Optimization and Engineering
https://doi.org/10.1007/s11081-023-09860-6

RESEARCH ARTICLE

Globally optimal scheduling of an electrochemical process via data-driven dynamic modeling and wavelet-based adaptive grid refinement

Chrysanthi Papadimitriou[1] · Tim Varelmann[1,5] · Christian Schröder[2] · Andreas Jupke[2] · Alexander Mitsos[1,3,4]

Received: 12 June 2023 / Revised: 2 October 2023 / Accepted: 2 October 2023

Abstract
Electrochemical recovery of succinic acid is an electricity intensive process with storable feeds and products, making its flexible operation promising for fluctuating electricity prices. We perform experiments of an electrolysis cell and use these to identify a data-driven model. We apply global dynamic optimization using discrete-time Hammerstein–Wiener models to solve the nonconvex offline scheduling problem to global optimality. We detect the method's high computational cost and propose an adaptive grid refinement algorithm for global optimization (AGRAGO), which uses a wavelet transform of the control time series and a refinement criterion based on Lagrangian multipliers. AGRAGO is used for the automatic optimal allocation of the

✉ Alexander Mitsos
amitsos@alum.mit.edu

Chrysanthi Papadimitriou
chrysanthi.papadimitriou@avt.rwth-aachen.de

Tim Varelmann
tim@bluebirdoptimization.com

Christian Schröder
christian.schroeder@avt.rwth-aachen.de

Andreas Jupke
andreas.jupke@avt.rwth-aachen.de

1 Process Systems Engineering (AVT.SVT), RWTH Aachen University, Forckenbeckstraße, 52074 Aachen, Germany

2 Fluid Process Engineering (AVT.FVT), RWTH Aachen University, Forckenbeckstraße, 52074 Aachen, Germany

3 JARA-CSD, Templergraben, 52056 Aachen, Germany

4 Institute of Energy and Climate Research, Energy Systems Engineering (IEK-10), Forschungszentrum Jülich GmbH, Wilhelm-Johnen-Straße, 52425 Jülich, Germany

5 Starenweg 13, 48429 Rheine, North Rhine-Westphalia, Germany

Published online: 24 November 2023

control variables in the grid to provide a globally optimal schedule within a given time frame. We demonstrate the applicability of AGRAGO while maintaining the high computational expenses of the solution method and detect superior results to uniform grid sampling indicating economic savings of 14.1%.

Keywords Electrified downstream processing · Demand-side-management · Discrete-time scheduling · Global dynamic optimization · Hammerstein–Wiener model · Adaptive grid refinement

1 Introduction

Demand-side-management (DSM) has long been acknowledged as an effective approach to ensuring stability and increasing efficiency in the power grid, as well as raising profit for electricity-intensive processes in economies where a high share of electricity is generated from intermittent renewable sources (Paulus and Borggrefe 2011; Mitsos et al. 2018). Traditional approaches to optimal scheduling do not explicitly account for the intrinsic process dynamics by relying on stationary process models. In case, however, the time scale of the process dynamics is similar to that of the prices, the former should be considered to ensure accuracy (Bhatia and Biegler 1996; Flores-Tlacuahuac and Grossmann 2006). In this direction, optimal process scheduling is applied, typically considering discrete-time formulations, due to the inherently discrete-time electricity prices (Zhang and Grossmann 2016).

Dynamic optimization problems arising in the field of scheduling are typically solved numerically with local optimization solution methods and with various techniques that simplify the system dynamics (Dias and Ierapetritou 2019). Such techniques (e.g., single perturbation analysis (Đukić and Sarić 2012), or scale-bridging techniques (Du et al. 2015)) reduce the problem size to a subset of "scheduling-relevan" variables; thus, they enable the consideration of even large-scale and complex dynamic systems, as well as sophisticated optimization methods, such as global optimization methods.

Various deterministic global optimization approaches for the solution of dynamic optimization problems have been introduced over the past years (Singer and Barton 2006; Scott and Barton 2013; Song and Khan 2022; Schaber et al. 2019; Floudas and Gounaris 2009). Global dynamic scheduling has been mainly performed via consideration of linear dynamics for the solution of a (MI)LP (Suresh and Chaudhuri 1993; Floudas and Lin 2005), or by exploiting special problem structures together with solution methods that allow for the use of standard optimization solvers (e.g., exploiting quasi-convexity of the relaxed NLP problem to solve a convex mixed-integer quadratic programming (MIQCP) (Chu and You 2012)). In this direction, nonlinear dynamic model simplification techniques, which consider the use of low-order scale-bridging models (SBMs) have been employed. Pattison et al. (2016) first explore the application of SBMs to account for the closed-loop dynamics of an air-separation unit (ASU) integrated in a demand response (DR) paradigm. Kelley et al. (2018a) propose a method for linearizing Hammerstein–Wiener and finite step response SBMs to reformulate the optimal scheduling problem as an MILP to be solved using stan-

dard MILP solvers, which they later apply to scheduling under dynamic constraints of an ASU (Kelley et al. 2018b). Following that, a method for further simplification of the optimal scheduling problem under dynamic process behavior formulated as an LP, which accounts for a linear representation of the process dynamics using a linear autoregressive (ARX) model has been proposed by Kelley et al. (2022). However, most chemical and energy processes are described by complex models, with resulting nonconvex and multimodal optimal operation problems (Chachuat et al. 2006) that may lead to intractable scheduling objectives in case of model simplification (Yang et al. 2014); thus, the global solution of such nonconvex scheduling problems is desired from an economic, safety, and environmental perspective.

One of the latest studies on the topic of deterministic global dynamic optimization, given by Kappatou et al. (2022), describes a framework for the global solution of dynamic optimization problems using Hammerstein–Wiener (HW) models for the expression of the nonlinear system dynamics. This approach suggests the utilization of the special structure of HW models to decouple the linear dynamics and the nonlinearities. A reformulation procedure accounting for this decoupling leads to the construction of the lower bounding problem. Then, the optimization problem, which considers a continuous state space formulation, can be solved to global optimality after control parametrization with a combination of a relaxation theory for ODEs (Singer 2004; Singer and Barton 2004) together with a convergent branch-and-bound (B&B) algorithm (standard relaxation techniques), and a numerical integration scheme similarly to following a single-shooting approach. The scaling of the method with the number of optimization variables discourages its real-time applicability to big problems (Kappatou et al. 2022; Chachuat et al. 2006).

In the present work, we aim at the optimal dynamic scheduling of a downstream electrochemical process for the recovery of succinic acid (Gausmann et al. 2020; Schröder et al. 2022). We conduct experiments of an electrolysis cell and acquire data for identifying a surrogate dynamic process model which considers a single input control and a single output variable. In order to obtain the global solution to the multimodal problem, we apply the approach of Kappatou et al. (2022), which we extend to global dynamic scheduling with discrete-time dynamics. Our goal is to identify the power consumption set-points resulting in energy-related cost minimization for a flexibly operating electrochemical process that participates in a day-ahead electricity market.

Early works on optimization have established a link between the dimensionality of the search space and the computational performance of the optimization algorithm, often referring to the scaling time phenomenon as "curse of dimensionality" (Bellman 1957). In the case of large-scale optimization problems, the exponential worst-case runtime of the B&B algorithm imposes difficulties in the global solution of such problems. As a result, parametrization of the control variables and computational time are linked (Chachuat et al. 2006). As presented in the work of Kappatou et al. (2022), solution time rises exponentially with the number of piecewise constant control segments that derive from the temporal discretization selected or proportionally the number of degrees of freedom (DoFs). Subsequently, the selection of relatively large scheduling horizons (e.g., a week or a year) or a fine control grid may lead to significantly high solution times, sometimes even higher than the scheduling horizon's length. A

method to tackle this problem is the application of a grid refinement technique that optimally allocates the setpoint change breakpoints of the control variable with the aim to decrease the number of optimization variables and thus the solution effort of the optimization problem (Schlegel et al. 2005).

Various methods of grid refinement for the solution of optimal control problems have been developed over the years (Chen et al. 2019; Zhao and Li 2019). Schlegel et al. (2005) introduce a wavelet-based adaptive refinement algorithm for the solution of dynamic optimization problems under continuous-time dynamics with the single shooting method. They later include an a priori investigation step of the existence of switching times and arcs to reformulate the problem as a multi-stage dynamic optimization problem where each stage corresponds to an arc (Schlegel and Marquardt 2006). The approach of Schlegel et al. (2005) is further extended by Schäfer et al. (2020a, b) to scheduling problems that account for quasi-stationary process models.

Grid refinement techniques that apply the simultaneous method have additionally been presented, such as the knot placement problem suggested by Cuthrell and Biegler (1987) and the finite element adaptation with error criteria enforcement introduced by Vasantharajan and Biegler (1990). Chen et al. (2012) develop a moving finite element method for the optimal allocation of the breakpoints for the control profiles to solve dynamic optimization problems based on direct transcription. The breakpoint determination results from an approximation error and a Hamiltonian-based termination criterion. For this, the optimization problem is formulated and analyzed as a multi-period dynamic optimization problem. The approach sets the basis for grid refinement with bilevel optimization (Weifeng Chen et al. 2014) and an adaptive mesh refinement strategy for the solution of singular optimal control problems based on partial moving grids using the simultaneous approach (Chen et al. 2019). While a wide collection of grid refinement algorithms exists, the formulation of the optimal control allocation problem as a multistage or a bilevel problem would additionally increase the computational expenses of the solution approach presented in this work. To this end, a grid refinement algorithm based on the work of Schlegel et al. (2005) is developed, as it allows for a sequential optimization approach of the global dynamic optimization problem as explained in Kappatou et al. (2022) for the solution of the optimal control problem possibly including state or cumulative control (see Sect. 3.1) constraints.

In this work, we propose an adaptive grid refinement algorithm for global optimization (AGRAGO) using HW models as an extension of the algorithms for wavelet-based grid adaptation presented in previous works (Schlegel et al. 2005; Schäfer et al. 2020b, a); the algorithm is implemented as an add-on tool to our in-house open-source software for deterministic global optimization, MAiNGO (Bongartz et al. 2018). AGRAGO enables the wavelet-based refinement of the control vector parametrization developed by Schlegel et al. (2005) for the solution of dynamic optimization problems with piecewise constant parametrization of the control variables to local or—if the problem formulation allows for it - global optimality. The algorithm uses the wavelet-based refinement approach considering the deletion and the insertion criterion developed and used by Schäfer et al. (2020a) and Schäfer et al. (2020b), respectively. The refinement strategy is coupled with the reduced-space modeling paradigm in MAiNGO to enable a nonuniform discretization that exposes only a few DoFs to the optimizer while performing the optimization step in the original parameterized control space.

The application of AGRAGO to dynamic scheduling aims at offering a compromise between the global optimization of simplified process models with fewer DoFs, and the local optimization of detailed models with finer control parametrization or larger optimization horizons; thus, a global optimum of a multimodal problem can be acquired by refining the control grid to reach the best allocation of the available DoFs. We herein apply dynamic optimization with AGRAGO to solve a nonconvex optimal scheduling problem. However, we underline that the superiority of the global solution over the solution obtained by local optimization and with a higher number of DoFs is not guaranteed.

The remainder of this article is structured as follows. We first describe the process to be operated under optimal scheduling and present the identification of the HW model describing the process (Sect. 2). The experimental data used for the model identification are being made available as Supplementary information. Then, we set the deterministic global dynamic scheduling problem with time-variable electricity prices in a reduced-space formulation and present the scheduling results (Sect. 3). Afterward, we introduce AGRAGO, which we utilize to solve the scheduling problem and present the results (Sect. 4). We finally draw conclusions on the offline global dynamic scheduling of the electrochemical process under discussion and the application of AGRAGO to the solution of the scheduling problem (Sect. 5). In the appendices we provide supporting information on parameters and results of the model fitting, optimization, and AGRAGO.

2 Data-driven dynamic modeling

2.1 Process description and experimentation

2.1.1 Process outline

Bio-based succinic acid production receives growing attention in the past few years due to its ability to replace succinic acid produced from fossil raw materials (Bozell and Petersen 2010; Song and Lee 2006). Nevertheless, the downstream processing of bio-based succinic acid is still challenging since it accounts for more than 60% of the total production costs (Bechthold et al. 2008). We recently proposed a downstream process in which succinic acid is recovered from aqueous solution by electrochemical pH-shift extraction, back-extraction and crystallization (Gausmann et al. 2020; Schröder et al. 2022), Fig. 1. This process is a strong candidate for DSM and potentially DSM can lead to sustainable succinic acid production. Herein, we investigate the anode chamber of the first electrolysis cell of this downstream process.

2.1.2 Experimental data acquisition

We conducted a series of experiments producing data that describe the process operation under a certain operating range to learn the HW model. We used an artificial aqueous solution containing disodium succinate and succinic acid as surrogate for the feed from the fermentation. The composition of disodium succinate and succinic acid

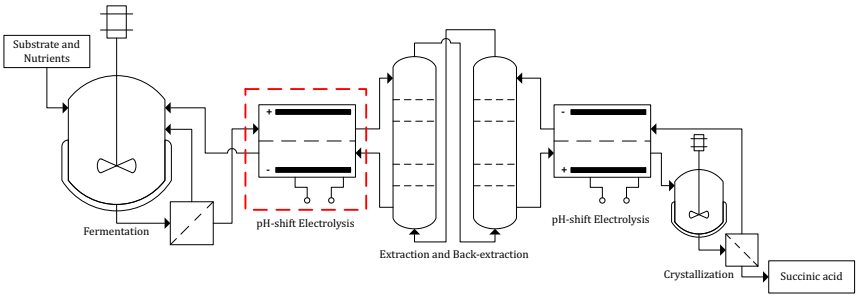


Fig. 1 Flowsheet of the succinic acid production process as derived from the work of Schröder et al. (2022). The part of the process considered for flexible operation is outlined in red

was chosen so that the aqueous solution has a pH-value of 6.8, which is a typical pH-value of succinic acid fermentation broth (Bradfield et al. 2015; Salvachúa et al. 2016; Thuy and Boontawan 2017). Disodium succinate (for synthesis) was supplied by Merck KGaA, Darmstadt, Germany, and succinic acid ($\leq$99wt%) was provided by Sigma-Aldrich, Saint Louis, United States. In total, approximately 47g L^{-1} succinate was present in the aqueous solution. Additionally, 14.2g L^{-1} sodium sulfate (99wt%) from Fisher Scientific GmbH, Schwerte, Germany, was added as background electrolyte to maintain a minimum conductivity of the model solution. The model solution was fed into the anode compartment and into the cathode compartment. The experiments were performed with a custom-made polytetrafluoroethylene electrolysis cell, which has an electrode area of 100cm^2. The electrolysis cell was already successfully employed in a similar downstream process of Gausmann et al. (2021). An iridium mixed metal oxide coated titanium anode and a platinum coated titanium cathode, both delivered by Magneto Special Anodes B.V., Schiedam, Netherlands, were used. A Fumasep FKL-PK-130 cation exchange membrane from Fumatech BWT GmbH, Bietigheim-Bissingen, Germany, was used as separator between anode and cathode. The electrolysis cell was operated under galvanostatic conditions with an EA-PSI 9040-120 power supply from EA Elektro-Automatik GmbH, Viersen, Germany.

Different set-point changes of current, meeting the safety requirements of the system, were applied to the electrolysis cell using a custom-made control interface and each experiment was conducted once. A rule-based control system was used to adjust the feed volume flow when a current density set-point change occurred, in order to achieve full protonation of the succinic acid. No feedback control system was applied in that case due to the self-regulating behavior of the current and flow. A current density of 0.12A cm^{-2} was selected as the nominal operating point. The current density was varied between 0.06A cm^{-2} and 0.15A cm^{-2}. The control interface was developed using LabVIEW from National Instruments, Austin, Texas, United States.

2.1.3 Model selection and data usage

We construct a data-driven dynamic model using the experimental data of the first electrolysis cell, as a process model that can well represent the potentials of integrat-

ing the entire process in the scheduling paradigm. For that, we utilize the available experimental data and disregard the consecutive process steps, to which experimentation has not yet been performed, to derive a process model to describe the flexible process operation. We, thus, first derive the set of "scheduling-relevant" variables, which are the process variables included in the product quality, production rate, process operating constraints, and scheduling objective (Pattison et al. 2016): the molar throughput of the anode chamber, the power consumption of the electrolysis cell, and the total acid production. For that purpose, the available volume flow and current data are converted into molar flow and power consumption, and the latter form the set of experimental data provided.

In the electrolytic cell no acid losses take place, the consecutive process steps are disregarded, and closed recycle process streams are applied to finally acquire a high-purity (Gausmann et al. 2020) acid product, safe—in terms of effect on the scheduling calculations—to assume that it is fully recovered; thus, no product quality, as well as operating constraints related to the consecutive process steps are accounted for, and the input molar flow can also correspond to the output molar flow of the acid. The operating points of the electrolysis cell are chosen in such a way that safety constraints are far from their bounds, and, thus, production rate constraints suffice to describe the flexible process operation. Possible revenues from applying DR paradigms are investigated considering the available data describing the lab-scale process. Although a different behavior is expected for the scaled-up process, such an investigation is out of the scope of this manuscript. Additional consideration of the entity of the operation units of the recovery process to construct a holistic process model will be part of future work.

A single input and single output (SISO) model of the process is built to consider the molar throughput (control variable) as the input and the power consumption as the output, given the control scheme applied, as presented in Sect. 2.1.2. The total acid production is used as a boundary condition in the optimization problem for flexible operation. The slow kinetics that govern the process dynamics are taken into account. Hence, dynamic modeling is considered. In addition, the high complexity of the reaction kinetics, including the water splitting electrolysis occurring in the electrolysis cell, motivates the use of nonlinear surrogate models that capture the intricate process dynamics without the need to identify a fully nonlinear mechanistic model (Tsay et al. 2019). A SISO HW model is used and optimal scheduling is performed using the approach of Kappatou et al. (2022) for deterministic global dynamic optimization with HW models.

2.2 Hammerstein–Wiener model identification

2.2.1 HW model framework

The HW models are data-driven models which consist of two nonlinear static transformations which precede (Hammerstein block) and follow (Wiener block) a linear dynamic system. They can be formulated for single or multiple inputs and outputs, considering the representation of the linear dynamics as a state-space model. A single-

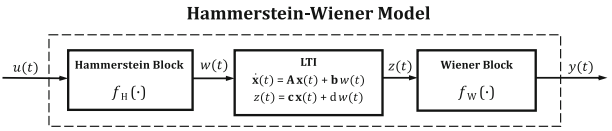


Fig. 2 SISO Hammerstein–Wiener model

input-single-output (SISO) HW model—used for the modeling of the electrolysis process as discussed in Sect. 2.1.1—is described by the following set of equations:

$$\begin{aligned} w(t) &= f_{\mathrm{H}}(u(t)), \quad \forall t \in [t_0, t_f] \\ \dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{b}w(t), \quad \forall t \in [t_0, t_f] \\ z(t) &= \mathbf{c}\mathbf{x}(t) + dw(t), \quad \forall t \in [t_0, t_f] \\ \mathbf{x}(t_0) &= \mathbf{x}_0 \\ y(t) &= f_{\mathrm{W}}(z(t)), \quad \forall t \in [t_0, t_f] \end{aligned} \tag{1}$$

where $u : [t_0, t_f] \to \mathbb{R}$ is the input and $y : [t_0, t_f] \to \mathbb{R}$ is the output of the HW model, $w : [t_0, t_f] \to \mathbb{R}$ is the input of the linear time-invariant (LTI) system, $\mathbf{x} : [t_0, t_f] \to \mathbb{R}^{n_x}$ are the states, $z : [t_0, t_f] \to \mathbb{R}$ is the output of the LTI system, $f_{\mathrm{H}} : \mathbb{R} \to \mathbb{R}$ is the input Hammerstein nonlinearity, $f_{\mathrm{W}} : \mathbb{R} \to \mathbb{R}$ is the input Wiener nonlinearity, $\mathbf{A} \in \mathbb{R}^{n_x \times n_x}$, $\mathbf{b} \in \mathbb{R}^{n_x \times 1}$, $\mathbf{c} \in \mathbb{R}^{1 \times n_x}$, $d \in \mathbb{R}$ are the matrices defining the linear state-space dynamical system, and $\mathbf{x}_0 \in \mathbb{R}^{n_x}$ are the initial conditions of the LTI. It is highlighted that the input variable $u(t)$ is bounded, so that $u(t) \in \mathbb{U}$, $\mathbb{U} \subset \mathbb{R}$ and $\mathbb{U}$ compact, due to its physical meaning for the case of the electrochemical process.

The structure of a SISO HW model is given in Fig. 2.

2.2.2 Data preprocessing

We use the collection of four experimental operating datasets of the process presented in Sect. 2.1.1 available to identify a HW model for offline scheduling and the dataset corresponding to one of the four experiments is given in Fig. 3 for illustrative purposes.

The first step for fitting a HW model to the available data is the data preprocessing, which results in a single set of equidistantly sampled data over time devoid of any noise. The preprocessing involves the smoothening of the dataset of around 6000 data points for each measured variable using a Simple Moving Average (SMA) with a window of 21 and, next, the linear interpolation between the smoothened data; therefore, we finally acquire an approximately ten-times smaller dataset of uniformly sampled data with a selected time step of 3min. The constant time step results from the minimization of the Bayesian information criterion (BIC) occurring from different samplings of the data, and it is ensured to be smaller than the inherent process dynamics (e.g., settling time). The experimental data, the smoothened data, and the final working data of a specific set-point change interval corresponding to the first experiment are given in Appendix A.1, Fig. 10.

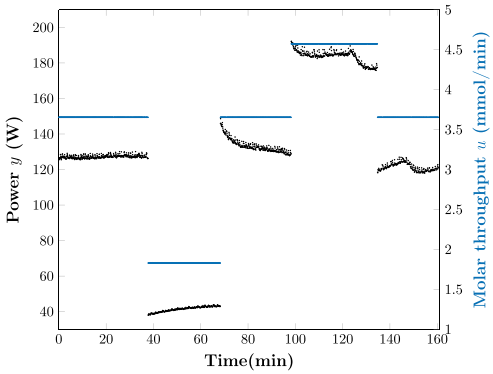


Fig. 3 Experimental data—experiment 1

2.2.3 Model identification

The identification of the HW model includes first the selection of a set of hyperparameters, namely the type and order of the Hammerstein and Wiener nonlinearities, as well as the parameters (number of zeros, number of poles, and input delay) of the LTI system. It should be noted here that a black-box identification approach for the HW models is followed in this work. This does not consider incorporating prior knowledge of the chemical system into the model; thus, the use of the model is limited to the range of experimentation. Grey-box identification approaches are also available (Mohammadi and Montazeri-Gh 2015) but are out of the scope of this manuscript.

An additional parameter to be specified is the use of either discrete- or continuous-time dynamics together with a numerical integration scheme. Note that most applications of dynamic scheduling in literature use discrete time, e.g., (Kondili et al. 1993; Velez and Maravelias 2013; Zhang et al. 2016), whereas continuous time was used by Kappatou et al. (2022) for the global dynamic optimization. The decision over the hyperparameters related to the Hammerstein, the Wiener, and the LTI block takes place through a standard training and validation procedure over a wide range of available options provided by the tool used (results not presented here) considering the minimization of the normalized Aikaike criterion (nAIC) and the BIC criterion, as well as considering a minimum accepted fitting to the training and test data of 95% (1-*NMSE*) (Tsay et al. 2019).

Our preliminary experimentation (results not shown) revealed that the computational performance of the optimization strongly depends on the HW model parametrization. First, depending on the nonlinearities used, the tightness of the constructed relaxations, here McCormick relaxations, varies (McCormick 1976). From the available options for the type of nonlinearities for the HW models training (e.g, piecewise constant, piecewise linear), continuous polynomial functions are considered, in order to avoid additional complexity by introducing binaries linked to the static function's breakpoints leading to the solution of an MINLP (given the available standard solvers). However, in our case the loose McCormick relaxations related to the fourth-degree polynomial Hammerstein function result in high solution times of the optimization problem. As an alternative to the fourth-degree polynomial, we use

an ANN which we train on the Hammerstein function—assuming negligible error in our computations for a fitting of 99.99% (1-$NMSE$)—with the aim to utilize tighter embedded ANN relaxations as provided by our tool MeLOn (Schweidtmann and Mitsos 2019) and thus, improve the computational performance of the optimization. The approach should not be confused with methodologies on handling the HW model accuracy.

Second, the computational performance of the scheduling problem depends on the type of the system dynamics, discrete- or continuous-time, with the latter introducing additional calculations related to the numerical integration of the dynamics. In fact, according to Kappatou et al. (2022), solution time exhibits a linear scaling with the number of steps in the integration scheme. In the same direction, a possibly small step length, thus higher number of steps, of the numerical solution of the ordinary differential equations, necessary—in many cases—to control numerical instability of the integration method (Cartwright and Piro 1992; Yang et al. 2008), may significantly increase the computational effort. Preliminary results show superior computational performance for the application of discrete-time dynamics. Therefore, we select discrete time, regardless of the resulting limitations related to the single possible state discretization of the control time series. We assume at this point that the error occurring from the transformation of discrete- to continuous-time dynamics and vice-versa is insignificant and thus not considered for the selection between the two. The effect of the model parametrization on the problem's solution times is utilized to provide feedback to the HW model identification process. The goal is a highly accurate HW model which does not degrade the computational performance of the optimization.

The HW model is obtained using the System Identification Toolbox (SIT) of MATLAB version R2021b, considering the first three experiments (70% of experimental data) to be the training data and the last experiment (30% of experimental data) as test data. The estimation parameters of the MATLAB SIT are kept at the default values. The parameters arising from the fitting of a HW model for a Hammerstein polynomial of 4th, a Wiener polynomial of 2nd degree, and an LTI of 3 zeros, 4 poles, no delay, and arbitrarily selected zero initial (non-physical) states, similar to Kappatou et al. (2022); Cai et al. (2018), to correspond to the standard experimental start-up steady-state—hereon the reference operating point—are given in Appendix A.2. The model hyperparameters and the model fitting results are presented in Appendix A.2, Table 1, and Appendix A.3, Fig. 11. Additionally, the ANN of one hidden layer with two neurons and the hyperbolic tangent activation function and of one output layer with the linear activation function is trained on the Hammerstein function in MATLAB. Overall, the modeling results exhibit good performance, leading to an accurate (1-$NMSE \geq$ 95%) prediction of the system's behavior.

3 Deterministic global dynamic scheduling

3.1 Problem formulation

The DR scheduling problem of the electrochemical process presented in Sect. 2.1.1 is formulated as a dynamic optimization problem in discrete time using the HW model identified in Sect. 2.2 to predict the open-loop response of the electricity consumption of the process to changes in the acid production. The high multimodality of the problem, as indicated by a maximum observed variation of 9% in the objective values resulting from local optimization (see Table 2 in Appendix B), motivates the application of global optimization. The problem is solved to global optimality considering the problem formulation for global dynamic optimization with HW models (Kappatou et al. 2022) for an electrochemical process taking part in a day-ahead electricity market. Scheduling is considered under a 24-hour profile of hourly changing German spot electricity prices of February 7, 2023, retrieved from Energy-Charts (Fraunhofer Institute for Solar Energy Systems ISE 2023).

The electricity costs are minimized by manipulating the molar throughput of the anode chamber subject to the electrochemical process dynamics and a minimum requirement in the daily acid production. The problem is solved to global optimality after piecewise-constant control parametrization, convex relaxation, and additional discretization of the dynamics. Piecewise-constant control parametrization is selected between others (e.g., piecewise-linear), with the goal to use a single control parameter for each control segment, thus keeping the number of DoFs and consequently the computational expenses, as explained in Sect. 1, lower. It is, additionally, an intuitive parametrization, as it matches that of the problem parameter, the electricity price. Note that the control vector parametrization (transition from an infinite-dimensional to a finite-dimensional optimization problem) is an inherent restriction; thus, global optimality refers to the selected control parametrization. In contrast to the work by Kappatou et al. (2022), we herein provide a problem formulation with discrete-time dynamics, as presented in Sect. 2.2.

The resulting problem formulation is

$$\min_{\hat{\mathbf{u}},\hat{\mathbf{w}}} T \cdot \sum_{t=1}^{n} \sum_{s=1}^{k} p_{s+k(t-1)} \cdot f_{\mathrm{W}}\left(\mathbf{c}_{\mathrm{D}} \hat{\mathbf{x}}_{s+k(t-1)} + d_{\mathrm{D}} f_{\mathrm{H}}(\hat{u}_t)\right) \tag{2a}$$

$$s.t. \quad \hat{w}_t = f_{\mathrm{H}}(\hat{u}_t) \qquad \forall t \in \{1, 2, \ldots, n\} \tag{2b}$$

$$u^{\mathrm{L}} \leq \hat{u}_t \leq u^{\mathrm{U}}, \qquad w^{\mathrm{L}} \leq \hat{w}_t \leq w^{\mathrm{U}} \qquad \forall t \in \{1, 2, \ldots, n\} \tag{2c}$$

$$m_{\mathrm{a}} \leq T \cdot \sum_{t=1}^{n} \hat{u}_t \tag{2d}$$

where $\hat{\mathbf{x}}_{s+k(t-1)}$ is the solution of:

$$\hat{\mathbf{x}}_{s+k(t-1)} = \mathbf{A}_{\mathrm{D}} \hat{\mathbf{x}}_{s+k(t-1)-1} + \mathbf{b}_{\mathrm{D}} \hat{w}_t \qquad \forall t \in \{1, 2, \ldots, n\}, \forall s \in \{1, 2, \ldots, k\}$$
$$\hat{\mathbf{x}}_0 = \mathbf{0}$$

the controls and states are parameterized as:

$$\hat{\mathbf{x}}_{s+k(t-1)} = \begin{pmatrix} \hat{x}_{1,s+k(t-1)} \\ \vdots \\ \hat{x}_{l,s+k(t-1)} \end{pmatrix} \in \mathbb{R}^l,$$

$$\hat{\mathbf{u}} = \begin{pmatrix} \hat{u}_1 \\ \vdots \\ \hat{u}_n \end{pmatrix} \in \mathbb{R}^n, \qquad \hat{\mathbf{w}} = \begin{pmatrix} \hat{w}_1 \\ \vdots \\ \hat{w}_n \end{pmatrix} \in \mathbb{R}^n,$$

and T is the scheduling horizon, n is the number of control intervals, $n \cdot k$ is the number of state intervals, and l is the number of states. k is the number of intervals in which a single control interval is split for the sampling of the dynamics, so that $T/(n \cdot k)$ is always the time-step of sampling the discrete-time dynamics. The control and state parametrization of n number of control and $n \cdot k$ number of state intervals is selected to allow for asynchronous control set-point and price change, giving the possibility of either more or less frequent control point shifts than price shifts. The only restriction of the formulation above is that a price change coincides with a state discretization point. m_a is the minimum allowed daily acid production, p_i, with $i \in \{1, 2, \ldots, n \cdot k\}$ is the price of the spot market at the ith scheduling horizon interval. $\hat{\mathbf{x}}_0$ are the initial state conditions, set to zero, to account for initialization from the reference operating point (see Sect. 2.2), and $\mathbf{A}_\mathrm{D}$, $\mathbf{b}_\mathrm{D}$, $\mathbf{c}_\mathrm{D}$, d_D, f_H and f_W are the parameters of the HW model given in Appendix A.2.

3.2 Computational scheduling performance with equidistant grids

The scheduling problem is implemented and solved using our in-house open-source software for deterministic global optimization, MAiNGO (Bongartz et al. 2018), based on McCormick relaxations (McCormick 1976; Mitsos et al. 2009; Tsoukalas and Mitsos 2014) with default settings unless otherwise stated. Appendix A.3 includes the configuration values deviating from the default ones (Table 3), and a guideline to the selection of the optimality tolerance. The ANN functions are accessible using the toolbox for machine learning models for optimization MeLOn (Schweidtmann and Mitsos 2019). All computations are conducted on an Intel® Xeon®CPU E5-2640 v3 @ 2.60GHz with 128 GB RAM and using a single server CPU.

For the optimization of the problem (2a)–(2d), we apply a set of problem parameters and the optimization configurations given in Table 3 (Appendix B). The problem is solved for a constant discretization of the dynamics of 3 min, a scheduling horizon of 24 h (T =1440min) of hourly changing spot electricity market prices, and n = 2, 3, 4, 6, 5, 8, 10 equidistant control time intervals.

From Fig. 4, we observe a general improvement of the economic objective value with the number of control discretization points. This is not guaranteed, though, in case of control set-point changes that do not coincide with a price set-point change (e.g., n = 5, 10), a smaller number of control set-point intervals that correspond to less highly-deviating consecutive prices (e.g., n = 3 compared to n = 4), or objective

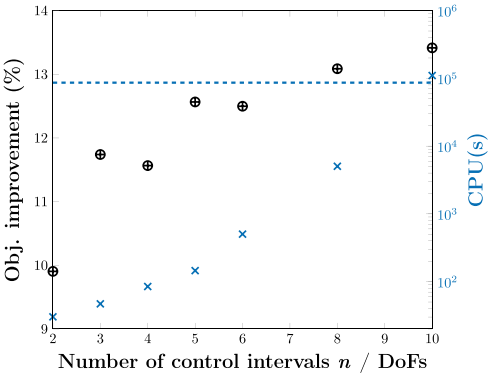


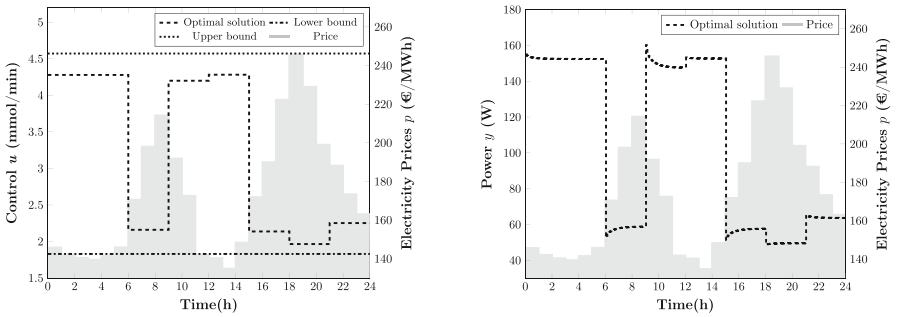
Fig. 4 Economic objective value improvement compared to non-scheduling operation (Crossed circles: ⊕) and computational time scaling (Crosses: ×) for different numbers of control discretization intervals n or DoFs. The dashed line (- - -) corresponds to the 12-hour computational time-limit

value differences that fall into the optimality gap used. The above indicate the effect of the control parametrization given a certain price profile on the optimization result.

We additionally note, in Fig. 4, a marginal (smaller than the optimality gap of the global optimization) change in the absolute objective values for $n > 5$. This behavior is a common feature of control parametrization methods (Chachuat 2007) and, although unknown a priori, it may indicate that minimal improvement can be expected from further grid resolution increase ($n > 10$). In the case of insignificant objective improvement via increasing grid resolution after a relatively coarse grid, global optimization considering coarser grids may potentially lead to superior solutions compared to local optimization for highly dense grids, in particular, if set-point and price changes can be aligned in time. However, there is no guarantee that low-resolution global optimization can outperform high-resolution local optimization in terms of cost objective value given the high computational complexity of the former compared to the latter. Although a comparison of global and local dynamic scheduling is out of the scope of this manuscript, further improvement by exploiting global optimization and better allocation of the available control discretization points is later investigated by applying the AGRAGO.

Figure 4 additionally presents the expected exponential scaling of the solution times with the number of equidistant piecewise constant control segments for each control variable $\hat{\mathbf{u}}$, $\hat{\mathbf{w}}$ (or equivalently with the total number of DoFs). This observation is associated with the scaling of B&B algorithms with the number of variables, as described by Chachuat et al. (2006) and later shown by Kappatou et al. (2022) for global dynamic optimization with HW models. Following the hourly discretization of the day-ahead market, a scheduling horizon of one day would be split into at least 24 control intervals. However, according to the scaling noted in Fig. 4, and considering the extrapolation of an exponential curve fitting the depicted data to $n = 24$, the global solution of a dynamic problem such as (2a)–(2d) would be intractable (CPU times in the order of centuries).

In order to acquire a computationally tractable solution, we consider the highest possible grid resolution that leads to computational times of less than a 12-hour threshold, which is the time frame between the price update (12 o'clock of the previous day) and the schedule implementations (24 o'clock of the day of implementation). Thus, the



(a) Molar throughput (b) Power consumption

Fig. 5 Offline optimal scheduling profiles. The dashed lines correspond to the control and power value, while the shaded areas correspond to the electricity price

problem is solved for a control parametrization of 8 piecewise constant equidistantly sampled segments, as the maximum resolution for tractable computations, and the optimal schedule (control and power profiles) of Fig. 5 is retrieved. The corresponding solution time amounts to approximately 1.5h, while offline scheduling results in a 13.1% drop in electricity-related costs compared to process operation under a fixed production rate.

From Fig. 5, we note an optimal power consumption fluctuation overall corresponding to the price changes. Additionally, we observe that the coarser time discretization of the controls compared to that of the price leads to suboptimal allocation of the production (control u) at some increments of the scheduling horizon. This allocation may result from two consecutive increments with high corresponding price deviation, which share the same control choice. For instance, a higher production during the 7th compared to the 10th hour would be economically more beneficial due to the higher electricity price of the latter. However, equidistant sampling does not account for this effect due to the coupling of the 7th, 8th and 9th, and the 10th, 11th and 12th hour, respectively, in two single control choices. These results indicate possible further revenues in the case of better assignment of the DoFs when considering the control grid.

We observe from Fig. 5 the dynamic power consumption response to the price fluctuation and draw attention to offsets in the power consumption (e.g., 10th hour). Such offsets might be closely related to production or safety constraints (Tsay et al. 2019), like for example limitations to high power consumption due to inefficiencies resulting from high voltage application (Swiegers et al. 2021), which would not be necessarily satisfied if process dynamics were disregarded. This type of constraints is not included in this case study, which follows a conservative process operation, but reflects to the general case applicable of the scheduling method presented. Thus, disregarding the inherent process dynamics to simplify the optimization problem (Seborg et al. 2016) is not advised due to the dynamic infeasibility of the suggested schedule, in the context of process constraint violation and unexpected costs (Simkoff and Baldea 2020). Another point that confirms the need to account for the process dynamics is the unrealistic cost calculations resulting from global optimization for scheduling under static behavior

(with an ANN model trained to the data for steady-state operation). In that case, steady state optimization could result to more than 15% underestimation of the costs resulting from scheduling, while additionally there would be no consistency to whether a cost over- or underestimation results from simulating the process under static behavior (see Table 4 in Appendix B). We note that more significant errors may result in case of faster set-point changes (e.g., real-time markets) (Baldea and Harjunkoski 2014; Dias et al. 2018).

Overall, Fig. 4 reveals the computational burden of the applied optimization method and provides an indication for acquiring high-quality solutions using global optimization and a relatively coarse control grid. Figure 5 highlights the additional possible benefits from a better allocation of the available DoFs with the aim of obtaining a better economic objective value in the same amount of computational time (or same n). In this direction, we explore further economic benefits by applying our grid refinement algorithm AGRAGO to the scheduling problem presented in Sect. 3.

4 Wavelet-based adaptive grid refinement algorithm for global optimization (AGRAGO)

4.1 Theoretical background

In this chapter we present the theory and application of a wavelet-based adaptive grid refinement algorithm used for the optimal grid allocation of the available DoFs for the solution of a dynamic optimization problem to global optimality. Such a wavelet-based refinement algorithm is first introduced by Schlegel et al. (2005) for the solution of dynamic optimization problems under continuous-time dynamics with single shooting and a piecewise constant control vector parametrization. The algorithm is an iterative procedure that includes: the wavelet transform of the solution of the previous iteration (lower resolution control grid), the wavelet coefficient insertion or activation ("grid point insertion"), and deletion or deactivation ("grid point elimination", or "grid point deletion"), and the re-solving of the optimization problem in the parameterized control vector space for the updated suggested control discretization scheme.

Schäfer et al. (2020a) apply the algorithm of Schlegel et al. (2005) to the scheduling of a compressed energy storage plant (CESP) considering quasi-stationary, instead of dynamic models, and on discrete, instead of continuous time in combination with a linear mapping of one DoF to multiple intervals with similar price values. For the solution of the problem, a reduced-spaced scheduling formulation is applied together with global optimization techniques resulting in a series of feasible near-optimal solution points further used as initial points of a local search for the full-dimensionality MINLP. The solution points correspond to nonincreasing—guaranteed for coefficient insertion only—objective values. The method demonstrates significant computing time reductions compared to the optimization problem with full temporal dimensionality.

Schäfer et al. (2020b) further extend their approach to account for the optimization problem's solution directly in the wavelet coefficients space and guarantee a unique allocation of DoFs to control intervals. This change in optimization variables implies an assignment procedure in which the wavelet coefficients of the unused wavelet basis

functions (*inactive* wavelet coefficients) are set to zero. The assignment replaces the representation of multiple control intervals of the previous works with one single DoF. The method also suggests using the Lagrangian multipliers of the constraints related to the wavelet coefficients truncation (Binder et al. 2000).

This replaces the addition of optimization variables to the set of the *active* coefficients based on a quantitative yet pure heuristic measure with a quantitative measure of the marginal objective value improvements, which enhances the algorithm's performance, especially in case of a high effect of ramping constraints on the optimal schedules. The method is applied in combination with a price-based linear mapping of a reduced set of new DoF onto the original DoFs to the scheduling of a cryogenic air separation unit (ASU). The problem is solved in a reduced space using global optimization techniques (B&B algorithm). The results show significant computational time savings compared to solution approaches of full temporal dimensionality, highlighting a linear scaling of the solution times with the horizon length. We note, though, that in the works of Schäfer et al. (2020b) and Schäfer et al. (2020a), the benefit of integrating the refinement procedure and the price-based linear mapping cannot be distinguished. Thus, there is no explicit measure of the effect of applying such a wavelet-based refinement strategy considering the solution time requirements.

4.2 Algorithm description

The high computational demands of global dynamic scheduling and the need for finer time series discretization that fully exploits the price fluctuations (as described in Sect. 3.2) lead to the application of a grid time series refinement algorithm. This algorithm can produce the same or even better results (objective value) using fewer DoFs, thus, speeding up the solution process by suggesting the use of a nonequidistantly sampled control grid. We herein propose an adaptive grid refinement algorithm for global optimization (AGRAGO) based on previous works presented in Sect. 4.1 (Schlegel et al. 2005; Schäfer et al. 2020a, b) to solve the global dynamic scheduling problem of the first electrolyzer of the acid recovery process. AGRAGO integrates the wavelet transformation of a current control grid solution of Schlegel et al. (2005) with the DoFs deletion criterion of Schäfer et al. (2020a) and the insertion criterion of Schäfer et al. (2020b) based on the Lagrangian multipliers associated with the deactivation of coefficients. In contrast to the work of Schäfer et al. (2020b), in AGRAGO, the optimization problem is solved in the original parameterized controls' space instead of the wavelet coefficients' space, as explained in Sect. 4.2.2. The refinement process of AGRAGO is automated and integrated into MAiNGO (Bongartz et al. 2018), where additional possibilities of insertion and deletion criteria presented in the previous works under discussion are added for use.

AGRAGO is an iterative procedure in which global optimization of the problem for a previously suggested control grid is performed, followed by post-processing of the optimization results. The latter is used to suggest a new control grid aiming to improve the optimization result in terms of objective value and which represents the control grid used for the next iteration. More specifically, the post-processing involves the suggestion of new wavelet coefficients, the wavelet transformation of the prior

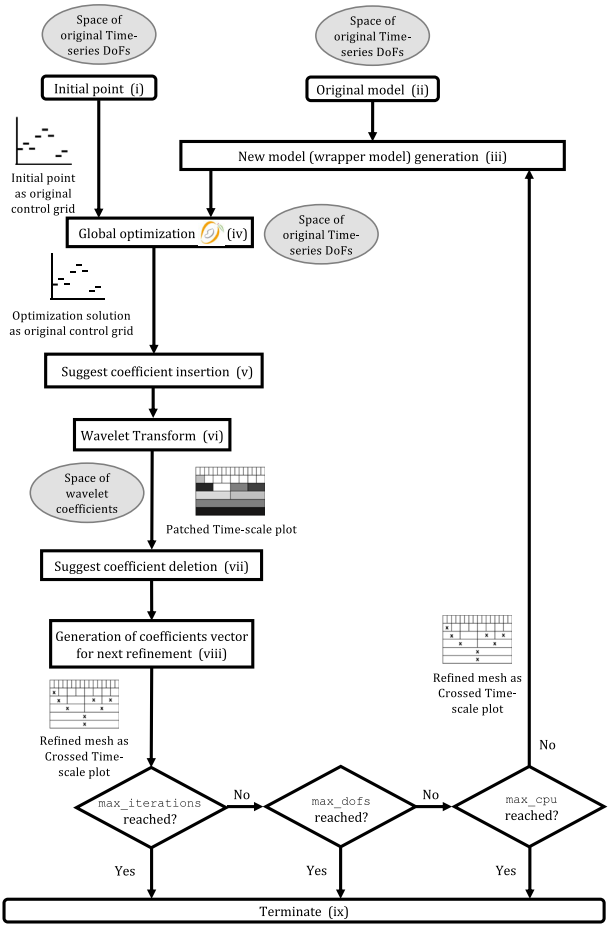


Fig. 6 Graphical representation of AGRAGO. The Crossed Time-scale plots represent the selection of wavelet coefficients (x: *active* coefficient), the Patched Time-scale plots correspond to the enumerated coefficients occurring after each global optimization step, and the wavelet transform of the resulting solution, and the original control grid points describe the current piecewise constant control solution. The algorithm terminates heuristically

solution, and the deletions of already *active* coefficients. The algorithm terminates heuristically. A graphical representation of AGRAGO is given in Fig. 6. The main steps of AGRAGO are further described in Sects. 4.2.1–4.2.4.

4.2.1 Wrapper-model generation

The first step of AGRAGO is the generation of a wrapper-model (new optimization problem) (Fig. 6 (iii)) which derives from the original optimization problem (Fig. 6 (ii)) with all piecewise constant control DoFs that correspond to the finest resolution permitted, and from the suggested new grid in the form of a wavelet coefficients' vector. This vector determines the equality constraints related to the activation and

deactivation of a coefficient. Further analysis of the wrapper model generation is given through a comprehensive example in Appendix C.1.

4.2.2 Global optimization

Next, global optimization (Fig. 6 (iv)) is performed in the original parameterized controls' space considering an initial point (Fig. 6 (i)) if available. Global optimization takes place in a reduced space using MAiNGO (Bongartz et al. 2018), and the optimization result is a new vector of piecewise-constant controls given the previously suggested discretization scheme.

It is highlighted that AGRAGO is designed to solve the optimization problem in the original space of the piecewise-constant parameterized control variables (as also done by Schlegel et al. (2005)), rather that mapping the problem to a different space in which the control parameters are expressed as wavelet coefficients (as in Schäfer et al. (2020b)). This is because the solution of the problem in the wavelet coefficients' space results in a higher computational effort. This is closely related to the enlargement of the search space, due to less tight box constraints of the wavelet coefficients compared to the original piecewise constant control variables. More specifically, the linear transformation of the DoFs of the piecewise control variables to wavelet coefficients entails that the value of each wavelet coefficient depends on the value of the coefficient that belongs to the lower level of refinement (e.g., see Eqs. (6)–(13) in Appendix C.1). This is also the case for the bounds of the wavelet coefficient variables. Given that, and since the bounds of the wavelet coefficients need to allow a switch from the lower bound interval value (corresponding to the lowest possible original control value) to the upper bound interval value (corresponding to the highest possible original control value), a cumulative effect takes place that enlarges the search space and increases the computational time of the global optimization step. A simple numerical example is given in in Appendix C.1. Results of performing global optimization in the original control variable and the wavelet coefficient space considering the same nonuniform grid support this view (see Sect. 4.3).

The application of AGRAGO to problem (2a)–(2d) suggests the addition of a set of equality constraints related to the deactivation of wavelet coefficients (Eq.(3e)), so that the optimal scheduling problem is reformulated as presented in Eqs. (3a)–(3e).

The wavelet transform requires that the number of control intervals is a power of two. More specifically, due to the hourly sampled electricity prices, a scheduling horizon $T = 2^N$, $N \in \{1, 2, 3, \ldots\}$ that represents the maximum possible resolution at the last refinement level is expected. This implies that if a non-power-of-two scheduling horizon (e.g., $T = 24$) is to be implemented, either the highest resolution should be selected as a power-of-two (e.g., 32) with the control and price-change points not coinciding (e.g., price-change each hour and control-change each 45min), or the non-power-of-two scheduling horizon (e.g., $T = 24$) should be decomposed into two (or more) subhorizons with a power-of-two number of intervals each (e.g., $T_1 = 16$ and $T_2 = 8$) (Schäfer et al. 2020a). For each subhorizon, the problem is solved separately. In the former case, the selection of the power-of-two highest resolution is related to the discrete-time dynamics, which may add a restriction (the minimum time-step is that of the state discretization) to choosing between different possible asynchronous

price and control step-change points. This asynchronous change implies that the price fluctuations are not fully exploited, and thus, further improvement in the objective could be expected in a different case. In the latter case, we detect additional DoFs related to the number and order of the subhorizons, as well as the adjustment of the constraints linked to the entire horizon. Such a constraint can be, for instance, a production goal over the entire scheduling horizon which should be broken up into the different subhorizons (see Eq. 3d). This prohibits the fully flexible operation of the process to which scheduling is applied; thus, additional improvement in the objective could be anticipated.

To tackle the above-mentioned issues, we suggest considering a non-power-of-two *highest possible resolution* for the refinement, which enables a synchronous price and control step change. It should be noted that since AGRAGO can be applied to the solution of different (global) dynamic optimization problems, the price set-point changes of the scheduling problem represent any set-point changes of a problem parameter (e.g., price profiles, set-points of control-tracking problems). The strategy suggests that the entire scheduling horizon (e.g., $T = 24$) is split into a user-specified number of subhorizons, namely *batches* (e.g., 3), which all consist of the same power-of-two number of grid intervals (e.g., 8). The refinement procedure remains the same, with the only difference being that the wavelet transform is performed separately for each solution of each *batch*. The graphical representation of a Crossed-Time-scale plot for an example of applying the *batch* approach is given in Fig. 14 (Appendix C.2).

The integration of the *batches* concept to the application of AGRAGO to problem (2a)–(2d) considering $q \in \mathbb{Z}$ *batches* of 2^ρ, $\rho \in \mathbb{Z}$ intervals each such that the *highest possible grid resolution* is $n = q \cdot 2^\rho$ leads to the reformulation of problem (2a)–(2d) which we present as problem (3a)–(3e):

$$\min_{\hat{\mathbf{u}},\hat{\mathbf{w}}} T \cdot \sum_{t=1}^{n} \sum_{s=1}^{k} p_{s+k(t-1)} \cdot f_{\mathrm{W}}\left(\mathbf{c}_{\mathrm{D}} \hat{\mathbf{x}}_{s+k(t-1)} + d_{\mathrm{D}} f_{\mathrm{H}}(\hat{u}_t)\right) \tag{3a}$$

$$s.t. \quad \hat{w}_t = f_{\mathrm{H}}(\hat{u}_t) \qquad \forall t \in \{1, 2, \ldots, n\} \tag{3b}$$

$$u^{\mathrm{L}} \leq \hat{u}_t \leq u^{\mathrm{U}}, \quad w^{\mathrm{L}} \leq \hat{w}_t \leq w^{\mathrm{U}} \qquad \forall t \in \{1, 2, \ldots, n\} \tag{3c}$$

$$m_{\mathrm{a}} \leq T \cdot \sum_{t=1}^{n} \hat{u}_t \tag{3d}$$

$$\mathbf{H}_{2^{\rho_{i,1:2^\rho}}} \hat{\mathbf{w}}_j = \mathbf{0} \qquad \forall j, i : \nu_{j,i} = 0 \tag{3e}$$

where $\mathbf{H}_{2^\rho}$ is the $2^\rho \times 2^\rho$ Haar matrix, $\boldsymbol{\nu} = \begin{pmatrix} \boldsymbol{\nu}_{j=1} \\ \vdots \\ \boldsymbol{\nu}_{j=q} \end{pmatrix} \in \mathbb{R}^n$ is the wavelet coefficient matrix, with $\boldsymbol{\nu}_j = \begin{pmatrix} \nu_{j,i=1} \\ \vdots \\ \nu_{j,i=2^\rho} \end{pmatrix} \in \mathbb{R}^{2^\rho}$, given the Haar transformation $\boldsymbol{\nu}_j = \mathbf{H}_{2^\rho} \hat{\mathbf{w}}_j$ and

$\hat{\mathbf{x}}$ is the solution of:

$$\hat{\mathbf{x}}_{s+k(t-1)} = \mathbf{A}_{\mathrm{D}}\hat{\mathbf{x}}_{s+k(t-1)-1} + \mathbf{b}_{\mathrm{D}}\hat{w}_t \qquad \forall t \in \{1, 2, \ldots, n\}, \forall s \in \{1, 2, \ldots, k\}$$
$$\hat{\mathbf{x}}_0 = \mathbf{0}$$

the controls and states are parameterized as:

$$\hat{\mathbf{x}}_{s+k(t-1)} = \begin{pmatrix} \hat{x}_{1,s+k(t-1)} \\ \vdots \\ \hat{x}_{l,s+k(t-1)} \end{pmatrix} \in \mathbb{R}^l,$$

$$\hat{\mathbf{u}} = \begin{pmatrix} \hat{u}_1 \\ \vdots \\ \hat{u}_n \end{pmatrix} \in \mathbb{R}^n, \qquad \hat{\mathbf{w}} = \begin{pmatrix} \hat{\mathbf{w}}_1 \\ \vdots \\ \hat{\mathbf{w}}_q \end{pmatrix} = \begin{pmatrix} \hat{w}_1 \\ \vdots \\ \hat{w}_n \end{pmatrix} \in \mathbb{R}^n$$

4.2.3 Wavelet coefficient insertion and deletion

The following step of AGRAGO is the analysis of the optimization solution of the previous refinement step in order to insert (Fig. 6 (v)) new DoFs for the next iteration of higher grid resolution using the "Sensitivity-based refinement" insertion technique (Schäfer et al. 2020b). For the coefficients' insertion, the Lagrangian multipliers related to the equality constraints describing the equality between neighboring controls (or, equivalently, the deactivation of a wavelet coefficient) are collected and used to suggest the addition of wavelet coefficients. The "Sensitivity-based refinement" insertion technique corresponds to the insertion method developed by Binder et al. (2000) and further extended by Schäfer et al. (2020b). This method is based on evaluating the gradients of the Lagrangian function of the dynamic optimization problem with respect to new prospective parameterization functions. Binder et al. (2000) focus on both wavelet and piecewise constant ("single scale") parameterization functions and highlight the possibility of a nonuniform piecewise constant parameterization of different interval lengths. The discretized in-time problem under consideration is solved for a certain control parametrization. The resulting optimal solution is used for retrieving the gradients of the Lagrangian at this certain optimal point. Those grid points, whose corresponding parameterization functions exhibit large Lagrangian gradients, are added to the control mesh. Schäfer et al. (2020b) have later implemented this approach, considering wavelet basis parameterization functions, through a post-processing step in order to obtain dual variables (Lagrangian multipliers) of the constraints fixing *inactive* wavelet coefficients to zero as local sensitivities of the optimal solution with respect to activating the coefficient. With this strategy, the *inactive* coefficients whose deactivation constraints have the largest absolute dual variable (marginal effect of constraint relaxing) are added in the next iteration. After the post-processing step, the reduced-space problem is solved in MAiNGO (Bongartz et al. 2018).

The approach of Schäfer et al. (2020b) is applied in this work for a grid refinement under piecewise constant control parametrization, except that objective sensitivities are considered as the Lagrangian multipliers of the equality constraints describing

the relation between the values of the discretized control variables. The correlation between the equality constraints of the deactivation of a wavelet coefficient and the piecewise constant control parametrization is given through an example of problem (3a)–(3e) in Appendix C.1. The number of wavelet coefficients suggested to be added at the next refinement step using the "Sensitivity-based refinement" method depends on a user-specified number with a default value of 1. The minimum increase in coefficients of one control variable that can occur between the refinement steps is specified by this number. The maximum increase is this number incremented by one, and it is independent of the coefficients deletion.

Other available insertion methods presented in literature propose the use of the Euclidean norm of the vector of the refined "boundary coefficients" (wavelet coefficients with at least one *inactive* corresponding coefficient on the next higher-resolution level) ("boundary signal") for the addition of new DoFs (Schlegel et al. 2005; Schäfer et al. 2020a). The "Sensitivity-based refinement" is another heuristic process for coefficient insertions which, in contrast to the other methods, provides a quantifiable indicator of the anticipated objective values improvement. Therefore, the latter is selected to be integrated into AGRAGO with a comparison over different coefficient insertion methods falling out of the scope of the present work.

After the suggestion of wavelet coefficients' insertion is conducted, a wavelet transformation (Fig. 6 (vi)) of the previous solution is performed, and the calculated wavelet coefficients are used to identify whether a particular coefficient should be deleted (Fig. 6 (vii)). A coefficient deletion occurs when the coefficient falls below a specific bound, as given in Eq.(4). This bound is a user-specified fraction (ϵ) of the norm of the wavelet coefficients vector (Schlegel et al. 2005).

$$|v_{j,i}| < \epsilon \times |\boldsymbol{v}| \tag{4}$$

Coefficient deactivation considers all variables that follow the same discretization scheme, or *synchronized* variables (Appendix C.1), and the algorithm reassures that when a coefficient is deactivated, the corresponding one of the *synchronized* variable is also deactivated. The need to account for the synchronization of the optimization variables results from the problem formulation of the dynamic optimization problem (detailed analysis by Kappatou et al. (2022)), which requires exposing two control-variable time-series to the optimizer that are linked with an equality constraint (see Eq. 3b). It is highlighted, in this regard, that AGRAGO allows users to decide whether to permit the reactivation of coefficients deleted in the previous refinement step. If the deletion criterion is not met for any currently *active* coefficient, and, thus, no deletion takes place between two recursive refinement steps, the insertion only increases the resolution of the initial mesh. More specifically, no shifting of the initial piecewise constant control grid takes place, but instead, new discretization points are added on top. This results in a non-decreasing objective between the two refinement steps, which can be guaranteed to the entity of the refinement steps in case no deletion occurs throughout the entire refinement procedure.

After the suggestion for coefficient insertion and deletion, the proposed coefficients are added and deleted (Fig. 6 (viii)) so that at the end of the step, a newly refined grid

is suggested in the form of a binary vector, where 0 corresponds to *inactive* and 1 to *active* wavelet coefficients.

4.2.4 Termination

The algorithm terminates heuristically after a user-specified number of refinement iterations (`max_iterations`) is executed, after a user-specified number of DoF (*active* wavelet coefficients) (`max_dofs`) is reached, if the refinement procedure exceeds a user-specified runtime limit (`max_cpu_time`), or if the *highest possible grid resolution* is reached. This preselected computational time limit is problem-dependent. For instance, in the case of scheduling of processes that take part in a day-ahead electricity market, the optimization solution should be obtained in less than 12h, as described in Sect. 3.2. Since the number of wavelet coefficients increases after each refinement step (unless differently selected by manipulating the user-specified number for insertion accordingly (see Sect. 4.2.3)), the computational time related to each optimization step is larger than that of the previous refinement iteration. Taking that into consideration and in order to avoid a refinement that will exceed the computational time limit, we terminate the refinement procedure when the total CPU time of the global optimization of all refinement steps conducted increased by the CPU time of the last refinement step's global optimization exceeds this preselected time limit. The final solution suggested by AGRAGO is the one resulting from the refinement step that leads to the lowest objective value.

4.3 Scheduling results using AGRAGO

AGRAGO is used for the solution of the scheduling problem (3a)–(3e) in a reduced space using MAiNGO (Bongartz et al. 2018) for the same setup as the one described in Sect. 3.2, and the same optimization parameters given in Table 3 (Appendix B). The control grid is parameterized using $n = 24$, and $q = 3$ (AGRAGO: $n = 24$), and $n = 32$, and $q = 1$ (AGRAGO: $n = 32$), and the following settings are used for the refinement: $\epsilon = 5 \cdot 10^{-3}$, `max_iterations`=12, `max_dofs`= 48, and `max_cpu`= 43,200 sec. In order to obtain the Lagrangian multipliers related to the equality constraints of Eq. (3e), which are used for the wavelet coefficient insertion step, we conduct a local search a posteriori beginning from the feasible point that gives the lowest objective using IPOPT (Wächter and Biegler 2006).

In addition to the problem (3a)–(3e) for $n = 24$ (AGRAGO: $n = 24$), and $n = 32$ (AGRAGO: $n = 32$), we solve the initial problem (2a)–(2d) applying the coefficient insertion approach of Schäfer et al. (2020b), the deletion criterion of Eq. (4) and optimization in the space of the wavelet coefficients for $n = 32$ (as described in Sect. 4.2.2) (refinement - optimization in wavelet c. space: $n = 32$). The three solution approaches are applied; the computational time scaling over the number of DoFs is illustrated in Fig. 7, and the objective value improvement over the number of DoFs is given in Fig. 8. The results of the equidistant sampling for the solution of problem (2a)–(2d) are added for comparison.

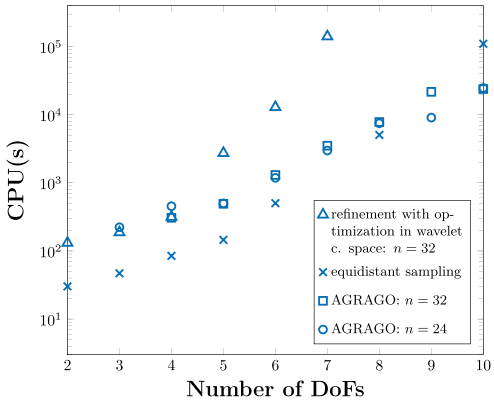


Fig. 7 Scaling of computational time with the number of DoFs for different methods of refinement. Triangles (△): Solution of problem (2a)–(2d) as presented by Schäfer et al. (2020b) (and deletion criterion of Eq.(4)) for different numbers of control intervals. Crosses (×): Solution of problem (2a)–(2d) using equidistant grid sampling for different numbers of control intervals n. Squares (□): Solution of problem (3a)–(3e) for $n = 32$, $q = 1$ and different numbers of control intervals. Circles (○): Solution of problem (3a)–(3e) for $n = 24$, $q = 3$ and different numbers of control set-point intervals

In Fig. 7, a general exponential trend of the CPU time with the number of DoFs, similar to Fig. 4 is observed. A steeper scaling, and generally larger CPU times are highlighted for optimization in the wavelet space (solution times increased by orders of magnitude) explained by the enlarged search space, as mentioned in Sect. 4.2.1, which supports the transition to performing optimization in the initial piecewise constant control space. We additionally note that AGRAGO has the same or better computational performance compared to equidistant sampling for more than 8 DoFs; however, for fewer DoFs, equidistant sampling outperforms AGRAGO, which arises from the different problem formulations implemented in both cases. As explained in Appendix C.1, AGRAGO uses the optimization variables at the *highest possible grid resolution* and equality constraints to indicate the deactivation of a coefficient. This formulation increases the computational expenses for small numbers of DoFs; the margin by which AGRAGO outperforms equidistant grids is expected to increase with increasing n values. However, this worse runtime performance for small numbers of DoFs is overall outperformed by the improved DoF allocation, as indicated in Fig. 8 for AGRAGO: $n = 24$.

From Fig. 8, we note the effect of the DoF allocation using AGRAGO on the objective value improvement. AGRAGO: $n = 32$ results in similar objective values to the case of the equidistant control grid. No practical improvement for a small number of DoFs, in this case, can be explained by the asynchronous price and control set-point change imposed by the non-power-of-two *highest grid resolution*, which endorses the application of the *batch* concept for scheduling under the price profile considered. On the other hand, AGRAGO: $n = 24$ results in a better allocation of the available DoFs compared to the equidistant sampling, as explained by the higher objective value improvements for a higher-than-4 number of DoFs, which supports the application of AGRAGO for an improved allocation of the available DoFs of the scheduling problem.

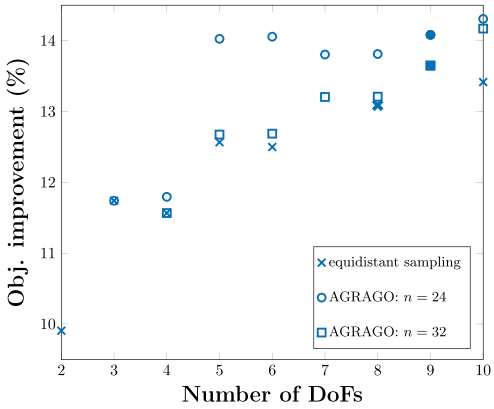


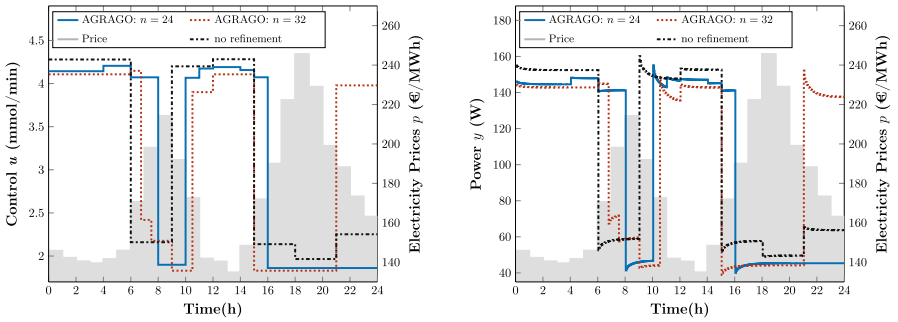
Fig. 8 Economic objective value improvement compared to non-scheduling operation with the number of DoFs for different methods and highest grid resolution of refinement. Crosses (×): Solution of problem (2a)–(2d) using equidistant grid sampling for different numbers of control intervals n. Circles (○): Solution of problem (3a)–(3e) for $n = 24$, $q = 3$ and different numbers of control set-point intervals. Squares (□): Solution of problem (3a)–(3e) for $n = 32$, $q = 1$ and different numbers of control intervals. The points with filled marks correspond to AGRAGO's suggestion and the point with the bold cross mark correspond to the solution of equidistant sampling

Particularly for 5 DoFs only, a 1.7% cost improvement compared to the equidistant sampling is detected.

We additionally observe, from Fig. 8, that with the progressive refinement of the control grid, the objective value overall drops (over-parametrization is avoided according to Binder et al. (2000) in contrast to uniform sampling) at a decreasing rate as it asymptotically approaches the value of the solution considering the *highest control grid resolution* allowed (24 or 32); equivalently the objective improvement increases with a decreasing rate. It is highlighted, though, that the objective is not necessarily improving from one refinement iteration to the next one due to the arbitrary wavelet coefficients' deletion (not present here) or due to solution changes that fall into the global optimality gap (1%) used in the B&B.

Especially for AGRAGO: $n = 24$, a marginal additional improvement ($< 1\%$) is detected for a number of DoFs higher than 4. We arbitrarily consider that since 6 recursive refinement steps result in absolute objective improvements smaller than the optimality gap used, the objective at the *highest possible grid resolution* is practically reached using only 5 DoFs (such stabilization is not identified for AGRAGO: $n = 32$); thus, refinement up to that point is assumed to provide the maximum benefit possible. This is a valuable feature of the application of AGRAGO, as it indicates the possibility of obtaining, with a significantly smaller number of DoFs, a solution with the same objective value as if a substantially higher resolution was used.

Figure 8, presents the solutions suggested by AGRAGO. For AGRAGO: $n = 24$ (●), the algorithm terminates after 7 iterations and 6 hours of CPU time (CPU of recursive optimization steps only), while for AGRAGO: $n = 32$ (■), the algorithm terminates after 6 iterations and 8 hours of CPU time. The highest economic improvement results from applying AGRAGO: $n = 24$ for 9 control intervals and amounts



(a) Molar throughput (b) Power consumption

Fig. 9 Offline optimal scheduling profiles for the grid suggestion of different control grid refinement approaches. Constant line (——): Solution of problem (3a)–(3e) for $n = 24$, $q = 3$, and 9 DoFs. Dotted line (····): Solution of problem (3a)–(3e) for $n = 32$, $q = 1$ and 9 DoFs. Dash-dotted line (-·-·-): Solution of problem (2a)–(2d) for $n = 8$ control intervals

to 14.1%. More specifically, economic savings of scheduling by applying AGRAGO: $n = 24$ considering the computational time limit are improved by 1.1% compared to the application of equidistant sampling for $n = 8$ (✖), as presented in Sect. 3.2. However, AGRAGO: $n = 32$ does not achieve acquiring a solution with significantly higher objective improvement (13.6%) compared to the equidistant sampling in the specified time limit.

Figure 9 gives the different control and power profiles that correspond to the solution points provided by the two AGRAGO refinement methods and a comparison to the optimal profile without refinement (see Sect. 3.2). The corresponding Crossed-Time scale plots are presented in Appendix C.4. In the case of AGRAGO: $n = 32$, we notice that most control set-point changes coincide with a price change regardless of the non-hourly discretized control grid, denoting that synchronous shifts are favored. However, single price time intervals share two contrasting control choices (e.g., 11th h), indicating additional potential savings from the application of the *batch* concept. According to the profile of the AGRAGO: $n = 24$ solution, and for 9 DoFs, it is more beneficial to drive the process to minimum production during the high and medium prices and more flexibly adjust the production during the hours of lower prices to fulfill the minimum daily production demand. It is not guaranteed, though, that this is the single optimal allocation of the available DoFs, as heuristic criteria are used for the selection of the grid, or that the profile resulting from the next refinement suggestion follows the same pattern.

5 Conclusions

Global optimization of dynamic scheduling problems aims at attaining the globally optimal schedule desired for maximizing economic benefits related to flexible dynamic process-operation problems that are nonconvex. In this work, we perform experimentation and data-driven dynamic modeling of an electrolyzer's anode chamber which is part of an electricity intensive process for the recovery of succinic acid (Schröder et al. 2022) to identify a HW model which describes the process operation. We then apply an approach for global optimization with HW models (Kappatou et al. 2022) considering discrete-time dynamics to solve the multimodal offline dynamic scheduling problem to global optimality accounting for the process integration in a day-ahead electricity market. We demonstrate the substantial economic benefits of such integration while guaranteeing the global optimality of the solution attained and the dynamic feasibility of the resulting schedule. We additionally highlight the computational limitations of the optimization method applied due to the unfavorable scaling of the solution time with the number of parameterized control segments related to the "curse of dimensionality" phenomenon. This poses a computational-time limit for the grid selection for the application of the method to large-scale problems to achieve real-time applicable computation and implementation of the schedule. While the worst-case exponential computational scaling of the B&B algorithm cannot be tackled, we propose a systematic reallocation of the available DoFs; this DoF reallocation results in schedules whose quality is indistinguishable (in terms of objective value) from other schedules computed with a higher number of DoFs and in higher solution times.

We propose an algorithm for dimensionality reduction of global dynamic optimization problems formulated in a reduced space to achieve substantial computational savings considering a trade-off between a fine control grid resolution and the exploitation of global methods for dynamic scheduling. The adaptive grid refinement algorithm for global optimization (AGRAGO) proposed follows a systematic wavelet-based analysis of the process control time series. The algorithm aims to iteratively suggest a refined nonequidistantly sampled control grid considering a piecewise constant parametrization of the control variables (Schlegel et al. 2005), alleviating limitations for the selection of optimization horizon. The algorithm performs global optimization in the parameterized control space for a current parametrization proceeded by a post-processing step that suggests insertion and deletions of piecewise constant control parameters. AGRAGO pinpoints refinement of the control grid based on sensitivity information from the Lagrangian multipliers (Schäfer et al. 2020b) related to the equality of neighboring controls. Then, the algorithm applies a wavelet transform of the control variables corresponding to a current solution, followed by a truncation of the wavelet transform. AGRAGO can be used as a control grid refinement method for global (as well as local) dynamic (or quasi-stationary) optimization.

We apply AGRAGO to the global dynamic scheduling of the electrochemical process considering both synchronous (by introducing the *batch* concept) and asynchronous control set-point and price change, and it is compared to a counterpart considering optimization in the space of wavelet coefficients. AGRAGO leads to significant computational time reduction compared to the counterpart, and AGRAGO of synchronous change results in a more effective DoF allocation. The latter addition-

ally prompts a quick convergence to a practically not further improved solution by additional refinement, highlighting the computational gain from optimizing directly over the piecewise control variables and supporting the flexibility of the optimization horizon selection to account for the synchronization of the price and control set-points. AGRAGO leads to a 14.1% cost reduction compared to nominal production, which is by 1.1% improved compared to scheduling under equidistant control grid sampling for the case study and time limitations under consideration. AGRAGO further provides automation in the decision of the DoFs allocation and the first high-quality solution in dramatically less time.

Future research should include additional experimentation and training of a more accurate either black- or grey-box process model. Moreover, it should consider the process units following the electrochemical cell at issue for the systematic flexibilization of the entire succinic separation process to reach additional cost reduction. Further investigation should focus on the process integration into an intraday electricity market by potentially following an economic nonlinear model predictive control (eNMPC) paradigm (Baldea and Harjunkoski 2014). Additionally, improvements in the scaling behavior of the global dynamic optimization approach should be investigated regarding methods for the construction of tighter relaxations for the lower bounding problem (e.g., (Castro 2015)) or the ODE system (e.g., (Sahlodin and Chachuat 2011)). A systematic comparison of first, the global and local optimization solution approaches, and second, the application of model linearization techniques (Kelley et al. 2018a) over the solution of the nonconvex problem to global optimality considering solution superiority and computational limitations for a methodical selection between them should be further analyzed. Furthermore, AGRAGO could be extended to allow for different discretization schemes of the control variables that are not synchronized and applied to different problems of steady-state or dynamic scheduling and control tracking problems, also incorporated in a moving horizon scheme (NMPC). Finally, the heuristic criteria of the DoF selection of recursive AGRAGO steps could be replaced by an optimization step while still accounting for the computational limitations of the applied method.

Supplementary information

Supplementary material associated with this article can be found, in the online version.

Acknowledgements This work was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation)—333849990/GRK2379 (IRTG Hierarchical and Hybrid Approaches in Modern Inverse Problems). Funding for the experimental setup and the conduct of experiments was provided by the Kopernikus project SynErgie by the Federal Ministry of Education and Research (BMBF) and by the project management organization Projektträger Jülich (PtJ). The authors gratefully acknowledge Julius Beerwerth for implementing a proof-of-concept of the refinement strategy and Jan Christoph Schulze, Susanne Sass, Dominik Bongartz and Chrysoula Dimitra Kappatou for valuable discussions and advice throughout the development of this work.

Author Contributions Open Access funding enabled and organized by Projekt DEAL. Chrysanthi Papadimitriou: Conceptualization (lead), Methodology (lead: data processing, data-driven modeling, global dynamic scheduling problem formulation, refinement algorithm), Formal analysis (lead), Software (lead: data-driven modeling implementation, scheduling problem implementation, adaptations to refinement algorithm implementation, implementation of refinement algorithm and scheduling code), Investigation (lead: data processing, data-driven modeling, global dynamic scheduling problem formulation, refinement algorithm), Data Curation, Writing - Original Draft (lead), Visualization (lead). Tim Varelmann: Methodology (supporting: refinement algorithm), Software (supporting: refinement algorithm implementation), Investigation (supporting: refinement algorithm), Writing—Review & Editing (supporting). Christian Pascal Schröder: Investigation (supporting: experimental setup and data acquisition), Formal analysis (supporting: experimental data acquisition), Writing—Original Draft (supporting: main contribution in section 2.1.), Visualization (supporting: data presentation in supplementary material). Andreas Jupke: Writing—Review and Editing (supporting: main contribution in chapter 2.1.), Supervision (supporting: experimental work), Funding acquisition (supporting: experimental work). Alexander Mitsos: Conceptualization (supporting), Writing—Review and Editing (lead), Supervision (lead), Funding acquisition (lead).

Declarations

Conflict of interest There are no conflicts to declare.

A Hammerstein–Wiener model fitting parameters & results

A.1 Data preprocessing

See Fig. 10.

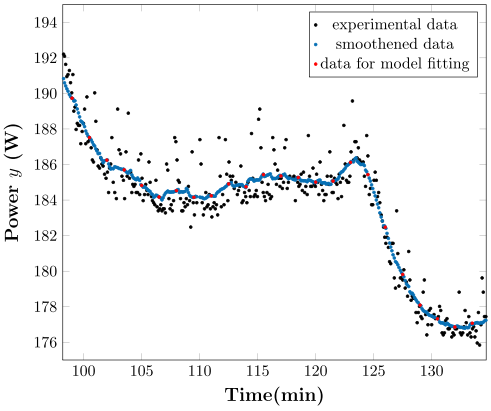


Fig. 10 Experimental, smoothened (SMA) & uniformly sampled (linear interpolation between the SMA) working data for a single set-point change of experiment 1

A.2 Identified model parameters

The parameters of the LTI system described in Eq. (1) expressed in discrete time as:

$$
\begin{aligned}
&w(t) = f_{\mathrm{H}}(u(t)), \quad \forall t \in [\,0, T\,] \\
&\mathbf{x}(t + \mathrm{T_s}) = \mathbf{A}_{\mathrm{D}}\mathbf{x}(t) + \mathbf{b}_{\mathrm{D}}w(t), \quad \forall t \in [\,0, T\,] \\
&z(t) = \mathbf{c}_{\mathrm{D}}\mathbf{x}(t) + d_{\mathrm{D}}w(t), \quad \forall t \in [\,0, T\,] \\
&\mathbf{x}(0) = \mathbf{0} \\
&y(t) = f_{\mathrm{W}}(z(t)), \quad \forall t \in [\,0, T\,]
\end{aligned}
$$

are given by:

$$
\mathbf{A}_{\mathrm{D}} = \begin{bmatrix} 2.0951 & -1.1263 & -0.01923 & 0.10064 \\ 1.6642 & -0.6744 & -0.0161 & 0.0605 \\ 0.2500 & 0 & 0 & 0 \\ 0 & 0.0313 & 0 & 0 \end{bmatrix}, \quad \mathbf{b}_{\mathrm{D}} = \begin{bmatrix} 0.6660 \\ 0.2500 \\ 0 \\ 0 \\ 0 \end{bmatrix},
$$

$$
\mathbf{c}_{\mathrm{D}} = \begin{bmatrix} -0.0738 & 0.0763 & -0.0644 & 0.2419 \end{bmatrix}, \quad d_{\mathrm{D}} = 1
$$

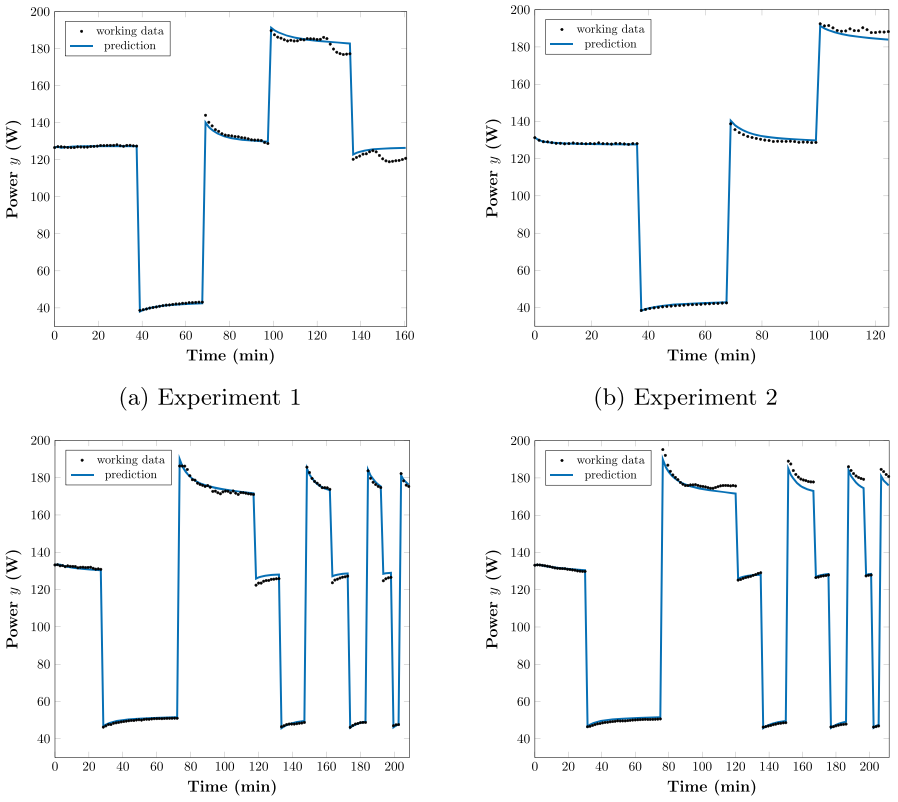
The input and output nonlinearities as presented in Eq. (1) are given, respectively by: $f_{\mathrm{H}}(u(t)) = 0.1837u^4(t) - 2.1060u^3(t) + 8.3186u^2(t) - 11.5370u(t) + 1.0399$, with $t \in [0, 1440]$, $u \in [1.830, 4.572]$, and $f_{\mathrm{W}}(z(t)) = 4.9567z^2(t) + 45.1037z(t) + 129.5721$, with $t \in [0, 1440]$

Table 1 Hammerstein–Wiener model hyperparameters and fitting results

H Polyn. degree	W Polyn. degree	Number of zeros of LTI	Number of poles of LTI	Delay of LTI	Training experiments 1/2/3 1-*NMSE* (%)	Test experiment 1 1-*NMSE*(%)	nAIC	BIC
4	2	3	4	–	94.6/96.4/96.5	98.7	1.6	1513.2

A.3 Fitting to data

See Fig. 11.



(a) Experiment 1

(b) Experiment 2

(c) Experiment 3

(d) Experiment 4

Fig. 11 HW model fitting to working data

B Optimization parameters and results

See Tables 2, 3 and 4.

Table 2 Objective value results for local optimization using miltistart with IPOPT 3.12.12 (Wächter and Biegler 2006)

Number of intervals (n)	Min Objective value	Max Objective value
4	41.19*	44.85
10	40.05*	43.76
24	39.85	41.63

Objective values noted with * additionally correspond to the global optimization solution obtained after convergence of the lower to the upper bound

Table 3 Global optimization parameters

Optimization parameter	Value
T(min)	1440
$n \cdot k$	960
m_a(mol min^{-1})	4.600
u^L(mol min^{-1})	1.830
u^U(mol min^{-1})	4.572
w^L (−)	−3.062
w^U (−)	1.149
Relative optimality tolerance	1×10^{-2}
Upper bounding problem solver	IPOPT 3.12.12 (Wächter and Biegler 2006)

Table 4 Objective value (Obj.) results of global optimization considering static process behavior and objective value results of steady-state simulation considering the GDO schedules

Number of intervals (n)	Obj. deviation from GDO solution for steady-state optimization (%)	Obj. deviation from GDO solution for steady-state simulation of the GDO solutions (%)
1	−14.7	−14.7
2	−7.7	−4.9
3	−14.2	−1.5
4	−16.7	−4.6
5	−15.2	−3.2
6	−14.6	−4.7
8	−15.4	−4.5
10	−14.9	3.1
12	−15.1	−4.5

The results are given in the form of deviations from the corresponding GDO objective values

For the selection of the optimality gap an a priori analysis takes place, which considers the scaling of the solution algorithm with the optimality tolerance used to obtain tractable solutions given the 12-hour computational time limitation, together with a an improvement from the AGRAGO application at least in the same range (here 1–2%) as the optimality gap (here 1%). Such an analysis is based on the B&B performance exhibiting quick convergence of the upper bound and gradual increase of the lower bound; thus no further improvement in the objective value result is expected from further decreasing the optimality tolerance for the B&B.

C Model generation and optimization in AGRAGO

C.1 Wrapper-model generation

The following example explains the wrapper model generation (Fig. 6 (iii)) step of AGRAGO. We assume piecewise constant control vector parametrization and a *highest control grid resolution* of 8 intervals. In case all wavelet coefficients are *active*, the control profile of the variable $\hat{\mathbf{w}}$ is that of Fig. 12. The Crossed Time-scale plot, which indicates by color (see Fig. 12) if a wavelet coefficient is *active* (x) or *inactive* (blank) is given in Fig. 13 (a) and the Patched Time-scale plot which illustrates by color intensity the absolute values of the coefficients is presented in Fig. 13b.

The wavelet coefficients of the vector $\boldsymbol{\nu}$ (for a single *batch* $j = 1$ and, thus j is disregarded, see problem (3a)–(3e)) are calculated using an 8-point Haar matrix. For illustrative purposes, the elements of $\boldsymbol{\nu}$ in Fig. 12 and Eqs. (5)–(13) result from an un-normalized 8-point Haar matrix, while in AGRAGO a normalized Haar matrix of the Haar wavelet transform is used.

$$\boldsymbol{\nu} = \mathbf{H_8}\hat{\mathbf{w}}, \quad \mathbf{H_8} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & 1 & -1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & -1 & -1 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix} \tag{5}$$

$$\nu_0 = w_0 + w_1 + w_2 + w_3 + w_4 + w_5 + w_6 + w_7 \tag{6}$$

$$\nu_1 = w_0 + w_1 + w_2 + w_3 - w_4 - w_5 - w_6 - w_7 \tag{7}$$

$$\nu_2 = w_0 + w_1 - w_2 - w_3 \tag{8}$$

$$\nu_3 = w_4 + w_5 - w_6 - w_7 \tag{9}$$

$$\nu_4 = w_0 - w_1 \tag{10}$$

$$\nu_5 = w_2 - w_3 \tag{11}$$

$$\nu_6 = w_4 - w_5 \tag{12}$$

$$\nu_7 = w_6 - w_7 \tag{13}$$

For the solution of a dynamic optimization problem, as the scheduling problem (2a)–(2d), in the space of the parameterized control variables considering the wavelet coefficients insertions and deletions proposed, we select to use the full control grid resolution and additional equality constraints in case a coarser resolution is applied. In the example presented here, the total number of DoFs is 16 (8 $\hat{w}$ and 8 $\hat{u}$), and each time a wavelet coefficient is *inactive* (zero value), an equality constraint is added. The additional equality constraints corresponding to a wavelet coefficient deactivation result from setting the given wavelet coefficients of Eqs. (6)–(13) to zero. It is underlined here that even though in the example of Eqs. (5)–(13) an un-normalized Haar matrix is used, the resulting equality constraints are the same as for using the

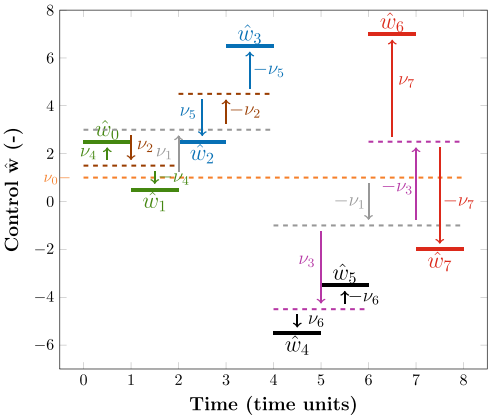
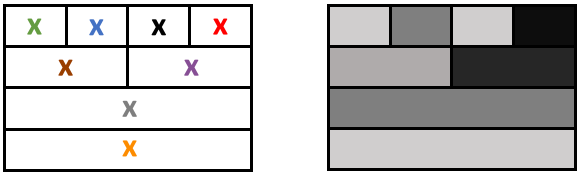


Fig. 12 Solution profile of the wavelet coefficients vector $\boldsymbol{\nu}$ for a highest resolution enabled of 8 intervals calculated as a wavelet representation of the control variable vector $\hat{\mathbf{w}}$. In the present case, all 8 wavelet coefficients are *active* (non-zero)



(a) Crossed Time-scale plot (b) Patched Time-scale plot

Fig. 13 Crossed **a** and Patched **b** Time-scale plot of the wavelet transformation of Fig. 12

normalized Haar matrix. An example of the zero-level coefficient expression using the normalized Haar matrix for comparison with Eq. 6 is given by:

$$\bar{\nu}_0 = \frac{1}{\sqrt{8}} \times (w_0 + w_1 + w_2 + w_3 + w_4 + w_5 + w_6 + w_7) \tag{14}$$

For suggesting a wavelet coefficient insertion when the Sensitivity-based refinement method is applied, we use the Lagrangian multipliers of the constraints related to the wavelet coefficients truncation, which in this case are equality constraints on the controls $\hat{w}$ resulting from Eqs. (6)–(13), instead of equality constraints on the coefficients ν, as presented in Schäfer et al. (2020b). The motivation behind this choice, as explained in Sect. 4.2.2, lies in the high computational expenses that derive from the enlargement of the search space in case the problem is solved in the wavelet coefficients' space. Characteristic is the example of the box constraints of the piecewise control parameters w_i which lie in the range of $w^U - w^L$ compared to the wavelet coefficients parameters, e.g., $\bar{\nu}_0$ which lies in the range of $\sqrt{8} \times (w^U - w^L)$ according to Eq. (14).

It should be pointed out here that control variables $\hat{\mathbf{w}}$ and $\hat{\mathbf{u}}$ follow the same piecewise constant discretization scheme, as they are not independent of each other, but instead, they are linked with an equality constraint. These two control variables are henceforth called *synchronized*. As $\hat{\mathbf{w}}$ and $\hat{\mathbf{u}}$ are synchronized, the addition of equality constraints related to the truncation of a wavelet coefficient refers to only one of the two control variables, in this case, $\hat{\mathbf{w}}$.

C.2 *Batch*-splitting of scheduling horizon

See Fig. 14.

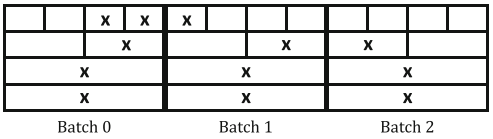


Fig. 14 Example of splitting a 24h scheduling horizon in three 8h *batches* as a Crossed-Time-scale plot for one of the *synchronized* control variables

C.3 AGRAGO parameters' selection

The user-specified parameters of AGRAGO can be selected using the following guidelines:

- `max_cpu`: It is related to the time-limitations of acquiring the optimal solution based on the optimization problem solved. For the case studies of this work, we consider the 12-hour threshold presented in Sect. 3.2.
- ϵ: It is a fraction of the norm of the wavelet coefficients' vector at a certain refinement iteration. Its values should be between 0 (no active coefficient is deleted) and 1 (all active coefficients are deleted). The value of the norm, as well as the coefficients depend on the range of the control values. Thus, no specific rule applies to which the value of ϵ should be. It is advisable, though that a value is selected through trial and error so that deletions are realized (high enough ϵ value), but avoiding continuous and unnecessary ones that may result in overall worse objective values (too high ϵ value).
- `max_dofs`: It is higher than 1, and at maximum equal to the number of DoFs at maximum resolution, which is a safe value to be used, as applied for the presented case studies. If there is no clear restriction or high effect of the time limitations (e.g., in case of steady-state global optimization with a few DoFs), a stricter value can be used.
- `max_iterations`: It can be higher or lower than the highest possible grid resolution depending on the initial number of active coefficients used and the

Table 5 AGRAGO parameters

AGRAGO parameter	Value for AGRAGO: $n = 24$	Value for AGRAGO: $n = 32$
`max_cpu`(sec)	43,200	43,200
ϵ	$5 \cdot 10^{-3}$	$5 \cdot 10^{-3}$
`max_dofs`	48	64
`max_iterations`	12	12

number of deleted and inserted coefficients throughout the refinement steps. The selection is similar to that of the `max_dofs`.

When applying AGRAGO to a global dynamic optimization problem as the one presented in our manuscript, we expect termination due to the CPU time limit reached, as the main computational restriction of our problem. For a different use of AGRAGO (e.g., global steady-state optimization), however, any of the other stopping criteria could be responsible for the algorithm termination.

C.4 AGRAGO applications

See Figs. 15 and 16.

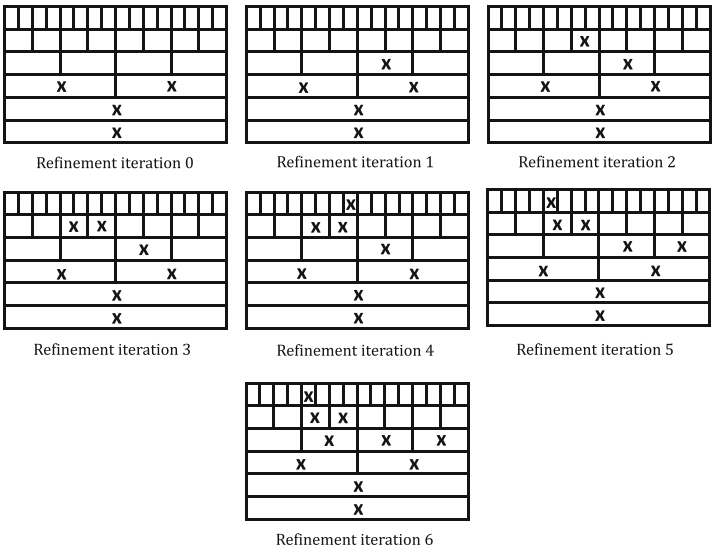


Fig. 15 AGRAGO: $n = 32$. Crossed-time scale plot for the different refinement steps for variable $\hat{\mathbf{w}}$

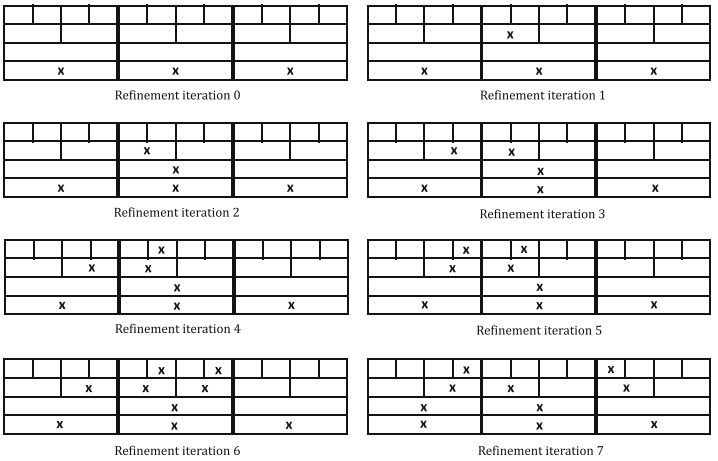


Fig. 16 AGRAGO: $n = 24$. Crossed-time scale plot for the different refinement steps for variable $\hat{\mathbf{w}}$

References

Baldea M, Harjunkoski I (2014) Integrated production scheduling and process control: A systematic review. Comput Chem Eng 71. https://doi.org/10.1016/j.compchemeng.2014.09.002

Bechthold I, Bretz K, Kabasci S et al (2008) Succinic acid: a new platform chemical for biobased polymers from renewable resources. Chem Eng Technol 31(5):647–654. https://doi.org/10.1002/ceat.200800063

Bellman R (1957) Dynamic programming, 1st edn. Princeton University Press, Princeton

Bhatia T, Biegler LT (1996) Dynamic optimization in the design and scheduling of multiproduct batch plants. Ind Eng Chem Res 35. https://doi.org/10.1021/ie950701i

Binder T, Cruse A, Villar CA, et al (2000) Dynamic optimization using a wavelet based adaptive control vector parameterization strategy. Comput Chem Eng 24. https://doi.org/10.1016/S0098-1354(00)00357-4

Bongartz D, Najman J, Sass S, et al (2018) MAiNGO-McCormick-based algorithm for mixed-integer nonlinear global optimization. http://permalink.avt.rwth-aachen.de/?id=729717

Bozell JJ, Petersen GR (2010) Technology development for the production of biobased products from biorefinery carbohydrates-the US Department of Energy's "Top 10" revisited. Green Chem 12:539–554. https://doi.org/10.1039/B922014C

Bradfield MFA, Mohagheghi A, Salvachúa D et al (2015) Continuous succinic acid production by actinobacillus succinogenes on xylose-enriched hydrolysate. Biotechnol Biofuels 8(1):181. https://doi.org/10.1186/s13068-015-0363-3

Cai H, Li P, Su C et al. (2018) Double-layered nonlinear model predictive control based on Hammerstein–Wiener model with disturbance rejection. Measurement and Control (United Kingdom) 51. https://doi.org/10.1177/0020294018785500

Cartwright JHE, Piro O (1992) The dynamics of Runge–Kutta methods. Int J Bifurcation Chaos 2. https://doi.org/10.1142/s0218127492000641

Castro PM (2015) Tightening piecewise McCormick relaxations for bilinear problems. Comput Chem Eng 72. https://doi.org/10.1016/j.compchemeng.2014.03.025

Chachuat B (2007) Nonlinear and dynamic optimization: From theory to practice. Tech. rep., Laboratoire d'Automatique, École Polytechnique Fédérale de Lausanne

Chachuat B, Singer AB, Barton PI (2006) Global methods for dynamic optimization and mixed-integer dynamic optimization. Ind Eng Chem Res 45. https://doi.org/10.1021/ie0601605

Chen W, Shao Z, Biegler LT (2014) A bilevel NLP sensitivity-based decomposition for dynamic optimization with moving finite elements. AIChE J 60. https://doi.org/10.1002/aic.14339

Chen W, Wang K, Shao Z, et al. (2012) Chapter 11: Direct transcription with moving finite elements. https://doi.org/10.1137/9781611972252.ch11
Chen W, Ren Y, Zhang G, et al. (2019) A simultaneous approach for singular optimal control based on partial moving grid. AIChE J 65. https://doi.org/10.1002/aic.16584
Chu Y, You F (2012) Integration of scheduling and control with online closed-loop implementation: Fast computational strategy and large-scale global optimization algorithm. Comput Chem Eng 47. https://doi.org/10.1016/j.compchemeng.2012.06.035
Cuthrell JE, Biegler LT (1987) On the optimization of differential-algebraic process systems. AIChE J 33. https://doi.org/10.1002/aic.690330804
Dias LS, Ierapetritou MG (2019) Optimal operation and control of intensified processes—challenges and opportunities. Curr Opin Chem Eng 25. https://doi.org/10.1016/j.coche.2018.12.008
Dias LS, Pattison RC, Tsay C et al (2018) A simulation-based optimization framework for integrating scheduling and model predictive control, and its application to air separation units. Comput Chem Eng 113. https://doi.org/10.1016/j.compchemeng.2018.03.009
Du J, Park J, Harjunkoski I, et al (2015) A time scale-bridging approach for integrating production scheduling and process control. Comput Chem Eng 79. https://doi.org/10.1016/j.compchemeng.2015.04.026
Flores-Tlacuahuac A, Grossmann IE (2006) Simultaneous cyclic scheduling and control of a multiproduct CSTR. Ind Eng Chem Res 45. https://doi.org/10.1021/ie051293d
Floudas CA, Gounaris CE (2009) A review of recent advances in global optimization. J Glob Optim 45. https://doi.org/10.1007/s10898-008-9332-8
Floudas CA, Lin X (2005) Mixed integer linear programming in process scheduling: Modeling, Algorithms, and Applications. Ann Oper Res 139. https://doi.org/10.1007/s10479-005-3446-x
Gausmann M, Kocks C, Doeker M et al (2020) Recovery of succinic acid by integrated multi-phase electrochemical ph-shift extraction and crystallization. Sep Purif Technol 240(116):489. https://doi.org/10.1016/j.seppur.2019.116489
Gausmann M, Kocks C, Pastoors J et al (2021) Electrochemical pH-T-Swing Separation of Itaconic Acid for Zero Salt Waste Downstream Processing. ACS Sustain Chem Eng 9(28):9336–9347. https://doi.org/10.1021/acssuschemeng.1c02194
Fraunhofer Institute for Solar Energy Systems ISE (2023) Energy charts. https://energy-charts.info. Accessed 25 March 2023
Kappatou CD, Bongartz D, Najman J, et al. (2022) Global dynamic optimization with Hammerstein–Wiener models embedded. J Glob Optim 84. https://doi.org/10.1007/s10898-022-01145-z
Kelley MT, Pattison RC, Baldick R et al. (2018) An efficient MILP framework for integrating nonlinear process dynamics and control in optimal production scheduling calculations. Comput Chem Eng 110. https://doi.org/10.1016/j.compchemeng.2017.11.021
Kelley MT, Pattison RC, Baldick R et al. (2018) An MILP framework for optimizing demand response operation of air separation units. Appl Energy 222. https://doi.org/10.1016/j.apenergy.2017.12.127
Kelley MT, Tsay C, Cao Y, et al. (2022) A data-driven linear formulation of the optimal demand response scheduling problem for an industrial air separation unit. Chem Eng Sci 252. https://doi.org/10.1016/j.ces.2022.117468
Kondili E, Pantelides CC, Sargent RW (1993) A general algorithm for short-term scheduling of batch operations-I.MILP formulation. Comput & Chem Eng 17. https://doi.org/10.1016/0098-1354(93)80015-F
McCormick GP (1976) Computability of global solutions to factorable nonconvex programs: Part I—Convex underestimating problems. Math Program 10. https://doi.org/10.1007/BF01580665
Mitsos A, Chachuat B, Barton PI (2009) McCormick-based relaxations of algorithms. SIAM J Optim 20. https://doi.org/10.1137/080717341
Mitsos A, Asprion N, Floudas CA et al. (2018) Challenges in process optimization for new feedstocks and energy sources. Comput Chem Eng 113. https://doi.org/10.1016/j.compchemeng.2018.03.013
Mohammadi E, Montazeri-Gh M (2015) A New Approach to the Gray-Box Identification of Wiener Models With the Application of Gas Turbine Engine Modeling. J Eng Gas Turbines Power 137. https://doi.org/10.1115/1.4029170
Pattison RC, Touretzky CR, Johansson T et al. (2016) Optimal Process Operations in Fast-Changing Electricity Markets: Framework for Scheduling with Low-Order Dynamic Models and an Air Separation Application. Ind Eng Chem Res 55. https://doi.org/10.1021/acs.iecr.5b03499
Paulus M, Borggrefe F (2011) The potential of demand-side management in energy-intensive industries for electricity markets in Germany. Appl Energy 88. https://doi.org/10.1016/j.apenergy.2010.03.017

Sahlodin AM, Chachuat B (2011) Convex/concave relaxations of parametric ODEs using Taylor models. Comput Chem Eng 35. https://doi.org/10.1016/j.compchemeng.2011.01.031
Salvachúa D, Mohagheghi A, Smith H et al (2016) Succinic acid production on xylose-enriched biorefinery streams by Actinobacillus succinogenes in batch fermentation. Biotechnol Biofuels 9(1):28–9347. https://doi.org/10.1186/s13068-016-0425-1
Schaber SD, Scott JK, Barton PI (2019) Convergence-order analysis for differential-inequalities-based bounds and relaxations of the solutions of ODEs. J Glob Optim 73. https://doi.org/10.1007/s10898-018-0691-5
Schäfer P, Schweidtmann AM, Lenz PH et al (2020) Wavelet-based grid-adaptation for nonlinear scheduling subject to time-variable electricity prices. Comput Chem Eng 132(106):598. https://doi.org/10.1016/j.compchemeng.2019.106598
Schäfer P, Schweidtmann AM, Mitsos A (2020) Nonlinear scheduling with time-variable electricity prices using sensitivity-based truncations of wavelet transforms. AIChE J 66(e16):986. https://doi.org/10.1002/aic.16986
Schlegel M, Marquardt W (2006). Detection and exploitation of the control switching structure in the solution of dynamic optimization problems. https://doi.org/10.1016/j.jprocont.2005.06.008
Schlegel M, Stockmann K, Binder T et al. (2005) Dynamic optimization using adaptive control vector parameterization. Comput & Chem Eng 29. https://doi.org/10.1016/j.compchemeng.2005.02.036
Schröder C, Gausmann M, Jupke A (2022) Lastflexible Extraktion biobasierter Carbonsäuren. In: Energieflexibilität in der deutschen Industrie–Band 2: Markt- und Stromsystem, Managementsysteme und Technologien energieflexibler Fabriken. Fraunhofer Verlag, Stuttgart, chap C.2, p 297–316, https://doi.org/10.24406/publica-258
Schweidtmann AM, Mitsos A (2019) Deterministic global optimization with artificial neural networks embedded. J Optim Theory Appl 180(3):925–948. https://doi.org/10.1007/s10957-018-1396-0
Scott JK, Barton PI (2013) Improved relaxations for the parametric solutions of ODEs using differential inequalities. J Global Optim 57. https://doi.org/10.1007/s10898-012-9909-0
Seborg DE, Edgar TF, Mellichamp DA et al (2016) Process dynamics and control. Wiley, Hoboken
Simkoff JM, Baldea M (2020) Stochastic scheduling and control using data-driven nonlinear dynamic models: Application to Demand Response Operation of a Chlor-Alkali Plant. Ind Eng Chem Res 59. https://doi.org/10.1021/acs.iecr.9b06866
Singer AB (2004) Global dynamic optimization. Ph.D. thesis, Massachusetts Institute of Technology
Singer AB, Barton PI (2004) Global solution of optimization problems with parameter-embedded linear dynamic systems. J Optim Theory Appl 121. https://doi.org/10.1023/B:JOTA.0000037606.79050.a7
Singer AB, Barton PI (2006) Global optimization with nonlinear ordinary differential equations. J Glob Optim 34. https://doi.org/10.1007/s10898-005-7074-4
Song H, Lee SY (2006) Production of succinic acid by bacterial fermentation. Enzyme Microb Technol 39(3):352–361. https://doi.org/10.1016/j.enzmictec.2005.11.043
Song Y, Khan KA (2022) Optimization-based convex relaxations for nonconvex parametric systems of ordinary differential equations. Math Program 196. https://doi.org/10.1007/s10107-021-01654-x
Suresh V, Chaudhuri D (1993) Dynamic scheduling-a survey of research. Int J Prod Econ 32. https://doi.org/10.1016/0925-5273(93)90007-8
Swiegers GF, Terrett RNL, Tsekouras G et al. (2021) Correction: The prospects of developing a highly energy-efficient water electrolyser by eliminating or mitigating bubble effects. Sustain Energy Fuels 5. https://doi.org/10.1039/d1se90039k
Thuy NTH, Boontawan A (2017) Production of very-high purity succinic acid from fermentation broth using microfiltration and nanofiltration-assisted crystallization. J Membr Sci 524:470–781. https://doi.org/10.1016/j.memsci.2016.11.073
Tsay C, Kumar A, Flores-Cerrillo J et al (2019) Optimal demand response scheduling of an industrial air separation unit using data-driven dynamic models. Comput Chem Eng 126. https://doi.org/10.1016/j.compchemeng.2019.03.022
Tsoukalas A, Mitsos A (2014) Multivariate McCormick relaxations. J Glob Optim 59. https://doi.org/10.1007/s10898-014-0176-0
Đukić S, Sarić A (2012) Dynamic model reduction: An overview of available techniques with application to power systems. Serbian J Electr Eng 9. https://doi.org/10.2298/sjee1202131d
Vasantharajan S, Biegler LT (1990) Simultaneous strategies for optimization of differential-algebraic systems with enforcement of error criteria. Comput Chem Eng 14. https://doi.org/10.1016/0098-1354(90)85005-U

Velez S, Maravelias CT (2013) Multiple and nonuniform time grids in discrete-time MIP models for chemical production scheduling. Comput Chem Eng 53. https://doi.org/10.1016/j.compchemeng.2013.01.014

Wächter A, Biegler LT (2006) On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. Math Program 106. https://doi.org/10.1007/s10107-004-0559-y

Yang D, Jin T, Wu Z, et al (2008) Dual stepsize explicit numerical integration method and applications. https://doi.org/10.1109/PES.2008.4596126

Yang Z, Li K, Foley A et al. (2014) Optimal scheduling methods to integrate plug-in electric vehicles with the power system: A review. https://doi.org/10.3182/20140824-6-za-1003.01804

Zhang Q, Grossmann IE (2016) Enterprise-wide optimization for industrial demand side management: Fundamentals, advances, and perspectives. Chem Eng Res Des 116. https://doi.org/10.1016/j.cherd.2016.10.006

Zhang Q, Sundaramoorthy A, Grossmann IE et al. (2016) A discrete-time scheduling model for continuous power-intensive process networks with various power contracts. Comput Chem Eng 84. https://doi.org/10.1016/j.compchemeng.2015.09.019

Zhao J, Li S (2019) Mars atmospheric entry trajectory optimization with maximum parachute deployment altitude using adaptive mesh refinement. Acta Astronaut 160. https://doi.org/10.1016/j.actaastro.2019.03.027