Chemical Engineering Journal 425 (2021) 131632

Contents lists available at ScienceDirect

Chemical Engineering Journal

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

A machine learning-based methodology for multi-parametric solution of chemical processes operation optimization under uncertainty

Ahmed Shokry [a,e,*], Sergio Medina-González [b], Piero Baraldi [c], Enrico Zio [c,d], Eric Moulines [a], Antonio Espuña [e]

[a] Center for Applied Mathematics, Ecole Polytechnique, Route de Saclay, 91120 Palaiseau, France
[b] Departamento de Posgrado, Facultad de Ciencias Químicas, Universidad Autónoma de Chihuahua, 31125 Chihuahua, Mexico
[c] Energy Department, Politecnico di Milano, Via Lambruschini 4, 20156 Milan, Italy
[d] MINES ParisTech, PSL Research University, CRC, Sophia Antipolis, France
[e] Department of Chemical Engineering, Universitat Politècnica de Catalunya, EEBE – Eduard Maristany, 14, 08019 Barcelona, Spain

ARTICLE INFO

Keywords:
Chemical processes
Operation optimization
Uncertainty
Multiparametric programming
Machine learning
Kriging
Gaussian process regression

ABSTRACT

Chemical process operation optimization aims at obtaining the optimal operating set-points by real-time solution of an optimization problem that embeds a steady-state model of the process. This task is challenged by unavoidable Uncertain Parameters (UPs) variations. MultiParametric Programming (MPP) is an approach for solving this challenge, where the optimal set-points must be updated online, reacting to sudden changes in the UPs. MPP provides algebraic functions describing the optimal solution as a function of the UPs, which allows alleviating large computational cost required for solving the optimization problem each time the UPs values vary. However, MPP applicability requires a well-constructed mathematical model of the process, which is not suited for process operation optimization, where complex, highly nonlinear and/or black-box models are usually used. To tackle this issue, this paper proposes a machine learning-based methodology for multiparametric solution of continuous optimization problems. The methodology relies on the offline development of data-driven models that accurately approximate the multiparametric behavior of the optimal solution over the UPs space. The models are developed using data generated by running the optimization using the original complex process model under different UPs values. The models are, then, used online to, quickly, predict the optimal solutions in response to UPs variation. The methodology is applied to benchmark examples and two case studies of process operation optimization. The results demonstrate the methodology effectiveness in terms of high prediction accuracy (less than 1% of NRMSE, in most cases), robustness to deal with problems of different natures (linear, bilinear, quadratic, nonlinear and/or black boxes) and significant reduction in the complexity of the solution procedure compared to traditional approaches (a minimum of 67% reduction in the optimization time).

1. Introduction

In the decision-making hierarchy of chemical plants management, plant-wide optimization, or process operation optimization, is a principle layer that receives, as inputs, the outcomes and decisions coming from the above layers (i.e., planning and scheduling) [20,54]. These inputs, basically, include forecasts of prices and demands, production rate targets over long time periods (weeks/days), assignment of resources to activities, (raw material allocation, tasks to units allocation, maintenance interventions, staffing), sequencing of activities and determination of starting and ending times for the execution over short periods of time [36,39].

The goal of the process operations optimization is to obtain the optimal values of the process variables (temperatures, pressures, compositions, flow rates, etc.) at which the plant and its units must operate to maximize certain performance criteria (e.g., efficiency, profit, operational cost), while satisfying all the constraints (equipment capacities, environmental restrictions, etc.) and requirements (product quality, production yields, safety, etc.) [2,3,62]. This is achieved by solving, in real time, an optimization problem, which embeds a detailed and rigorous steady-state model of the process [56]. Depending on the model characteristics, such as its structure, transparency (e.g., white, gray,

* Corresponding author.
E-mail address: ahmed.shokry@polytechnique.edu (A. Shokry).

https://doi.org/10.1016/j.cej.2021.131632

Available online 10 August 2021

black-box) and availability of derivative information, and on the formulation of the objective(s) and constraints of the optimization problem, different algorithms can be used, like derivative-free algorithms (e.g., Genetic Algorithms), where the explicit values of the objective(s) function are used to direct the optimization search, or derivative-based algorithms (e.g., interior point algorithms), where the optimization search is directed based on the derivatives of the objective (s) with respect to the decision variables [4,55].

In order to maximize the reliability of the optimal set points when they are implemented in the plant, the process-model mismatch is minimized by updating the process model parameters (e.g., heat transfer coefficients, catalyst activities, distillation column tray efficiencies) before performing the operation optimization, relying on reconciled estimates of the steady-state measurements of the plant variables [2,3,12]. These estimates are often obtained by applying data reconciliation and gross error detection techniques to the real data collected by the sensors in order to reduce the effect of random errors and sensor faults (bias, drifting, miss-calibration, total failure, etc.), respectively [5].

Recently, there has been growing interest to use complex and high-fidelity mathematical models of the process in the operation optimization task. Although these models are able to capture more detailed features and sophisticated characteristics of the process and, consequently, provide more accurate estimation of its behavior, they show many challenging characteristics which lead to practical drawbacks [14,40,52], such as

i) high nonlinearity and complexity, due to the sophisticated phenomena typically involved (thermodynamics, reactions kinetics, heat and mass transfer, etc.) and to the large number of equations contained in the analytical or First Principle Models (FPMs). For example, a full-scale model of a refinery could contain more than a million of equations [21],
ii) complex model architectures, which, in many cases, appear to the user as black-boxes with no access to the embedded first principle equations [4],
iii) large computational cost required for the model simulation due to the complexity of the utilized numerical solution procedures –e.g., iterative schemes and/or integration techniques- [17].

These challenging characteristics inherent to the FPMs of chemical processes represent an obstacle to their use for the operation optimization, especially for large scale and/or fast dynamic processes [29]. For example, the optimization of a full-scale petrochemical plant (crude oil and gas treatment facilities, refineries, etc.) based on its FPM model requires several hours to obtain the optimal solution, and in many cases, the optimization process may fail to converge [26,55].

To overcome the above drawbacks and cope with the above challenges, Surrogate Based Optimization (SBO) approaches have been proposed and have received significant attention in the chemical process industry area [52]. The basic idea of SBO is to use the original complex FPM to generate input–output data points by "Computer Experiments", and use them to develop accurate, but simple and fast-running, data-driven models ("metamodels" or "surrogate models"). Then, these data-driven models are used in replacement of the complex FPM in optimization problem [43]. Two classes of SBO approaches can be identified. The first one is based on the development of global metamodels approximating the entire modular simulator or flow-sheet of the process (the FPM), i.e., the input and output variables of the metamodels are the main input and output variables of interest selected over the entire process flow-sheet [6,7,30,42–44,57,59]. The second class of approaches is based on partitioning the simulation model into different units or subgroups of units, for each of which a dedicated surrogate model is developed; then, the different surrogates are aggregated to constitute the final approximate model of the process [4,21,51,52,55]. A comprehensive review about metamodeling and SBO methods applied in the chemical process engineering area can be found in [26]. One challenge for the applications of SBO approaches to process operation optimization comes from the fact that surrogate models are trained on data generated by FPMs with values of parameters and conditions predefined so as to match the real process behavior. So, any sudden and uncertain change in these parameters and conditions in the real process makes the surrogate models and, consequently, the obtained optimal solution based on their analysis, no longer valid/realistic.

The presence of uncertainty in the process is unavoidable at various levels [24], including inherent physical properties (e.g., kinetic rates, heat transfer constants) [8,14,40] and process fluctuations (e.g., feed streams properties like temperatures, pressures and concentrations, recipe variations, processing time, equipment efficiencies) [38,45], as well as external uncertainty (such as resources, prices, demands) [32]. Many methods have been developed for handling uncertainty in optimization problems, most of them can be categorized into two main approaches: proactive and reactive [37]. The proactive approach aims at providing conservative optimal decisions, which minimize the consequences of the uncertainty on the performance measures of the system (i.e., objective(s)) [24]. Stochastic programming and robust optimization are among the most popular proactive methods [18]. In stochastic programming methods, the UPs are treated as stochastic variables with "a-priori" known probability distribution functions, whose parameters are estimated from historical data. Then, the goal becomes to identify the optimal decision variables that maximize/minimize the expected value of the objective function(s) and achieve feasibility over the distribution of the Ups [32]. Robust optimization methods deal with unknown but bounded UPs and aim at finding robust optima that ensures the feasibility of the solution and the immunity of the performance measure over the entire range of realizations of the Ups [40]. The three main limitations associated to the use of these methods are: *i*) the large computational effort required to obtain the optimal solution, since these methods imply the analysis of a large number of scenarios, which grows with the number of UPs, *ii*) the need of a complete knowledge of the characteristics of the UPs to identify their types and probability distributions, which is unrealistic especially in dynamic environments and *iii*) the fact that the provided solution becomes suboptimal for most of the realizations of the uncertainties during the operation/production [32,48].

The reactive approach, instead, is considered when it is necessary to promptly provide online update of the optimal values of the decision variables in response to real-time changes of the UPs values, which can be measured once unveiled. Since the reactive approach is able to provide the optimal solution for each realization of the UPs, it is preferred in dynamic production environments [50]. Among the reactive methods, MPP is considered as one of the most powerful tools, as it is able to explicitly map and explain the effect of the parametric uncertainty and variability on optimal decisions [46]. The general MPP problem is expressed as

$$\left.\begin{array}{c} \min\limits_{x} Z(x,\theta) \\ S.T. \\ g_l(x,\theta) \le 0, \quad l = 1,2,\ldots L \\ x \in R^K, \quad \theta \in R^k, \\ lb_x \le x \le ub_x, \quad lb_\theta \le \theta \le ub_\theta \end{array}\right\} \tag{1}$$

Where $x \in R^K$ is the vector of decision variables, $\theta \in R^k$ is a set of uncertain parameters and $g_l(x,\theta)$ is a set of constraints. Generally speaking, MPP algorithms rely on the basic Sensitivity Analysis (SA) theory, but considering a different goal [46]. Whereas SA gives the optimal solution in a local vicinity of the nominal UPs values, MPP provides the optimal solution behavior over the entire space of the Ups [50]. The MPP solution is based on partitioning the UPs space into subdomains called critical regions, where each critical region is uniquely associated to a different active set of constraints and yields a different set of Karush-Kuhn-Tucker (KKT) conditions than the other regions. The set of

equations derived from the KKT conditions are analytically solved, leading to the explicit construction of the optimal solution over the entire UPs space [41,46]. This solution is in the form of a number of P simple mathematical relations describing each of the optimal decision variables x^* and the objective Z^*as functions of the UPs θ, and is expressed as follows:

$$\left.\begin{array}{c} Z_p^* = \mathscr{F}_{0,p}(\theta) \\ x_{ip}^* = \mathscr{F}_{i,p}(\theta), \;\; i = 1,2,\ldots,K, \\ lb_{\theta_p} \le \theta \le ub_{\theta_p}, \;\; p = 1,2,\ldots P \end{array}\right\} \quad (2)$$

where the p^{th} relation is only valid over the p^{th} critical region of the UPs [11].

MPP offers outstanding capabilities [48], which include : *i*) its solution provides simple mathematical functions mapping the optimal decisions variables and objective(s) over the entire space of UPs, *ii*) once uncertainty is unveiled, optimal decisions can be readily and immediately calculated by these simple functions, avoiding the large computational cost associated to repetitive optimization procedure, *iii*) MPP is not only able to handle uncertainty related to the process conditions, but also to the optimization problem parameters (e.g., relative weights or importance of the different objectives). More information on the MPP theory and on the different MPP algorithms developed for problems of different natures (linear, quadratic, nonlinear, mixed-integer, convex, local, global, etc.) can be found in [48,50].

Despite the attractive characteristics of MPP, its successful application is conditioned by two main requirements: the first one is the deep and complex mathematical programming knowledge required for the development of such formulations, while the second is the need to a process model matching some very strict mathematical requirements [1,31,53]. These requirements hinder the smooth application of MPP to process operation optimization in practice, where complex, highly nonlinear and/or black-box models need to be considered (e.g., modular process simulators).

To tackle these limitations, the use of data-driven or Machine Learning (ML) techniques for the solution of MPP problems has recently emerged as a feasible alternative, where ML models are built to capture the mapping (embedded in the optimization problem) between the optimal decisions and the UPs. This research direction has been considered in [58], where kriging and Artificial Neural Network (ANN) models have been used to solve explicit Model Predictive Control (MPC) problems. Specifically, ML techniques are employed to build control laws that approximate the optimal control variables that must be applied to the process at future time step as a function of the current values of the state variables. The training data used to develop these ML-based control laws are generated by solving the MPC problem several times, considering different values of the initial state variables. Katz et al. [27] and Katz et al. [28] used multilayer ANNs with ReLU activation functions as surrogate models that approximate highly nonlinear objective and/or constraints, and, then, the MPP problem is traditionally solved on the basis of the surrogate models. Medina-González et al. [37] have used kriging models for approximating the multiparametric solution of the multi-objective optimization of a bio-based energy chain subjected to uncertainties. In more detail, they employed kriging models to approximate the optimal behavior of each decision variable as a function of the UPs that include electricity demands, environmental conditions and social dynamics. This set of decision variables was composed by the amount of energy that must be produced at each of the considered production plants along each of the considered successive time periods, besides the associated economic benefit of each plant. Luper, et al. [33] have proposed a similar approach for supporting reactive scheduling in a real batch plant for synthesis of polyols, which consists of several units working in parallel. The specific aim of their study was the development of a tool for the online adaptation of the optimal scheduling decisions to avoid overlaps between the emissions generated from the plant units due to frequent and uncertain changes in the units' starting times and task-unit assignment. Therefore, they utilized one kriging metamodel to approximate each optimal management decision as a function of UPs including equipment starting times and task-unit assignment. Shokry and Espuña [60] have employed kriging metamodels for the dynamic optimization of a batch process, which approximate the relations between the optimal time profiles of the batch control variables and UPs including kinetics properties and initial conditions of the batch. Shokry et al. [61] and Lupera et al. [34] have addressed the solution of mixed-integer optimization problems by using a combination of regression (kriging) and classification (ANN) techniques to approximate the optimal continuous and integer variables, respectively, as a function of the UPs. They have applied the methodology to a simple planning problem. Lupera et al. [35] have proceeded to apply the latter method to successfully solve mixed-integer reactive scheduling problem: in their work, they considered an energy production system which is composed of several Combined Heat and Power (CHP) units subjected to UPs variation, such as heat demand, electricity demand and units' startup and shutdown states. They have utilized ML models for regression and classification to approximate continuous and discrete optimal scheduling decisions as a function of UPs.

In these previous works, each of the developed ML-based method for the solution of MPP problems has been tailored and evaluated in the framework of a particular application of interest, involving an optimization problem of a specific nature, which was, in most of the cases, linear. Also, they have not addressed the optimization of the process and/or unit operations, where the benefits of such methodology, if successfully applied, would be significant, because the update of the optimal setpoints in response to UPs variations is typically required within very tight time slots (minutes or seconds), whereas, the models of the process are often complex, nonlinear and/or in the form of computationally expensive black-boxes.

This work contributes with the development of a novel data-driven methodology for the solution of general continuous MPP problems. The proposed methodology is based on the combination of *i*) Design Of Computer Experiment techniques (DOCE) techniques [22] to uniformly sample over the UPs domain, *ii*) state-of-art optimization tools to solve the problem with respect to each of the UPs combinations and *iii*) kriging metamodels which are trained using the generated input–output data (i.e., UPs values-optimal variables and objective values) to approximate the multiparametric behavior of the optimal solutions over the entire space of the UPs. The performance of the methodology is assessed by its application to five MPP benchmark examples of different nature (linear, bilinear, quadratic and nonlinear optimization problems) and to two case studies, which include the operation optimization of a utility plant modeled using a black-box modular process simulator and of a batch reactor.

In this sense, the specific novelties and characteristics of this proposal are:

i) the generality and applicability of the proposed methodology which, unlike the reviewed data-driven method for the solution of MPP problems, is aimed at solving different types of continuous optimization problems (i.e., linear, bilinear, quadratic, nonlinear, black-box) in a systematic way,
ii) the capacity of the proposed methodology to address process and unit operation optimization problems, where highly nonlinear and/or black-box models are typically used,
iii) a novel utilization of the kriging metamodel to approximate the optimal behavior of a system, unlike most of the chemical engineering literature, in which the kriging model is used to approximate the response of the actual system.

The rest of the work is organized as follows. Section 2 formulates the problem. Section 3 describes the steps of the proposed methodology and the involved techniques. Section 4 assesses the proposed approach

through its application to different benchmark continuous MPP problems and to two case studies of process operation optimization. Section 5 concludes the work and highlights possible future directions of research.

2. Problem statement

The proposed methodology is aimed at overcoming the difficulties of solving process operation optimization problems using classical/mathematical MPP formulations. Generally speaking, the problem (Eq. (3)) consists on finding the optimal values of the decisions variables $x \in R^K$ that maximize an objective function $Z(x,\theta)$ representing a performance index of the process. The problem is subjected to set of constraints $g_l(x,\theta), l = 1,..,L$, and is influenced by a set of bounded UPs $\theta \in R^k$ [4,47,50]. Notice that the objective Z and/or the constraints g_l could involve complex, nonlinear and/or black-box relations.

$$\left.\begin{array}{c} \min\limits_{x} Z(x,\theta) \\ S.T. \\ g_l(x,\theta) \le 0, \;\; l = 1,2,\ldots L \\ x \in R^K, \;\; \theta \in R^k \\ lb_x \le x \le ub_x, \;\; lb_\theta \le \theta \le ub_\theta \end{array}\right\} \quad (3)$$

In this work, we consider situations, such as process and unit operation optimization, in which classical MPP approaches cannot be applied due to the high nonlinearity and complexity of the process model, and consequently, the exact solutions in Eq. (2) cannot be attained. Alternatively, the proposed methodology develops ML models that act as accurate data-driven multiparametric relations, which are referred to as "MultiParametric Metamodels (MPM)", and expressed as follows:

$$\left.\begin{array}{c} \hat{Z}^* = f_0(\theta) \\ \hat{x}_i^* = f_i(\theta) \\ lb_{\theta_p} \le \theta \le ub_{\theta_p}, \;\; i = 1,2,\ldots,K \end{array}\right\} \quad (4)$$

where, $f_0, f_i, i = 1,2,..K$ are supervised ML regression models efficiently built to approximate the optimal objective function, $\hat{Z}^*$, and the decision variables, $\hat{x}_i^*$, as functions of the UPs, θ.

3. MPMs for continuous optimization under uncertainty

The steps of the proposed methodology are schematically illustrated in Fig. 1, and are detailed in the following subsections.

3.1. UPs sampling by DOCE techniques

In this step, the optimization problem and the involved FPM are explored in order to identify the influencing UPs, $\theta \in R^k$, and estimate their bounds $lb_\theta \le \theta \le ub_\theta$ (MPM input domain). Then, the goal becomes the selection of a proper set of combinations of the UPs values, $[\theta]_{n\times k}$, (i.e. the MPMs inputs) that uniformly cover the entire UPs space in order to collect information about the optimal solution behavior over all the subregions of the global domain.

Many DOCE techniques are available in the literature such as Latin hypercube sampling [16,17], low discrepancy sequences as Hammersley technique [22], space-filling designs and sequential or adaptive sampling [26]. Most of these DOCE techniques show both desired characteristics and limitation in terms of the uniformity of the generated sampling plan and of the required computational cost [17]. More details about different DOCE techniques can be found in [22]. In this work, a hybrid technique of Hammersley sequence and full factorial design is

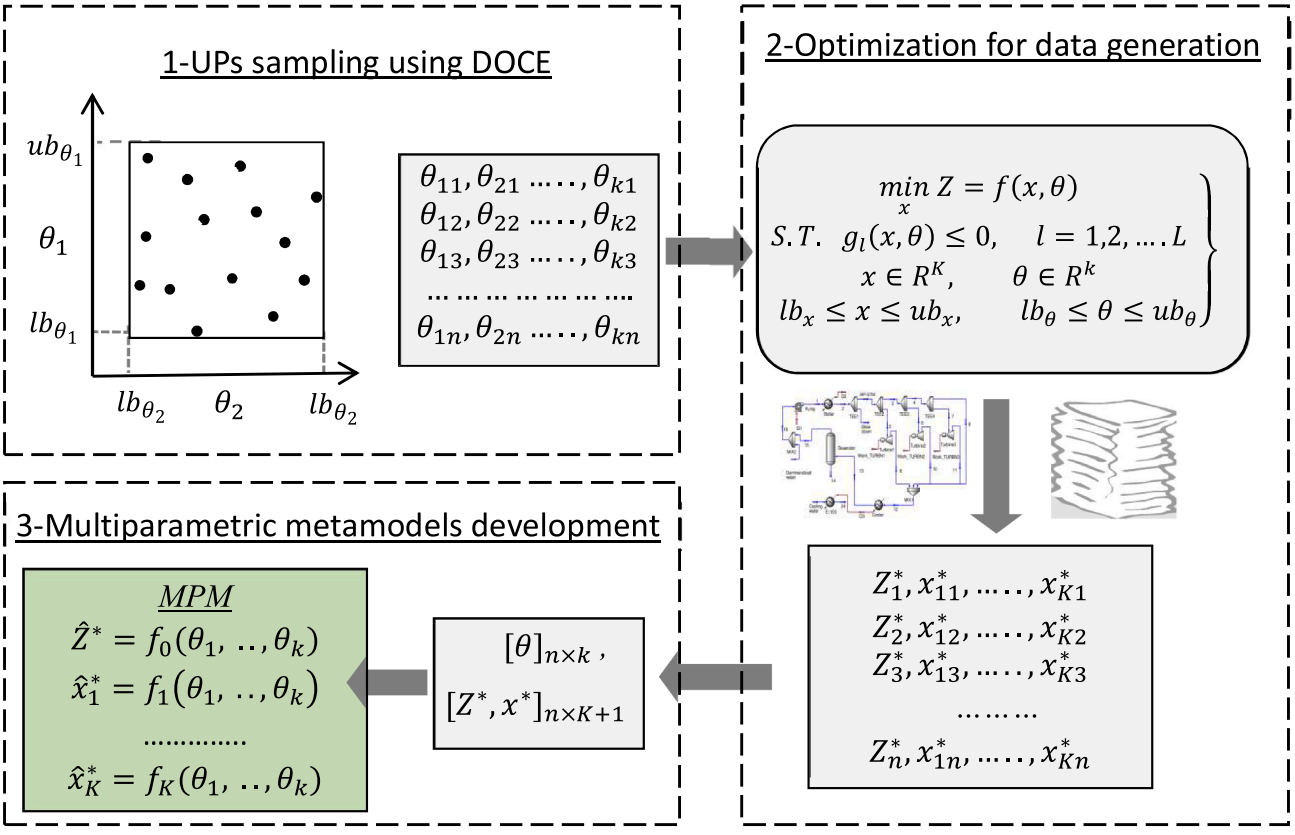

Fig. 1. Schematic representation of the proposed methodology.

used, as it achieves high uniformity with low computational cost [22]. The idea behind this hybrid technique is to employ the factorial design to compensate the limited ability of the Hammersley sequence to select sample points near the bounds and vertices of the input space, while at the same time, exploiting the high uniformity of the samples set of the Hammersley sequence over the bulk of the input space.

The number of required sample points, n, is proportional to the number of UPs, k, influencing the optimization problem and depends, also, on the complexity of the multiparametric behavior of the optimal solutions. On the other hand, as n increases, more computational effort is required for performing optimization runs and for the MPM training. So, the modeler should carefully balance this trade-off. The effect of the selection of the number of training patterns to be used, n, on the MPMs accuracy and on the computational effort required for their training and validation will be illustrated and quantified through the first application (Section 4.1).

3.2. Optimization for training data generation

Once a good sampling plan, $[\theta]_{n\times k}$, is obtained, the optimization problem must be solved $n$times, each time considering one of the generated UPs combinations, so as to obtain the matrix of the optimal values of the objective and decision variables $[Z^*, x^*]_{n\times K+1}$. State-of-art optimization algorithms can be used to solve the optimization problem depending on its characteristics [2,3]. Particularly, this work addresses different types of continuous optimization problems that include linear, quadratic and nonlinear ones, and employs the algorithms included in the Matlab optimization toolbox for their solutions. Specifically, for linear problems, the solver "*linprog*", based on a dual simplex algorithm, is used; for quadratic problems, the solver "*quadprog*", based on an interior-point-convex algorithm, is selected; and for nonlinear problems, the optimizer "*fmincon*", based on a sequential quadratic programming algorithm, is adopted. Default values for the optimization algorithms parameters (e.g., maximum number of function evaluations, tolerance on the constraint violation, termination tolerance on the first-order optimality, termination tolerance on decisions variables) are used in all cases.

3.3. MultiParametric metamodels development

The generated input ($[\theta]_{n\times k}$) and output ($[Z^*, x^*]_{n\times K+1}$) data are used to train a number of $K+1$ kriging-based MPMs, $f_0, f_1, \ldots, f_K$ (see Fig. 1), each of them is approximating the optimal behavior of each of the objective and decision variables as a function of all UPs.

For one optimal decision variable x^*, the kriging technique assumes the predictor $\widehat{x}^*(\theta) = \mu_{ok} + \mathbb{Z}(\theta)$, where the constant term μ_{ok} represents the main trend of the system to be approximated, and $\mathbb{Z}(\theta)$ is a deviation/residual from that trend, which accounts for detailed complex behavior of the system that could not be captured via the main trend μ_{ok} [25]. The residual $\mathbb{Z}(\theta)$ is modeled as a stochastic Gaussian process with expected value $E(\mathbb{Z}(\theta)) = 0$ and covariance between two residuals $cov(\mathbb{Z}(\theta_i), \mathbb{Z}(\theta_j))$ that only depends on the corresponding input values θ_i, θ_j. Thus, it can be calculated as: $cov(\mathbb{Z}(\theta_i), \mathbb{Z}(\theta_j)) = \sigma_{ok}^2 R(\theta_i, \theta_j)$, being σ_{ok}^2 the process variance and $R(\theta_i, \theta_j)$ a correlation function, $R(\theta_i, \theta_j) = exp\left(-\sum_{\ell=1}^{k} \xi_\ell \left|\theta_{i,\ell} - \theta_{j,\ell}\right|^{p_\ell}\right) + \delta_{ij}\lambda$, where ξ_ℓ are the model hyper-parameters, δ_{ij} is the Kronecker delta, p_ℓ are smoothing parameters and λ is a regularization constant that enables the kriging predictor to regress noisy data [15].

In order to estimate the values of these parameters $[\mu_{ok}, \sigma_{ok}^2, \xi_\ell, p_\ell, \lambda]$, the likelihood function of the observed data $[x^*]_{n\times 1}$ is maximized. The kriging predictor (Eq. (5)) and its estimated error (Eq. (6)) are obtained by deriving the augmented likelihood function of both the original training data set and a new interpolating point $(\theta_{new}, x^*_{new})$. In both equations, $[r]_{n\times 1}$ is the vector of correlations between the new point to be predicted θ_{new} and the original training data points, and calculated as $R(\theta_i, \theta_{new})$, and $[1]_{n\times 1}$ is the identity vector [4,25].

$$\widehat{x}^*(\theta_{new}) = \mu_{ok} + r^T R^{-1}(x^* - 1\mu_{ok}) \tag{5}$$

$$\widehat{s}^2(\theta_{new}) = \sigma_{ok}^2\left(1 + \lambda - r^T R^{-1} r + \left(1 - 1^T R^{-1} r\right)^{-1} \Big/ \left(1^T R^{-1} 1\right)\right) \tag{6}$$

In practice, the maximization of the concentrated log-likelihood function is computationally challenging because of the high effort associated to the repetitive calculation of the correlation matrix inverse $[R]_{n\times n}^{-1}$ during the optimization iterations, which quickly grows with the size of the training data set and/or the model input dimensionality. Besides, the nature of the concentrated log-likelihood function itself is quite complicated because it is flat near the optimum. More details about these computational challenges, and the numerical methods and optimization techniques to overcome or reduce these obstacles can be found in [13,16].

After the training of the MPMs, they are validated using a new and different dataset, $[\theta^v]_{n^v\times k}$ $[Z^{*,v}, x^{*,v}]_{n^v\times K}$, where n^v is the number of samples. The MPMs are used to estimate the values of the optimal decision variables and objective, $[\widehat{Z}^{*,v}, \widehat{x}^{*,v}]_{n^v\times K}$, which are compared to their exact counterparts $[Z^{*,v}, x^{*,v}]_{n^v\times K}$ in order to calculate an accuracy measure, such as the Normalized Room Mean Square Error (NRMSE) (Eq. (7)), for each of the $K+1$ MPMs.

$$NRMSE = 100^* \frac{RMSE}{\left(x^{*,v,max} - x^{*,v,min}\right)} \tag{7}$$

$$RMSE = \sqrt{\frac{1}{n^v}\sum_{i=1}^{n^v}\left(x^{*,v,i} - \widehat{x}^{*,v,i}\right)^2}$$

Notice that this work adopts the Kriging technique due to its widely reported advantages, especially its ability to capture highly nonlinear input–output mapping requiring a relatively low number of training data and its independence of any manual selection of specific structure or parameters. Nevertheless, any other efficient ML method can be used such as ANNs, Support Vector Regression (SVR), etc.

4. Benchmarks and applications

In this section, five benchmark examples with different characteristics selected from the MPP literature are used to assess the proposed methodology in terms of its accuracy of estimating the optimal solutions and its applicability to different types of continuous optimization problems, including linear, bilinear, quadratic and nonlinear cases. The methodology is, then, applied to two additional case studies involving the operation optimization of a utility plant and a batch reactor.

4.1. Linear optimization: Refinery blending problem

A refinery blending and production process [49,50] receives raw materials including two types of crude oils with flowrates x_1 and x_2 (bbl/day), which are processed in order to produce four types of products, namely, Gasoline, Kerosene, Fuel-oil and Residuals. It is required to select the optimal flowrates x_1 and x_2 that maximize the profit Z ($\$/day$). The optimization problem (Eq. (8)) is subjected to three constraints associated to the maximum allowable production rates of the Gasoline, Kerosene and Fuel-oil from each crude oil type, and is affected by two UPs, θ_1 and θ_2, which are the maximum allowable production rates (bbl/day) of the Gasoline and Kerosene, respectively:

$$\left.\begin{array}{c} MinZ = 8.1x_1 + 10.8x_2 \\ \underset{x}{S.T}: 0.80x_1 + 0.44x_2 \le 24000 + \theta_1 \\ 0.05x_1 + 0.10x_2 \le 2000 + \theta_2 \\ 0.10x_1 + 0.36x_2 \le 6000 \\ x_1 \ge 0,\ \ x_2 \ge 0 \\ 0 \ge \theta_1 \ge 6000,\ \ 0 \ge \theta_2 \ge 500 \end{array}\right\} \quad (8)$$

The methodology is applied following the steps described in Section 2. In this problem, different training sets of different sizes are used to build the MPMs to show the effect of the training set size on the MPMs accuracy and on the computational const of training and prediction. First, over the space of the UPs [0: 60000 *bbl/day*, 0: 500 *bbl/day*], five different sampling plans with different sizes, $([\theta_1,\theta_2]_{15\times2}$, $[\theta_1,\theta_2]_{30\times2}$, $[\theta_1,\theta_2]_{45\times2}$, $[\theta_1,\theta_2]_{60\times2}$, $[\theta_1,\theta_2]_{75\times2})$, are designed by means of the hybrid technique of Hammersley sequence and two-levels fractional factorial design (Fig. 2). For each of these sampling plans, the LP problem is solved to obtain the corresponding matrixes of the optimal objective and variables values, $[Z^*,x_1^*,x_2^*]_{15\times3},\ldots,\ldots,\ldots.[Z^*,x_1^*,x_2^*]_{75\times3}$. The optimization problem is solved using the "*linprog*" optimizer of the Matlab optimization toolbox, based on a dual simplex algorithm. Finally, using each of the five input–output training datasets (e.g., $[\theta_1,\theta_2]_{15\times2}-[Z^*,x_1^*,x_2^*]_{15\times3}$), three metamodels are fitted to approximate the optimal objective and decision variables, $\widehat{Z}^* = f_0(\theta_1,\theta_2), \widehat{x}_1^* = f_1(\theta_1,\theta_2), \widehat{x}_2^* = f_2(\theta_1,\theta_2)$.

One validation dataset $\left[\theta_1^v,\theta_2^v\right]_{400\times2}-[Z^{*,v},x_1^{*,v},x_2^{*,v}]_{400\times3}$ is generated and used to assess the performances of the five sets of MPMs, f_0,f_1,f_2, trained with the five different datasets. It is worth highlighting that the validation set is in the form of a uniform grid of 20 × 20 over the UPs space, so as to achieve a credible assessment of the MPMs predictions in all the local regions of the UPs. The NRMSE of the MPMs prediction is calculated by comparing their estimated outputs $\left[\widehat{Z}^{*,v},\widehat{x}_1^{*,v},\widehat{x}_2^{*,v}\right]_{400\times3}$ with the exact ones $[Z^{*,v},x_1^{*,v},x_2^{*,v}]_{400\times3}$ provided by the rigorous optimization itself.

Fig. 3 shows that the NRMSE of the MPMs decreases when the training dataset size increases, while the required computational efforts for the MPMs training and for prediction (of the validation set) increase. The Figure also indicates that even with few (i.e., 15) training patterns, the MPMs are able to achieve a satisfactory accuracy (less than 3 % of NRMSE). Table 1 illustrates the NRMSE and computational efforts required for the MPMs training and validation when the largest training dataset (75 patterns) is considered.

Fig. 3 and Table 1 shows *i)* the high accuracy of the MPMs trained by the dataset containing 75 instances (NRMSE less than 0.5 % of all the MPMs), *ii)* affordable "offline" computational effort required for training the MPMs (2.79 s = 0.86 + 0.80 + 1.13) and *iii)* very low computational effort required to estimate the optimal solutions of the validation dataset (0.00127 s = 0.51/400), saving 75.7 % ((2.1-0.51)/2.1) of the computational effort required for the "wait and see" optimization procedure. It is also worth to emphasize that only one multiparametric relation (i.e., a MPM) is able to describe the optimal solution behavior over the entire space of the UPs, which makes its online usage to react to the UPs variation easier and more flexible than using several multiparametric relations. In the literature [50], the conventional MPP approach provides a solution for the same problem that divides the UPs space into two critical regions and, consequently, two sets of mathematical parametric functions are obtained each of them is valid only for one of the two partitions of the UPs space.

As an additional assessment of the performance of the developed MPMs, the deterministic multiparametric solution provided in [50] is used to calculate the optimal objective and decision variables values for the validation dataset and compared with the one obtained by the MPMs in Fig. 4.

4.2. Bilinear optimization

In this mathematical example (Eq. (9)) [23], a bilinear objective function is to be minimized, subjected to two linear constraints which are affected by one uncertain parameter:

$$\left.\begin{array}{c} MinZ = x_1x_2 \\ \underset{x}{S.T.}: 2x_1 + x_2 \ge \theta \\ x_1 + 3x_2 \ge 0.5\theta \\ -1 \le x_1,x_2 \le 1,\ \ 0 \le \theta \le 1 \end{array}\right\} \quad (9)$$

The proposed method is applied starting by designing a sampling plan $[\theta]_{60\times1}$ over the space [0: 1] of the UP (i.e., the MPMs input). The optimization problem is solved 60 times considering the values of the UP in the sampling plan, to obtain the corresponding optimal decision variables and objective values (i.e., the MPMs outputs) $[Z^*,x_1^*,x_2^*]_{60\times3}$. The "*fmincon*" optimizer of the Matlab optimzation toolbox, based on a sequential quadratic programming algorithm, is used. This input–output training dataset is used to build three MPMs, one for each of the optimal objective and decision variables ($\widehat{Z}^* = f_0(\theta), \widehat{x}_1^* = f_1(\theta), \widehat{x}_2^* = f_2(\theta)$). A different validation dataset, including a uniform grid of 150 samples, is generated $([\theta^v]_{150\times1}-[Z^{*,v},x_1^{*,v},x_2^{*,v}]_{150\times3})$ and the trained MPMs are used to estimate the optimal solutions $[\widehat{x}_1^{*,v},\widehat{x}_2^{*,v},\widehat{Z}^{*,v}]_{150*3}$.

Table 1 reports the results obtained by the proposed methodology, which indicate the very high accuracy of the three MPMs (NRMSE of 0.01 %, 0.8 % and 0.4 %) and, also, the significant reduction (98.9%) in the required computational effort with respect to the use of standard optimization tools. Also, the deterministic MPP solution provided in [23] is used to calculate the optimal objective and variables of the validation dataset, and the extend of matching between both solutions is

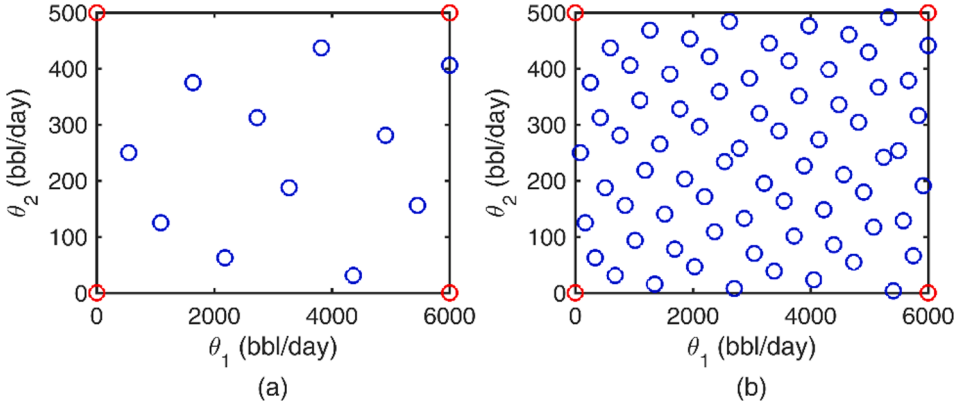

Fig. 2. Two of the five sampling plans of the UPs ((a) $n = 15$ and (b) $n = 75$): Blue circles indicate UPs combinations generated by the Hammersley techniques, whereas red circles refer to those generated by the two-levels full factorial design.

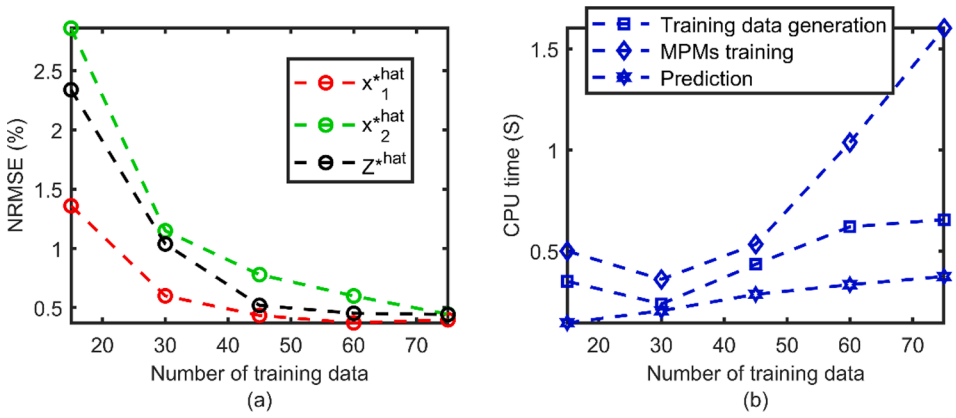

Fig. 3. (a) NRMSE and (b) computational effort required for training data generation, MPMs training and prediction times as a function of the training dataset size.

Table 1
NRMSE (%) of the MPMs and computational effort for training and validation.

Problem	*MPMs*	*Training*				*Validation*			*NRMSE (%)*
		No. of training samples	*CPU Time (s)**			*No. of validation samples*	*CPU Time (s)**		
			Ups Sampling	*Optimization for data generation*	*Fitting*		*Optimization for data generation*	*Prediction*	
Refinery blending problem	$\widehat{Z}^*$	75	0.002	0.53	0.86	400	2.10	0.51	0.44
	$\widehat{x}_1^*$				0.80				0.39
	$\widehat{x}_2^*$				1.13				0.44
Bilinear objective with linear constraints	$\widehat{Z}^*$	60	0.012	6.9	0.53	150	16.20	0.17	0.01
	$\widehat{x}_1^*$				1.08				0.80
	$\widehat{x}_2^*$				1.56				0.40
Quadratic optimization	$\widehat{Z}^*$	80	0.015	0.815	5.76	400	4.51	1.49	0.00011
	$\widehat{x}_1^*$				2.52				0.02813
	$\widehat{x}_2^*$				4.46				0.01853
Quadratic optimization: milk surplus problem	$\widehat{Z}^*$	80	0.023	1.07	1.47	400	8.17	1.59	0.00350
	$\widehat{x}_1^*$				1.52				0.00020
	$\widehat{x}_2^*$				1.42				0.00016
	$\widehat{x}_1^*$				1.39				0.00021
	$\widehat{x}_2^*$				155				0.00010
Nonlinear optimization	$\widehat{Z}^*$	140	0.039	4.67	4.60	400	13.85	1.60	0.62
	$\widehat{x}_1^*$				4.00				1.5
	$\widehat{x}_2^*$				6.00				1.9
Operational cost minimization of utility plant	$\widehat{C}^*$	70	0.015	48.4	1.20	400	300.80	0.20	0.31
	$\widehat{S}_F^*$				1.40				0.61
	$\widehat{S}_T^*$				1.07				0.77
Operational cost minimization of batch reactor	$\widehat{C}_B^*$	150	0.038	861	14.0	300	4887.00	3.88	0.08
	$\widehat{V}_0{}^*$				17.5				1.50
	$\widehat{F}^*$				0.66				*NA*
	$\widehat{T}^*$				11.1				1.45

*Intel core i5-6200U CPU@2.3 GHz.

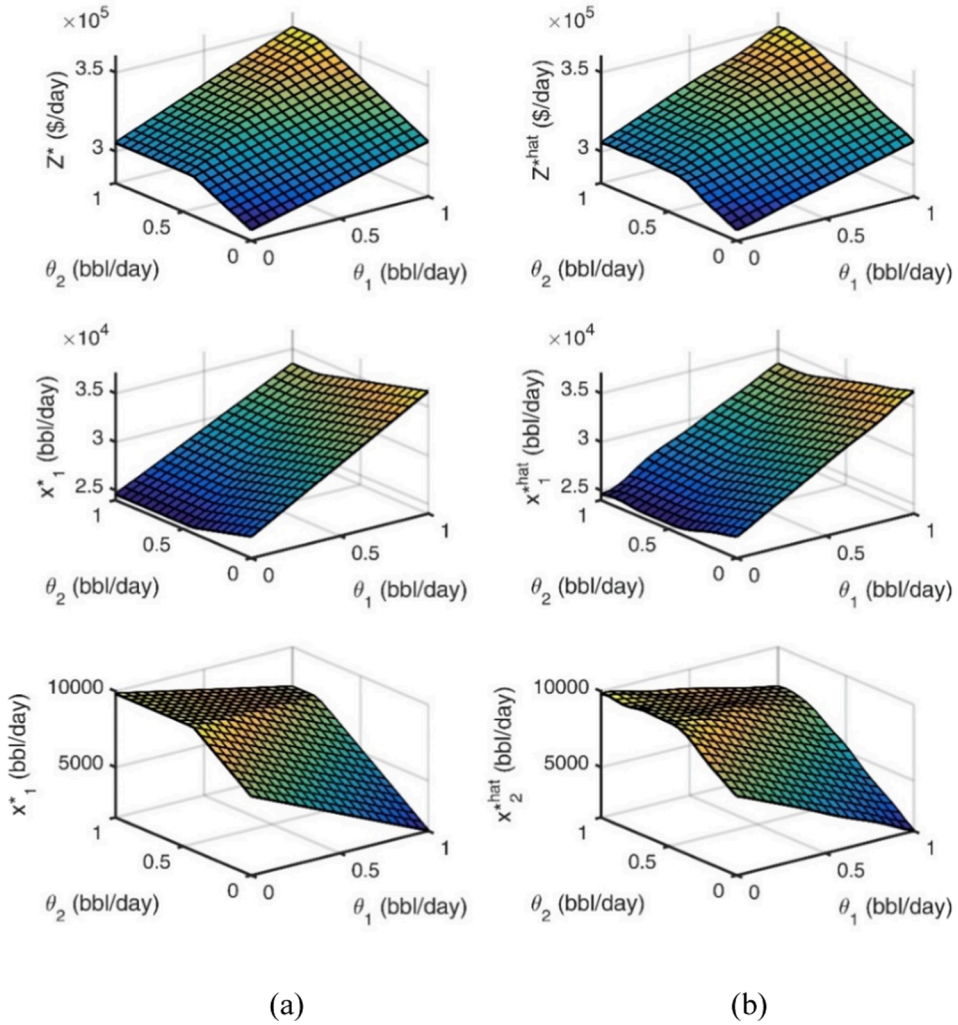

Fig. 4. Refinery blending problem: comparison between (a) the results obtained by the classical MPP solution provided in [50] and (b) the results provided by the proposed methodology.

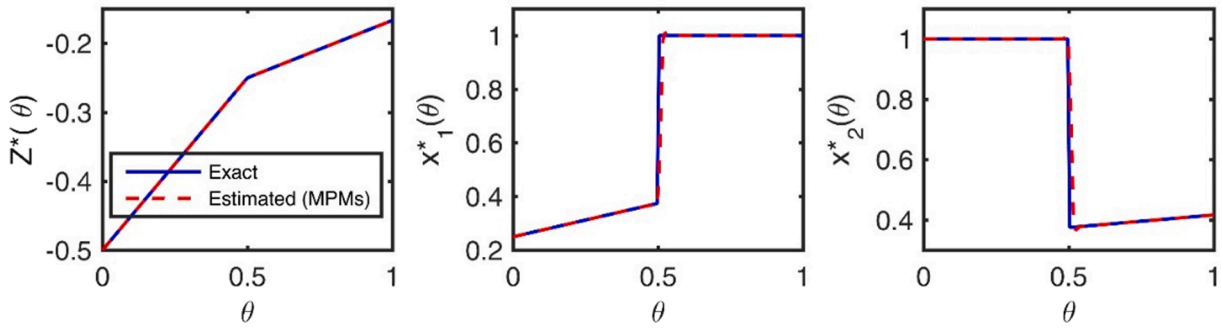

Fig. 5. Bilinear optimization problem: comparison between the solutions obtained by the classical MPP provided in [23] (blue solid lines) and the solutions provided by the proposed methodology (red dashed lines).

shown in Fig. 5.

4.3. Quadratic optimization

The second application [10] involves the minimization of a quadratic objective function subjected to a set of six constraints, which are affected by two UPs in their right-hand side (Eq. (10)).

$$\left.\begin{array}{l} \underset{x}{Min} Z = c^T[x_1,x_2]^T + 0.5[x_1,x_2]Q[x_1,x_2]^T \\ S.T.: A[x_1,x_2]^T \le b + F[\theta_1,\theta_2]^T \\ -1 \le x_1,x_2 \le 1,\ \ 0 \le \theta_1,\theta_2 \le 1 \end{array}\right\} \tag{10}$$

$$c = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, Q\begin{bmatrix} 0.0196 & 0.0063 \\ 0.0063 & 0.0199 \end{bmatrix}$$

$$b = \begin{bmatrix} 0.417425 \\ 3.582575 \\ 0.413225 \\ 0.467075 \\ 1.090200 \\ 2.909800 \end{bmatrix},\ A = \begin{bmatrix} 1 & 0 \\ -1 & 0 \\ -0.0609 & 0 \\ -0.0064 & 0 \\ 0 & 1 \\ 0 & -1 \end{bmatrix},\ F = \begin{bmatrix} 3.16515 & 3.7546 \\ -3.16515 & -3.7546 \\ 0.17355 & -0.2717 \\ 0.06585 & 0.4714 \\ 1.81960 & -3.2841 \\ -1.81960 & 3.2841 \end{bmatrix}$$

The same procedure is considered. A sampling plan $[\theta_1, \theta_2]_{80\times2}$ is designed over the UPs space. The problem is solved 80 times to obtain the matrix $[Z^*, x_1^*, x_2^*]_{80\times3}$, using the "*quadprog*" optimizer of the Matlab optimization toolbox based on an interior-point-convex algorithm. Using this input–output dataset, three MPMs are fitted: $\widehat{Z}^* = f_0(\theta_1, \theta_2)$, $\widehat{x}_1^* = f_1(\theta_1, \theta_2)$, $\widehat{x}_2^* = f_2(\theta_1, \theta_2)$. The validation is accomplished using a different dataset including 400 samples $([\theta_1^v, \theta_2^v]_{400\times2} - [Z^{*,v}, x_1^{*,v}, x_2^{*,v}]_{400\times3})$, where the MPMs are used to estimate the outputs $[\widehat{x}_1^{*,v}, \widehat{x}_2^{*,v}, \widehat{Z}^{*,v}]_{400*3}$ and the NRMSE of the MPMs predictions is calculated (Table 1). In this case, the accuracy of the MPMs (NRMSE of 0.0001 %, 0.028 % and 0.018 %) is significantly higher than that of the MPMs in the previous two examples, which can be explained by the smooth and continuous multiparametric behavior of the optimal solutions over the entire UPs space (Fig. 6), which is relatively easy to capture by the data-driven models. On the contrary, the multiparametric behavior of the optimal solution in the previous two cases shows discrete features, which represent a challenge for the data-driven models.

The MPMs were also able to save a considerable percentage (67 %) of the computational effort required to calculate the optimal solutions through real optimization, but the saving percentage is not as high as in the previous two cases. This is, again, due to the relative simplicity of the optimization problem solution (i.e., a quadratic objective function subjected to linear constraints).

The deterministic multiparametric solution obtained by [10] for the same problem, divides the UPs space into four critical regions, and a different set of mathematical multiparametric functions are used to calculate the optimal solutions over each region.

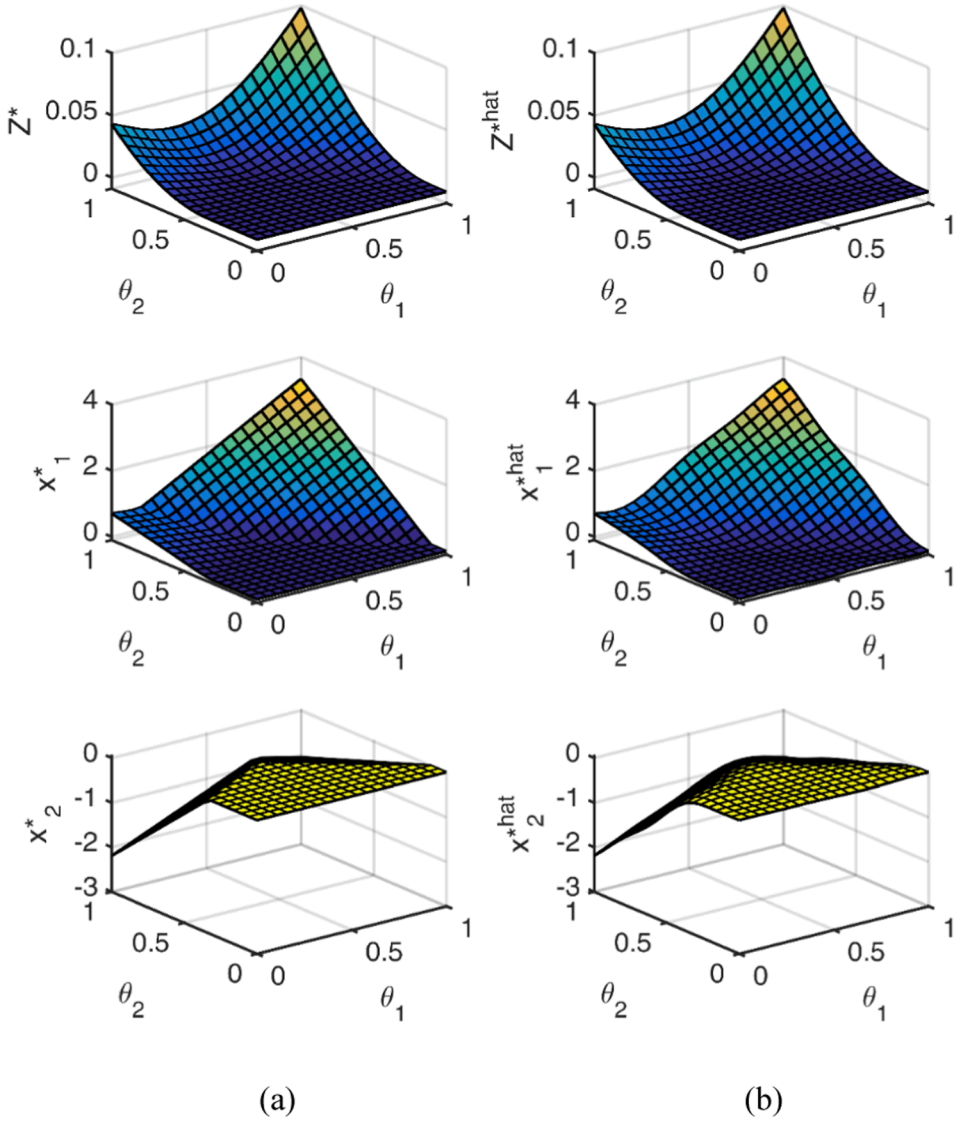

Fig. 6. Quadratic optimization problem: comparison between (a) the results obtained by the classical MPP solution provided in [10] and (b) the results provided by the proposed methodology.

4.4. Quadratic optimization: Milk surplus problem

This application considers a Dutch agriculture cooperative company that produces four products including milk for direct consumption, butter, fat and cheese with prices x_1, x_2, x_3 and x_4, respectively [50]. The consumer demand from each product is modelled as an inverse function of the product price. The cooperative company must decide the optimal prices (that indirectly set the optimal quantities of products to be delivered) that maximize the profit, assuming this will not affect the production costs. The optimization problem is subjected to production capacity constraints and an escalation of the price constraint, and is influenced by four UPs, $\theta_1, \theta_2, \theta_3$ and θ_4, related to the consumer demand, and another UP θ_5, associated to the escalation of the prices. Notice that, in this optimization problem, the UPs are affecting both of the constraints and the objective function, as follows:

$$\min_x Z = -1.2338x_1^2 - 0.0203x_2^2 - 0.0136x_3^2 - 0.0027x_4^2 + 0.0031x_3x_4 + 2139x_1 + 135x_2 + 103x_3 + 19x_4 + x_1\theta_1 + x_2\theta_2 + x_3\theta_3 + x_4\theta_4$$

$S.T.$:

$$-0.0321x_1 - 0.0162x_2 - 0.0038x_3 - 0.0002x_4 \le -80.5 - 0.026\theta_1 - 0.800\theta_2 - 0.306\theta_3 - 0.245\theta_4,$$

$$-0.1061x_1 - 0.0004x_2 - 0.0034x_3 - 0.0006x_4 \le 26.6 - 0.086\theta_1 - 0.020\theta_2 - 0.297\theta_3 - 0.371\theta_4,$$

$$1.2334x_1 \le 2139 + \theta_1,$$

$$0.0203x_2 \le 135 + \theta_2,$$

$$0.0136x_3 - 0.0015x_4 \le 103 + \theta_3,$$

$$-0.0016x_3 + 0.0027x_4 \le 19 + \theta_4,$$

$$0.0163x_1 + 0.0003x_2 + 0.0006x_3 + 0.0002x_4 \le 10 + \theta_5,$$

$$-150 \le \theta_1 \le 150,$$

$$-5 \le \theta_2 \le 5,$$

$$-6 \le \theta_3 \le 6,$$

$$-2 \le \theta_4 \le 2,$$

$$-1 \le \theta_5 \le 1$$

The methodology is, again, applied following the same procedure. A sampling plan, $[\theta_1, \theta_2, \theta_3, \theta_4, \theta_5]_{80\times5}$, is created including different combinations of the UPs values, which are selected by the hybrid DOCE method within the known bounds [−150:150, −5:5, −6:6, −2:2, −1:1]. The quadratic optimization problem is solved 80 times using the "*quadprog*" optimizer of the Matlab optimization toolbox, based on the interior-point-convex algorithm, to yield the output data $[Z^*, x_1^*, x_2^*, x_3^*, x_4^*]_{80\times5}$. Five MPMs, $\widehat{Z}^* = f_0(\theta_1, \theta_2, \theta_3, \theta_4, \theta_5)$, $\widehat{x}_i^* = f_i(\theta_1, \theta_2, \theta_3, \theta_4, \theta_5)$, $i = 1, \ldots 4$, are fitted to approximate the optimal profit and prices as a function of the demand and the price escalation uncertainties.

The MPMs performances are assessed relying on a new dataset, $([\theta_1^v, \theta_2^v, \theta_3^v, \theta_4^v, \theta_5^v]_{400\times5} - [Z^{*,v}, x_1^{*,v}, x_2^{*,v}, x_3^{*,v}, x_4^{*,v}]_{400\times5})$, and their accuracies, in terms of NRMSE, are reported in Table 1. As in the third case (Section 4.3), a significantly high prediction accuracy of the MPMs is obtained (a maximum NRMSE of 0.0035 %), which can be justified by similar reasons. Notice that despite the high dimensionality of the optimization problem (four decision variables) and the high number of the UPs (five) with respect to the other examples, the size of the training dataset required to achieve such high accuracy is on the same order of magnitude (80 points).

Classical MPP approaches [50] provide a deterministic solution to this problem that is characterized by two critical regions over the UPs

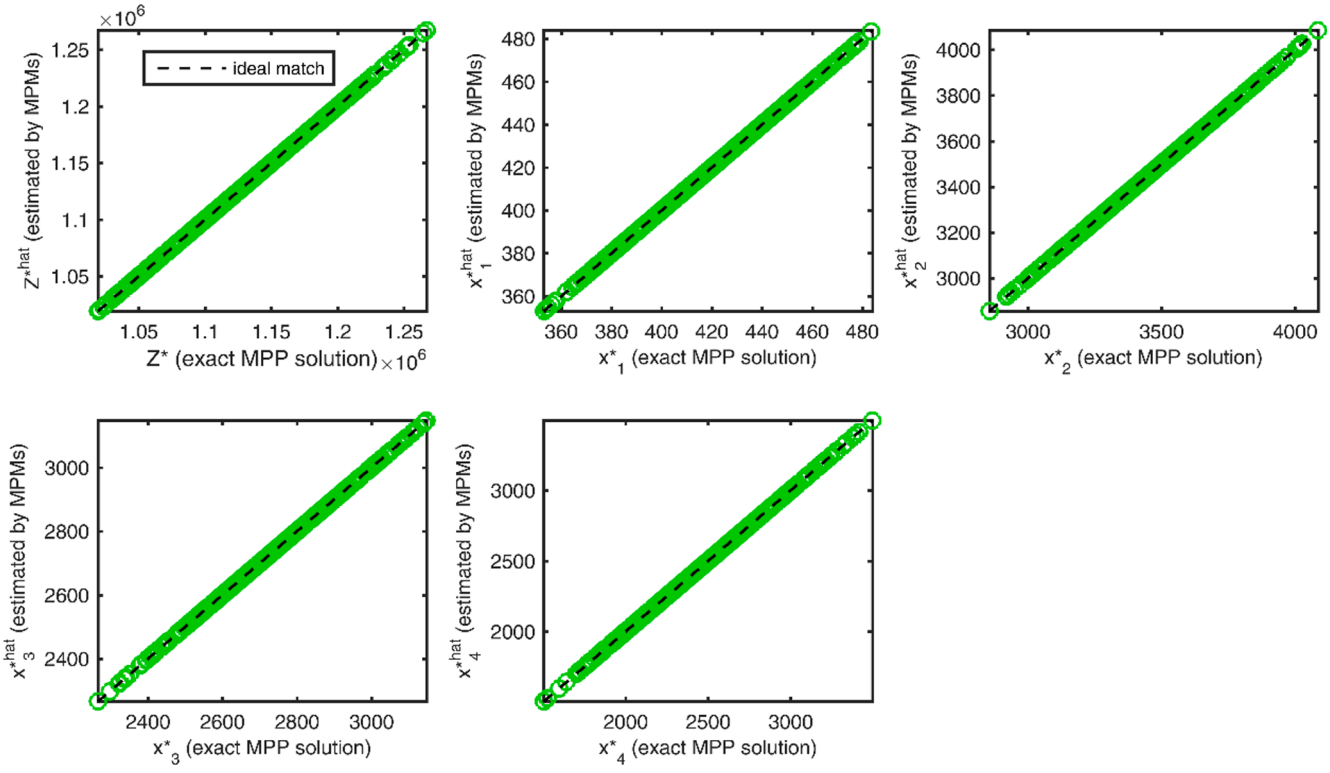

Fig. 7. Milk surplus problem: comparison between the solutions obtained by the classical MPP provided in [50] and the solutions provided by the proposed methodology.

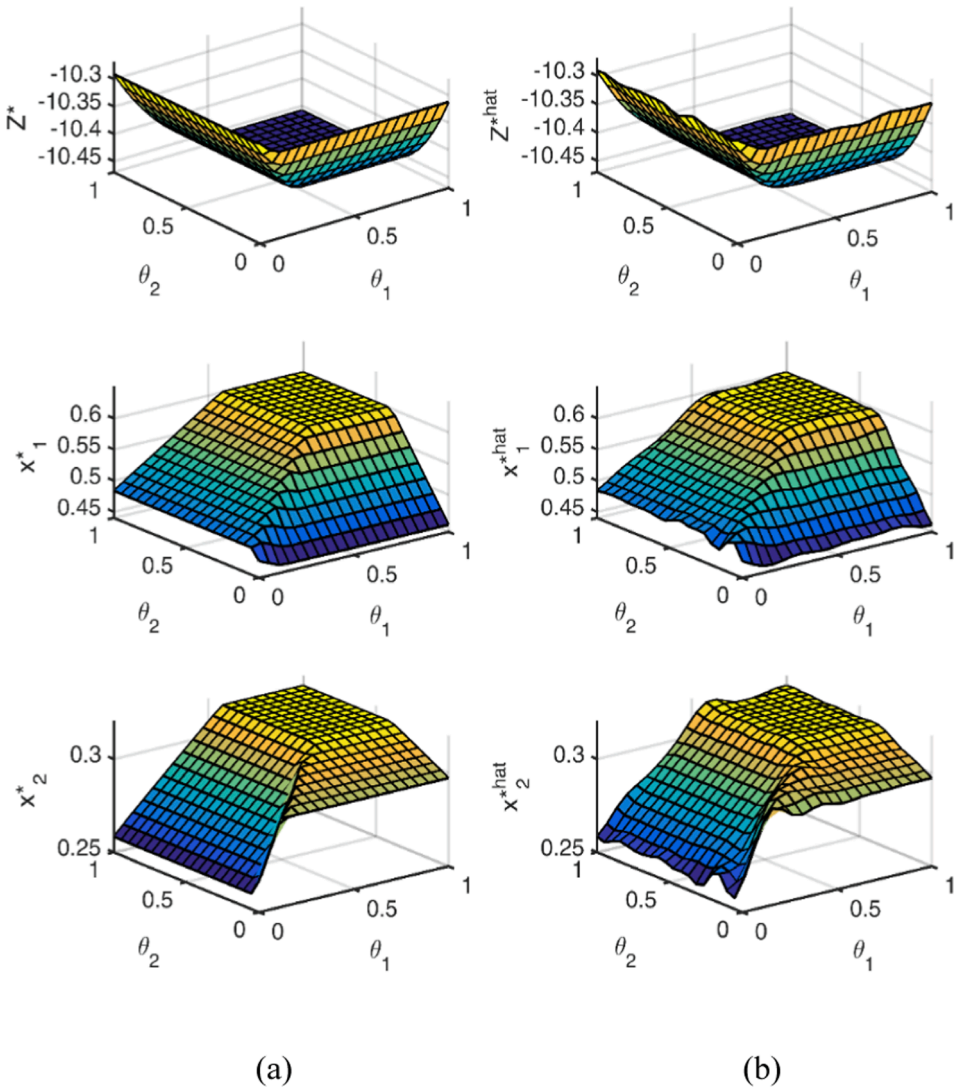

Fig. 8. Nonlinear optimization problem: comparison between (a) the results obtained by the classical MPP solution provided in [9] and (b) the results provided by the proposed methodology.

space. Fig. 7 shows a comparison between the deterministic solution and the approximate one provided by the proposed methodology, using the validation dataset.

4.5. Nonlinear optimization

The last multiparametric optimization benchmark example [9] includes a nonlinear objective subjected to two constraints, each one involving an uncertain parameter in the right-hand side (Eq. (11)).

$$\left.\begin{array}{c} \underset{x}{MinZ} = x_1^3 + 2x_1^2 - 5x_1 + x_2^2 - 3x_2 - 6 \\ S.T: \\ 2.0x_1 + x_2 \le 2.5 + \theta_1 \\ 0.5x_1 + x_2 \le 1.5 + \theta_2 \\ 0 \ge \theta_1 \ge 1, \;\; 0 \ge \theta_2 \ge 1 \end{array}\right\} \tag{11}$$

Three MPMs, $\widehat{Z}^* = f_0(\theta_1, \theta_2), \widehat{x}_1^* = f_1(\theta_1, \theta_2)$ and $\widehat{x}_2^* = f_2(\theta_1, \theta_2)$, are fitted using the input–output training dataset, $[\theta_1, \theta_2]_{140\times 2} - [Z^*, x_1^*, x_2^*]_{140\times 3}$, generated as explained earlier for the other examples. The performance of the MPMs is evaluated using a new validation dataset, $\left[\theta_1^v, \theta_2^v\right]_{400\times 2} - [Z^{*,v}, x_1^{*,v}, x_2^{*,v}]_{400\times 3}$, and the results are shown in Table 1. Fig. 8 compares the approximated multiparametric solution obtained by the MPMs to the deterministic one obtain by [9,50]. In [9], different classical MPP algorithms have been applied to the problem, and the best one was the quadratic approximation algorithm that partitioned the UPs into four critical regions.

In this application, the size of the training dataset required to fit the MPMs is relatively high and, although the resulted accuracy is good (NRMSE of 0.62 %, 1.5 % and 1.9 %), it is not as high as in the previous examples, where the NRMSE is less than 1 % in the worst cases. Again, this is because of the challenging discrete or piecewise characteristics of the multiparametric solution, which can be clearly noticed in Fig. 8.

4.6. Case Study 1: Optimization of a utility system operation

A utility system (Fig. 9) supplying mechanical energy to an industrial process is considered. The system is composed of a boiler (*E-1*) that receives water and supplies high pressure steam to a steam turbine (*T1*), whose outlet steam is condensed to water that is fed-back to the boiler inlet by a pump (*P-1*).

The objective is to minimize the system operational cost, *C*, which includes the costs of the energies consumed by the boiler, *T1*, and the pumps, *E-1* and *E-2*, (*Q2*, *Q1* and *Q5* respectively), and the cooling water cost. The operational cost is modeled as a function of the boiler outlet

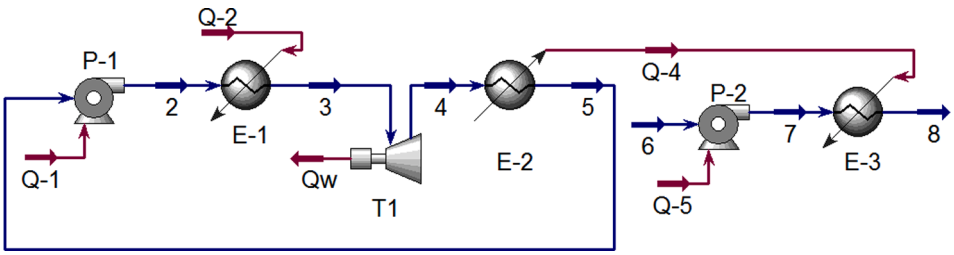

Fig. 9. Utility system model.

steam flowrate (S_F) and temperature (S_T). However, two uncertain parameters, θ_1 and θ_2, affect the system, which are: *i*) the power demand that must be satisfied by the turbine (Qw) and varies in the range [53000, 57000] kW, and *ii*) the turbine efficiency that varies in the range [75, 95] %.

The system is simple to describe, but complex thermodynamic relations have to be considered, so the natural way to model the system consists on using a sequential-modular simulator (in this case, ASPEN HYSYS [63] modeling and simulation environment has been chosen, see Fig. 9). Then, a black box (simulation-based) optimization procedure is the straight-forward optimization alternative. In any case, due to this complex thermodynamic/mathematical behavior, optimization under uncertainty will be hard to be managed through conventional multi-parametric mathematical programming approaches.

The problem is formulated as:

$$\left.\begin{array}{c} \underset{S_F,S_t}{MinC} = f(S_F, S_t, \theta_1, \theta_2) \\ S.T: \\ \text{the process model} \\ Qw(S_F, S_t, \theta_2) \le \theta_1 \\ 36000 < S_F < 79200\ kgMole/hr,\ \ 162 < S_t < 360C^o \end{array}\right\} \quad (12)$$

A sampling plan$[\theta_1, \theta_2]_{70\times2}$, is designed over the domain [53000:57000, 75:95]. The black-box simulation-based optimization problem (Eq. (12)) is solved 70 times (the "*Fmincon*" Matlab optimizer is used) to obtain the optimal values of the objective and decision variables$[C^*, S_F^*, S_T^*]_{70\times3}$. Using this dataset, three MPMs are trained, which approximate the optimal behavior of the operational cost, steam flowrate and temperature as a function of the power demand and turbine efficiency: $\widehat{C}^* = f_0(\theta_1, \theta_2), \widehat{S}_F^* = f_1(\theta_1, \theta_2), \widehat{S}_T^* = f_2(\theta_1, \theta_2)$. The validation is performed using another dataset including 400 samples, $[\theta_1^v, \theta_2^v]_{400\times2} - [C^{*,v}, S_F^{*,v}, S_T^{*,v}]_{400\times3}$, where the three MPMs are employed to

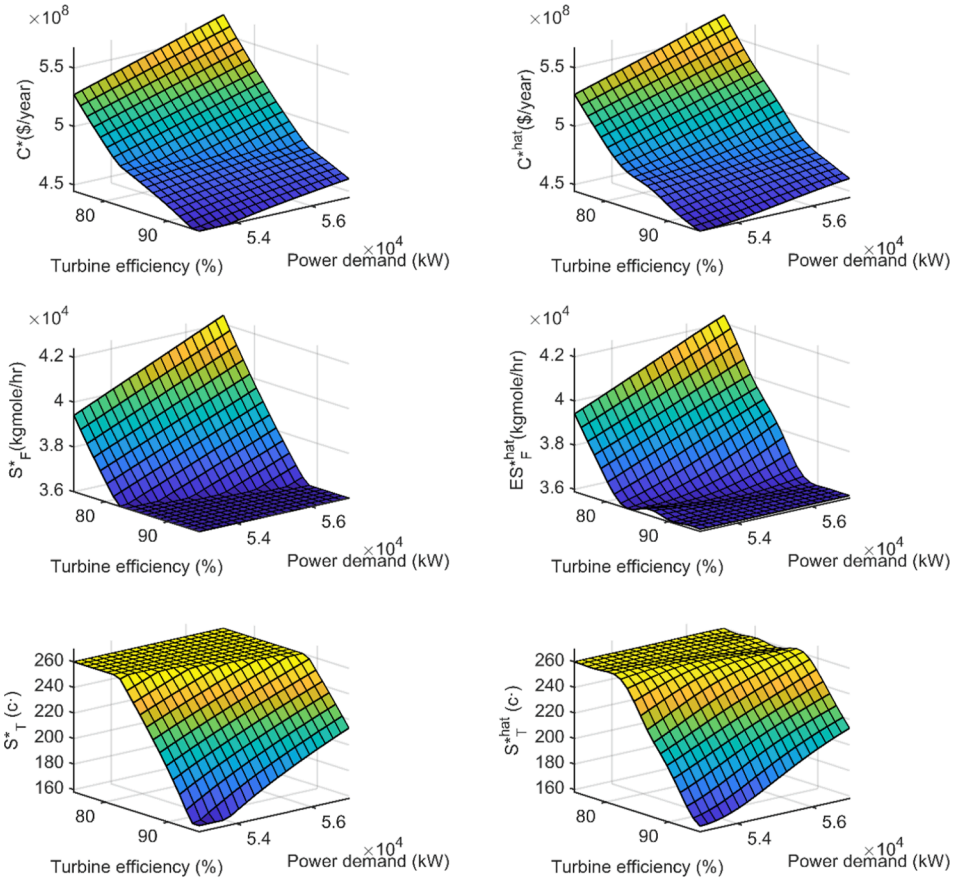

Fig. 10. (a) Exact versus (b) approximate multiparametric behavior of the utility system.

predict the outputs $[\widehat{C}^{*,v}, \widehat{S}_F^{*,v}, \widehat{S}_T^{*,v}]_{400\times3}$ and the NRMSE is calculated (Table 1).

Table 1 and Fig. 10 show the high potential of the method: the optimal decisions are accurately predicted via simple interpolations using the MPMs, in a very short time (0.2 s, using a regular personal computer), saving the significant computational effort required by the simulation-based optimization (300.8 s). Thus, the method represents a powerful tool to promptly manage the UPs variations during the process online operations.

4.7. Case Study 2: Optimization of a batch reactor operation

The second case study [19] considers a hypothetical scenario in which an engineer is charged with starting up a new chemical process based on the reactions:

$$A\rightarrow B, \quad r_1 = k_1 C_A, \quad k_1 = 0.05\ s^{-1}$$

$$B\rightarrow C, \quad r_2 = k_2 C_B, \quad k_2 = \theta_1 k_1\ s^{-1}$$

$$A\rightarrow C, \quad r_3 = k_3 C_A, \quad k_3 = \theta_2 k_1\ s^{-1}$$

where A is the reactant, B is the desired product, C is a secondary undesired product, $r_i, i = 1,2,3$ are the respective reaction rates, $k_i, i = 1,2,3$ are the reaction rate constants, C_A is the reactant concentration and C_B is the desired product concentration. It has been assumed that the reaction constants of the two side-reactions, k_2, k_3, are not precisely estimated, so they will be represented by two UPs θ_1 and θ_2.

The process is to be run in a batch reactor with maximum capacity $V_{max} = 1000\ L$, on which an automatic feeding and emptying system is installed, with flowrate $0 \le F \le F_{max} = 1\ L/s$. The process can be operated in two scenarios:

- A complete batch mode, where the process starts with the reactor full of reactants, i.e., the initial volume equals to the reactor maximum capacity: $V_0 = V_{max} = 1000\ L$.
- A hybrid mode of fed-batch then batch, according to which, the process starts with initial volume, V_0, of the reactant less than the maximum capacity, i.e., $V_0 < V_{max}$. Then, the rest of the reactant volume ($V_{max} - V_0$) is continuously fed into the batch with constant flowrate, F, until the time τ at which the reactor is full ($V(\tau) = V_{max}$); then the flow is shut-off and the process continues in a batch mode.

Given that the duration of the batch is T, the time $\tau = (V_{max} - V_0)/F$ (i.e., for a full batch mode scenario $\tau = 0$) and the feed concentration $C_{A0} = 2M$, hence the process can be modeled as:

for the time $0 \le t < \tau$

$$\frac{dC_A^{'}}{dt} = C_{A0}\ F - (1+\alpha_2)\ k_1 C_A^{'}, \quad C_A^{'}(0) = C_{A0} V_0$$

$$\frac{dC_B^{'}}{dt} = k_1 C_A^{'} - \alpha_1\ k_1 C_B^{'}, \quad C_B^{'}(0) = 0$$

$$\frac{dV^{'}}{dt} = F, \quad V^{'}(0) = V_0$$

for time $\tau \le t \le T$

$$\frac{dC_A}{dt} = -(1+\alpha_2) k_1 C_A, \quad C_A(\tau) = C_A^{'}(\tau)$$

$$\frac{dC_B}{dt} = k_1 C_A - \alpha_1 k_1 C_B, \quad C_B(\tau) = C_B^{'}(\tau)$$

$$\frac{dV}{dt} = 0, \quad V(\tau) = V^{'}(\tau)$$

where the dash superscript is used to distinguish the process variables during the fed-batch period. The process total time, T_P, is the summation of the time required to fill the reactor with the initial volume of reactant, V_0, plus the batch duration, T, plus the time to empty the batch, i.e., $T_P = \frac{V_0}{F_{max}} + T + \frac{V_{max}}{F_{max}}$. The objective of the batch operation is to select the optimal values of the decision variables V_0, F and T that maximize the amount of the desired product B produced per unit of time, considering bounded uncertain parameters θ_1 and θ_2.

$$\left.\begin{array}{c} \underset{V_0,T,F}{\text{Max}}\ C_B(V_0, \mathrm{F}, T)/T_P \\ S.T.: \\ the\ \ process\ \ model \\ 0 \le F \le F_{max} \\ 0 \le V_0 \le max \\ 0 \le \tau \le T \\ 0 \le \theta_1, \quad \theta_2 \le 5 \end{array}\right\} \tag{13}$$

Before the application of the proposed method, the correctness of the optimization procedure is checked by solving the problem considering the nominal values of the UPs $\theta_1 = 0.05$ and $\theta_2 = 0$, as in [19], and exactly the same solution of the nominal problem ($V_0^* = 978.7$ L, $F^* = 1\ L/s$, $T^* = 27.7\ s$) is obtained.

The proposed method is straightforwardly applied with the same steps previously illustrated. A sampling of 150 points is designed, the optimization problem (Eq. (13)) is solved (using the Matlab "*Fmincon*" algorithm) to obtain the input–output training data $[\theta_1, \theta_2]_{150\times2} - [C_B^*, V_0^*, F^*, T^*]_{150\times4}$ and, finally, four MPMs, $C_B^* = f_0(\theta_1, \theta_2), V_0^* = f_1(\theta_1, \theta_2), F^* = f_2(\theta_1, \theta_2)$ and $T^* = f_3(\theta_1, \theta_2)$, are trained. The MPMs validation is accomplished in the same way as previously mentioned, considering a new input–output dataset $[\theta_1^v, \theta_2^v]_{300\times2} - [C_B^{*,v}, V_0^{*,v}, F^{*,v}, T^{*,v}]_{300\times4}$ different from the training one.

The performance, shown in Fig. 11 and reported in Table 1, further emphasizes the methodology capabilities in terms of high prediction accuracy and significant reduction in the optimization time (99.92 % = (4887–3.88)/4887). Notice that the optimal flowrate is insensitive to the UPs variation (see Fig. 11) and always takes the maximum allowable value, which make sense because using the maximum flowrate minimizes the time of filling and emptying the reactor and, consequently, maximizes the objective function (maximum desired product B produced per unit time). Also, since the variability range of F^* is almost zero, the calculation of its NRMSE is meaningless (see Eq. (7)), as it will lead to an extremely high "numerical" value that is not expressing the actual performance of the MPM.

The Table, as a whole, also shows that the method advantages increase as the optimization problem complexity increases: in examples 1, 3 and 4, simple linear and quadratic optimization problems are solved, and the computational effort is reduced by 75.7 % (=100 × (2.1–0.51)/2.1), 67 % and 80 %, respectively. However, as the problem complexity increases, as in example 2 (bilinear objective function including a saddle behavior) and in example 5, the percentage of the computational effort saved rises to 98.9 % and 88.2 %, respectively. Finally, when the optimization problem involves a complex, nonlinear and/or black-box model (case studies), to which the conventional multiparametric mathematical programming approaches are not applicable, the ratios of the reduced computational effort reaches to 99.9 %.

Finally, the criticalities of the proposed method are inherently associated to the general drawbacks of ML models. The first one is that the provided solution is approximated without any explicit mathematical proof of optimality, unlike classical MPP methods. However, given that the optimality of the data, which are used to train the ML models, is guaranteed because they are generated by state-of-art optimization techniques, the point becomes to which extent these ML models are able to capture the exact multiparametric behavior, and this can be assessed by an offline validation step. Using the proposed procedure, the offline validation has proven the very high accuracy of the developed MPMs (less than 1 % of NRMSE in most of the cases presented in this work). The second criticality is the computational burden required for the training of the ML models, which arises as the training data size and/or

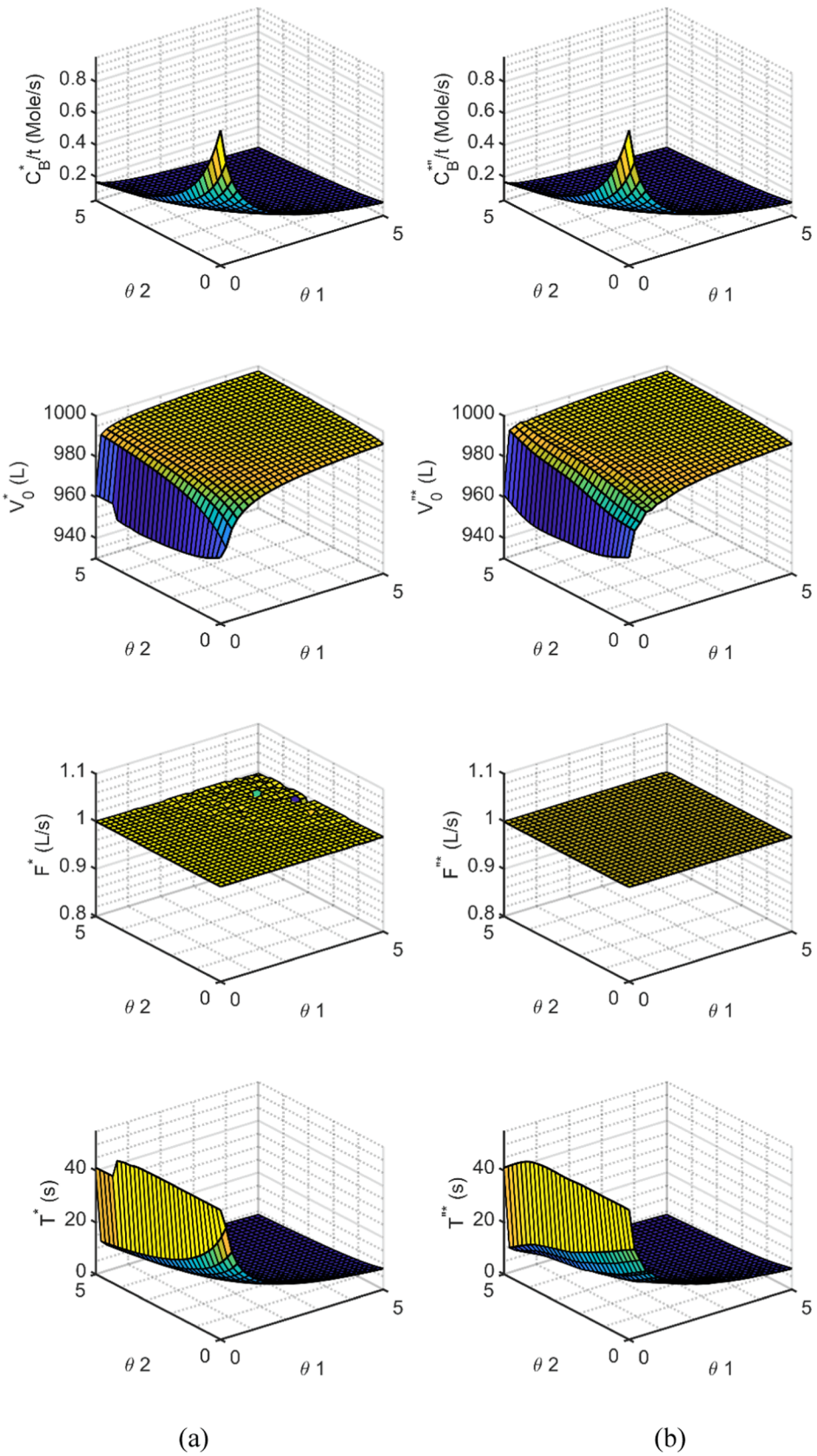

Fig. 11. (a) Exact versus (b) approximate multiparametric behavior of the batch reactor.

dimensions increase. Nevertheless, the MPMs are developed and validated offline, which makes the computational burden required for their training and validation quite affordable. The third criticality is associated to the discrete multiparametric behavior of some decision variables (e.g., see Section 4.5), for which the performance of the ML models is relatively reduced.

5. Conclusions

This paper presents an efficient ML-based methodology for multiparametric optimization of general continuous problems subjected to uncertainties, with special emphasis on chemical processes operation optimization problems, where the applications of conventional MPP approaches can be difficult due to the complexity, high nonlinearity and low transparency of the process models that must be considered at this operating phase. The method combines different statistical and mathematical tools including DOCE techniques, state-of-art optimization algorithms and machine learning models.

The proposed method has been tested on five benchmark MPP examples including linear, bilinear, quadratic and nonlinear problems and applied to two case studies of process operation optimization. The results show that the method is able to approximate the multiparametric solutions using a relatively small number of training data, with very good accuracy (less than 1 % of NRMSE, in most application cases). More importantly, significant differences with the results of the standard MPP appear, which are: *i*) in all the tested cases, a single MPM was enough to correctly reproduce the multiparametric behavior of the optimal solution over the whole UPs domain, which makes the online usage of the MPMs simpler and more flexible than using several mathematical functions (provided by classical MPP approaches) each one of them is applicable to a certain partition of the UPs space, *ii*) the method is able to solve problems of different types (linear, bilinear, quadratic, nonlinear) in a systematic and robust way, instead of using many types of conventional MPP algorithms, each one is applicable to a specific problem type, and *iii*) the method is capable of solving process operation optimization problems where complex, black-box and/or highly nonlinear models must be considered, providing a huge reduction in computational effort required for the online optimization (a ratio of 99.9 %).

Future research will investigate the extension of the methodology capabilities for improving the modeling of the multiparametric behavior of continuous variables that show significant/discrete changes over the UPs space (e.g., the example in Section 4.5), the solution of mixed-integer optimization problems and the quantification of uncertainty in the MPMs predictions.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] A. Bemporad, M. Morari, V. Dua, E.N. Pistikopoulos, The Explicit Linear Quadratic Regulator for Constrained Systems, Automatica 38 (2002) 3–20.
[2] L. Biegler, Nonlinear Programming: Concepts, Algorithms, and Applications to Chemical Processes, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2010.
[3] L.T. Biegler, Nonlinear Programming: Concepts, Algorithms, and Applications to Chemical Processes, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2010.
[4] J.A. Caballero, I.E. Grossmann, An algorithm for the use of surrogate models in modular flowsheet optimization, AIChE J. 54 (2008) 2633–2650.
[5] M.N.R. Chaudhary, Real Time Optimization of Chemical Processes, Master thesis, Curtin University of Technology, Bentley, Western Australia, 2009.
[6] G. Chia, S. Hua, Y. Yanga, T. Chen, Response surface methodology with prediction uncertainty: A multi-objective optimisation approach, Chem. Eng. Res. Des. 90 (2012) 1235–1244.
[7] E. Davis, M. Ierapetritou, A kriging method for the solution of nonlinear programs with black-box functions, AIChE 53 (2007) 2001–2012.
[8] N.A. Diangelakis, B. Burnak, J. Katz, E.N. Pistikopoulos, Process design and control optimization: A simultaneous approach by multi-parametric programming, AIChE J. 63 (2017) 4827–4846.
[9] L.F. Domínguez, D.A. Narciso, E.N. Pistikopoulos, Recent advances in multiparametric nonlinear programming, Comput. Chem. Eng. 34 (2010) 707–716.
[10] V. Dua, N.A. Bozinis, E.N. Pistikopoulos, A multiparametric programming approach for mixed-integer quadratic engineering problems, Comput. Chem. Eng. 26 (2002) 715–733.
[11] V. Dua, E. Pistikopoulos, Algorithms for the Solution of Multiparametric Mixed-Integer Nonlinear Optimization Problems, Ind. Eng. Chem. Res. 38 (1999) 3976–3987.
[12] G. Fadda, Supervision and Diagnosis of Industrial Systems. PhD. Cagliari: Universit'a degli Studi Di Cagliari, 2017.
[13] K.-T. Fang, R. Li, A. Sudjianto, Design and Modelling for Computer Experiment, Chapman and Hall/CRC, New York, 2005.
[14] T. Flemming, M. Bartl, P. Li, Set-Point Optimization for Closed-Loop Control Systems under Uncertainty, Ind. Eng. Chem. Res. 46 (2007) 4930–4942.
[15] A. Forrester, A. Keane, Recent advances in surrogate-based optimization, Prog. Aerosp. Sci. 45 (2009) 50–79.
[16] A. Forrester, A. Sobester, A. Keane, Engineering Design via Surrogate Modelling: A Practical Guide, John Wiley and Sons, Southampton, UK, 2008.
[17] S.S. Garud, I.A. Karimi, M. Kraftb, Design of computer experiments: A review, Comput. Chem. Eng. 106 (2017) 71–95.
[18] I.E. Grossmann, et al., Recent advances in mathematical programming techniques for the optimization of process systems under uncertainty, Comput. Chem. Eng. 91 (2016) 3–14.
[19] E.T. Hale, S.J. Qin, Multi-Parametric Nonlinear Programming and the Evaluation of Implicit Optimization Model Adequacy, IFAC Proc. Vol. 37 (2004) 449–454.
[20] B. Hauptman, V. Jovan, An approach to process production reactive scheduling, ISA Trans. 43 (2004) 305–318.
[21] C. Henao, C. Maravelias, Surrogate-based superstructure optimization framework, AIChE J. 57 (2011) 1216–1232.
[22] M. Ibrahim, et al., Impact of Sampling Technique on the Performance of Surrogate Models Generated with Artificial Neural Network (ANN): A Case Study for a Natural Gas Stabilization Unit, Energies 12 (2019) 1–12.
[23] H. Ichihara, H. Anai, A symbolic-numeric approach to multi-parametric programming for control design, in: Orlando, proceeding of the 50th IEEE Conference on Decision and Control and European Control Conference, 2012, pp. 7152–7157.
[24] Y. Jiao, H. Su, W. Hou, Z. Liao, Optimization of refinery hydrogen network based on chance constrained programming, Chem. Eng. Res. Des. 90 (2012) 1553–1567.
[25] D.R. Jones, M. Schonlau, W.J. Welch, Efficient global optimization of expensive black-box functions, J. Global Optim. 13 (1998) 455–492.
[26] O. Kajero, et al., Meta-modelling in chemical process system engineering, J. Taiwan Inst. Chem. Eng. 73 (2017) 135–145.
[27] J. Katz, I. Pappas, S. Avraamidou, E.N. Pistikopoulos, Integrating deep learning models and multiparametric programming, Comput. Chem. Eng. 136 (2020) 106801.
[28] J. Katz, I. Pappas, S. Avraamidou, E.N. Pistikopoulos, Integrating Deep Learning and Explicit MPC for Advanced Process Control. Denver, CO, USA, Published in: 2020 American Control Conference, 2020b, pp. 3559–3564.
[29] J.D. Kelly, D. Zyngier, Unit-operation nonlinear modeling for planning and scheduling applications, Optim. Eng. 18 (2017) 133–154.
[30] S. Kempf, B. Forget, L.-W. Hu, Kriging-based algorithm for nuclear reactor neutronic design optimization, Nucl. Eng. Des. 247 (2012) 248–253.
[31] K. Kouramas, N.P. Faisca, C. Panos, E.N. Pistikopoulos, Explicit/multi-parametric model predictive control (MPC) of linear discrete-time systems by dynamic and multi-parametric programming, Automatica 47 (2011) 1638–1645.
[32] Z. Li, Process Operations With Uncertainty and Integration Considerations, Rutgers, The State University of New Jersey, New Brunswick, New Jersey, 2010.
[33] G. Lupera, A. Shokry, G. Campanya, A. Espuña, Application of the Meta-MultiParametric Methodology to the Control of Emissions in the Industry under Continuous and Discrete Uncertain Parameters, Chem. Eng. Res. Des. 155 (2016) 365–373.
[34] G. Lupera, A. Shokry, G. M.Kopanos, A. Espuña, Mixed-integer multiparametric Metamodeling: A machine learning tool applied to reactive scheduling, Comput. Aided Chem. Eng. 43 (2018) (2018) 163–168.
[35] G. Lupera, et al., Ordinary Kriging: A machine learning tool applied to mixed-integer multiparametric approach, Comput. Aided Chem. Eng. 43 (2018) 531–536.
[36] A.G. Marchetti, A. Ferramosca, A. González, Steady-state target optimization designs for integrating real-time optimization and model predictive control, J. Process Control 24 (2014) 129–145.
[37] S. Medina-González, et al., Optimal management of bio-based energy supply chains under parametric uncertainty through a data-driven decision-support framework, Comput. Ind. Eng. 139 (2020) 105561.
[38] G. Mesfin, M. Shuhaimi, A chance constrained approach for a gas processing plant with uncertain feed conditions, Comput. Chem. Eng. 34 (2010) 1256–1267.
[39] D. Muller, et al., Real-Time Optimization in the Chemical Processing Industry, Chem. Ing. Tech. 89 (2017) 1464–1470.
[40] A. Norbert, et al., Simulation and Multi-criteria Optimization under Uncertain Model Parameters of a Cumene Process, Chem. Ing. Tech. 89 (2017) 665–674.
[41] R. Oberdieck, et al., On multi-parametric programming and its applications in process systems engineering, Chem. Eng. Res. Des. 116 (2016) 61–82.

[42] L. Ochoa-Estopier, et al., Industrial Application of Surrogate Models to Optimize Crude Oil Distillation Units, Chem. Eng. Trans. 69 (2018) 289–294.
[43] L. Ochoa-Estopier, M. Jobson, Optimization of Heat-Integrated Crude Oil Distillation Systems. Part I: The Distillation Model, Ind. Eng. Chem. Res. 54 (2015) 4988–5000.
[44] K. Palmer, M. Realef, Optimization and validation of steady-state flowsheet simulation metamodels, Trans IChemE 80 (2002) 773–782.
[45] M.M. Papathanasiou, et al., Control of a dual mode separation process via multi-parametric Model Predictive Control, IFAC PapersOnLine 52 (2019) 988–993.
[46] I. Pappas, et al., Multiparametric Programming in Process Systems Engineering: Recent Developments and Path Forward, Front. Chem. Eng. 2 (2021) 620168.
[47] E. Pistikopoulos, Uncertainty in process design and operations, Comput. Chem. Eng. 19 (1995) 553–563.
[48] E.N. Pistikopoulos, Perspectives in Multiparametric Programming and Explicit Model Predictive Control, AIChE J. 55 (2008) 1918–1925.
[49] E.N. Pistikopoulos, et al., On-line optimization via off-line parametric optimization tools, Comput. Chem. Eng. 26 (2002) 175–185.
[50] E.N. Pistikopoulos, M.C. Georgiadis, V. Dua, Multi-Parametric Programming: Volume 1: Theory, Algorithms, and Applications, Wiley-VCH Verlag GmbH & Co, Weinheim, 2007.
[51] N. Quirante, J. Caballero, Large scale optimization of a sour water stripping plant usingsurrogate models, Comput. Chem. Eng. 92 (2016) 143–162.
[52] N. Quirante, J. Javaloyes-Antón, J.A. Caballero, Hybrid simulation-equation based synthesis ofchemical processes, Chem. Eng. Res. Des. 132 (2018) 766–784.
[53] P. Rivotti, R.S.C. Lambert, E.N. Pistikopoulos, Combined model approximation techniques and multiparametric programming for explicit nonlinear model predictive control, Comput. Chem. Eng. 42 (2012) 277–287.
[54] B. Roffel, B. Betlem, Advanced Practical Process Control, Springer-Verlag, Berlin Heidelberg, Berlin, 2004.
[55] D.D. Salback, A Surrogate Model Approach to Refinery-Wide Optimization, PhD thesis, Texas Tech University, Texas, 2004.
[56] G. Shao, L. Hasan, C. Martin-Villalba, P. Denno, Standards-based integration of advanced process control and optimization, J. Ind. Informat. Integrat. 13 (2019) 1–12.
[57] A. Shokry, A.D. Bojarski, A. Espuña, Using surrogate models for process design and optimization, Uncertainty Model. Knowledge Eng. Decision Making 7 (2012) 483–488.
[58] A. Shokry, C. Dombayci, A. Espuña, Multiparametric Metamodels for Model Predictive Control of Chemical Processes, Comput. Aided Chem. Eng. 38 (2016) 1955–1960.
[59] A. Shokry, A. Espuña, Applying Metamodels and Sequential Sampling for Constrained Optimization of Process Operations, Lect. Notes Comput. Sci. 8468 (2014) 396–407.
[60] A. Shokry, A. Espuña, Dynamic Optimization of Batch Processes under Uncertainty via Meta-MultiParametric Approach, Comput. Aided Chem. Eng. 40 (2017) 2215–2220.
[61] A. Shokry, S. Medina-González, A. Espuña, Mixed-Integer MultiParametric Approach based on Machine Learning Techniques, Comput. Aided Chem. Eng. 40 (2017) 451–456.
[62] M. Vaccari, G. Pannocchia, A Modifier-Adaptation Strategy towards Offset-Free Economic MPC, Processes 5 (2017) 1–22.
[63] Aspentech, 2020. Aspen-Hysys. https://www.aspentech.com/en/products/engineering/aspen-hysys.