

Article

Multi Set-Point Explicit Model Predictive Control for Nonlinear Process Systems

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

Centre for Process Systems Engineering, Department of Chemical Engineering, University College London, Torrington Place, London WC1E 7JE, UK; v.charitopoulos@ucl.ac.uk (V.M.C.); l.papageorgiou@ucl.ac.uk (L.G.P.)

* Correspondence: v.dua@ucl.ac.uk

Abstract: In this article, we introduce a novel framework for the design of multi set-point nonlinear explicit controllers for process systems engineering problems where the set-points are treated as uncertain parameters simultaneously with the initial state of the dynamical system at each sampling instance. To this end, an algorithm for a special class of multi-parametric nonlinear programming problems with uncertain parameters on the right-hand side of the constraints and the cost coefficients of the objective function is presented. The algorithm is based on computed algebra methods for symbolic manipulation that enable an analytical solution of the optimality conditions of the underlying multi-parametric nonlinear program. A notable property of the presented algorithm is the computation of exact, in general nonconvex, critical regions that results in potentially great computational savings through a reduction in the number of convex approximate critical regions.

Keywords: multi-parametric programming; explicit MPC; enterprise-wide optimisation; set-point tracking; algebraic geometry



Citation: Charitopoulos, V.M.; Papageorgiou, L.G.; Dua, V. Multi Set-Point Explicit Model Predictive Control for Nonlinear Process Systems. *Processes* **2021**, *9*, 1156. <https://doi.org/10.3390/pr9071156>

Academic Editor: Luis Puigjaner

Received: 9 March 2021

Accepted: 13 May 2021

Published: 2 July 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

High-fidelity and computationally efficient optimisation models are key for profitable decision making in process industries and have been the focus of extensive research over the years [1]. In recent years, the need for exploiting and explicitly considering interdependencies throughout the different layers of decision making has been underpinned by the enterprise-wide optimisation (EWO) concept [2]. Stemming from the progressively volatile and competitive market conditions, it is imperative for process industries to operate with agility in order to maximise their profitability [3]. EWO is aiming at increased profitability and resilience in process operations through the integration and simultaneous optimisation of existing information streams. Nonetheless, it comes at a considerable cost. Because of the multiple scales considered, EWO leads to computational challenges, thus preventing practitioners from harnessing the potential benefits such wide integration has to offer. Particularly, incorporating control considerations in an EWO fashion results in (mixed integer) nonconvex problems which are hard to solve.

By the same token, control considerations are ubiquitous in EWO problems. Figure 1 showcases how real-time optimisation and production scheduling exchange information with the layer of APC because of their interdependent decisions.

Real-time optimisation is concerned with the manipulation of systems' dynamics in order to achieve optimised profitability and operations. On the other hand, production scheduling determines the optimal allocation of resources for the completion of competing tasks. As indicated by Figure 1, both RTO and process scheduling exchange information with the layer of APC so as to achieve optimal dynamic operations. To this end, the research community has proposed different methods for their integration.

A common shortfall when focusing on integrating RTO and APC is that two different models are employed for the optimisation of the same system. Typically, a locally linear model of the initial nonlinear dynamics is used at the APC because of the need for fast

solution rates while RTO considers the original nonlinear model. This leads subsequently to issues related to suboptimal trajectories and non-reachable states [4].

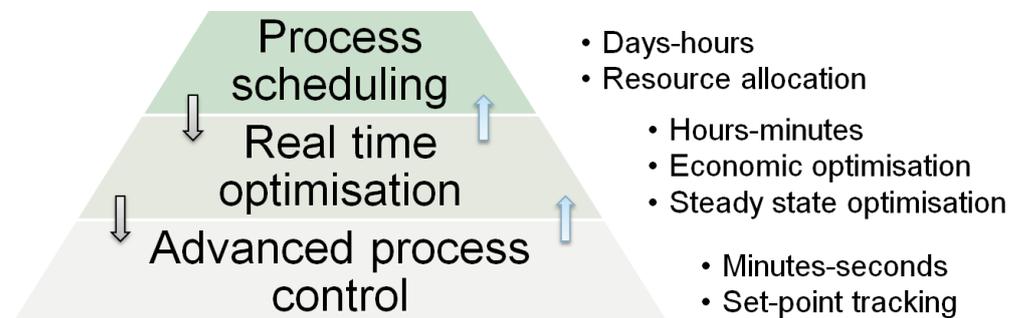


Figure 1. Interaction of APC with different layers of decision making in process industries.

Darby et al. [5], through their literature review regarding the integration of RTO and MPC, suggested that for a successful integration, common issues such as model mismatch among the layers of RTO and APC should be eliminated. Nonetheless, in real industrial processes, model degradation and other factors can result in model mismatch, so the consideration of parameter estimation and data reconciliation functionalities is needed to integrate RTO and MPC, as indicated by Figure 2.

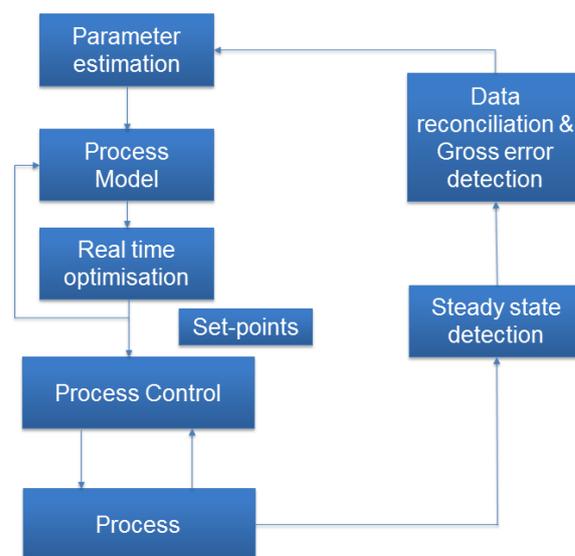


Figure 2. Interaction between advanced process control and real-time optimisation.

The interaction between real-time optimisation and model predictive control can be categorised broadly into three classes: (i) dynamic RTO (d-RTO), (ii) static RTO (s-RTO) and (iii) economic model predictive control (e-MPC). Both s-RTO and d-RTO are two-layer schemes where reference trajectories are passed to the layer of APC in the form of set-points [6]. While under the static real-time optimisation paradigm, the optimisation problem is solved at specific instances whenever new data become available or when steady state is achieved, in the d-RTO paradigm, the system's transient behaviour is explicitly considered, thus resulting in dynamic optimisation problems. e-MPC [7] refers to single-layer strategies which are incorporated into the control structure economic considerations. In that spirit, De Souza et al. [8] proposed the inclusion of the gradient of the economic objective function into the MPC objective as a single-layer strategy. Considering uncertain systems, Chachuat et al. [9] examined alternative model adaptation strategies.

This article is motivated by the abovementioned issues and aims at introducing a method for designing multi set-point explicit controllers for nonlinear systems through recent advances in multi-parametric programming. Multi-parametric programming (mp-P) has received considerable attention from the process systems engineering community because of its unique ability to aid in the design of explicit model predictive controllers and thus shift the computational burden associated with offline control [10]. We examine a case of multi-parametric nonlinear programs (mp-NLPs) that involve both endogenous uncertainty, in the form of left-hand side parameters (LHS), as well as exogenous uncertainty in the cost coefficient of the objective function (OFC), and, on the right-hand side of the constraints (RHS), uncertain parameters on the right-hand side (RHS). In engineering problems, LHS uncertainty arises from variations in model coefficients, due to parameter estimation errors or model mismatch; OFC uncertainty arises due to fluctuation in market prices or control penalties while RHS uncertainty can be due to varying system exogenous factors. The contribution of the present work is a novel framework for the design of multi set-point explicit controllers for nonlinear process systems.

The remainder of the article is organised as follows: in Section 2, an overview of the field of multi-parametric programming and explicit MPC is given, and then, in Section 3, the proposed algorithm is detailed and a framework for multi set-point explicit controllers is introduced. In Section 4, two case studies are examined so as to illustrate the main computational steps of the proposed methodology and, lastly, in Section 5, concluding remarks are made.

2. Background

2.1. Multi-Parametric Programming

Overall, multi-parametric programming problems are concerned with the effect of parametric perturbations on the optimal solution of an optimisation problem. Consider the following optimisation problem:

$$z(\theta) = \begin{array}{ll} \min_{\mathbf{x} \in \mathbb{R}^{n_x}} & f(\mathbf{x}, \theta) \\ \text{subject to :} & \mathbf{g}(\mathbf{x}, \theta) \leq 0 \\ & \theta \in \mathbb{R}^{n_\theta} \end{array} \quad (1)$$

where θ stands for the vector of uncertain parameters, which is n_θ -dimensional, \mathbf{x} is the n_x -dimensional vector of continuous decision variables, $\mathbf{g}(\mathbf{x}, \theta)$ is the vector of inequality constraints and f is the objective function as a mapping $\mathbb{R}^{n_x \times n_\theta} \rightarrow \mathbb{R}$, both of which are assumed to be \mathcal{C}^2 (twice continuously differentiable). Problem (1) is a multi-parametric program and its solution results in the partition of the parametric \mathbb{R}^{n_θ} -space into a number of regions, also known as critical regions (CRs). Within each CR, the optimal solution and the objective value are given as functions of the uncertain parameters, i.e., $\mathbf{x}(\theta)$ and $\mathbf{z}(\theta)$, respectively. Even though mp-P has been studied quite actively, the class of multi-parametric nonlinear programming problems remains one of the most challenging ones [11,12]. Depending on the convexity of the nonlinear functions that form Problem (1), different solution techniques have been proposed in the literature to date.

Advances in the algorithms and theory of parametric nonlinear programs (p-NLPs) date back to the early works of Fiacco [13] and Bank et al. [14]. More specifically, in the books of Bank et al. [14] and Fiacco [13], a collection of the early research works for parametric NLPs can be found and invaluable theoretical foundations for some classes of convex p-NLPs with perturbations in the OFC and the right-hand side of the constraints are provided. Even though the term “parametric nonlinear optimisation/programming” was widely established from the aforementioned works, the early works on numerical stability analysis of NLPs by [15,16] and the work of Robinson [17,18] on generalised equations provided a significant way of studying the effect of parametric variations on the optimal solution of p-NLPs. Kypraris [19] studied the uniqueness and differentiability of solutions of parametric nonlinear complementarity problems while in Ralph and Dempe [20], the

directional derivatives of parametric nonlinear programs were used to characterise their explicit solution. However, the first algorithm for the multi-parametric case of convex NLPs was due to Dua and Pistikopoulos [21]. The authors, based on the findings about the convexity properties of the parametric value function ($z(\theta)$), devised an iterative procedure in which the integer variables were fixed by the solution of a primal mixed integer nonlinear program (MINLP) and the resulting mp-NLP was then transformed into an mp-LP following the outer approximation idea. Because of the value function's convexity property, the maximum error of the approximation occurs at the vertices of the CRs and if the error is greater than the prespecified tolerance, the CR is partitioned again; otherwise, integer and parametric cuts are implemented and then the algorithm iterates until the primal MINLP is infeasible. The same algorithm was revisited by Acevedo and Salgueiro [22], where the authors proposed heuristics to improve its computational efficiency while quadratic approximations were studied by Johansen [23] and Domínguez and Pistikopoulos [24]. An approximate algorithm for the solution of convex mp-NLPs was proposed by Johansen [25], who proposed the consecutive subdivision of the parametric space in hyper-rectangles and the interpolation of the parametric solution through the solution of $2^{n_{\theta}}$ NLPs at each step. Further approaches involve the geometric vertex search by Narciso [26] and sub-gradient methods by Leverenz et al. [27]. For the nonconvex cases, Dua et al. [28] developed suitable parametric under/overestimators which were then incorporated into a spatial branch and bound routine for the global optimisation of the nonconvex problem within ϵ -tolerance. For a more thorough discussion on the algorithms that have been proposed for the solution of mp-NLPs, the interested reader is directed to the review of Domínguez et al. [29] while Hale [30], in her doctoral thesis, also offers a thorough discussion on several classes of parametric optimisation. Fotiou et al. [31,32] initially studied the polynomial multi-parametric programming problem with application to control, however, their approach did not include the definition of final nonconvex CRs, while the mixed integer polynomial case was studied by Charitopoulos and Dua [33] and a procedure for the computation of exact nonconvex CRs was presented. Despite the aforementioned research effort, mp-NLP problems remain one of the most difficult to tackle and, as illustrated in Table 1, all the aforementioned algorithms can handle uncertain parameters only on the right-hand side (RHS) of the constraints. Recently, Pappas et al. [34], by generalising the basic sensitivity theorem of Fiacco [13], devised an algorithm for the exact solution of multi-parametric quadratically constrained quadratic programs.

Table 1. Summary of multi-parametric nonlinear programming algorithms.

mp-NLP Solution Techniques	RHS	LHS	OFC	Comments
Dua and Pistikopoulos [21]	✓	-	-	Convex
Johansen [23]	✓	-	-	Convex
Acevedo and Salgueiro [22]	✓	-	-	Convex
Johansen [25]	✓	-	-	Convex
Dua et al. [28]	✓	-	-	Nonconvex
Fotiou et al. [31]	✓	-	-	Polynomial
Narciso [26]	✓	-	-	Convex
Domínguez and Pistikopoulos [24]	✓	-	-	Convex
Charitopoulos and Dua [33]	✓	-	-	Polynomial
Pappas et al. [34]	✓	-	-	QCQPs

Among the wide range of applications that multi-parametric programming has been applied to, the invention of explicit model predictive control (mp-MPC) is undoubtedly the most dominant area where mp-P has had the biggest impact [10,12,35]. The main concept of mp-MPC is that instead of solving the optimisation problems related to standard MPC at each sampling instance, the state of the system is treated as an uncertain parameter and an mp-P can be solved offline to derive the explicit control solution once and for all [10,36].

The general formulation of mp-MPC for discrete time systems is shown by (2):

$$\left. \begin{array}{l} \\ \\ \\ \\ \\ \\ \end{array} \right\} \begin{array}{l} \Phi(x(t_k)) = \min_{\mathbf{u}} \sum_{i=0}^{P_H-1} \mathcal{L}(x_t, u_t) + E(x_N) \\ \text{subject to: } x_{t|t=0} = x(t_k) \\ x_{t+1} = f(x_t, u_t) \quad t = 0, 1, \dots, P_H - 1 \\ y_{t+1} = h(x_t, u_t) \quad t = 0, 1, \dots, P_H - 1 \\ \mathbf{A}x_t \leq \boldsymbol{\alpha} \quad t = 0, 1, \dots, P_H \\ \mathbf{B}y_t \leq \boldsymbol{\beta} \quad t = 0, 1, \dots, P_H \\ \mathbf{C}u_t \leq \boldsymbol{\gamma} \quad t = 0, 1, \dots, P_H \end{array} \quad (2)$$

where \mathbf{x}_t , \mathbf{u}_t , \mathbf{z}_t are the state, control input and system output vectors, respectively, at every sampling point, t , and are n_x , n_u , n_y -dimensional. \mathbf{A} , \mathbf{B} , \mathbf{C} are matrices of appropriate dimensions and $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$, $\boldsymbol{\gamma}$ vectors of pertinent dimensions which represent inequality constraints for the state, output and control inputs while $\mathcal{L}: \mathbb{R}^{n_x+n_u} \rightarrow \mathbb{R}$ is a stage cost and $E: \mathbb{R}^{n_x} \rightarrow \mathbb{R}$ is a terminal cost function. The repetitive solution of Problem (2) provides the optimal cost $\Phi(\mathbf{x}(t_k))$ and the optimisation vector, i.e., the sequence of optimal control inputs $\mathbf{u}^* = [\mathbf{u}_1^*, \mathbf{u}_2^*, \dots, \mathbf{u}_{P_H-1}^*]$ over the finite prediction horizon P_H . Compared to the conventional model predictive control fashion, in which an optimisation problem is solved at each sampling point, through the mp-MPC notion, the explicit control law is calculated offline once and for all. The solution of the resulting mp-P problem results in the optimal control inputs as explicit functions of the (uncertain) parameters, i.e., the state of the system at each sampling instance, along with the corresponding CRs, as shown by Equation (3).

$$\mathbf{u}^* = \begin{cases} v_1(\mathbf{x}(t_k)) & \text{if } \mathbf{x}(t_k) \in CR_1 \\ v_2(\mathbf{x}(t_k)) & \text{if } \mathbf{x}(t_k) \in CR_2 \\ \vdots & \vdots \\ v_\omega(\mathbf{x}(t_k)) & \text{if } \mathbf{x}(t_k) \in CR_\omega \end{cases} \quad (3)$$

For the case of MPC for linear systems, instead of solving a quadratic program at each sampling instance, the explicit MPC requires the offline solution of an mp-QP while online, so only simple function evaluations are required [10,37,38]. This concept is also known as online via offline optimisation and it is shown in Figure 3.

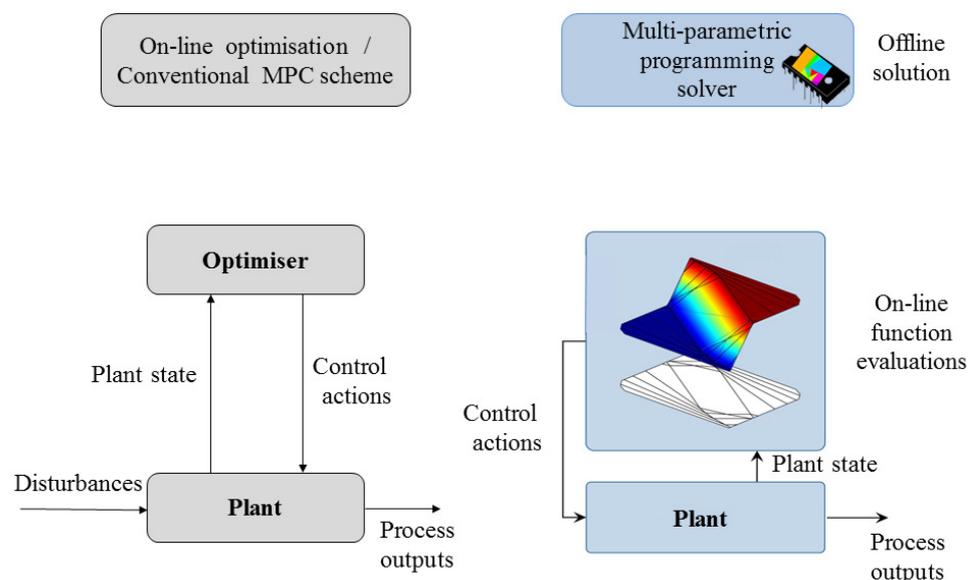


Figure 3. Online via offline optimisation framework [10,39].

Even though mp-MPC is the niche area of mp-P, designing mp-MPCs of nonlinear systems for set-point tracking is still a computationally strenuous task as one has to design an mp-MPC for each set-point based on the algorithms that exist in the literature to date [40]. Next, we review two mathematical techniques that will enable the development of novel multi set-point explicit controllers though the algorithm we propose in the present work.

2.2. Computer Algebra

2.2.1. Gröbner Bases Theory

The idea of the present work is to devise an algorithm for the solution of mp-NLPs from an analytical and not numerical perspective, and the reason is two-fold. Firstly, for the case that we are interested in, i.e., mp-NLPs with combined uncertainty on the RHS and OFC, no theoretical foundations exist for the computation of the explicit solution like the basic sensitivity theorem of Fiacco [13] which serves as the basis of the numerical mp-P approaches. Secondly, because of the nonconvex nature of the parametric problem, the numerical solution would require global optimisation techniques similar to the ones presented in Dua et al. [28] and would lead to an explosion in the number of convex approximate CRs.

Gröbner bases theory was introduced by the doctoral research of Bruno Buchberger [41] as a method of analytically solving systems of multivariate polynomial equations. In brief, Gröbner bases and the Buchberger algorithm can be considered as the polynomial counterpart of Gaussian elimination for the case of nonlinear systems. Before formally defining what a Gröbner basis is, it would be useful to define some preliminary concepts.

Definition 1. Power products

Let \mathbb{R} be any field and let $\mathbb{R}[x_1, \dots, x_n]$ be the ring of polynomials in n -indeterminates. Any polynomial can be described as a sum of terms of the form: $\alpha x_1^{\beta_1} \dots x_n^{\beta_n}$ with $\alpha \in \mathbb{R}$ and $\beta_i \in \mathbb{N}$, $i = 1, \dots, n$ and the term $x_1^{\beta_1} \dots x_n^{\beta_n}$ is called a power product.

Definition 2. Term order

A term order is defined with regard to a set of power products (T_n) and imposes a total order $<$ on the set in compliance with the conditions below:

1. $1 < x^\beta$ for all $x^\beta \in T_n$
2. If $x^\alpha < x^\beta \rightarrow x^\alpha x^\gamma < x^\beta x^\gamma$, for all $x^\gamma \in T_n$

A number of alternative power product orderings exist but the most commonly employed is the the lexicographic one due to its computational efficiency [42]. Lastly, the notion of *ideals* is crucial within the Gröbner bases theory.

Definition 3. Ideals

Let \mathbb{R} be a field and $\mathbb{R}[x_1, x_2, \dots, x_n]$ be a ring over the field of n -variate polynomials. Let a finite subset of the field, $G = \{g_1, g_2, \dots, g_t\}$, then an ideal I can be generated by G as follows:

$$I = \left\{ \sum_{i=1}^n u_i g_i \mid u_i \in \mathbb{R}[x_1, x_2, \dots, x_n], g_i \in \mathbb{R}, \forall i \right\}$$

For problems that accept analytical solutions, their existence is guaranteed by the Hilbert basis theorem [43], which also guarantees that algorithms used to compute Gröbner bases can terminate in a finite number of steps.

Definition 4. Gröbner basis [41]

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 $g \in I$, such that $g \neq 0$, there exists $i \in \{1, \dots, t\}$ such that $lp(g_i)$ divides $lp(g)$, where $lp(\cdot)$ stands for the leading power product of a polynomial function.

For the calculation of Gröbner bases, apart from Buchberger's algorithm, Faugère has presented algorithms F4 [44] and F5 [45] as two variants of another algorithm. They exploit concepts from linear algebra and represent polynomials using matrix forms, thus enabling successive truncated Gröbner bases to be created. Lastly, software implementations of different algorithms that compute Gröbner bases computations can be found in freely available CAS such as Singular, SymPy and SageMath as well as commercial tools like Maple and Mathematica.

2.2.2. Cylindrical Algebraic Decomposition (CAD)

The notion of cylindrical algebraic decomposition was presented by Collins in 1975 [46] in an effort to solve the problem of quantifier elimination over real closed fields. In this article, as will be shown later on, CADs are used for computing nonconvex regions in the space of parameters. Thus, we provide the following definitions for ease of exposition in the algorithmic steps that we detail later on in the manuscript.

Definition 5. *Semi-Algebraic Sets* [43].

Let $\mathbb{R}[x_1, x_2, \dots, x_n]$ indicate the ring of polynomials in n -indeterminates with real coefficients. If, for example, a subset S of \mathbb{R}^n can be developed by a finite number of applications of the complementation, union and intersection operations, it is called semi-algebraic and can have the following form:

$$\{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{g}(\mathbf{x}) \leq 0\}, \text{ where } \mathbf{g} \in \mathbb{R}[X]$$

Definition 6. *Standard atomic formula*

A formula that includes a functional relation over a polynomial ring in either of the ways shown below is called a standard atomic formula:

$$g(\mathbf{x}) = 0, g(\mathbf{x}) \neq 0, g(\mathbf{x}) < 0, g(\mathbf{x}) > 0, g(\mathbf{x}) \leq 0, g(\mathbf{x}) \geq 0$$

Proposition 1 ([43]). *Semi-algebraic sets of \mathbb{R}^n can be written as a finite union of semi-algebraic sets of the form:*

$$\{\mathbf{x} \in \mathbb{R}^{n_x} \mid g_1(\mathbf{x}) = \dots = g_\omega(\mathbf{x}) = 0, g_{\omega+1}(\mathbf{x}) > 0, \dots, g_t(\mathbf{x}) > 0\}$$

where $g_1, \dots, g_\omega, g_{\omega+1}, \dots, g_t$ are in $g \in \mathbb{R}[X]$.

The proof of the proposition can be found in the book of Bochnak et al. [43].

Using the definitions and propositions given above, in summary, one can use CAD routines to compute the solution to polynomial inequalities. In the process of computations, one partitions the related space over finite cells and qualifies whether or not standard atomic formulas hold. A comprehensive exposition on the solution of polynomial inequalities using CAD is given at the book of Jirstrand [47].

3. Algorithms and MPC

3.1. Multi Set-Point Explicit Controller via Multi-Parametric Programming

In the context of multi-parametric model predictive control, the state of the system at each sampling point is treated as an uncertain parameter and as a result an mp-P with RHS uncertainty arises [10,37,38,40]. Its solution results in the explicit control law, i.e., the control decisions as explicit functions of the system's initial conditions at a sampling instance along with the related CRs.

In many applications, however, particularly those related to continuous manufacturing, there is a great need for fast calculations in order to communicate decisions between the different layers of decision making in an effective manner. For instance, set-point tracking goals for APC are, most of the time, passed down from either the functionality of process scheduling or RTO [39,48]. In these cases, it becomes obvious that explicit MPC

can provide a significant advantage in computational time by treating the set-points or estimated model inputs as uncertain parameters. One way to design such explicit controllers, assuming the existence of the n_u set-point, is to solve n_u mp-P problems and thus design n_u mp-MPCs. Another way, which has not been investigated in the literature, is to consider the set-points and/or model inputs as uncertain parameters and thus derive a multi set-point explicit MPC. Conceptually, by doing so, we would design a layered controller as given by Figure 4.

Conventionally, when mp-MPC is employed for set-point tracking of nonlinear systems, one would have to compute a different mp-MPC for each of those set-points as well as account for any time delay in the offline solution of the related mp-P should a new set-point arise. Stringent market regulations and an increasingly volatile market environment lead process industries to constantly optimise their operations and give rise to new set-points from a control perspective which in turn hinder the deployment of mp-MPC. As illustrated by Figure 4, in this article, by considering the set-points as uncertain parameters which lie within prespecified bounds, we overcome the abovementioned drawback of explicit MPC since, by solving one mp-NLP, we can design a “multi set-point” mp-MPC for nonlinear systems.

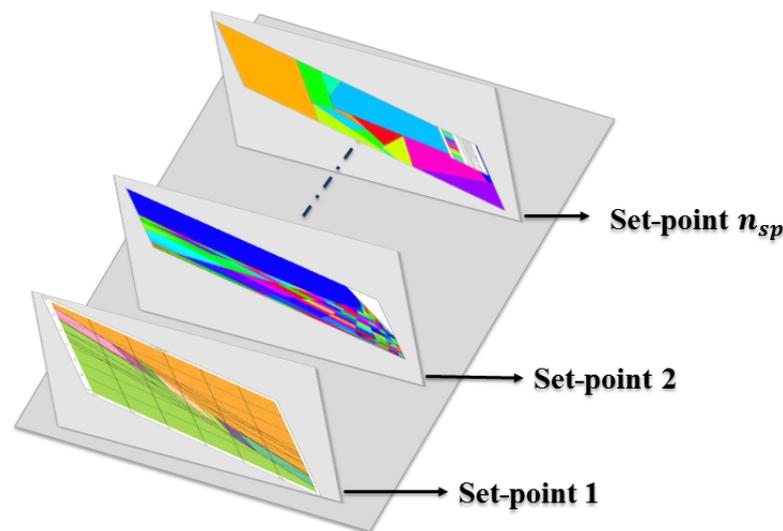


Figure 4. Concept of a multi set-point mp-MPC setting where instead of separate explicit controllers one designs a universal explicit MPC for all the set-points.

The design problem of a multi set-point mp-MPC can be formulated as in (4).

$$\left\{ \begin{array}{l} Y(x(t_k), x_{sp}) = \min_{\mathbf{u}} \sum_{t=0}^{P_H-1} \mathcal{L}(x_t, u_t, x_{sp}) + E(x_N, x_{sp}) \\ \text{subject to: } \quad x_{t|t=0} = x(t_k) \\ \quad \quad \quad x_{t+1} = f(x_t, u_t) \quad t = 0, 1, \dots, P_H - 1 \\ \quad \quad \quad y_{t+1} = h(x_t, u_t) \quad t = 0, 1, \dots, P_H - 1 \\ \quad \quad \quad \mathbf{A}x_t \leq \boldsymbol{\alpha} \quad t = 0, 1, \dots, P_H \\ \quad \quad \quad \mathbf{B}y_t \leq \boldsymbol{\beta} \quad t = 0, 1, \dots, P_H \\ \quad \quad \quad \mathbf{C}u_t \leq \boldsymbol{\gamma} \quad t = 0, 1, \dots, P_H \\ \quad \quad \quad x_{sp} \in \mathbb{R}^{n_{sp}} \end{array} \right. \quad (4)$$

The notation adopted is the same as in the previous section, aside from the following: we treat the set-points together with the initial state of the system as uncertain parameters, thus resulting in a problem with simultaneous variations on the RHS and OFC. As indicated by (4), we consider both the initial states ($x(t_k)$) and the various set-points (x_{sp}) as uncertain

parameters. Notice that within an EWO framework, the deployment of such universal controllers is of great importance since they allow for rapid communication between the layer of control with RTO and scheduling. For instance, when integrating scheduling with control, the changeover times as well as the production rates are immediate results of the dynamic decisions made through the control system. In the next section, an algorithm for the design of such “multi set-point” explicit MPC is presented.

3.2. Solution Algorithm for Analytical mp-NLPs with Global Uncertainty

Here, we present an algorithm that can solve multi-parametric nonlinear programs with non transcendental nonlinear terms, i.e., nonlinear terms that have closed-form solutions. The proposed method can be seen as a generalisation of the our previous work on the solution of multi-parametric mixed integer polynomial programs [33] as well as the algorithm of Fotiou et al. [32] for multi-parametric polynomial programs. However, both of these methods were employed only for instances that the uncertainty is present on the right-hand side of the constraints and the latter does not compute the critical regions associated with each explicit solution.

The main idea of the algorithm proposed herein can be explained as follows: given a multi-parametric nonlinear program with analytical terms, or terms that can be expressed in a nontranscendental fashion, derive the first order KKT conditions and compute its solution using Gröbner bases by treating the uncertain parameters as symbols. The output of this step is a collection of candidate solutions which are explicit in θ and encompass: global and local optima as well as infeasible solutions. For these solutions, examine their dual and primal feasibility along with a constraint qualification so as to remove infeasible candidate solutions. Lastly, in order to report only the globally optimal solutions, perform a comparison procedure [33].

Problem (1) details a general formulation of multi-parametric programs. The case that f and/or g are analytically nonlinear and the uncertain parameters are in the OFCs along with the RHS and LHS of the constraints is used.

Deriving the 1st order KKT conditions of Problem (1) returns a system of equations that is square and is given by Equations (5) and (6).

$$\nabla_{\mathbf{x}} L(\mathbf{x}, \theta) = \mathbf{0} \quad (5)$$

$$\lambda^T \mathbf{g}(\mathbf{x}, \theta) = \mathbf{0} \quad (6)$$

$L(\mathbf{x}, \theta)$ is the Lagrangian function of Problem (1), $\nabla_{\mathbf{x}}$ is the nabla operator with respect to the decision variables and λ are the Lagrange multipliers corresponding to the constraints. Because of the assumption that the nonlinearities have an analytical solution, Gröbner bases can be employed for the solution of the square system of equations because of its elimination property. Even though a tailored implementation of one of the already existing algorithms for computing Gröbner bases may be advantageous from a computational standpoint, it is beyond the scope of the present work and thus Mathematica 10 was employed as the computer algebra software in which the calculations were performed.

By solving the system of Equations (5) and (6), a number of candidate solution sets are returned. Note that although the original optimisation problem involves n_x variables, in the current step, the variables for which we compute the explicit solution are $n_x + n_g$. The candidate solutions include the Lagrange multipliers together with the optimisation variables as explicit functions of the uncertain parameters, i.e., $\lambda(\theta)$ and $\mathbf{x}(\theta)$, respectively.

Definition 7. Candidate solutions [49]

A solution of the Problem (1) is said to be candidate if it satisfies the system of Equations (5) and (6) along with a constraint qualification, e.g., linear independence constraint qualification [15].

In this part of the algorithm, due to strict complementary slackness, the collection of candidate solutions indicates the active and inactive constraints for each solution. Until this

step, the set of solutions computed may be infeasible, local or global optima. By evaluating the primal and dual feasibility of the candidate solutions, the infeasible solutions can be rejected, i.e., Equations (7) and (8).

$$\mathbf{g}(\boldsymbol{\theta}) \leq \mathbf{0} \implies \text{feasibility conditions} \quad (7)$$

$$\boldsymbol{\lambda}(\boldsymbol{\theta}) \geq \mathbf{0} \implies \text{optimality conditions} \quad (8)$$

Conditions (7) and (8) are evaluated by substituting the explicit expressions of the optimisation variables and they form a collection of parametric constraints. If for a candidate solution there exists a subset of the initial parametric space such that the aforementioned inequalities are satisfied, then this region is called the CR of the candidate feasible solution; otherwise, the candidate solution is infeasible and thus removed from further consideration. Note that the evaluation performed at this step, from a computer algebra perspective, is equivalent to computation of the corresponding CAD.

Definition 8. *Critical region [39,49]*

A critical region (CR) is a partition of the parametric space where Conditions (7) and (8) are satisfied for a specific candidate solution. A critical region is characterised by a set of inactive/active constraints and can be discontinuous or nonconvex.

In Algorithm 1, the pseudo-code of the presented method is given. The comparison procedure is outlined in [33].

3.3. Illustrative Example

The proposed methodology will be motivated through the following modified example by Domínguez et al. [29].

$$\min_{x_1, x_2} x + 2x_1^2 - 5x_1 + x_2^2 - 3\theta_1 x_2 - 6$$

Subject to:

$$2x_1 + x_2 \leq 2.4 - \theta_2 \quad (9)$$

$$0.5\theta_3 x_1 + x_2 \leq 1.5$$

$$x_1 \geq 0, x_2 \geq 0$$

$$0 \leq \theta_1 \leq 6, 0 \leq \theta_2 \leq 4, 0 \leq \theta_3 \leq 2$$

In the beginning, the first order KKT conditions of (9) are formulated and we derive the square system of Equations (10) and (14)

$$\nabla_{\mathbf{x}} L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\theta}) = \mathbf{0} \quad (10)$$

$$\lambda_1(2x_1 + x_2 - 2.5 + \theta_2) = 0 \quad (11)$$

$$\lambda_2(0.5\theta_3 x_1 + x_2 - 1.5) = 0 \quad (12)$$

$$\lambda_3(-x_1) = 0 \quad (13)$$

$$\lambda_4(-x_2) = 0 \quad (14)$$

where $L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\theta})$ is the Lagrangian of Problem (9). Equations (10) and (14) can be analytically solved through symbolic computations which return the dual and primal variables as functions of the uncertain parameters, i.e., $\boldsymbol{\lambda}(\boldsymbol{\theta})$ and $\mathbf{x}(\boldsymbol{\theta})$. Systems (10)–(14) are solved in 0.006 s and, as shown in Table A1, fifteen candidate solutions are computed.

The primal and dual feasibility of the candidate solutions is examined by computing the CAD of the related disjunctions. If the result of this step is “False” the solution violates feasibility (in either the primal or dual sense), otherwise, a collection of explicit inequalities is returned which characterises the candidate solution’s CR. The CAD computations of this example take 5.33 s and nine of them are nonempty. Despite the fact that nine candidate

solutions are primal and dual feasible, their global optimality is not guaranteed given the nonconvex nature of the problem. To this effect, for those regions, a new set of CAD computations is performed so as to identify overlaps.

Algorithm 1: mp-NLP under global uncertainty

Input: f, g, x, θ
Output: $x(\theta), CRs$

- 1 Formulate 1st order KKT conditions of mp-NLP
- 2 Solve the 1st order KKT conditions of mp-NLP using Gröbner Bases
- 3 $TEMP \leftarrow solutions$, i.e., $x(\theta), \lambda(\theta), z(\theta)$
- 4 **if** $TEMP = \emptyset$ **then**
- 5 \lfloor mp-NLP is infeasible.
- 6 **else**
- 7 **for** $(i \in range(1, \dots, Length[TEMP]))$ {
- 8 Evaluate with a first order constraint qualification, e.g. Linear Independence Constraint Qualification (LICQ)
- 9 Evaluate with primal and dual feasibility conditions (**Cylindrical Algebraic Decomposition computation**):
- 10 $CR = \{\exists \theta \text{ such that } [\lambda_i(\theta) \geq 0] \wedge [g_i(\theta) \leq 0]\}$
- 11 **if** $CR = \emptyset$ **then** Candidate solution i is infeasible and discard from TEMP.
- 12 **else**
- 13 \lfloor Candidate solution i is feasible and append CR_i to TEMP.
- 14 $i+=1$
- 15 **for** $(k, j) \in range(1, \dots, Length[TEMP]) \wedge k \neq j$ {
- 16 Identify if any overlapping CRs exist (**Cylindrical Algebraic Decomposition computation**):
- 17 $CR_{int} = \{\theta | CR_k \wedge CR_j\}$
- 18 **if** $CR_{int} = \emptyset$ **then** The two CRs are not overlapping
- 19 **else**
- 20 \lfloor Follow the **comparison procedure** from Charitopoulos et al. (2016) so as to remove the overlaps
- 21 Collect the final non-overlapping CRs and the corresponding explicit solutions, i.e., $x(\theta)$

It was found that CR_{10} and CR_{11} were overlapping, as shown by Figure 5, where the overlap (CR_{int}) is shown as the dark partition in between the two CRs.

For the elimination of the resulting overlap, the comparison procedure is invoked and the logic disjunction, as illustrated below, is used for the CAD, as shown by Equations (15) and (16).

$$\exists \theta | \{\theta_1, \theta_2, \theta_3\} \in CR_{int} \wedge z_{CR10}(\theta) \leq z_{CR11}(\theta) \quad (15)$$

or

$$\exists \theta | \{\theta_1, \theta_2, \theta_3\} \in CR_{int} \wedge z_{CR10}(\theta) \geq z_{CR11}(\theta) \quad (16)$$

where $z_{CR_i}(\theta)$ denotes the optimal explicit value within CR_i . The result of this step is partitioning of the parametric space where each explicit solution is the globally optimal. In this case, CR_{10} was shown to be dominant within the common parametric space and thus the overlap was subtracted from CR_{11} . The algorithm terminates once no more overlapping critical regions are identified. In Table 2, an overview of the explicit solutions is presented, while in Figure 6, the final critical regions are shown. Practically, one would consult the CR column of Table 2 to identify based on the uncertain parameter values where the

uncertainty is realised, i.e., CR1 or CR2, and then the optimal cost can be computed by evaluating the corresponding expression from the “Explicit solution” column.

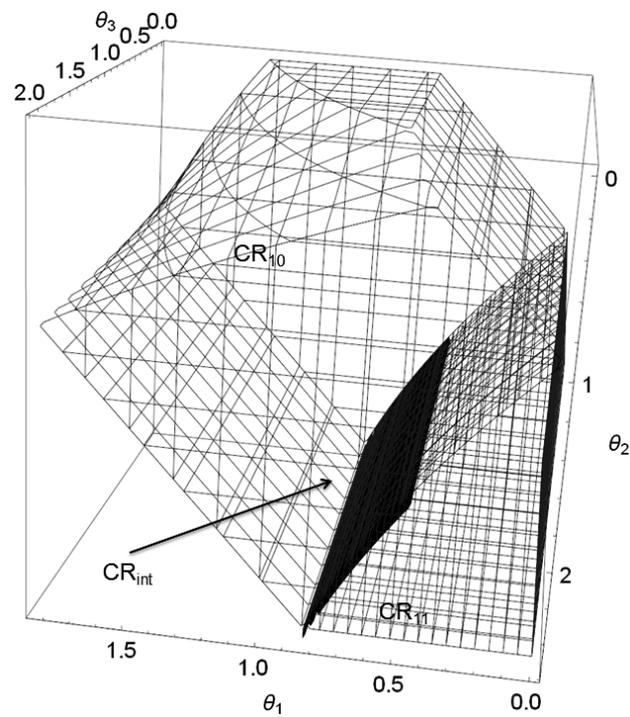


Figure 5. Instance of overlap between CRs.

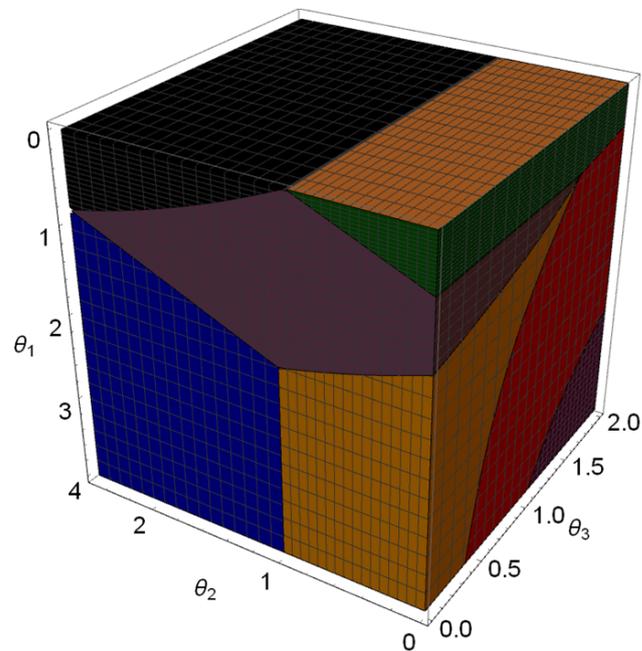


Figure 6. Visualition of the critical regions for motivating example.

Table 2. Optimal explicit solutions of motivating example.

CRs	Explicit Solution
$CR_1 = \begin{cases} 2.667 \leq \theta_1 \leq 4. \\ 0 \leq \theta_2 \leq 1 \\ \frac{10}{3\theta_1 - 3} \leq \theta_3 \leq 2 \end{cases}$	$z_1(\theta) = -3.75 - 4.5\theta_1$
$CR_2 = \begin{cases} \begin{cases} 0.666667\sqrt{19 - 18\theta_1} + \theta_2 \geq 3.8333 \\ 0 \leq \theta_1 \leq 2.005302497 \cdot 10^{-16} \\ 1.5\theta_1 + \theta_2 \leq 0.927401 \\ 0 \leq \theta_3 \leq 2 \end{cases} \\ \begin{cases} 3.2118 \cdot 10^{-8} \sqrt{8.186 \times 10^{15} - 7.7552 \times 10^{15}\theta_1} + \theta_2 \geq 3.8333 \\ 0 \leq \theta_1 \leq 0.8333 \\ \theta_2 \leq 2.5 \\ 0 \leq \theta_3 \leq 2 \end{cases} \end{cases}$	$z_2(\theta) = 0.015625(5 - 2\theta_2)^3 + 0.125(5 - 2\theta_2)^2 - 1.25(5 - 2\theta_2) - 6$

4. Case Studies

Here, we examine two case studies from process systems and their corresponding explicit controllers are designed. The computational experiments were performed on a workstation with 24 GB RAM, a 3.80 GHz processor and a Windows 10 64-bit operating system. For the symbolic calculation, the computer algebra system that was employed was Mathematica 10.2.

4.1. Multiple-Input Multiple-Output Non-Isothermal CSTR

We examine a non-isothermal MIMO multi-product CSTR where the decomposition reaction $A \rightarrow R$ happens under the kinetic law: $-\mathcal{R}_b = k_r C_b$. Additional details on the design and kinetics can be found in Camacho and Alba [50] and the data used for this case study can be found in Table 3. The system has two control inputs: the liquid (F_l) and coolant (F_c) flow rates, whereas the system's states are the temperature of the liquid (T_l) and the concentration of the decomposition product (C_b). Using the mass and energy balances, the dynamic model of the system is derived as given by Equations (17) and (18).

$$\frac{d(V_l C_b)}{dt} = V_l k_r (C_{a0} - C_b) - F_l C_b \quad (17)$$

$$\frac{d(V_l \rho_l C_{pl} T_l)}{dt} = F_l \rho_l C_{pl} T_{l0} - F_l \rho_l C_{pl} T_l + F_c \rho_c C_{pc} (T_{c0} - T_c) + V_l k_r (C_{a0} - C_b) H \quad (18)$$

Table 3. Data of the multiple-input multiple-output CSTR case study.

k_r reaction constant	26 $\frac{1}{h}$
V_l tank volume	24 L
ρ_l liquid density	800 $\frac{kg}{m^3}$
ρ_c coolant density	1000 $\frac{kg}{m^3}$
C_{pl} specific heat of liquid	3 $\frac{kJ}{kg \cdot K}$
C_{pc} specific heat of coolant	4.19 $\frac{kJ}{kg \cdot K}$
T_{l0} entering liquid temperature	283 K
T_{c0} inlet coolant temperature	273 K
T_c outlet coolant temperature	303 K
C_{a0} initial concentration of the reactant	4 $\frac{mol}{L}$

Firstly, the systems (17) and (18) are transformed into an algebraic one in order to design the explicit controllers. Using the forward Euler method, the MPC problem of the discretised system is shown by Equations (19) and (20).

$$\min_{\mathbf{u}} J(\boldsymbol{\theta}) = \sum_{t=0}^{P_H} \left\| \mathbf{x}(t) - \mathbf{x}_{ref} \right\|_2 \tag{19}$$

Subject to:

$$\begin{cases} C_{b_{t+1}} = C_{b_t} + \frac{h_e(k_r(C_{a0}-C_{b_t})-F_t C_{b_t})}{V_l}, 0 \leq t \leq P_H - 1 \\ T_{l_{t+1}} = T_{l_t} + h_e \frac{F_t \rho_l C_{pl} T_{l0} - F_t \rho_l C_{pl} T_{l_t} + F_{c_t} \rho_c C_{pc} (T_{c0} - T_c) + V_l k_r (C_{a0} - C_{b_t})}{V_l \rho_l C_{pl}} \\ 0.8 \leq C_{b_t} \leq 3.5, 0 \leq t \leq P_H \\ 280 \leq T_{l_t} \leq 400, 0 \leq t \leq P_H \\ 0 \leq F_{c_t} \leq 1000, 0 \leq t \leq P_H \\ 0 \leq F_{l_t} \leq 2000, 0 \leq t \leq P_H \\ C_{b_{t=0}} = \theta_1, T_{l_{t=0}} = \theta_2 \\ C_b^{ref} = \theta_3, T_l^{ref} = \theta_4, \end{cases} \tag{20}$$

By employing the presented solution algorithm for mp-NLPs, the related KKT system is solved analytically using Gröbner bases. It takes 0.76 s to compute 29 candidate solutions explicit in θ_1, θ_2 and a collection is shown in Table 4.

Table 4. Collection of candidate solutions of MIMO mp-MPC.

Candidate Solution	F_c	F_l
1	0	0
2	2000	$-0.2011\theta_1 - 38.178\theta_2 - 0.00796\theta_4 + 10807.5$
3	$\frac{23376(\theta_1 - 0.714579)}{\theta_1}$	$\frac{-0.2011(\theta_1^2 + 2219.4\theta_1\theta_2 - 628088.\theta_1 - 1585.96\theta_2 + 448827)}{\theta_1}$
4	$\frac{23376.\theta_1 - 24000.\theta_3 + 2496.}{\theta_1}$	$\frac{-0.201\theta_1^2 - 446.32\theta_1\theta_2 + 126307\theta_1 + 458.23\theta_2\theta_3 - 47.66\theta_2 - 129680\theta_3 + 13486.7}{\theta_1}$

As mentioned in the previous section, the proposed algorithm can facilitate both continuous as well as discrete set-points for the solution of the resulting problem. For the MIMO case study, we consider 8 different set-points for which the explicit control law is derived. In Table A2, we outline the explicit solutions for the different set-points while the optimal partition of the uncertainty space is shown in Figure 7.

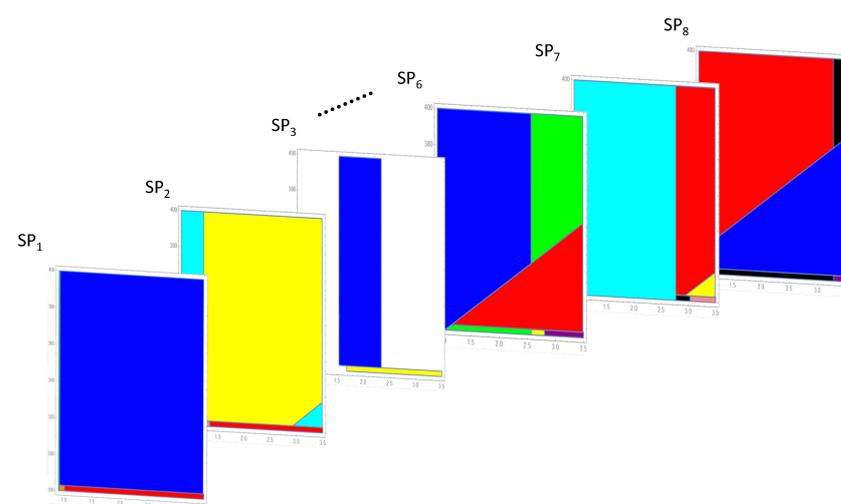


Figure 7. Critical regions for the mp-MPC controller for the MIMO CSTR.

We validate the performance of the explicit multi set-point controller by examining the transition between two steady states. We benchmark the controller’s predictions against the globally optimal solution as computed using the BARON 14.4 solver. In Figure 8, the control and state evolution can be seen.

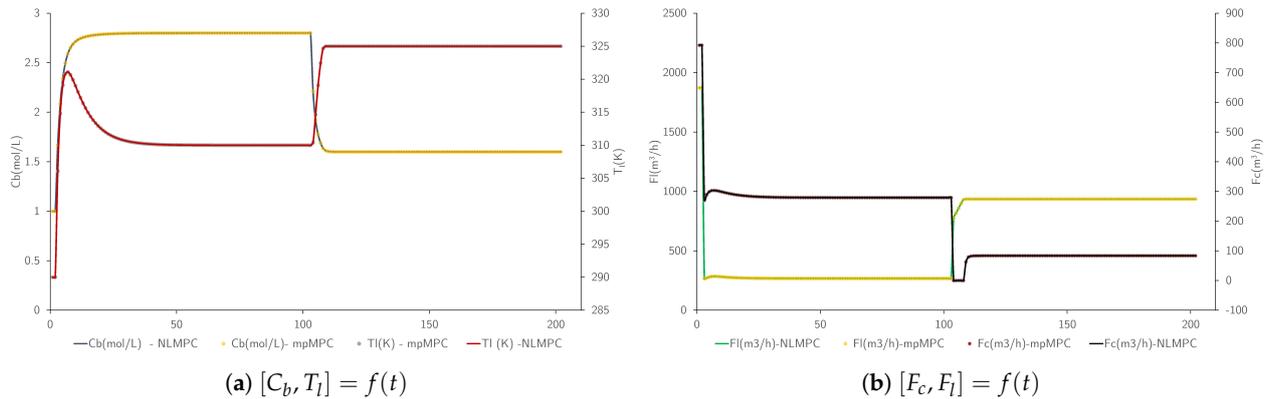


Figure 8. Comparative graphs of the control performance of the mp-MPC vs. NLMPC for case study 1.

4.2. Isothermal Polymerisation CSTR

Next, we examine the design of a multi set-point controller for the grade transition problem with polymerisation CSTRs. The model nonlinearities involve bilinear terms and square roots. The free radical polymerisation reaction happens in a CSTR that operates isothermally at 335 K, where methyl methacrylate (MMA) is produced [51,52]. The mathematical model is given by Equations (21) and (25). The system has 4 state variables, i.e., the concentrations of the monomer (C_m) and the initiator (C_l), the dead chains' molar concentration (D_0) and the dead chains' mass concentration (D_l). The control input is the flow rate of the initiator (F_l) and one output, i.e., the molecular weight of the polymer produced (y). In the multiple steady states, different polymeric grades can be produced corresponding to alternative molecular weights. We provide the notation for this system in Table 5 and model parameter values can be found in Table 6.

$$\frac{dC_m}{dt} = -(k_p + k_{fm}) \sqrt{\frac{2f^*k_l C_l}{k_{Td} + k_{Tc}}} C_m + \frac{F(C_{min} - C_m)}{V} \quad (21)$$

$$\frac{dC_l}{dt} = \frac{F_l C_{lin} - F C_l}{V} - k_l C_l \quad (22)$$

$$\frac{dD_0}{dt} = (0.5k_{Tc} + k_{Td}) \frac{2f^*k_l C_l}{k_{Td} + k_{Tc}} C_m + k_{fm} \sqrt{\frac{2f^*k_l C_l}{k_{Td} + k_{Tc}}} C_m - \frac{F D_0}{V} \quad (23)$$

$$\frac{dD_l}{dt} = M_m (k_p + k_{fm}) \sqrt{\frac{2f^*k_l C_l}{k_{Td} + k_{Tc}}} C_m - \frac{F D_l}{V} \quad (24)$$

$$y = \frac{D_l}{D_0} \quad (25)$$

Table 5. MMA CSTR notation.

C_m (kmol/m ³)	state: monomer concentration
C_l (kmol/m ³)	state: initiator concentration
D_0 (kmol/m ³)	state: molar concentration of dead chains
D_l (kg/m ³)	state: mass concentration of dead chains
F_l (m ³ /h)	control: initiator flow rate
$y = D_l/D_0$	output: molecular weight

Table 6. Model parameters for the MMA polymerisation reactor.

$F = 10.0 \text{ m}^3/\text{h}$	monomer flow rate
$V = 10.0 \text{ m}^3$	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
$C_{I_{in}} = 8.00 \text{ kmol}/\text{m}^3$	inlet initiator concentration
$C_{m_{in}} = 6.00 \text{ kmol}/\text{m}^3$	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
$k_l = 1.02 \times 10^{-1} \text{ h}^{-1}$	initiation rate constant
$M_m = 100.12 \text{ kg}/\text{kmol}$	molecular weight of monomer

The model nonlinearities are not transcendental and, thus, the presented method for the design of the multi set-point mp-MPC can be used. This polymerisation system has been examined intensively by the research community and it has been noted that online computation of its optimal control law can be challenging due to numerical instabilities arising because of scaling issues [52]. As a trade-off between computational complexity and stability of the integration scheme, we employ the forward Euler method with a step size of $h = 36 \text{ s}$.

Following the proposed method, the globally optimal solutions are computed, whereas employing off-the-self global optimisation solvers for online implementation leads to extensive computational times. In Table 7, the results using the BARON 14.4 solver in GAMS with different prediction horizons are given.

Table 7. Computational effort for varying prediction horizons using BARON 14.4 [53].

Prediction Horizon	CPU (s)
1	603
2	1906
5	3600 [†]
10	3600 [†]
20	3600 [†]

[†] Reached time limit.

In this case study, eight uncertain parameters were considered, one for each state and set-point. In Table 8, we provide a mapping of the uncertain parameters of the control problem. Whilst having the set-point as continuous, uncertain parameters increase the computational complexity of the mp-P and one could argue that in the context of systems integration where the APC receives data by the real-time optimisation functionality, the same set-points may not always be realised. In such cases, following conventional mp-MPC frameworks, the explicit laws would have to be recomputed from the beginning (mp-P solution and implementation of the explicit solutions, possibly in a microchip), whereas, following the proposed framework, if the bounds of the set-points remain within the prespecified ranges, then the same multilayer controller can be readily used.

Table 8. MMA CSTR uncertain parameters.

Parameter	Range	Notation
θ_1	$\in [0, 5]$	$C_{m t=0}$
θ_2	$\in [0, 0.5]$	$C_{l t=0}$
θ_3	$\in [0, 0.05]$	$D_{0 t=0}$
θ_4	$\in [0, 300]$	$D_{l t=0}$
θ_5	$\in [0, 5]$	Set-points for C_m
θ_6	$\in [0, 0.5]$	Set-points for C_l
θ_7	$\in [0, 0.05]$	Set-points for D_0
θ_8	$\in [0, 300]$	Set-points for D_l

The resulting mp-NLP involves one optimisation variable, eight uncertain parameters and ten constraints when a prediction horizon of unity is considered. Overall, we seek analytical solutions to eleven variables, i.e., the Lagrange multipliers and the optimisation variables. The solution of the mp-NLP returns five candidate solutions, as shown in Table A3.

Substituting the explicit expressions into the constraints, the feasible region is projected into the uncertainty space. Candidate solutions that satisfy primal/dual feasibility are considered for the next step of the algorithm; otherwise, they are discarded as infeasible. For instance, the 6th candidate solution violates dual feasibility as any value of θ_6 would result in negative λ_8 .

Subsequently, the explicit inequalities for the remaining solutions are examined. The intersection of the feasible regions defined by the parametric inequalities defines the critical regions of the candidate solution. Because of the nonconvex nature of the problem, it is likely that explicit solutions may be valid in the same uncertainty space, thus overlapping. In order to compute only the global explicit solutions, we employ the comparison procedure. Three overlapping solutions were identified. An example of the inequalities defining the overlap between CR_1 , CR_2 is shown by Equation (26).

$$CR_{int} := CR_1 \cap CR_2 = \begin{cases} 0 \leq \theta_3 \leq 0.05 \\ 0 \leq \theta_4 \leq 300 \\ 0 \leq \theta_5 \leq 5 \\ 0 \leq \theta_7 \leq 0.05 \\ 0 \leq \theta_4 \leq 300 \\ \theta_2 + 0.00242674 = 1.01114\theta_6 \\ 0 \leq \theta_2 \leq 0.199802 \\ 0 \leq \theta_1 \leq 2.72345 \end{cases} \quad \begin{cases} 4.9899 \leq \theta_1 \leq 5 \\ 1617.74 + \frac{40280.2}{\theta_1^2} \leq \frac{16144.7}{\theta_1} + \theta_2 \\ \theta_2 \leq \frac{1.48197}{\theta_1^2} \end{cases} \quad (26)$$

For illustration purposes, the mathematical definition of CR_2 is given by Equation (A1) in the Appendix A. Due to the extensive set of inequalities defining the rest of the CRs, we do not detail them in the manuscript for the sake of space.

After the algorithm's convergence and with the optimal explicit solutions reported, the performance of the multi set-point mp-MPC's explicit control law is compared to that of conventional MPC. The solution of the online MPC is found by implementing the related NLP in GAMS and solving it to global optimality using BARON 14.4. As can be seen in Figure 9, the state and control evolution of the system are in perfect agreement when the two schemes are compared, thus highlighting the accuracy and correctness of the proposed framework while in Figure 10 the stability of the resulting control policy can be envisaged.

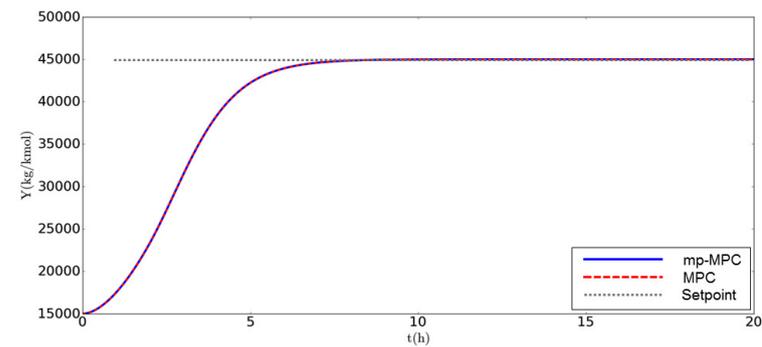
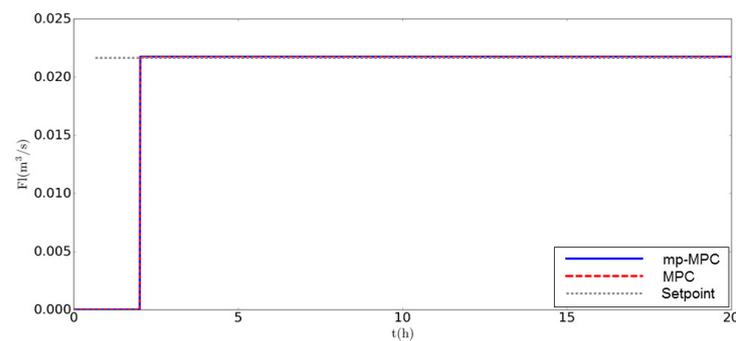
(a) $y = f(t)$ (b) $F_l = f(t)$

Figure 9. Plots comparing the control solutions computed by the proposed method (mp-MPC) and online NLMPC for the polymerisation CSTR.

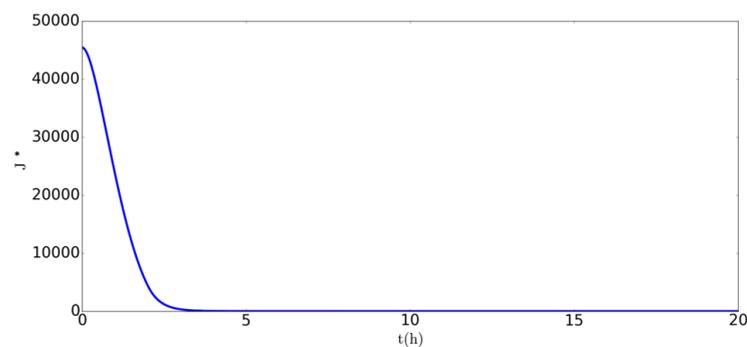


Figure 10. Graph of the control cost function vs. time for the polymerisation CSTR.

We assess the performance of the controller for set-point tracking between two set-points. At the beginning, we assume the CSTR to be operated at a steady state of $y = 15,000 \frac{kg}{kmol}$ and then controlled towards $y = 45,000 \frac{kg}{kmol}$ where it is regulated for nine hours to produce a specific polymer grade. Next, the controller steers the system to the next set-point ($y = 19,250 \frac{kg}{kmol}$), in which steady state another polymer is produced. The performance of the set-point tracking can be seen in Figure 11.

Finally, with respect to the scalability of the proposed method, a number of systems and prediction horizon settings were examined and, as shown in Table 9, for the current state of the art in computer algebra software, only small- to medium-scale systems can be efficiently facilitated.

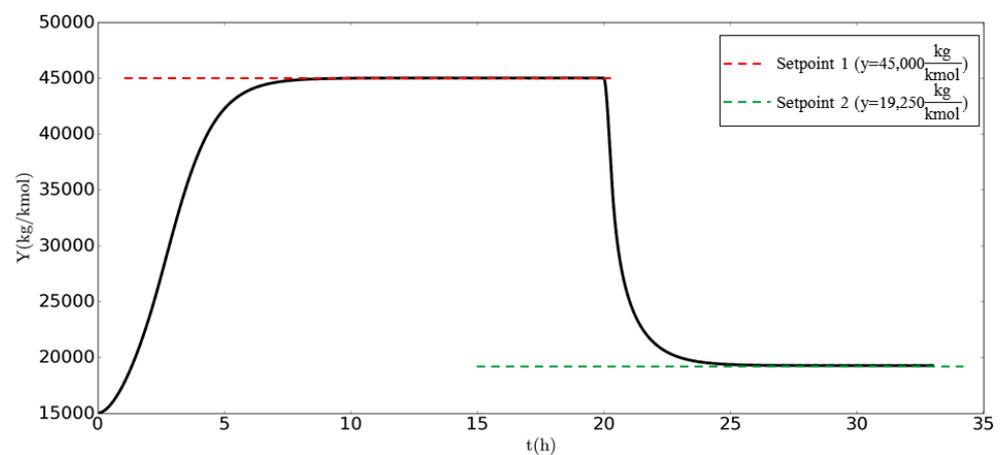


Figure 11. Set-point tracking performance of the polymerisation reactor case study.

Table 9. Computational statistics of the proposed method for different case studies.

Case Study	Problem Statistics			Uncertainty	CPU (s)
	n_x	n_g	n_θ		
SISO CSTR ($P_H = 1$)	1	2	2	OFC, RHS	1.32
SISO CSTR ($P_H = 2$)	2	4	2	OFC, RHS	65.8
SISO CSTR ($P_H = 2$)	3	6	2	OFC, RHS	4652
MMA CSTR ($P_H = 1$)	1	10	8	OFC, RHS	1.86
MMA CSTR ($P_H = 1$)	1	10	9	OFC, RHS, LHS	4.86
MMA CSTR ($P_H = 2$)	2	20	8	OFC, RHS	Memory limit
MIMO CSTR ($P_H = 1$)	2	8	4	OFC, RHS	2.76
MIMO CSTR ($P_H = 2$)	4	16	4	OFC, RHS	192.75
MIMO CSTR ($P_H = 3$)	6	24	4	OFC, RHS	5929

5. Conclusions

We have presented a computer algebra-based algorithm for the analytical solution of mp-NLPs that involve uncertain parameters on the RHS and OFC as well as the LHS of the constraints. In the first step, Gröbner bases are used for symbolically expressing the optimisation variables and the Lagrange multipliers as functions of the uncertain parameters. Next, by computing cylindrical algebraic decompositions, the globally optimal CRs are defined. Building upon the proposed algorithm, we introduce a framework for the design of multi set-point explicit MPC for nonlinear systems. The proposed technique expands the scope of mp-MPCs, as we illustrate that it is feasible to design a single “multi-layer” controller for capturing set-point tracking problems and potentially new model parameter estimations. Ongoing research focuses on the latter and how current progress in algebraic geometry can alleviate the related computational burden and allow for solutions of large-scale studies. Specifically, the application of machine learning techniques for faster evaluations of standard atomic formulas and thus reductions in the computational expense of CAD calculations is a promising direction.

Author Contributions: Conceptualization: V.M.C. and V.D.; Data curation: V.M.C.; Formal Analysis: V.M.C.; Funding acquisition: L.G.P. and V.D.; Investigation: V.M.C.; Methodology V.M.C., L.G.P. and V.D.; Visualization: V.M.C.; Writing—original draft: V.M.C.; Writing—review & editing: V.M.C. and V.D. All authors have read and agreed to the published version of the manuscript.

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

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

Conflicts of Interest: The authors declare no conflict of interest.

Nomenclature

APC	Advanced Process Control
CAD	Cylindrical Algebraic Decomposition
CAS	Computer Algebra Software
CR	Critical Region
CSTR	Continuous Stirred Tank Reactor
e-MPC	Economic Model Predictive Control
EWO	Enterprise Wide Optimisation
KKT	Karush–Kuhn–Tucker
LHS	Left-hand Side
MIMO	Multiple Input Multiple Output
MINLP	Mixed Integer Nonlinear Program
MMA	Methyl Methacrylate
mp-(NLP)	Multi-parametric (Nonlinear) Program
NLMPC	Nonlinear Model Predictive Control
OFC	Objective Function Coefficient
RHS	Right-hand Side
(s/d)-RTO	(Static/Dynamic) Real-time Optimisation

Appendix A

Table A1. Candidate solutions of motivating example.

	x_1	x_2	λ_1
1	-2.12	0	0
2	-2.12	$1.5\theta_1$	0
3	0	1.5	0
4	0	$0.5(5 - 2\theta_2)$	$3\theta_1 + 2\theta_2 - 5$
5	0	0	0
6	0	$1.5\theta_1$	0
7	0.786	0	0
8	0.786	$1.5\theta_1$	0
9	$-0.577\sqrt{-6\theta_1 - 4\theta_2 + 27} - 2$	$0.5(2.309\sqrt{-6\theta_1 - 4\theta_2 + 27} - 2\theta_2 + 13)$	$0.333(-6.9282\sqrt{-6\theta_1 - 4\theta_2 + 27} + 9\theta_1 + 6\theta_2 - 39)$
10	$0.333(1.732\sqrt{-6\theta_1 - 4\theta_2 + 27} - 6)$	$0.166(-6.928\sqrt{-6\theta_1 - 4\theta_2 + 27} - 6\theta_2 + 39)$	$2.309\sqrt{-6\theta_1 - 4\theta_2 + 27} + 3\theta_1 + 2\theta_2 - 13$
14	$0.0833\left(-\sqrt{\theta_3^4 - 72\theta_1\theta_3 + 16\theta_3^2 + 72\theta_3 + 304 - \theta_3^2} - 8\right)$	$0.0416\left(\theta_3\sqrt{\theta_3^4 - 72\theta_1\theta_3 + 16\theta_3^2 + 72\theta_3 + 304 + \theta_3 + 8\theta_3 + 36}\right)$	0
15	$0.0833\left(\sqrt{\theta_3^4 - 72\theta_1\theta_3 + 16\theta_3^2 + 72\theta_3 + 304 - \theta_3^2} - 8\right)$	$0.0416\left(-\theta_3\sqrt{\theta_3^4 - 72\theta_1\theta_3 + 16\theta_3^2 + 72\theta_3 + 304 + \theta_3 + 8\theta_3 + 36}\right)$	0

	λ_2	λ_3	λ_4
1	0	0	$-3\theta_1$
2	0	0	0
3	$3(\theta_1 - 1)$	$0.5(3\theta_1\theta_3 - 3\theta_3 - 10)$	0
4	0	$6\theta_1 + 4\theta_2 - 15$	0
5	0	-5	$-3\theta_1$
6	0	-5	0
7	0	0	$-3\theta_1$
8	0	0	0
9	0	0	0
10	0	0	0
14	$0.0833(-\theta_3\sqrt{\theta_3^4 - 72\theta_1\theta_3 + 16\theta_3^2 + 72\theta_3 + 304 + 36\theta_1 - \theta_3^2} - 8\theta_3 - 36)$	0	0
15	$0.0833(\theta_3\sqrt{\theta_3^4 - 72\theta_1\theta_3 + 16\theta_3^2 + 72\theta_3 + 304 + 36\theta_1 - \theta_3^2} - 8\theta_3 - 36)$	0	0

Table A2. Final CRs and explicit solutions for $P_H = 1$ of the MIMO CSTR for two of the set-points.

Set-Point	Explicit Solution
$(\theta_3 = 1, \theta_4 = 290)$	$(\theta_1, \theta_2) \in CR_1^{sp1} \begin{cases} F_c(t=0) = 0 \\ F_l(t=0) = 0.00796\theta_2 - 0.2011\theta_1 - 0.00796\theta_4 + 0.8044 \end{cases}$
	$(\theta_1, \theta_2) \in CR_2^{sp1} \begin{cases} F_c(t=0) = 2000 \\ F_l(t=0) = 10807.5 - 0.2011\theta_1 - 38.178\theta_1 - 0.00796\theta_4 \end{cases}$
	$(\theta_1, \theta_2) \in CR_3^{sp1} \begin{cases} F_c(t=0) = \frac{23376(\theta_1 - 3.487)}{\theta_1} \\ F_l(t=0) = 0 \end{cases}$
	$(\theta_1, \theta_2) \in CR_4^{sp1} \begin{cases} F_c(t=0) = \frac{23376\theta_1^2 + \theta_1(-3.495 \cdot 10^{10}\theta_2 - 24000\theta_3 + 9.889 \cdot 10^{12}) + \theta_2 \times 10^9(1.38\theta_2 - 1.38\theta_4 - 251) + 3.913 \times 10^{11}\theta_4 - 3.95 \times 10^{13}}{\theta_1^2 + 3.318 \times 10^9\theta_2^2 - 1.878 \times 10^{12}\theta_2 + 2.658 \times 10^{14}} \\ F_l(t=0) = 0 \end{cases}$
	$(\theta_1, \theta_2) \in CR_5^{sp1} \begin{cases} F_c(t=0) = \frac{23376\theta_1 - 24000\theta_3 + 2496}{\theta_1} \\ F_l(t=0) = \frac{-0.2011\theta_1^2 + \theta_1(-446.31\theta_2 - 0.00796\theta_4 + 126309) + \theta_2(458.24\theta_3 - 47.66) - 129680\theta_3 + 13486.7}{\theta_1} \end{cases}$
$(\theta_3 = 1.6, \theta_4 = 325)$	$(\theta_1, \theta_2) \in CR_1^{sp2} \begin{cases} F_c(t=0) = 2000 \\ F_l(t=0) = 10807.5 - 0.2011\theta_1 - 38.178\theta_1 - 0.00796\theta_4 \end{cases}$
	$(\theta_1, \theta_2) \in CR_2^{sp2} \begin{cases} F_c(t=0) = \frac{23376(\theta_1 - 3.487)}{\theta_1} \\ F_l(t=0) = 0 \end{cases}$

14. Bank, B.; Guddart, J.; Klatte, D.; Kummer, B.; Tammer, K. *Non-Linear Parametric Optimization*; Springer Academic: Berlin/Heidelberg, Germany, 1983.
15. Kojima, M. *Strongly Stable Stationary Solutions in Nonlinear Programs*; Academic Press: Cambridge, MA, USA, 1980; Volume 43, pp. 93–138.
16. Levitin, E. Differentiability with respect to a parameter of the optimal value in parametric problems of mathematical programming. *Cybern. Syst. Anal.* **1976**, *12*, 46–64.
17. Robinson, S.M. Strongly regular generalized equations. *Math. Oper. Res.* **1980**, *5*, 43–62. [[CrossRef](#)]
18. Robinson, S.M. Generalized equations and their solutions, Part I: Basic theory. In *Point-to-Set Maps and Mathematical Programming*; Springer: Berlin/Heidelberg, Germany, 1979; pp. 128–141.
19. Kyparisis, J. Uniqueness and differentiability of solutions of parametric nonlinear complementarity problems. *Math. Prog.* **1986**, *36*, 105–113. [[CrossRef](#)]
20. Ralph, D.; Dempe, S. Directional derivatives of the solution of a parametric nonlinear program. *Math. Prog.* **1995**, *70*, 159–172. [[CrossRef](#)]
21. Dua, V.; Pistikopoulos, E.N. Algorithms for the solution of multiparametric mixed-integer nonlinear optimization problems. *Ind. Eng. Chem. Res.* **1999**, *38*, 3976–3987. [[CrossRef](#)]
22. Acevedo, J.; Salgueiro, M. An efficient algorithm for convex multiparametric nonlinear programming problems. *Ind. Eng. Chem. Res.* **2003**, *42*, 5883–5890. [[CrossRef](#)]
23. Johansen, T.A. On multi-parametric nonlinear programming and explicit nonlinear model predictive control. In Proceedings of the 41st IEEE Conference on Decision and Control, Las Vegas, NV, USA, 10–13 December 2002; Volume 3, pp. 2768–2773.
24. Domínguez, L.F.; Pistikopoulos, E.N. A quadratic approximation-based algorithm for the solution of multiparametric mixed-integer nonlinear programming problems. *AIChE J.* **2013**, *59*, 483–495. [[CrossRef](#)]
25. Johansen, T.A. Approximate explicit receding horizon control of constrained nonlinear systems. *Automatica* **2004**, *40*, 293–300. [[CrossRef](#)]
26. Narciso, D.A. *Developments in Nonlinear Multiparametric Programming and Control*. Ph.D. Thesis, Imperial College, London, UK, 2009.
27. Leverenz, J.; Xu, M.; Wiecek, M.M. Multiparametric optimization for multidisciplinary engineering design. *Struct. Multidiscipl. Optim.* **2016**, *54*, 795–810. [[CrossRef](#)]
28. Dua, V.; Papalexandri, K.P.; Pistikopoulos, E.N. Global optimization issues in multiparametric continuous and mixed-integer optimization problems. *J. Glob. Optim.* **2004**, *30*, 59–89. [[CrossRef](#)]
29. Domínguez, L.F.; Narciso, D.A.; Pistikopoulos, E.N. Recent advances in multiparametric nonlinear programming. *Comput. Chem. Eng.* **2010**, *34*, 707–716. [[CrossRef](#)]
30. Hale, E.T. *Numerical Methods for d-Parametric Nonlinear Programming with Chemical Process Control and Optimization Applications*. Ph.D. Thesis, The University of Texas at Austin, Austin, TX, USA, 2005.
31. Fotiou, I.A.; Parrilo, P.A.; Morari, M. Nonlinear parametric optimization using cylindrical algebraic decomposition. In Proceedings of the 44th IEEE Conference on Decision and Control and 2005 European Control Conference, Seville, Spain, 15 December 2005; pp. 3735–3740.
32. Fotiou, I.A.; Rostalski, P.; Parrilo, P.A.; Morari, M. Parametric optimization and optimal control using algebraic geometry methods. *Int. J. Control* **2006**, *79*, 1340–1358. [[CrossRef](#)]
33. Charitopoulos, V.M.; Dua, V. Explicit model predictive control of hybrid systems and multiparametric mixed integer polynomial programming. *AIChE J.* **2016**, *62*, 3441–3460. [[CrossRef](#)]
34. Pappas, I.; Diangelakis, N.A.; Pistikopoulos, E.N. The exact solution of multiparametric quadratically constrained quadratic programming problems. *J. Glob. Optim.* **2020**, *1*–27. [[CrossRef](#)]
35. Charitopoulos, V.M.; Dua, V. A unified framework for model-based multi-objective linear process and energy optimisation under uncertainty. *Appl. Energy* **2017**, *186*, 539–548. [[CrossRef](#)]
36. Bretti, G.; Piccoli, B. A tracking algorithm for car paths on road networks. *SIAM J. Appl. Dyn. Syst.* **2008**, *7*, 510–531. [[CrossRef](#)]
37. Patrinos, P.; Sarimveis, H. A new algorithm for solving convex parametric quadratic programs based on graphical derivatives of solution mappings. *Automatica* **2010**, *46*, 1405–1418. [[CrossRef](#)]
38. Sun, M.; Chachuat, B.; Pistikopoulos, E.N. Design of multi-parametric NCO tracking controllers for linear dynamic systems. *Comput. Chem. Eng.* **2016**, *92*, 64–77. [[CrossRef](#)]
39. Charitopoulos, V.M. *Uncertainty-Aware Integration of Control with Process Operations and Multi-Parametric Programming under Global Uncertainty*; Springer Nature: Berlin/Heidelberg, Germany, 2020.
40. Pappas, I.; Kenefake, D.; Burnak, B.; Avraamidou, S.; Ganesh, H.S.; Katz, J.; Diangelakis, N.A.; Pistikopoulos, E.N. Multiparametric Programming in Process Systems Engineering: Recent Developments and Path Forward. *Front. Chem. Eng.* **2021**, *2*. [[CrossRef](#)]
41. Buchberger, B. Bruno Buchberger’s PhD thesis 1965: An algorithm for finding the basis elements of the residue class ring of a zero dimensional polynomial ideal. *J. Symb. Comput.* **2006**, *41*, 475–511. [[CrossRef](#)]
42. Lazard, D. Gröbner bases, Gaussian elimination and resolution of systems of algebraic equations. In *Computer Algebra: EUROCAL83, European Computer Algebra Conference London*; van Hulzen, J.A., Ed.; Springer: Berlin/Heidelberg, Germany, 1983; pp. 146–156.

43. Bochnak, J.; Coste, M.; Roy, M.F. *Real Algebraic Geometry*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2013; Volume 36, pp. 23–54.
44. Faugere, J.C. A new efficient algorithm for computing Gröbner bases (F4). *J. Pure Appl. Algebr.* **1999**, *139*, 61–88. [[CrossRef](#)]
45. Faugere, J.C. *Computing Gröbner Basis without Reduction to Zero (F5)*; Technical Report; LIP6: Paris, France, 1998.
46. Collins, G.E. Quantifier elimination for real closed fields by cylindrical algebraic decomposition. In *Automata Theory and Formal Languages 2nd GI Conference*; Springer: Berlin/Heidelberg, Germany, 1975; pp. 134–183.
47. Jirstrand, M. *Cylindrical Algebraic Decomposition—An Introduction*; Linköping University: Linköping, Sweden, 1995.
48. Charitopoulos, V.M.; Dua, V.; Papageorgiou, L.G. Traveling Salesman Problem-Based Integration of Planning, Scheduling, and Optimal Control for Continuous Processes. *Ind. Eng. Chem. Res.* **2017**, *56*, 11186–11205. [[CrossRef](#)]
49. Charitopoulos, V.M.; Papageorgiou, L.G.; Dua, V. Multi-parametric linear programming under global uncertainty. *AIChE J.* **2017**, *63*, 3871–3895. [[CrossRef](#)]
50. Camacho, E.F.; Alba, C.B. *Model Predictive Control*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2013.
51. Daoutidis, P.; Soroush, M.; Kravaris, C. Feedforward/feedback control of multivariable nonlinear processes. *AIChE J.* **1990**, *36*, 1471–1484. [[CrossRef](#)]
52. BenAmor, S.; Doyle, F.J.; McFarlane, R. Polymer grade transition control using advanced real-time optimization software. *J. Process Control* **2004**, *14*, 349–364. [[CrossRef](#)]
53. Kılınç, M.R.; Sahinidis, N.V. Exploiting integrality in the global optimization of mixed-integer nonlinear programming problems with BARON. *Optim. Meth. Soft.* **2018**, *33*, 540–562. [[CrossRef](#)]