling time is 36 s and thus it requires to solve the iPSC in times less than 36 s. The parameters for the rescheduling framework in this case study were set as $\omega = 10^{-6}$, $\epsilon = \pm 5\%$. The first rescheduling action takes place during the production of product O, when a disturbance that exceeds

the threshold is realised and the rescheduling mechanism is triggered. Following the steps of Algorithm 1 a new duplicate product is generated and the reduced iPSC solved with fixed the past decisions, in this case the production time of O that was achieved and the relevant binary decisions with respect the order of O in the schedule. Solving the reduced iPSC takes 28.83 s, slightly increased when compared to the nominal case but this increase is due to the introduction of the two dummy products in the reduced iPSC, i.e. the duplicate of O and the dummy disturbance product. As shown in the Gantt chart in Fig. 14, the reschedule results in interrupting the production of O and move to the production of N while compared to the nominal plan in period p_1 the additional production of product G was chosen too. The next disturbance is detected during the transition from J to I and once again the rescheduling mechanism is activated but this time the system is controlled towards the completion of the nominal transition while the related computational time is 7.96 s. Similarly to planning period 1, in period 2 another reschedule takes place due to dynamic disruption during the transition from product E to product G and the system is resumed to its nominal schedule with the completion of the original transition. During period 3 the no dynamic disturbance that exceeded the relevant threshold was detected. Period 4 involved a reschedule during the production of product K, 5.36 h after its initialisation and the rescheduling framework resulted in resuming its production. Finally, during period 6 another instance



(a) Nominal gantt chart for the entire planning horizon.

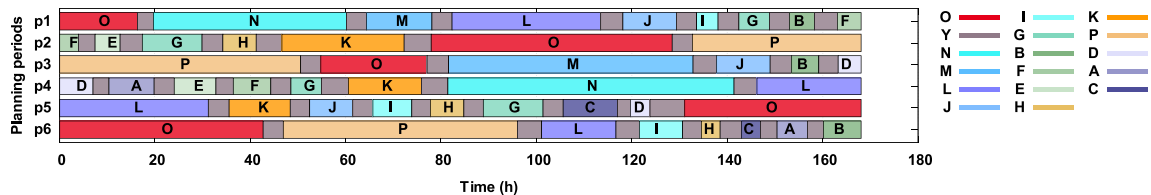
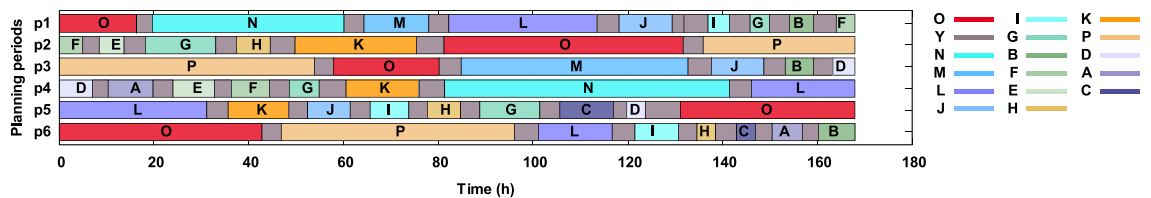
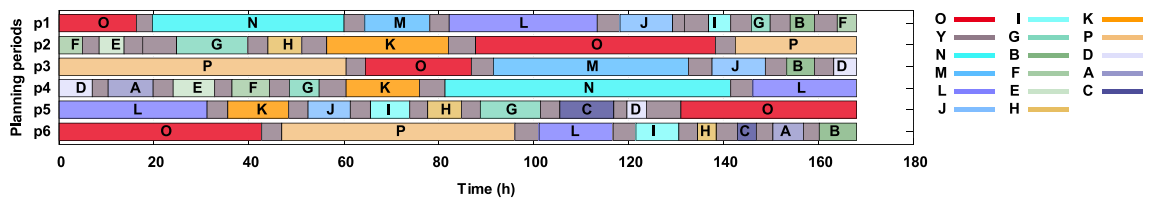
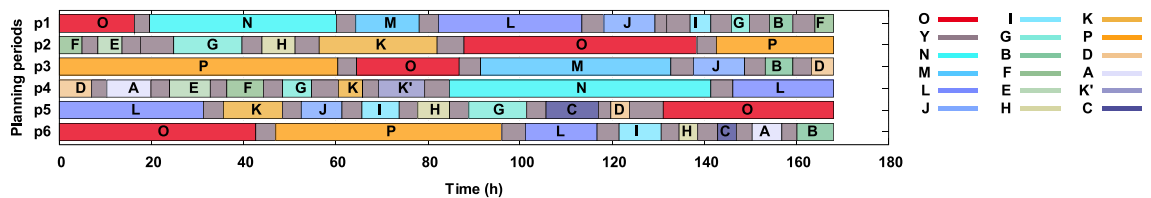
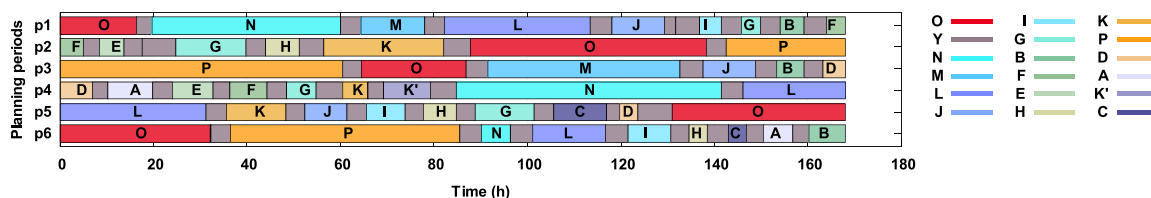
(b) Gantt chart after the 1st reschedule during p₁. A disturbance during the production of product O resulted in interruption while product G was selected to be additionally produced instead towards the end of p₁.(c) Gantt chart after the 2nd reschedule during p₁. Disturbance during the changeover from J to I which resulted in reduction of production time for G during p₁.(d) Gantt chart after the reschedule during p₂. Disturbance during the changeover from E to G which resulted in reduction of production time for P during p₂.(e) Gantt chart after the reschedule during p₄. Disturbance during the production of K which resulted in restoration of its production (K') from E to G while a reduction of production time for N during p₄ was rescheduled.(f) Gantt chart after the reschedule during p₆. A disturbance during the production of product O resulted in interruption while product N was selected to be additionally produced instead during p₆.**Fig. 14.** Gantt charts of all the reschedules for the MMA case study where the gray blocks represent changeover periods.

Table 12
Overview of the impact of the rescheduling mechanism on different factors of the iPSC.

Rescheduling Run	Profit (rmu)	Transition cost (rmu)	Backlog cost (rmu)	$\frac{\sum_i T_{ip}}{\sum_{i,j} T_{ip}^{trans} + T_{ip}^{trans}}$
Nominal	$1.876 \cdot 10^6$	714.63	553,094.75	4.86
1st Reschedule P ₁	$1.851 \cdot 10^6$	691.83	562,172.88	4.62
2nd Reschedule P ₁	$1.811 \cdot 10^6$	714.63	599,572.34	4.51
Reschedule P ₂	$1.767 \cdot 10^6$	753.58	639,595.41	4.32
P ₃	$1.767 \cdot 10^6$	753.58	639,595.41	4.32
Reschedule P ₄	$1.748 \cdot 10^6$	775.78	651,992.14	4.23
P ₅	$1.748 \cdot 10^6$	775.78	651,992.14	4.23
Reschedule P ₆	$1.703 \cdot 10^6$	850.34	692,827.74	4.11

of dynamic disruption during the production of O was detected. The graphs of the system's dynamics are given in Fig. 15 while the closed loop results from a scheduling perspective are given in Fig. 14.

Next based on the solution of the MMA case study, a discussion on the impact that the rescheduling mechanism has on the overall iPSC is conducted and some key correlations are underlined.

4.2.2. Impact of the rescheduling mechanism on the overall iPSC

The integrated nature of the iPSC gives rise to several synergies as well as hidden interdependences that can prove to be rather crucial for the optimal operations. In Table 12, a number of indicating factors from the iPSC problem are given against the different rescheduling runs that took place in the MMA case study. The general trends, as expected, is that the profit and overall production time to overall transition time ratio are decreasing functions of the rescheduling runs while on the other hand opposite behavior is exhibited by the transition cost and backlog cost of the overall plan. If we have a closer look at the results presented in Table 12 we notice that different rescheduling runs have different impact on the factors under examination.

Firstly, the reduction in profit during the second rescheduling run of the period P₁ is significantly greater than the decrease from the first rescheduling. The key difference between the two reschedules is the timing that the disturbance was detected. In the first instance, the disturbance was detected early during the first planning period but during a production time while the second reschedule took place during the transition from product J to product I. It seems that disturbances during transition periods have greater impact on the profit due to increased production of off-spec material. The timing of the disturbance seems to play, in this case, a secondary role as the first planning period had since the nominal plan an end time of 168 h which is the upper bound on time per planning period. The next interesting observation is that the transition cost during the first reschedule decreases slightly when compared to the nominal case. One would expect the transition cost to be monotonically increasing function of the rescheduling runs but from studying the economic data of the case study the following can explain the unexpected drop in transition cost. From the occurrence of the disruption in the dynamics, the transition from product O to N is benefited and thus happens in less time and thus cost. Moreover, product G that is inserted for production is the one that has the fastest transition time/less transition cost from product I and also its price is similar to the price of product O whose production time was reduced.

Finally, the greatest change in the indicating factors is observed in the last rescheduling during the sixth period and there are a

number of reasons that result in this. Demand that is backlogged during the end of the planning horizon does not allow for recovery of the lost sales since the model cannot allocate the backlogged demand in subsequent periods as in the reschedulings that happened in the earlier periods. Overall the following can be characterised as important factors that affect the optimal operation of the production facility within the iPSC scope: (i) the timing of the disturbance, i.e. if it happens at the beginning/middle/end of the planning period, (ii) the occurrence of the disturbance, i.e. if the disruption happens during the production or the transition time can result in loss of production but also in extended waste due to off-spec materials, (iii) the existence of idle time during each planning period and (iv) the complex trade-offs that stem from the economics of the iPSC, e.g. it might be more preferable to lose production of a product that has lower selling price when compared to a more premium grade. Finally, a graphical illustration of the economic indicators is given in Fig. 16.

5. Concluding remarks and future research directions

A novel framework for the closed-loop implementation of iPSC with rescheduling considerations has been presented. The key features of this framework involve: (i) a TSP model for the decisions at the planning and scheduling level where immediate sequence was used instead of the time-slots that have been proposed in the literature, (ii) a meta-modelling approach that allows the decision maker to take into consideration the transition cost for each transition and (iii) a novel model based controller for the closed-loop implementation of the control strategy. For the control level, we present the novel concept of multi-setpoint explicit controllers for a special class of nonlinear dynamic systems. The case studies illustrate that the use of multi-setpoint explicit controllers within the closed-loop framework enable real-time implementation of the decisions made by the integrated problem in the face of dynamic disruptions, something that had not appeared in the literature so far.

The case studies examined indicate that under the consideration of uncertain process dynamics, the inherent interdependence of the integrated problem manifests itself, since a disruption in the process dynamics can result in major changes on the decisions from the levels of scheduling and planning in order to guarantee optimal operations. To this end, currently in our group we examine the development of a hybrid methodology that accounts for multi-scale uncertainties throughout the three levels of decision making and investigate potential trade-offs and synergies that the integrated problem exhibits.

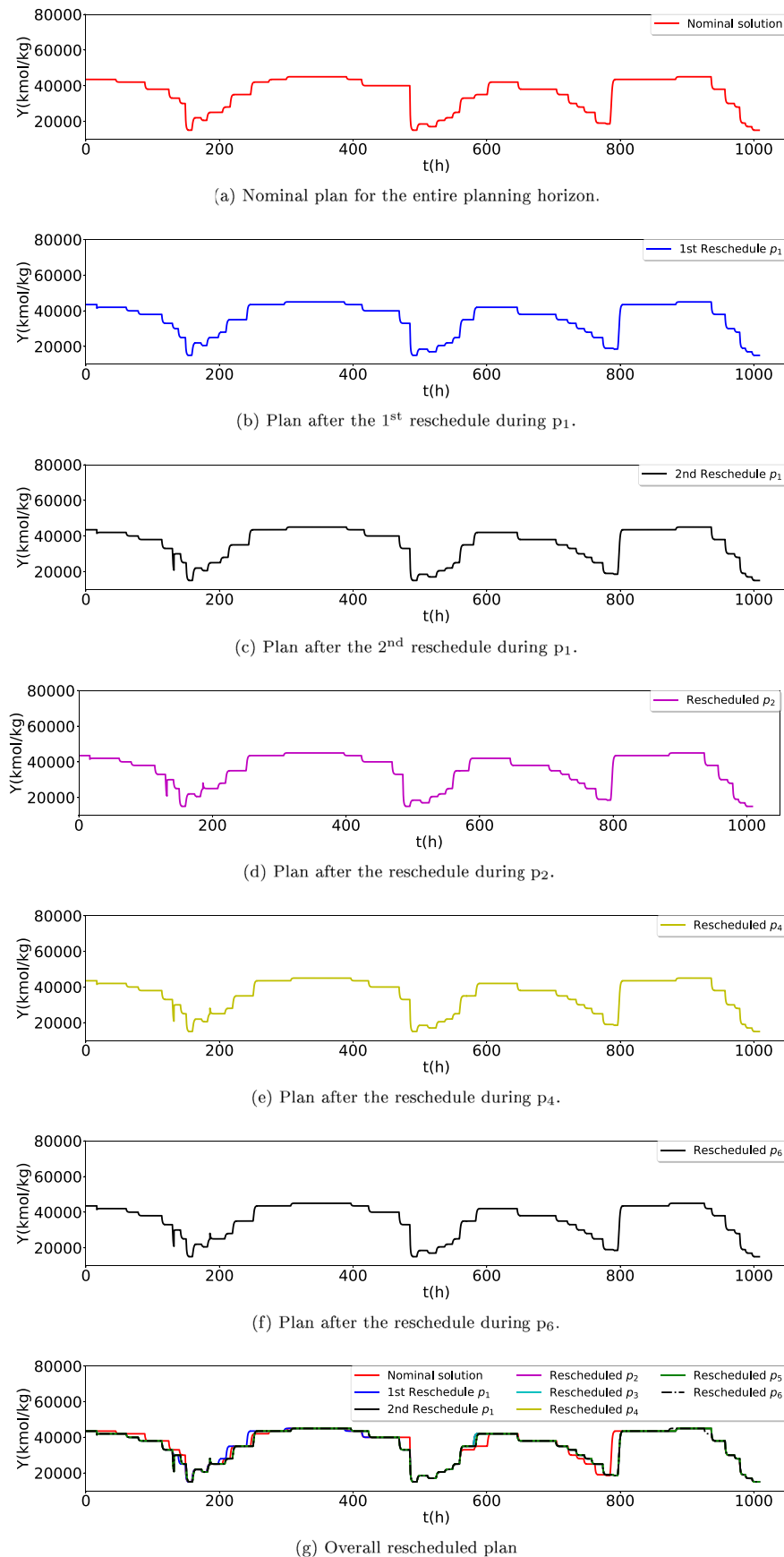


Fig. 15. $Y = f(t)$ plots indicative of the system's dynamics over the entire planning horizon.

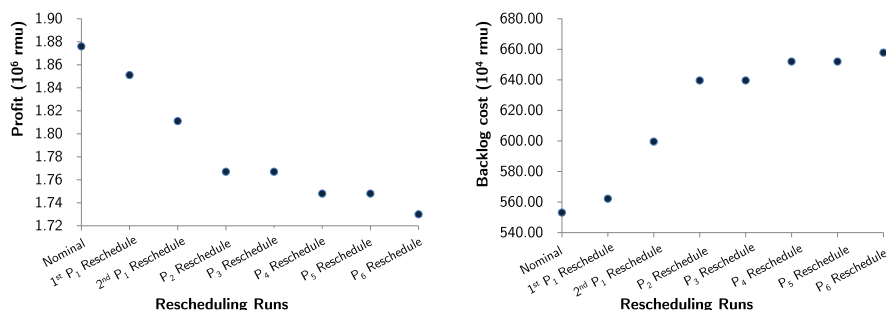


Fig. 16. Visual representation of the effect of the rescheduling mechanism on the economic performance of the iPSC.

Acknowledgments

The authors gratefully acknowledge financial support from EPSRC grants EP/M027856/1 and EP/M028240/1.

References

- Aguirre, A.M., Liu, S., Papageorgiou, L.G., 2017. Mixed integer linear programming based approaches for medium-term planning and scheduling in multiproduct multistage continuous plants. *Ind. Eng. Chem. Res.* 56 (19), 5636–5651.
- Aytug, H., Lawley, M.A., McKay, K., Mohan, S., Uzsoy, R., 2005. Executing production schedules in the face of uncertainties: a review and some future directions. *Eur. J. Oper. Res.* 161 (1), 86–110.
- Baldea, M., Du, J., Park, J., Harjunkski, I., 2015. Integrated production scheduling and model predictive control of continuous processes. *AIChE J.* 61 (12), 4179–4190.
- Bemporad, A., Morari, M., Dua, V., Pistikopoulos, E.N., 2002. The explicit linear quadratic regulator for constrained systems. *Automatica* 38 (1), 3–20.
- Bochnak, J., Coste, M., Roy, M.-F., 2013. *Real Algebraic Geometry*, Vol. 36. Springer Science & Business Media.
- Buchberger, B., 2006. Bruno buchberger's Ph.D. thesis 1965: an algorithm for finding the basis elements of the residue class ring of a zero dimensional polynomial ideal. *J. Symb. Comput.* 41 (3–4), 475–511.
- Castro, P.M., Barbosa-Póvoa, A.P., Matos, H.A., Novais, A.Q., 2004. Simple continuous-time formulation for short-term scheduling of batch and continuous processes. *Ind. Eng. Chem. Res.* 43 (1), 105–118.
- Charitopoulos, V.M., Dua, V., 2016. Explicit model predictive control of hybrid systems and multiparametric mixed integer polynomial programming. *AIChE J.* 62 (9), 3441–3460.
- Charitopoulos, V.M., Dua, V., Papageorgiou, L.G., 2017a. Closed loop integration of planning, scheduling and control via exact multi-parametric nonlinear programming. In: *Computer Aided Chemical Engineering*, Vol. 40. Elsevier, pp. 1273–1278.
- Charitopoulos, V.M., Dua, V., Papageorgiou, L.G., 2017b. Traveling salesman problem-based integration of planning, scheduling, and optimal control for continuous processes. *Ind. Eng. Chem. Res.* 56 (39), 11186–11205.
- Charitopoulos, V.M., Papageorgiou, L.G., Dua, V., 2017c. Multi-parametric linear programming under global uncertainty. *AIChE J.* 63 (9), 3871–3895.
- Charitopoulos, V.M., Papageorgiou, L.G., Dua, V., 2017d. Nonlinear model-based process operation under uncertainty using exact parametric programming. *Engineering* 3 (2), 202–213.
- Charitopoulos, V.M., Papageorgiou, L.G., Dua, V., 2018. Multi-parametric mixed integer linear programming under global uncertainty. *Comput. Chem. Eng.* <https://doi.org/10.1016/j.compchemeng.2018.04.015>.
- Charitopoulos, V.M., Papageorgiou, L.G., Dua, V., 2018. A methodology for the design of multi-setpoint explicit controllers. In preparation.
- Chu, Y., You, F., 2012. Integration of scheduling and control with online closed-loop implementation: fast computational strategy and large-scale global optimization algorithm. *Comput. Chem. Eng.* 47, 248–268.
- Chu, Y., You, F., 2015. Model-based integration of control and operations: overview, challenges, advances, and opportunities. *Comput. Chem. Eng.* 83, 2–20.
- Dias, L.S., Ierapetritou, M.G., 2016. Integration of scheduling and control under uncertainties: review and challenges. *Chem. Eng. Res. Des.* 116, 98–113.
- Dogan, M.E., Grossmann, I.E., 2006. A decomposition method for the simultaneous planning and scheduling of single-stage continuous multiproduct plants. *Ind. Eng. Chem. Res.* 45 (1), 299–315.
- Dominguez, L.F., Narciso, D.A., Pistikopoulos, E.N., 2010. Recent advances in multi-parametric nonlinear programming. *Comput. Chem. Eng.* 34 (5), 707–716.
- Dua, V., Kouramas, K., Dua, V., Pistikopoulos, E.N., 2008. MPC On a chip - recent advances on the application of multi-parametric model-based control. *Comput. Chem. Eng.* 32 (4), 754–765.
- Dua, V., 2015. Mixed integer polynomial programming. *Comput. Chem. Eng.* 72, 387–394.
- Flores-Tlacuahuac, A., Grossmann, I.E., 2006. Simultaneous cyclic scheduling and control of a multiproduct CSTR. *Ind. Eng. Chem. Res.* 45 (20), 6698–6712.
- Fotiu, I.A., Parrilo, P.A., Morari, M., 2005. Nonlinear parametric optimization using cylindrical algebraic decomposition. In: *Proceedings of the Forty Fourth IEEE Conference on Decision and Control and European Control Conference*. IEEE, pp. 3735–3740.
- Grossmann, I.E., 2012. Advances in mathematical programming models for enterprise-wide optimization. *Comput. Chem. Eng.* 47, 2–18.
- Gupta, D., Maravelias, C.T., Wassick, J.M., 2016. From rescheduling to online scheduling. *Chem. Eng. Res. Des.* 116, 83–97.
- Gutierrez-Limon, M.A., Flores-Tlacuahuac, A., Grossmann, I.E., 2014. MINLP Formulation for simultaneous planning, scheduling, and control of short-period single-unit processing systems. *Ind. Eng. Chem. Res.* 53 (38), 14679–14694.
- Gutierrez-Limon, M.A., Flores-Tlacuahuac, A., Grossmann, I.E., 2016. A reactive optimization strategy for the simultaneous planning, scheduling and control of short-period continuous reactors. *Comput. Chem. Eng.* 84, 507–515.
- Harjunkski, I., Maravelias, C.T., Bongers, P., Castro, P.M., Engell, S., Grossmann, I.E., Hooker, J., Méndez, C., Sand, G., Wassick, J., 2014. Scope for industrial applications of production scheduling models and solution methods. *Comput. Chem. Eng.* 62, 161–193.
- Kanakamedala, K.B., Reklaitis, G.V., Venkatasubramanian, V., 1994. Reactive schedule modification in multipurpose batch chemical plants. *Ind. Eng. Chem. Res.* 33 (1), 77–90.
- Kopanos, G.M., Capon-Garcia, E., Espuna, A., Puigjaner, L., 2008. Costs for rescheduling actions: a critical issue for reducing the gap between scheduling theory and practice. *Ind. Eng. Chem. Res.* 47 (22), 8785–8795.
- Kopanos, G.M., Pistikopoulos, E.N., 2014. Reactive scheduling by a multiparametric programming rolling horizon framework: a case of a network of combined heat and power units. *Ind. Eng. Chem. Res.* 53 (11), 4366–4386.
- Kopanos, G.M., Puigjaner, L., Maravelias, C.T., 2010. Production planning and scheduling of parallel continuous processes with product families. *Ind. Eng. Chem. Res.* 50 (3), 1369–1378.
- Li, Z., Ierapetritou, M.G., 2008. Reactive scheduling using parametric programming. *AIChE J.* 54 (10), 2610–2623.
- Liu, S., Pinto, J.M., Papageorgiou, L.G., 2008. A TSP-based MILP model for medium-term planning of single-stage continuous multiproduct plants. *Ind. Eng. Chem. Res.* 47 (20), 7733–7743.
- Maravelias, C.T., Sung, C., 2009. Integration of production planning and scheduling: overview, challenges and opportunities. *Comput. Chem. Eng.* 33 (12), 1919–1930.
- Mendez, C.A., Jaime, C., 2004. An MILP framework for batch reactive scheduling with limited discrete resources. *Comput. Chem. Eng.* 28 (6), 1059–1068.
- Novas, J.M., Henning, G.P., 2010. Reactive scheduling framework based on domain knowledge and constraint programming. *Comput. Chem. Eng.* 34 (12), 2129–2148.
- Oberdieck, R., Diangelakis, N.A., Nascu, I., Papathanasiou, M.M., Sun, M., Avraamidou, S., Pistikopoulos, E.N., 2016. On multi-parametric programming and its applications in process systems engineering. *Chem. Eng. Res. Des.* 116, 61–82.
- Pistikopoulos, E.N., Diangelakis, N.A., 2016. Towards the integration of process design, control and scheduling: are we getting closer? *Comput. Chem. Eng.* 91, 85–92.
- Pistikopoulos, E.N., Diangelakis, N.A., Oberdieck, R., Papathanasiou, M.M., Nascu, I., Sun, M., 2015. PAROC-An integrated framework and software platform for the optimisation and advanced model-based control of process systems. *Chem. Eng. Sci.* 136, 115–138.
- Sahinidis, N.V., 1996. BARON: A general purpose global optimization software package. *J. Glob. Optim.* 8 (2), 201–205.
- Shi, H., Chu, Y., You, F., 2015. Novel optimization model and efficient solution method for integrating dynamic optimization with process operations of continuous manufacturing processes. *Ind. Eng. Chem. Res.* 54 (7), 2167–2187.
- Subramanian, K., Maravelias, C.T., Rawlings, J.B., 2012. A state-space model for chemical production scheduling. *Comput. Chem. Eng.* 47, 97–110.
- Vin, J.P., Ierapetritou, M.G., 2000. A new approach for efficient rescheduling of multiproduct batch plants. *Ind. Eng. Chem. Res.* 39 (11), 4228–4238.
- Zhuge, J., Ierapetritou, M.G., 2012. Integration of scheduling and control with closed loop implementation. *Ind. Eng. Chem. Res.* 51 (25), 8550–8565.

Zhuge, J., Ierapetritou, M.G., 2014. Integration of scheduling and control for batch processes using multi-parametric model predictive control. *AIChE J.* 60 (9), 3169–3183.

Zhuge, J., Ierapetritou, M.G., 2015. An integrated framework for scheduling and control using fast model predictive control. *AIChE J.* 61 (10), 3304–3319.

Zhuge, J., Ierapetritou, M.G., 2016. A decomposition approach for the solution of scheduling including process dynamics of continuous processes. *Ind. Eng. Chem. Res.* 55 (5), 1266–1280.