Computers and Chemical Engineering 193 (2025) 108917

Contents lists available at ScienceDirect

Computers and Chemical Engineering

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

Review

Digital twins in process engineering: An overview on computational and numerical methods

Luisa Peterson [a], Ion Victor Gosea [a], Peter Benner [a,b], Kai Sundmacher [a,b,*]

[a] Max Planck Institute for Dynamics of Complex Technical Systems, Sandtorstr. 1, 39106 Magdeburg, Germany
[b] Otto von Guericke University Magdeburg, Germany

ARTICLE INFO

Keywords:
Digital twin
Process- and chemical engineering
First-principle models
Data-driven models
Scientific machine learning
Reduced-order modeling

ABSTRACT

A digital twin (DT) is an automation strategy that combines a physical plant with an adaptive real-time simulation environment, where both are connected by bidirectional communication. In process engineering, DTs promise real-time monitoring, prediction of future conditions, predictive maintenance, process optimization, and control. However, the full implementation of DTs often fails in reality. To address this issue, we first examine various definitions of DTs and its core components, followed by a review of its current applications in process engineering. We then turn to the computational and numerical challenges for building the simulation environments necessary for successful DTs implementation

Contents

* Corresponding author.
E-mail addresses: peterson@mpi-magdeburg.mpg.de (L. Peterson), gosea@mpi-magdeburg.mpg.de (I.V. Gosea), benner@mpi-magdeburg.mpg.de (P. Benner), sundmacher@mpi-magdeburg.mpg.de (K. Sundmacher).

https://doi.org/10.1016/j.compchemeng.2024.108917
Received 12 February 2024; Received in revised form 4 November 2024; Accepted 5 November 2024
Available online 16 November 2024

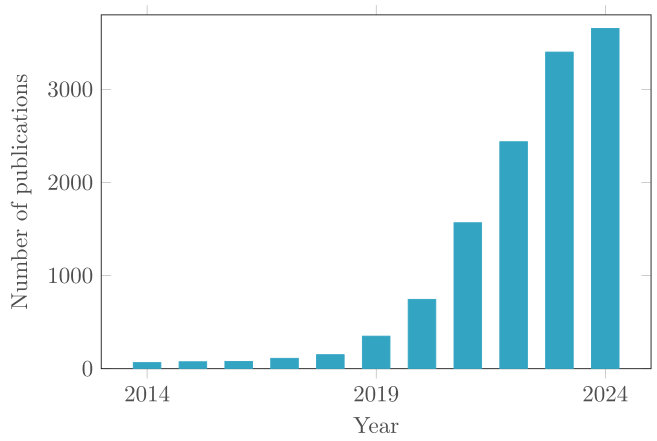

Fig. 1. Number of research works on DTs (published in English) according to the Web of Science over the last 10 years (2014 to October 2024).

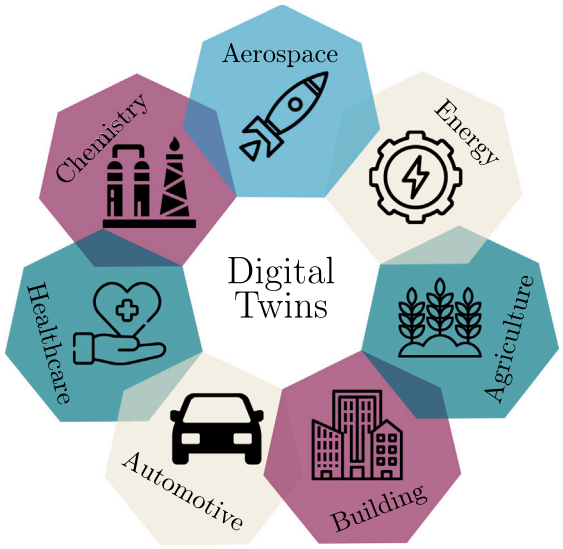

Fig. 2. A schematic representation of the diverse applications of DTs across multiple industries. This illustration emphasizes the cross-industry nature of DTs and the potential for synergy and knowledge transfer between different fields.

1. Introduction

In recent years, industrial sectors have been under increasing pressure to optimize operations, improve efficiency, and reduce downtime. The rise of automation technologies, along with advances in data analytics and simulation, has paved the way for innovative solutions to these challenges. Among these, the digital twin (DT) concept has emerged as a key strategy for improving process control, operational insight, and risk management.

The growing interest in DTs is reflected in the increasing number of publications, as shown in Fig. 1, which tracks English-language publications on DTs from 2014 to 2024. The data show a steady increase in research activity, and projections suggest that this upward trend will continue. Beyond academia, DTs is gaining traction in various industries, including chemistry, healthcare, and energy, as illustrated in Fig. 2. This widespread adoption highlights the potential for knowledge transfer and synergies between different sectors (History Digital Twin, 2023).

Despite the growing attention, the full potential of DTs remains largely unrealized. Many of the proposed benefits are still difficult to achieve in practice. Bridging this gap requires closer collaboration between academia and industry to align theoretical advances with real-world applications (National Academy of Engineering and National Academies of Sciences, Engineering, and Medicine, 2023).

This paper attempts to explore the current state of DTs in process engineering, with a focus on computational and numerical aspects. Following this introduction, Section 2 discusses key definitions and enablers of DTs. Section 3 reviews the existing literature on DTs in process engineering. Next, Section 4 outlines numerical methods for model development with examples from process engineering. Finally, Section 5 summarizes key findings and suggests directions for future research.

2. Definitions and key enablers

This section presents various definitions of DT from the existing literature. Then, the key components that must come together to form a DT are presented.

2.1. Definitions of digital twins

Despite being a prominent technological trend, the definition of DTs varies widely both between and within disciplines. In addition, the applications of DTs vary significantly across the various stages of product lifecycle management, including product design, engineering, procurement, construction, and plant operations (Örs et al., 2020). To highlight this inconsistency, Wright and Davidson (2020) humorously reference a quote from the fictional character Humpty Dumpty in Lewis Carroll's Through the Looking Glass: *"When I use a word, it means whatever I want it to mean"*.

To provide a basic overview, we will explore the most important definitions of DTs below. For a detailed historical overview of the development and definitions of DTs, see History Digital Twin (2023). The concept of DTs was introduced by Michael Grieves during an industry presentation in 2002. Grieves defined DTs as consisting of three essential components: (1) a *physical entity in physical space*, (2) a *virtual entity in virtual space*, and (3) a *link between the physical and virtual entities that enables the exchange of data and information*. Importantly, according to this definition, a DT is not just the virtual part, but a system that integrates all three elements. Nevertheless, DT is often used as a buzzword for (very detailed) models. Clarifying this distinction is essential to avoid mischaracterizing DTs as a fleeting trend (Wright and Davidson, 2020).

Building on Grieves' original definition, Jones et al. (2020) propose an expanded, standardized nomenclature for the DT concept. They define the *physical entity* as the object, the *physical environment* as its conditions, and the *physical processes* as the operations performed within this environment. Accordingly, they introduce *virtual entity*, *virtual environment*, and *virtual processes* to match the naming of the physical counterparts. Interactions are enabled by *physical-virtual connections* and *virtual-physical connections*, where a complete circuit requires both connections. If the virtual-physical connection is missing, the concept is referred to as a *digital shadow*, whereas if both connections are missing, it is called a *digital model* (Kritzinger et al., 2018).

Newrzella et al. (2021) introduce cross-sectoral terms with a five-dimensional DT concept based on Grieves' three dimensions. These include (1) the *physical entity*, (2) its *characteristics*, (3) the *form of communication*, (4) the *virtual entity*, and (5) the *user-specific value added*. Various user-specific values, ranging from process design and optimization, risk reduction and maintenance (Zalai et al., 2013), to product quality prediction (Chen et al., 2020) are reported (Mabkhot et al., 2018).

Talkhestani et al. (2019) compare different DT definitions, focusing on architectures. Their analysis reveals common requirements, including a realistic digital representation of a physical entity with

all relevant information. This information must include all operational characteristics, as well as organizational and technical details from the design phase. The authors further emphasized the need for permanent synchronization and interfaces between DTs for co-simulation and data exchange.

A broader definition focusing on the digital entity is provided by Glaessgen and Stargel (2012) and Tao et al. (2018). They describe it as an integrated, multi-physics, multi-scale, and probabilistic simulation of a complex product that uses the best available physical models and real-time sensor data to mirror the life cycle of its physical counterpart. In this view, a DT serves as a digital representation of a physical asset, process, or system, with real-time synchronization being critical. In addition, the authors emphasize the role of artificial intelligence (AI) and machine learning in improving decision-making and asset performance within DT systems. The level of sophistication required for the digital entity is a point of debate in the literature. While some advocate maximum detail and complexity, others argue for simplicity to avoid excessive system complexity (National Academy of Engineering and National Academies of Sciences, Engineering, and Medicine, 2023; Ferrari and Willcox, 2024; Willcox and Segundo, 2024).

Several DT subcategories have been proposed, but these categories are not widely accepted. Firstly, an *operational* DT includes data management, process modeling, optimization, and production planning (Örs et al., 2020). In contrast, an *executable* DT is an autonomous model integrated into operational environments that represents only those aspects relevant to its specific application (Hartmann and van der Auweraer, 2022; Eppinger et al., 2021). Additionally, a *hybrid* DT combines physics-based and data-driven methods (Chinesta et al., 2020; Kapteyn et al., 2022). *Structural* DTs apply to systems that change over time due to environmental or operational factors. Examples include wind turbines, nuclear reactors, gas turbine engines, and civil infrastructure. Finally, *virtual* DTs are traditional numerical models based on different discretization methods that do not continuously assimilate data (Abramovici et al., 2017; Jones et al., 2020; Chinesta et al., 2020).

For the remainder of this paper, we define a DT as an automation strategy that connects a physical plant to an adaptive real-time simulation environment through bidirectional communication. This environment may include multiple models with different characteristics. Our definition emphasizes that a DT is not just a detailed model, but a broad and flexible concept. Following National Academy of Engineering and National Academies of Sciences, Engineering, and Medicine (2023), we emphasize that DT models should be fit for purpose, avoiding unnecessary complexity. In addition, we adopt the terminology of Jones et al. (2020) to describe the relationship between the physical and virtual components of DTs.

The process engineering application of this definition is illustrated in Fig. 3. In this context, the connection between the physical and digital environments is always mediated by some form of actuation. To ensure the safe operation of the automated system, both human monitoring and intervention are always possible.

2.2. Enablers of digital twins

For DTs to work effectively in scientific and industrial settings, several key components need to be in place. These components are outlined below:

- **Data acquisition and connectivity**: Reliable data acquisition systems with strong connectivity gather real-time data from physical assets. This involves using sensors (Kapteyn and Willcox, 2022), internet of things (IoT) devices, and other tools to collect accurate information.
- **Data integration**: Integrating data from sensors, historical records, and external databases provides a complete view of asset behavior. Semantic standards and communication protocols ensure that these models can interact effectively. One such semantic standard is the Asset Administration Shell, which provides a digital interface for physical assets, ensuring consistent data representation and interaction across systems (Industrie 4.0 Glossar, 2023; Wagner et al., 2017; Park et al., 2021). In industrial automation, OPC-UA is a widely used protocol for secure and reliable machine-to-machine communication. It enables seamless data exchange between devices and systems, regardless of manufacturer, and supports both real-time and historical data (Greppi, 2010; Redeker et al., 2021).
- **Data analytics**: Advanced data analysis extracts valuable information from the collected data. These methods help detect patterns, identify anomalies, and create predictive models that guide decision-making and optimization in DTs (Kapteyn et al., 2020; Kapteyn and Willcox, 2022). Machine learning methods are commonly employed to perform these tasks effectively.
- **Modeling and simulation**: The development of accurate and reliable simulation models that replicate the behavior and characteristics of physical assets is at the heart of DT. These models serve as the basis for virtual testing, optimization, and "what-if" scenarios within the DT environment. This DT component is discussed in detail in Section 4.
- **Co-simulation**: The integration of multiple models from different domains improves the accuracy of DTs. Co-simulation allows models — for example, of subsystems or with different levels of detail — to interact in real time. Each model works with its native solver, and a common interface ensures seamless data exchange between them. In this way, subsystems that form a coupled problem are modeled and simulated in a distributed manner (Talkhestani et al., 2019).
- **Cloud computing and big data infrastructure**: The scalability and processing power of cloud computing and big data infrastructure manage the large volumes of data generated. These technologies enable the storage, processing, and analysis of large data sets (Kapteyn and Willcox, 2022; Data Digital Twin, 2023).
- **Cybersecurity and privacy**: Because DTs often handles sensitive or critical data, strong cybersecurity measures and privacy protocols are necessary. Protecting the integrity, confidentiality, and availability of data maintains trust and ensures the secure operation of DTs systems (Balta et al., 2023).
- **Visualization and user interfaces**: Visualization and interactive user interfaces make DTs usable and effective. This improves decision-making and process understanding.

3. Digital twin applications in process engineering

The enablers mentioned in Section 2.2 have become more and more available for process engineering applications, which increases the exploration of DTs (Kockmann, 2018; Bamberg et al., 2021). These DTs promise several benefits such as real-time monitoring, predictive maintenance, process design, optimization, control, and deeper understanding of the system (Bamberg et al., 2021; Yang et al., 2020). Overall, they are expected to improve efficiency, reduce time-to-market, save costs, and increase productivity (Kockmann, 2018).

In the process industry, the concept of DTs has roots that predate the formal definition of the term itself (Bequette, 2019). In particular, in the 1990s, Natori and O'Young (1996) envision an advanced closed-loop chemical plant that integrated multiple technologies and incorporated environmental and sustainability considerations, even before the term DT was coined. These authors also pioneered the first steps in implementing real-time capabilities in existing plants.

Despite early efforts, DT terminology remains inconsistent within the field of process engineering, and the full implementation of a

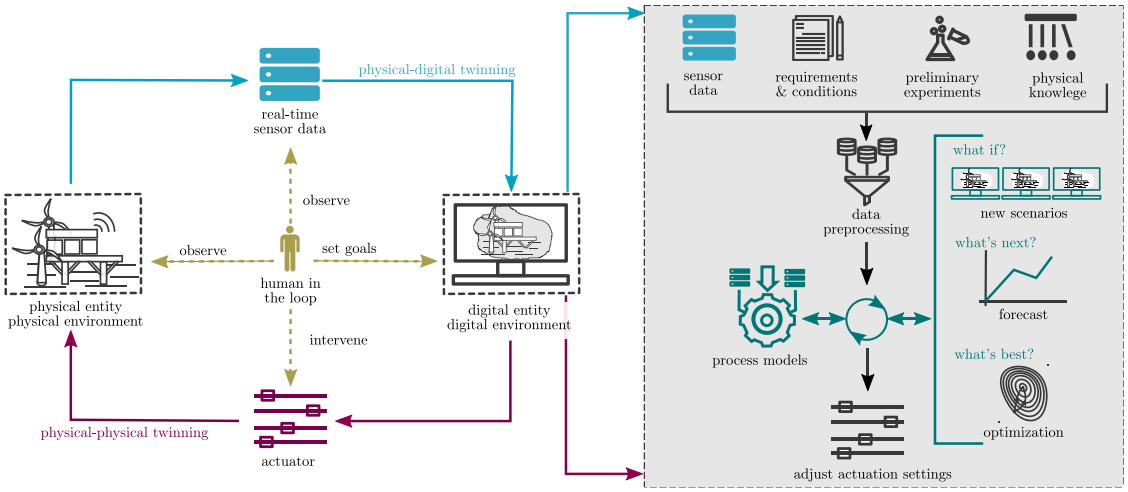

Fig. 3. Schematic of the DT concept in process engineering, illustrating the integrated flow between physical and digital environments. The digital entity includes data preprocessing, process modeling, and scenario exploration. An actuator enables real-time control and predictive maintenance by connecting the digital and physical environments.

DT system remains a challenge. In addition, comprehensive studies are difficult to find because the term DT is often used synonymously with model. Despite these challenges, recent research highlights the potential of the DT concept. For example, He et al. (2019) propose a data-driven DT for chemical processes that includes monitoring, diagnostics, and control. The system is optimized online to increase safety and reduce variability. Similarly, Min et al. (2019) present a DT for the petrochemical industry that leverages IoT data, machine learning, and a feedback loop. This DT provides a real-time replica of the process, leading to cost and efficiency optimization. In addition, Kender et al. (2021) describe a DT for an air separation unit that integrates historical and real-time data, and demonstrates its economic viability through two case studies. Future implementations include the development of a DT for an offshore power-to-X process (Dittler et al., 2022; Rentschler et al., 2023).

Several studies focus on the development of individual DT components, each contributing to different aspects of the field. Below are some of the most important and recent works. Aversano et al. (2019) present reduced-order models (ROMs) for the simulation of combustion systems within a DT. Lopez et al. (2020) introduce a soft sensor for monitoring cellulose-to-ethanol fermentation, creating a *digital shadow*. In the field of battery management, Wu et al. (2020) review the use of models, data, and AI, and propose an unvalidated framework aimed at improving control and extending the life of lithium-ion batteries. Scaling up solid oxide fuel cells using pilot scale models and data is explored by Kang et al. (2021). Meanwhile, Helgers et al. (2022) develop a dynamic metabolic model for the production of HIV gag virus-like particles that can be integrated into a DT for model-based control with process-analytical detectors. In addition, Wang et al. (2023a) propose a noise-accounting model identification approach focused on dynamic data, which is demonstrated using the Rössler attractor and a diesel hydrotreating unit. Edington et al. (2023) present a state-dependent weighting technique for model integration, demonstrated using synthetic data in a two-tank, single-pump system. Moreover, Wang et al. (2023b) develop a DT for predicting composite quality by integrating a static autoclave model and a dynamic composite model, validated with experimental data. Hassan et al. (2024) employ a virtual reality-based bioreactor simulation that trains operators by simulating rare events and evaluating their responses. Finally, Soesanto et al. (2024) refine models of a primary separation vessel to more accurately reflect the actual separation process, using industry data and techniques such as Bayesian optimization, step testing, and sensitivity analysis to parameterize the model.

In addition to developing custom DTs, the process industry has explored commercial software (de Beer and Depew, 2021). Companies like AspenTech (Aspentech Digital Twin, 2023), gPROMS (PSE Digital Twin, 2023), and AVEVA (AVEVA Digital Twin, 2023) offer software for real-time simulation, optimization, analytics, and control. The solutions use machine learning and AI algorithms, offering a comprehensive approach to DT technology.

Several resources are recommended for further reading. Herwig et al. (2021a,b) provide a comprehensive two-volume series on DTs in biomanufacturing. In addition, Thelen et al. (2022, 2023) offer a two-part series that reviews DTs, including a case study on battery modeling and a discussion of uncertainty quantification. (Guo et al., 2023) explore the maturity of DTs components in manufacturing, with electrochemical machining as a key example. Vassiliadis et al. (2024) provide a process systems engineering perspective, emphasizing the need for improved modeling and simulation tools to fully realize the potential of DTs. Similarly, Walmsley et al. (2024) discusse the role of self-adaptive DTs in energy-intensive industries, emphasizing its importance for optimizing energy efficiency. Finally, Bizon (2023) focuses on simulation methods in process engineering, with particular attention to model-order reductions (MORs) techniques.

4. Numerical methods for digital twins in process engineering

This section examines the simulation environment within DTs. Models used in DTs face several challenges, both cross-industry and process-engineering specific, especially when integrated into high-volume production environments (Willcox and Segundo, 2024). These challenges include managing large datasets, handling diverse models, addressing prediction discrepancies, and facilitating autonomous updates. In addition, issues such as model transparency, privacy, cybersecurity, ethical considerations, and human-machine interaction must be addressed (Balta et al., 2023; Hartmann and van der Auweraer, 2022).

In process engineering, modeling presents unique complexities due to the high dimensionality and interconnected nature of physical and chemical processes at multiple scales (Bhutani et al., 2006; Niederer et al., 2021). Data scarcity, resulting from the static nature of process engineering and limited dynamic data collection, complicates the modeling process. Measurement uncertainties introduce additional inaccuracies. Despite these difficulties, accurate simulations must predict system behavior and optimize performance.

To address these challenges, we provide an overview of mathematical tools and methods for constructing DTs, covering both theoretical and practical aspects. To address these challenges, we provide an overview of mathematical tools and methods for constructing DTs, covering both theoretical and practical aspects, and discuss their applications in process engineering and related fields.

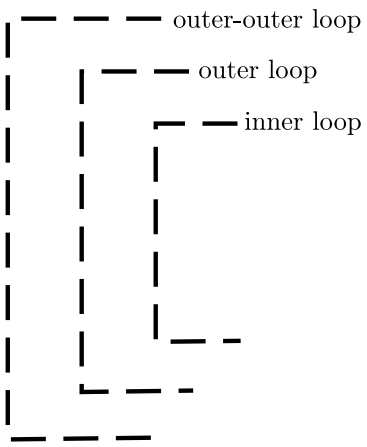

Fig. 4. Hierarchical loop concept in the DT development (Asch, 2022).

This section is organized around the hierarchical loop concept illustrated in Fig. 4, which includes three interrelated loops: the physical problem-solving loop (inner loop), the optimization loop (outer loop), and the decision loop (outer-outer loop). The physical problem-solving loop focuses on modeling the physical problem itself. The optimization loop includes techniques such as parameter estimation, uncertainty quantification (UQ), co-simulation, control, and solving inverse problems. We will first explore the inner loop before moving on to the outer loops. The outer-outer loop is primarily concerned with goals and strategies rather than mathematical methods, and will not be discussed here.

4.1. Physical problem-solving loop

The physical problem-solving loop involves models describing the underlying physical system. The solutions of those models are called simulations. Simulations allow for accurate evaluation of the system response and thereby reduce the number of experimental tests. An example of a generic simulation is given in Fig. 5.

Modeling dynamic systems such as chemical plants typically involves either *mechanistic models*, *data-driven models*, or a combination of both. Mechanistic models are based on physical laws, while data-driven models are based on empirical or measured data. In many cases, mechanistic models may be too computationally intensive to solve directly, leading to the use of surrogate models that approximate the original problem and reduce computational requirements. Surrogate models can range from ROMs to data-driven models trained on mechanistic data. However, data-driven methods are not limited to surrogates; they are also powerful stand-alone tools. ROMs are discussed in Section 4.1.4, while methods for building data-driven models and surrogates can be found in Sections 4.1.2 and 4.1.3. For a thorough overview of surrogate models in chemical engineering, see McBride and Sundmacher (2019) and Peterson et al. (2024b).

4.1.1. Equations in mechanistic models: linear, nonlinear, differential, and stochastic forms

In process engineering, mechanistic models use mathematical equations to describe chemical systems based on physical principles such as conservation laws, thermodynamics, and transport kinetics. To accurately model chemical processes, hierarchical frameworks represent the system at different levels of abstraction, e.g. from molecular to production scale (Freund and Sundmacher, 2011).

However, mechanistic models are challenged by knowledge gaps and algorithmic inaccuracies that lead to mismatches with real-world processes. Building mechanistic models is often time and resource-consuming, especially for complex systems with poorly understood interactions. In addition, solving mechanistic models is computationally demanding. However, when well implemented, mechanistic models effectively simulate complex chemical processes and provide strong predictive capabilities (Bhutani et al., 2006; Pantelides and Renfro, 2013).

In the realm of mechanistic models, we encounter two main categories: *Unit-based* and *function-based* models. *Unit-based* models focus on individual plant units such as reactors and heat exchangers. These models are often presented visually as flow diagrams that provide a clear picture of all unit operations. In contrast, *function-based* models encapsulate the overall behavior of the chemical system. They account for the flow of mass, energy, and momentum across system boundaries. Two examples of *function-based* models are the elementary process function (EPF) method and the FluxMax approach. The EPF method dynamically optimizes its process path by manipulating Lagrangian matter elements (Freund and Sundmacher, 2008; Peschel et al., 2010). The FluxMax approach segments the state space into discrete thermodynamic states and models the transition between them with elementary processes. This results in a network flow optimization problem (Liesche et al., 2019; Schack et al., 2020; Svitnič and Sundmacher, 2022).

Despite the variety of approaches to mechanistic modeling, the same types of equations appear: *algebraic equations*, *differential equations*, and *stochastic equations*. For each of these types of equations, there are established numerical methods for solving or estimating their solutions. An overview of possible solutions for all types of equations is given in Table 1.

Systems of linear and nonlinear equations: Systems of equations are divided into two categories: *Linear* and *nonlinear*. *Linear equations*, represented as $\mathbf{Ax} = \mathbf{b}$, are characterized by direct relationships between variables. Here, $\mathbf{A} \in \mathbb{R}^{m \times n}$ is the coefficient matrix, $\mathbf{x} \in \mathbb{R}^n$ the vector of variables, and $\mathbf{b} \in \mathbb{R}^m$ the vector of known constants. The power of each variable in a linear equation is one, which results in a straight line when plotted in two dimensions. Solving linear equations involves finding values of $\mathbf{x}$ that (approximately) satisfy the equation. Methods for solving linear equations include direct techniques, such as Gaussian elimination, and iterative approaches, such as Newton's method (Golub and van Loan, 2013). The choice of the numerical scheme depends on factors such as the desired spatial and temporal accuracy, requiring careful consideration of shocks, breaking points, and other discontinuities (Rheinboldt, 1998).

Nonlinear equations represent more complicated interactions between variables. They are expressed as $\mathbf{G}(\mathbf{x}) = \mathbf{0}$, where $\mathbf{G} : \mathbb{R}^n \to \mathbb{R}^m$ is a vector-valued function. Solving nonlinear equations involves finding the vector $\mathbf{x} \in \mathbb{R}^n$ that (approximately) satisfies $\mathbf{G} = \mathbf{0}$. Nonlinear equations may involve higher powers of variables, exponential functions, logarithms, or other mathematical functions that introduce complexity beyond simple proportionality. As a result, the graph of a nonlinear equation takes on a variety of shapes, such as parabolas, circles, or exponential curves, depending on the specific form of the equation. Methods for solving nonlinear equations include bracketing methods such as the bisection method, iterative methods such as Newton–Raphson and secant methods, and optimization techniques such as Gauss–Newton and gradient descent algorithms. Each is tailored to the specific characteristics of the problem.

Process engineering applications: In process engineering, systems of equations model complex phenomena such as chemical reactions, phase equilibria, and physical interactions. For example, heat transfer calculations in heat exchangers demonstrate linear behavior, while temperature-dependent reaction rates, described by the Arrhenius equation, illustrate nonlinear behavior.

Differential equations: Differential equations describe a system's dynamics, e.g., how variables change with factors such as temperature or pressure. *Ordinary differential equations (ODEs)* are differential equations dependent on one single independent variable. A system of n

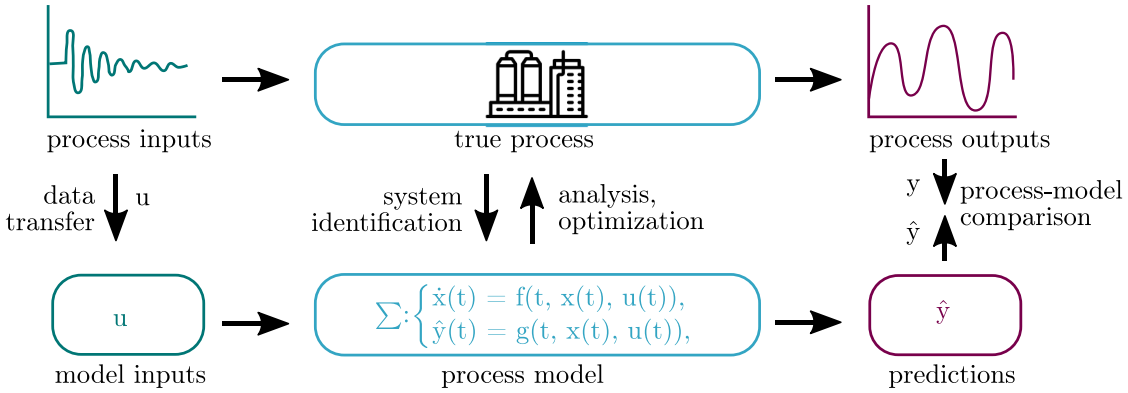

Fig. 5. Diagram describing the simulation of dynamical systems.

Table 1
Overview of equation types and solution techniques.

Equation type	Solution techniques
Linear equations	Direct methods (e.g., Gaussian elimination), Iterative methods (e.g., Newton's method)
Nonlinear equations	Bracketing methods (e.g., bisection), Iterative methods (e.g., Newton–Raphson, secant methods), Optimization methods (e.g., Gauss–Newton, gradient descent)
ODEs	Forward methods (e.g., Euler, Runge–Kutta), Backward methods (e.g., implicit Euler, implicit Runge–Kutta), Machine learning methods (e.g., PINN)
PDEs	Discretization methods (e.g., FDM, FEM, FVM), Machine learning methods (e.g., PINN)
DAEs	Index reduction methods (e.g. Pantelides' algorithm) Direct discretization methods
Stochastic equations	Strong approximation methods (e.g., Euler–Maruyama method), Weak approximation methods (e.g., MC simulations)

first-order ordinary differential equations (ODEs) takes the following form:

$$\begin{cases} \frac{d}{dt}\mathbf{x}(t) = F(t, \mathbf{x}(t)), \quad t \in I, \\ \mathbf{x}(t_0) = \mathbf{x}_0, \end{cases} \tag{1}$$

In this equation, $F : \mathbb{R} \times \mathbb{R}^n \to \mathbb{R}^n$ is a vector-valued function, $\mathbf{x} \in \mathbb{R}^n$ is the variable vector, I is the interval of interest, and $\mathbf{x}_0$ is the initial condition at time $t = t_0$. Solving ODEs involves various numerical methods, including forward and backward approaches. Choosing a method depends on accuracy requirements and whether the problem is stiff or non-stiff (Golub and van Loan, 2013). In addition, artificial neural networks (ANNs)-based methods, such as physics-informed neural networks (PINNs), solve complex or high-dimensional ODEs by approximating the solution with neural networks.

Partial differential equation (PDE) involve multiple variables and their partial derivatives, modeling dynamic systems with spatial and temporal variations. While PDEs are ideal for such modeling, many process models are simplified to differential-algebraic equations (DAEs) or ODEs by assuming steady states or negligible spatial variations. Various methods solve PDEs, with discretization being the most common. The finite difference method (FDM) approximates the PDE derivatives by finite differences and converts them into algebraic equations. The finite volume method (FVM) integrates PDEs over control volumes and calculates boundary flows to satisfy conservation laws, making it ideal for mass, momentum, or energy conservation simulations. Finite element method (FEM) subdivides the domain into elements and uses variational methods. In addition, as with ODEs, ANN methods can solve PDEs.

Partial differential equations (PDEs) consist of differential and algebraic equations and typically arise from the spatial discretization of PDEs. They capture the interplay between conservation laws or state-dependent equations. Unlike ODEs, which rely on integration, DAEs involve differentiation in their solution, requiring specialized numerical techniques (Ascher and Petzold, 1998a; Kunkel and Mehrmann, 2006; Ascher and Petzold, 1998b). While index reduction methods simplify DAEs by converting algebraic constraints into differential relations, direct discretization methods are also applicable, as mentioned in Chapter 10 of Ascher and Petzold (1998b). Reduced DAEs are solved using numerical integrators like Euler or Runge–Kutta methods, chosen based on the system's stiffness.

Process engineering applications: In process engineering, differential equations are critical for modeling conservation laws such as mass, energy, and momentum balances. Recent advances include Ji et al. (2021), which employ a PINN framework for stiff chemical kinetics, and Langiu et al. (2021), which present a differential–algebraic modeling framework for optimizing energy systems.

Stochastic Equations: Stochastic differential equations model processes influenced by random forces or fluctuations, capturing behaviors affected by inherent variability, environmental noise, and uncertain parameters. A general form of a stochastic differential equation is:

$$\frac{d\mathbf{Z}}{dt} = \mathcal{F}(\mathbf{Z}, t) + \mathcal{G}(\mathbf{Z}, t) \cdot \eta(t), \tag{2}$$

where $\mathbf{Z}$ is the systems' state variable, t is time, $\mathcal{F}(\mathbf{Z}, t)$ is the deterministic part, $\mathcal{G}(Z, t)$ is the stochastic part, and $\eta(t)$ is the random noise term. Solving stochastic differential equations involves strong approximation methods, such as the Euler–Maruyama method for approximating the

Table 2
Overview of data-driven modeling categories.

Category	Description
Supervised learning	Identifies patterns or relationships in labeled data. Methods include ANN, GP, decision tree, random forest, SVM.
Unsupervised learning	Identifies patterns or relationships in unlabeled data. Methods include clustering, PCA, and self-organizing maps.
Reinforcement learning	An agent learns decision-making by interacting with the environment.

solution path, and weak approximation methods, such as Monte Carlo (MC) simulation for estimating probability distributions.

Process engineering applications: Process engineers use stochastic equations to study noisy chemical reactions, analyze system variability, and assess robustness, reliability, and risk (Ripley, 2009; Prévôt and Röckner, 2007). Stochastic differential equations are useful for constructing Markov processes with predetermined statistical properties that model random disturbances in process plants (King, 1974). In Pineda and Stamatakis (2018), stochastic modeling of surface reactions using the Langevin equation generates stochastic realizations of the Fokker–Planck equation, considering test cases of a single species for adsorption, desorption, reaction, or diffusion on a lattice.

4.1.2. Data-driven models

Data-driven models use historical, sensor, and simulation data to predict complex system behavior through statistical techniques. The data characteristics and the process being modeled determine the best data-driven approach. For an overview of the data-driven category, see Table 2.

Data-driven models are increasingly popular in process engineering (Pirdashti et al., 2013; Kockmann, 2018; Venkatasubramanian, 2019; Shang and You, 2019; Md Nor et al., 2020; Schweidtmann et al., 2021; Pistikopoulos et al., 2021) due to their ability to be built without prior knowledge of underlying physics. They effectively handle nonlinear relationships, a common challenge for mechanistic models, and offer faster decision-making due to computational efficiency. However, challenges include ensuring model reliability, addressing confidence limits, and dealing with ethical concerns like data security (Wang et al., 2021; Pistikopoulos et al., 2021). Additionally, data-driven models often rely on sparse, noisy data and might miss critical phenomena such as faulty operations (Sansana et al., 2021). Design of experiments (DOE) helps address these issues by systematically collecting relevant data to improve model quality and reliability (Pan et al., 2022; Sharma and Liu, 2022).

Data-driven model infrastructure: Robust data-driven models require a structured infrastructure for *data storage*, *preprocessing*, *parameter optimization*, *feature selection*, and *model evaluation* (Raschka and Mirjalili, 2019). Efficient data storage systems like databases or data warehouses manage large datasets. *Data preprocessing* includes cleaning, normalization, noise reduction, and dimensionality reduction to prepare data for model training. After preprocessing, *feature selection* identifies relevant variables, improving accuracy and performance. During model development, *optimization algorithms*, such as stochastic gradient descent (SGD), adjust model parameters based on loss gradients (Kingma and Ba, 2014; Lydia and Francis, 2019), affecting model accuracy. *Model evaluation* applies metrics like accuracy and precision to assess generalizability, with ongoing validation and monitoring essential to detect performance changes.

Supervised learning: Supervised learning builds data-driven models using labeled data sets, where the data contains corresponding outcomes for the model to predict. These models include *regression*, *classification*, and *active learning*. *Regression models* predict continuous outcomes, such as chemical process yield, while *classification models* classify inputs into discrete classes, such as determining process states. *Active Learning Models* optimize data labeling by selecting the most informative data points for training. Supervised learning algorithms can be adapted to different tasks by modifying learning objectives and loss functions, with common methods including

- ANNs, which model nonlinear relationships using layers of interconnected nodes (neurons) that process inputs through activation functions.
- Gaussian processess (GPs), which provide a probabilistic approach, useful for limited data sets, and offer predictions with confidence intervals.
- Decision trees, which model complex decision processes by recursively splitting the data into branches based on feature values.
- Random forests, which combine multiple decision trees trained on random subsets of the data to improve accuracy and reduce the risk of overfitting.
- Support vector machines (SVMs), which identifies the optimal hyperplane for class separation or regression, with a margin of error.

Process engineering applications: Supervised learning methods have proven to be highly effective in modeling, prediction, and optimization in various process engineering applications. For example, Willis et al. (1991) provide a comprehensive review of ANNs used in chemical engineering, while Deringer et al. (2021) review GPs regression methods for materials science. In addition, Ma and Wang (2009) automate decision trees for chemical process control, and Partopour et al. (2018) apply random forests to microkinetic models. The use of support vector machines (SVMs) in chemical property prediction is explored in Ivanciuc (2007). Research in active learning demonstrates how the iterative design of experiments can significantly reduce the burden of reaction screening (Eyke et al., 2020).

Unsupervised learning: Unsupervised learning deals with unlabeled datasets, aiming to uncover hidden patterns and relationships without predefined labels. This is valuable when the data distribution is unknown or manual labeling is impractical. Typical methods are:

- Clustering, which groups data points based on their similarities to reveal natural groupings or trends within the data.
- Principle components analysis (PCA), which reduces the dimensionality of the data by projecting it into a lower-dimensional space while preserving as much variance as possible.
- Self-organizing maps, which create a low-dimensional representation of high-dimensional data by mapping similar data points close together to help identify patterns.

Process engineering applications: Unsupervised learning methods discover patterns in unlabeled process data. For example, Sancho et al. (2022) use k-means clustering to classify crude oil, while Lin et al. (2000) apply nonlinear PCA for process monitoring. Furthermore, Chen and Yan (2012) extend self-organizing maps for troubleshooting in chemical processes.

Reinforcement learning: Reinforcement learning focuses on decision-making and control in dynamic environments. Unlike supervised learning, it involves an agent that learns through interactions with its

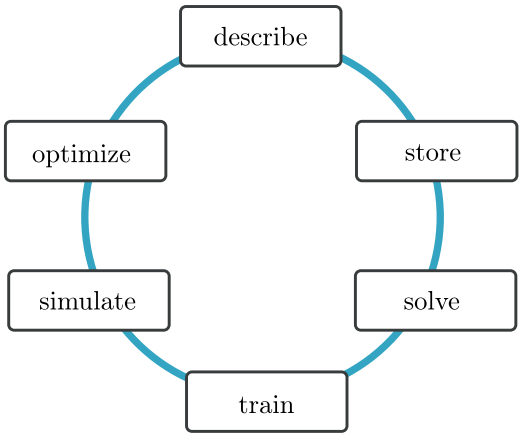

Fig. 6. Circular description of some of the needed steps for SciML.

environment, receiving rewards or penalties to develop strategies for achieving goals.

Process engineering applications: Shin et al. (2019) review the impact of reinforcement learning on process control, while Hubbs et al. (2020) demonstrate its application to production scheduling, addressing uncertainty, and enhancing operational efficiency.

4.1.3. Scientific machine learning

Scientific machine learning (SciML) integrates traditional computational sciences with machine learning techniques to improve model robustness, interpretability, and reliability (Karpatne et al., 2017; Sharma and Liu, 2022; Karniadakis et al., 2021). Fig. 6 illustrates the key steps in a typical SciML setup. The importance of SciML is highlighted in several studies (Karpatne et al., 2017; Willard et al., 2020; Karniadakis et al., 2021; Scientific machine learning, 2023), with practical applications in aerospace and medical engineering (Kapteyn et al., 2020, 2021; Kapteyn and Willcox, 2022; Kapteyn et al., 2022).

Several methods integrate scientific knowledge with data-driven models throughout the modeling process. These techniques span from initial input selection to final model evaluation:

- Feature selection chooses features that capture the essential physical processes of the system. This ensures that the model focuses on relevant data (Schubert et al., 1994; Sharma and Liu, 2022).
- Synthetic data generation supplements real data with synthetic points created using physical principles. This approach reduces the need for additional measurements and is useful when data is limited (Sharma and Liu, 2022).
- Loss function design incorporates physical constraints through penalty terms, ensuring that the model adheres to known physical laws and limits (Schubert et al., 1994; Karpatne et al., 2017; Sharma and Liu, 2022).
- Model architecture must be carefully chosen to ensure optimal performance and accuracy. For instance, ANN models can be designed to include symmetries and invariances (Karpatne et al., 2017; Karniadakis et al., 2021).
- Pre-training with synthetic data accelerates learning and improves model robustness (Serra et al., 2003; Karpatne et al., 2017; Sharma and Liu, 2022).
- Hybrid modeling integrates physical information into data-driven models. Mechanism correction uses deviations between model and measured states as error signals, while mechanism estimation predicts process parameters that are challenging to derive from first principles (von Stosch et al., 2014).

For effective SciML implementation, advanced computational tools are essential. These tools should integrate smoothly with existing scientific simulations (Essential Tools of Machine Learning, 2023; Asch, 2022) and include:

- Structured linear algebra efficiently manages and manipulates large matrices and vectors common in scientific computing.
- Mixed precision arithmetic number types speed up computations while maintaining accuracy, especially in large-scale simulations.
- Differentiable programming utilizes automatic differentiation tools to compute derivatives efficiently, which is essential for training neural networks and solving differential equations.
- Equation solvers handle algebraic and differential equations in complex systems modeled in SciML (see Section 4.1.1).
- Probabilistic programming aids in UQ and global sensitivity analysis by quantifying uncertainty and assessing model reliability (see Section 4.2.4).
- Optimization techniques train models and fine-tune hyperparameters to achieve optimal performance (see Section 4.2.1).
- Model benchmarking involves using standardized benchmark datasets to compare and evaluate different models, ensuring they meet performance standards (Thiyagalingam et al., 2022).

In the following, we will delve into two key areas within SciML: identifying governing equations and integrating physical principles with ANNs.

Identification of governing equations: Two primary techniques, *symbolic regression* and *sparse regression*, are widely used to identify governing equations. Both methods decipher the dynamics of the system, particularly the right-hand side of differential equations. *Symbolic Regression* generates a set of candidate symbolic functions and compares them with the numerical derivatives to select the best-fitting function. Initially, random candidate functions are refined using evolutionary algorithms like genetic programming via expression trees.

Sparse regression, specifically the SINDy approach, discovers parsimonious models by applying sparsity-promoting regression on the derivatives to a library of candidate nonlinear functions. This library, which must be created manually, may contain pre-existing information. A limitation of the original SINDy approach is that it requires direct access to or approximation of gradient information in the time domain, as does symbolic regression. This leads to numerical errors from gradient approximation, a problem that is addressed in Goyal and Benner (2022) by incorporating classical numerical integration schemes (e.g., Runge–Kutta) into the sparse identification process, thereby eliminating the need for derivative information collection. Another solution for the gradient approximation problem is proposed in Both et al. (2021), where the sparse regression is done within ANNs with automatic differentiation. In Forootani and Benner (2024), the curse of dimensionality and large datasets is addressed by solving the most informative samples via discrete empirical interpolation method (DEIM) as preprocessing for the SINDy algorithm.

Process engineering applications: Symbolic regression is used by de Carvalho Servia et al. (2023) for model generation in an isomerization case study. In Bhadriraju et al. (2020) and Bhadriraju et al. (2019), SINDy derives multiple models from historical data for different input scenarios. These models, along with their training data, are then used to construct a deep ANN. This ANN is integrated into a model predictive control (MPC) framework for closed-loop operation, facilitating the identification and control of a continuous stirred tank reactor (CSTR).

Combination of physical principles and ANNs: ANNs handle noisy data, perform automatic differentiation, and act as flexible, meshless models. The *Universal Approximation Theorem* demonstrates this by showing that a feed-forward network with a single hidden layer and a finite number of neurons can approximate continuous functions on

compact subsets of $\mathbb{R}^n$ under certain conditions on the activation function. However, sometimes ANNs face convergence issues, especially for long-time integration and nonlinear problems, and the determination of the optimal network architecture is uncertain. As a result, the combination of ANNs-based solutions with traditional physics-based methods is increasingly being explored to improve performance and reliability.

In this context, we present four approaches. (1) *Neural ODEs* parameterizes the derivative of the hidden state with a neural network, allowing the integration of the hidden state over time (Chen et al., 2018). (2) *Neural Operators* learn the map $\mathcal{G}$ between infinite-dimensional function spaces and serve as discretization-free surrogates for PDEs (Li et al., 2020). (3) *PINNs* encode the differential equation within the neural network, allowing it to learn the solution as it fits observed data (Raissi et al., 2017, 2019; Cuomo et al., 2022). Finally, (4) *group equivariant convolutional networks* use convolutional layers that preserve the translational structure of the data, exploit symmetries, and require less training data while enforcing structure (Cohen and Welling, 2016). These networks maintain transformations that preserve system dynamics, such as invariance under time shifts, time reversal, spatial shifts, spatial reflection, rotations, and coordinate rescaling.

Process engineering applications: Recent advances in process engineering have demonstrated the growing importance of the interaction between physical principles and ANNs. In particular, hybrid modeling has made remarkable progress, as reviewed in von Stosch et al. (2014), which highlights applications in biochemistry and chemical engineering. For example, Raccuglia et al. (2016) train models to predict crystallization outcomes and aid material discovery using reaction data, while Wu et al. (2019a) develop a machine learning predictive control scheme with real-time updates for non-isothermal reactors. In addition, Bangi and Kwon (2020) combine first principles with deep ANNs to model hydraulic fracturing, and Wang et al. (2021) integrates deep learning with d-band theory to predict surface reactivity of transition metals. In contrast, Peterson et al. (2024a) compare hybrid models to data-driven approaches in a CSTR case study, challenging assumptions about the enduring superiority of hybrid modeling.

In the realm of neural ODEs, Owoyele and Pal (2022) apply this approach to chemical kinetics by modeling hydrogen-air self-ignition, incorporating adjustments to ANN weights for more accurate predictions. Similarly, neural operators are used by Goswami et al. (2024) to solve stiff chemical kinetics problems, including applications to combustion kinetics and the three-species ROBERS problem. PINNs have also been explored in solving stiff chemical kinetic systems, as demonstrated by Ji et al. (2021). Furthermore, Qiao et al. (2021), apply group equivariant convolutional networks to quantum chemistry, and Kaba and Ravanbakhsh (2022) use these networks to incorporate crystalline symmetries in materials science predictions.

4.1.4. Model order reduction

Outer-loop applications require fast and accurate processing of potentially large-scale, complex dynamical systems. With advances in modern computing environments and high-performance clusters, the limits of computational feasibility are continuously pushed.

Reduced-order models (ROMs) addresses the computational challenges of large dynamical systems by intersecting disciplines such as control, systems theory, approximation theory, and numerical linear algebra. The core idea of MOR is to replace complex, large-scale systems, typically described by many ODEs, with reduced-dimensional systems governed by well-understood dynamics. Effective MOR approaches ensure that ROMs exhibit reduced complexity, reliability, and preservation of essential properties of the original system. As highlighted in Asch (2022), a DT benefits significantly from a finely tuned MOR. The lack of a universal MOR methodology for all DTs requires a tailored approach to ensure that the ROM is calibrated for reliable and accurate results. However, this customization may hinder the development and implementation of DTs (Asch, 2022).

For a comprehensive review of the variety of MOR methods that have been proposed and continuously developed over the last two decades, we refer the reader to the reviews of Baur et al. (2014) and Benner et al. (2015) as well as the books of Antoulas (2005), Benner et al. (2017), Antoulas et al. (2020) and Benner et al. (2021). A survey of MOR methods for DAEs is given in Benner and Stykel (2017). In Eason and Biegler (2021), developments and practical applications of ROMs in process engineering are emphasized, especially in process optimization.

MOR methods are subdivided into two categories: *intrusive* and *non-intrusive methods*. *Intrusive methods* necessitate the processing of the governing equations of the full order model (FOM), while *non-intrusive methods* operate on the input–output or snapshot data of the system, thereby avoiding making changes to the underlying simulation code. However, not all methods are unequivocally categorized into one of these two subdivisions. Table 3 lists all the methods described in this section.

Proper orthogonal decomposition: A prime example of resistance to categorization is POD. The categorization of POD as either intrusive or non-intrusive depends on its implementation within the MOR process. For example, when POD is used in conjunction with Galerkin projection, which requires access to and modification of the governing equations, it is considered an intrusive method. Conversely, when POD is used in a data-driven manner, such as in the snapshot method introduced by Sirovich (1987), it is considered non-intrusive because it operates on snapshot data and does not require modifications to the underlying model. POD, closely related to PCA or Karhunen-Loève expansion, serves two purposes. On the one hand, it acts as a statistical tool, adept at extracting significant structures or patterns from large data sets. On the other hand, it reduces complexity, being able to use a minimal number of POD modes to approximate a given dynamical system to a desired level of accuracy. Once a sufficient number of POD modes are identified, these modes can be used to transform the data into a latent dimensional space. This reduced data can then be used as input to standard data-driven models (Ammar et al., 2024; Chinesta et al., 2020).

Process engineering applications: In Abdullah et al. (2021), a nonlinear identification approach (based on nonlinear PCA and system identification approaches) computes a dynamic model of first-order nonlinear ODEs describing the temporal evolution of the slow process states for two test cases: An isothermal multi-reaction batch reactor and a non-isothermal jacketed CSTR. Agarwal et al. (2009) develop a ROM based on POD, which is a low-dimensional approximation to a dynamic PDE-based model with application to pressure swing adsorption (PSA) processes (for a dynamically coupled PDE-based model of a two-bed, four-step PSA process for the separation of hydrogen from methane). In Bremer et al. (2016, 2017), a hybrid approach (known as POD-DEIM (Chaturantabut and Sorensen, 2010)), is applied to a dynamic, two-dimensional reactor model for catalytic CO_2-methanation. The method is snapshot-based, i.e. it uses snapshots of the time-domain evolution of the system state.

I Classical, intrusive methods

Traditional MOR involves access to the full model and reduction by intrusive methods, typically projection-based. As shown in the linear setup in Fig. 7, the main idea behind MOR (and in particular the structure-preserving classes of methods) is to approximate the original (potentially) large-scale model described by n state variables by a ROM of (much) lower dimension, denoted by r, and with a structure similar to that of the original.

The classical setup of linear, time-invariant dynamical systems is described by a state-space representation with first-order dynamics. Within this framework, the dynamical system with homogeneous (zero) initial conditions is characterized by equations in both the time (left) and frequency (right) domains, as shown below:

$$\begin{cases} \frac{d}{dt}\mathbf{x}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t), \\ \mathbf{y}(t) = \mathbf{C}\mathbf{x}(t), \ \mathbf{x}(t) = \mathbf{0}. \end{cases} \iff \begin{cases} s\mathbf{x}(s) = \mathbf{A}\mathbf{x}(s) + \mathbf{B}\mathbf{u}(s), \\ \mathbf{y}(s) = \mathbf{C}\mathbf{x}(s). \end{cases} \tag{3}$$

Table 3
Overview of equation types and solution techniques.

Category	Methods
Intrusive	Balanced truncation (BT), moment matching (MM), reduced basis (RB)
Non-intrusive	System identification (SI), subspace identification (SSI) (e.g., MOESP, N4SID), dynamic mode decomposition (DMD) operator inference (OpInf), Frequency domain analysis methods
Both	Proper orthogonal decomposition (POD)

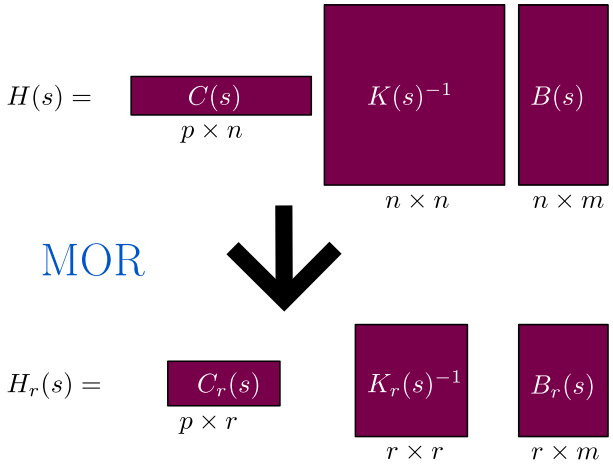

Fig. 7. Projection-based MOR in a particular setup (the linear time-invariant case).

The system matrices $\mathbf{A} \in \mathbb{R}^{n\times n}$, $\mathbf{B} \in \mathbb{R}^{n\times m}$, and $\mathbf{C} \in \mathbb{R}^{p\times n}$ capture the dynamics and relationships between the control inputs $\mathbf{u}(t) \in \mathbb{R}^m$, observed outputs $\mathbf{y}(t) \in \mathbb{R}^p$, and state variables $\mathbf{x}(t) \in \mathbb{R}^n$.

Using a frequency domain analysis, the Laplace transform is applied to Eq. (3), resulting in $\mathbf{x}(s) = (s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B}\mathbf{u}(s)$. This allows us to express the transfer function of the system as $\mathbf{H}(s) = \mathbf{C}\mathbf{K}(s)^{-1}\mathbf{B}$, where $\mathbf{K}(s) = (s\mathbf{I} - \mathbf{A})^{-1}$. The Laplace transform is a mathematical integration technique commonly used in control systems and signal processing to convert differential equations into algebraic equations in the frequency domain. By analyzing the system's response to different input signals and designing controllers based on the transformed equations, specific tasks are achieved based on performance criteria.

When modeling systems that incorporate state delays and derivative observations adjustments are required. In process engineering, delayed ODEs result from time delays in system response due to transport phenomena, reaction kinetics, or feedback loops. For example, Roussel (1996) studies delayed variable enzyme catalysis and Oregonator models. These modified equations are expressed, to illustrate the required adjustments, as follows:

$$\begin{cases} \frac{d}{dt}\mathbf{x}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{A}_1\mathbf{x}(t-\tau) + \mathbf{B}\mathbf{u}(t), \\ \mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{C}_1\frac{d}{dt}\mathbf{x}(t). \end{cases} \iff \begin{cases} s\mathbf{x}(s) = (\mathbf{A} + e^{-\tau s}\mathbf{A}_1)\mathbf{x}(s) + \mathbf{B}\mathbf{u}(s), \\ \mathbf{y}(s) = (\mathbf{C} + s\mathbf{C}_1)\mathbf{x}(s). \end{cases} \tag{4}$$

Similarly, as before, using the Laplace transformation and solving for $\mathbf{x}(s)$, we get that $\mathbf{x}(s) = (s\mathbf{I} - \mathbf{A} - e^{-\tau s}\mathbf{A}_1)^{-1}\mathbf{B}\mathbf{u}(s)$. So the transfer function are written as $\mathbf{H}(s) = \frac{\mathbf{y}(s)}{\mathbf{u}(s)} = \mathbf{C}(s)\mathbf{K}(s)^{-1}\mathbf{B}$, where $\mathbf{K}(s) = (s\mathbf{I} - \mathbf{A} - e^{-\tau s}\mathbf{A}_1)^{-1}$ and $\mathbf{C}(s) = \mathbf{C} + s\mathbf{C}_1$.

Among the various intrusive methods in the family of MOR approaches, *balanced truncation (BT)* and *moment matching (MM)* are widely used, established techniques from system and control theory (Antoulas, 2005).

Balanced truncation: BT computes ROMs by eliminating states that have a negligible impact on the output of the system. BT-type methods use basic but important concepts from control theory, such as controllability, observability, and Hankel singular values (Antoulas, 2005). In the case of linear systems, they are shown to preserve the asymptotic stability of the system and also provide explicit ways to quantify the approximation error. BT-type methods have been combined with Bayesian inference (Qian et al., 2022b), used in the context of data assimilation (DA) (Lawless et al., 2008; König and Freitag, 2022), and extended to a realization-free data-driven formulation (from input–output data) (Gosea et al., 2022). For further reading, we refer the reader to Chapters 4, 5, 6, and 7 in Antoulas (2005), and to Chapter 6 in Benner et al. (2017).

Process engineering applications: In their study, Hahn and Edgar (2002) apply BT to a CSTR case study, proposing the computation of the balancing transformation matrix by empirical Gramians. Dones et al. (2011) extend BT to nonlinear systems, recommending the inclusion of all states in the balancing outputs and the use of a linearizing static transformation of the states. Their methodology is applied to a distillation column model.

Moment matching: In MM, the moments of the original system are matched to the reduced ones. These approximation techniques are computationally efficient through the use of Krylov subspace methods (Grimme, 1997), which are a class of methods used to solve linear systems and eigenvalue problems. With readily available algorithms from numerical linear algebra, MM is easy to implement. For a better understanding of interpolation-based MOR methods, we refer the readers to Chapter 8 in Antoulas et al. (2020), and Chapters 7 and 8 in Benner et al. (2017).

Process engineering applications: In Li et al. (2014), two Krylov subspace methods are proposed to speed up the computation of cyclic steady states for moving bed processes simulated with linear isotherms. Full and partial update schemes for the derivation of ROMs for a glucose-fructose separation are proposed in Yue et al. (2014). In a related approach, Shih and Shieh (1978) use MM to reduce the order of multivariable continuous and discrete systems, highlighting its utility in control system design.

Reduced basis: Traditional MOR methods, such as modal analysis or system theoretical model reduction of linear time-invariant systems, may not be directly applicable to PDEs since they do not effectively handle the spatial and temporal complexities inherent in these equations. A popular approach in the context of PDEs is the RB method. The RB projects the high-fidelity problem onto a subspace composed of relevant basis functions (Quarteroni et al., 2015). For single-domain systems, the RB method focuses on reducing the size of the system-level finite element approximation space. This is particularly beneficial when dealing with large and complex engineering systems that require discrete time steps. However, a limitation of these RB-based approaches is that the full system-level problem can still be quite large, posing computational challenges for complex engineering systems.

Process engineering applications: In the context of DT applications in aerospace engineering, Kapteyn et al. (2020) adopt the static-condensation RB method (Huynh et al., 2013) to model lithium-ion batteries with spatially resolved porous electrodes. The application of the RB method to multiscale problems is described in Ohlberger et al. (2016).

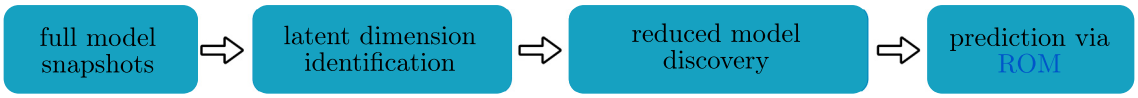

Fig. 8. A diagram illustrating non-intrusive construction of ROMs.

II Data-based, non-intrusive methods

An alternative approach to classical (intrusive) methods of MOR, which rely on explicit access to a large-scale model, is the use of data-driven (non-intrusive) techniques. Unlike intrusive methods, data-driven ROMs do not require explicit knowledge of the model structure or matrices. Instead, low-order models are directly computed from time-domain data. This makes non-intrusive methods suitable for reduction and system identification tasks where the underlying mathematical model is not known or easily accessible. The general construction of non-intrusive ROMs can be seen in Fig. 8. In the following, we highlight several prominent specific examples.

System identification: System identification (SI) encompasses various methods for deriving mathematical models of dynamic systems from observed input–output data. Acting as a bridge between practical applications and the mathematical domain of systems and control theory, SI encompasses a wide range of techniques and methodologies. These methods vary according to the structure and behavior of the models, and include linear, nonlinear, hybrid, nonparametric, and other approaches. For further reading, we recommend the influential book by Ljung (1999), which provides a comprehensive overview of established SI methods. References such as Gevers (2006), Verhaegen and Verdult (2007), and Pintelon and Schoukens (2012) further explore related aspects.

Subspace identification (SSI) is a special form of SI that is based on discrete linear systems. Among the many proposed methods, two notable ones are the MOESP algorithm (Verhaegen and Dewilde, 1992) and the N4SID approach (Van Overschee and De Moor, 1994). A comprehensive review of subspace system identification methods, including MOESP and N4SID, is provided by Favoreel et al. (2000). We further refer the reader to Qin (2006) for an extensive overview of methods developed by the mid-2000s. Extensions of SSI (e.g., applications to bilinear control systems) are proposed in Favoreel et al. (1999).

Process engineering applications: Building on the MOESP method, Borjas and Garcia (2011) introduce the MON4SID algorithm, which computes state sequences and the extended observability matrix. The authors apply SI methods to the Shell benchmark process and demonstrate that a linear model can accurately describe a nonlinear system within a certain operating range. Addressing ill-conditioning in SI, Hachicha et al. (2014) highlight the robustness of these methods for parameter estimation in process control, emphasizing the MOESP algorithm's ability to handle the Hankel block, which significantly improves parameter estimation and has important implications for process engineering. Finally, Kumari et al. (2021) present a k-nearest neighbor-based parametric ROMs that incorporates MOESP to improve numerical robustness in response to parameter changes, as demonstrated in a case study of a supercritical CO_2 release.

Dynamic mode decomposition: Dynamic mode decomposition (DMD) extracts the dominant dynamic modes of a system by analyzing time-domain snapshots of state variables (Schmid, 2010). In particular, DMD is closely related to the concept of the Koopman operator (Koopman and von Neumann, 1932), a mathematical tool that describes the evolution of scalar observables (functionals of measurable state variables) in infinite time in an infinite-dimensional Hilbert space (a mathematical space equipped with inner products and completeness properties). Through the extended DMD approach presented in Williams et al. (2015), the leading Koopman eigenfunctions are identified and a finite-dimensional representation of the underlying linear dynamics is approximated. A comprehensive understanding of DMD, including its variants such as DMD with control (DMDc) (Proctor et al., 2016), which incorporates control signals into the dynamics equation, are found in Kutz et al. (2016).

Process engineering applications: DMD is often studied and applied in the field of fluid dynamics (Schmid, 2022), which is related to process engineering, but there are also applications to process engineering itself. For example, Narasingam and Kwon (2017) propose a modification of DMDc to capture local dynamics through temporal clustering of snapshot data using mixed-integer nonlinear programming, which is applied to the feedback control of hydraulic fracturing processes to compute optimal pumping schedules. In addition, Velegar et al. (2024) present an optimized DMD algorithm for constructing an adaptive and computationally efficient prediction tool for global atmospheric chemical dynamics.

Operator inference: Operator inference (OpInf) builds ROMs in continuous time (Peherstorfer and Willcox, 2016). The basic idea is to construct polynomials of a given order in the reduced coordinates. Learning the matrices of the polynomials is theoretically possible using the least squares method. However, since the problem is often ill-conditioned and the stability of the ROM is not guaranteed, parameterization guidelines (Goyal and Benner, 2022; Pontes Duff et al., 2024) or large regularization terms (Qian et al., 2022a) are often required. For detailed principles, innovations, and applications of OpInf, we refer the reader to Benner et al. (2020, 2022), and also to the review paper of Kramer et al. (2024).

Process engineering applications: OpInf has been applied to process engineering problems, such as a single-injector rocket combustion model (McQuarrie et al., 2021) or the methanation of a carbon dioxide reactor (Peterson et al., 2024c; Gosea et al., 2024).

Frequency domain analysis methods: Frequency domain analysis methods approximate and analyze the behavior of a system using its transfer function, which provides insight into the system's response to a range of frequencies actuated by the control input. One notable approach is the *Loewner framework* (Mayo and Antoulas, 2007), which uses interpolation to construct ROMs using rational approximation. By selecting interpolation points, the Loewner framework recovers the dynamics of the system and provides a priori estimates of the complexity of the system (encoded in the singular values of the Loewner matrix). A major advantage of the Loewner approach is its directness and ease of implementation, avoiding complicated optimization schemes. Another method is the *vector fitting algorithm* (Gustavsen and Semlyen, 1999), which uses a linearized least-squares fitting approach. This algorithm iteratively adjusts the parameters of the rational functions to minimize the mismatch between the model and the actual data, resulting in an accurate recovery of the transfer function. In addition, the *adaptive Antoulas-Anderson (AAA) algorithm* (Nakatsukasa et al., 2018) combines elements of both interpolation and least squares fitting. It aims at finding an approximation by iteratively adjusting the model based on the greedily selected interpolation points and on the least squares fit.

Process engineering applications: The Loewner framework has recently been applied to electrochemistry test cases, by fitting ROMs constructed from impedance data corresponding to models of polymer electrolyte membrane (PEM) fuel cells (Sorrentino et al., 2023; Gosea et al., 2023) and of lithium-ion batteries (Rüther et al., 2023). In addition, the vector fitting algorithm has been used in Hu et al. (2012) for battery thermal management for high-power applications (in the context of hybrid and electric vehicles).

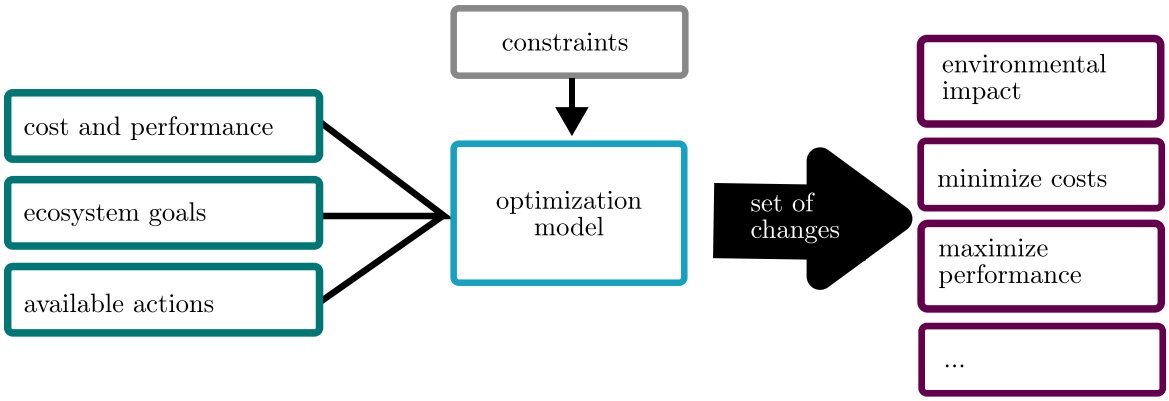

Fig. 9. A generic optimization scheme.

4.2. Optimization loop

The optimization loop contains methods that repeatedly invoke the model of the physical problem. Many of these methods are based on optimization (e.g., parameter fitting, scheduling optimal control), hence the name. In this section, we will discuss optimization, inverse problems, DA, UQ, and control.

4.2.1. Optimization

As stated in Asch (2022), optimization is fundamental to the development of DT. Typically, optimization involves finding the minimum or maximum of a given function. Fig. 9 shows a generic optimization scheme. Several steps are needed when formulating an optimization problem:

- Construct a model that describes the underlying process or plant and the associated optimization problem.
- Define a suitable objective (cost) function.
- Choose an optimization algorithm.
- Verify (potential) optimality conditions
- Perform sensitivity analysis of the results.

Optimization problems come in many forms, ranging from discrete to continuous, constrained to unconstrained, and local to global. Traditional optimization methods rely on having an explicit function or numerical approximation to optimize. However, when such explicit knowledge is lacking, finite difference approximations or the ability to compute the gradient of a quadratic cost function are used. Classical optimization methods typically provide only local minimum guarantees, while achieving a global minimum often requires specialized techniques such as simulated annealing, genetic algorithms, or swarm optimization (Mitchell, 1998).

Optimization tools are divided into three main categories: First-order or *gradient methods*, second-order or *Newtonian methods*, and *stochastic methods*. *Gradient methods* are typically used when the objective function is smooth and its gradient is easy to compute. *Newtonian methods* are used in cases where the objective function is highly nonlinear and second-order information (i.e., the Hessian) is available and can improve convergence speed. *Stochastic methods* are useful in scenarios where the objective function is noisy, high-dimensional, or computationally expensive to evaluate, as they can efficiently explore the search space without relying on gradient information. Extensive discussions and detailed descriptions of prominent optimization approaches can be found in Nocedal and Wright (1999) and Locatelli and Schoen (2013). In the field of process optimization, a comprehensive coverage of nonlinear programming methods, covering both steady-state and dynamic process optimization, is provided in Biegler (2010).

Process engineering applications: Process engineering often involves complex problems with multiple solutions that interact economically and in terms of performance, making it difficult to intuitively identify the optimal solution. To address this growing complexity, the Integrated Platform for Advanced Process Modeling and Simulation (IDAES) provides a framework that combines process simulators and algebraic modeling languages (Lee et al., 2021). Tailored for process flowsheet optimization, the platform uses advanced solvers and techniques to model and optimize dynamic, interconnected systems. It supports structure-based design and optimization under uncertainty, providing the flexibility to adapt to industrial needs.

Several notable works highlight optimization in process engineering. Genetic algorithms have been applied to computer-aided molecular design in Venkatasubramanian et al. (1994), while hybrid approaches to modeling intracellular dynamics in biochemical applications are explored in del Rio-Chanona et al. (2016), Teixeira et al. (2006) and Zhang et al. (2019a). A gray-box modeling approach for chemical process optimization is proposed in Asprion et al. (2019), and Kumar et al. (2016) optimize furnace temperature distribution in a steam methane reformer using ROM. In Lima et al. (2007), a symbolic representation of fundamental equations for hybrid mechanistic-empirical models is developed and applied to the optimization of the Williams-Otto benchmark reactor. In renewable energy-to-chemicals applications, Uebbing et al. (2020) optimize power-to-methane processes via superstructure optimization, focusing on heat integration and gas grid requirements. Uebbing et al. (2021) optimize PSA processes with reduced models and a trust-region filtering method, significantly reducing computational time. In addition, Garmatter et al. (2021) develop a benchmark structure for syngas production that addresses the complexity of dynamic modeling.

4.2.2. Inverse problems

Mathematically, two problems are considered inverse if solving one requires addressing all or part of the solution to the other. One is called a *direct problem* and the other is called an *inverse problem*. The *direct problem*, also known as the forward problem, involves solving the physical equations to obtain the dependent variable of interest when all other quantities are known. These problems are solved using established tools from numerical linear algebra in the linear case, or iterative descent methods in the nonlinear case. In contrast, the *inverse problem* aims to go from the effects to the cause by reconstructing the unknown quantities based on observed data. The interaction of the forward and inverse models is shown in Fig. 10.

Inverse problems are further classified as deterministic or statistical, and linear or nonlinear, depending on the type of algebraic or differential operators involved. Solving *inverse problems* is typically more challenging than solving *direct problems* because they involve incomplete information, measurement errors, and noise, which can lead to ill-posed solutions with unstable or non-unique properties. Addressing these challenges requires specially designed algorithms that deal with errors in the measurement data.

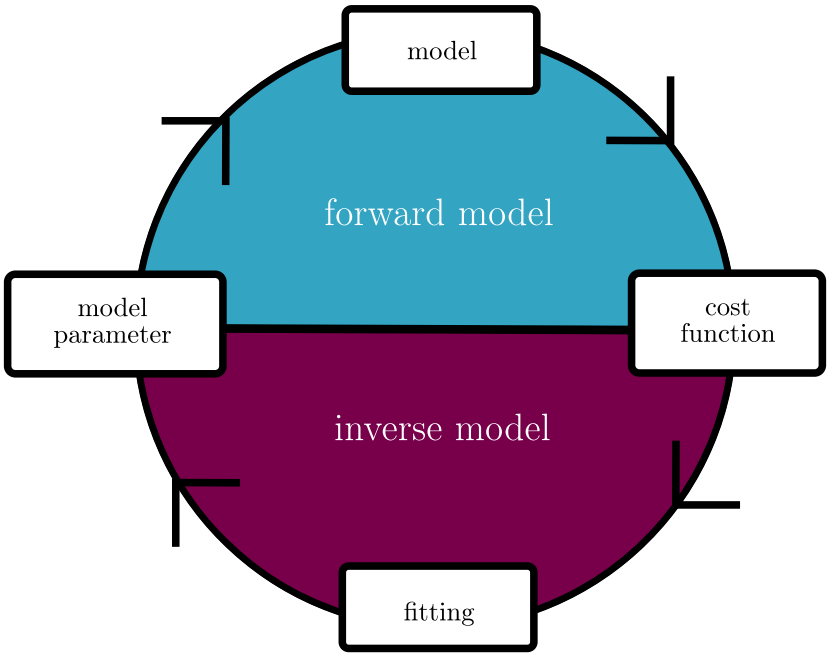

Fig. 10. A generic schematic for inverse modeling.

We follow the general mathematical formulation of an inverse problem from Asch (2022), given by the equality $\mathbf{d} = g(\mathbf{m})$, where $\mathbf{m}$ is a vector of parameters in the model parameter space $\mathcal{M}$, $\mathbf{d}$ is a vector of data values in the data space $\mathcal{D}$, and g is an operator that describes the forward model. The operator g is defined by differential, algebraic, integral, or matrix equations. In practical scenarios, the data values are often corrupted by additive (Gaussian) noise, leading to the modification of the relationship, as: $\mathbf{d} = g(\mathbf{m}) + \eta$. Then, the data vectors are connected to the prediction vector using the relation $\mathbf{y} = h(\mathbf{d}) = h(g(\mathbf{m}))$, where $\mathbf{y}$ represents the vector of predictions or observations, $\mathcal{H}$ is the observation space, and h is the observation operator.

For a comprehensive understanding of the available regularization techniques and strategies in inverse problems, we recommend reading Kirsch (2011). In addition, the review by Ghattas and Willcox (2021) highlights significant advances in efficient algorithms for inverse problems related to MOR of large models. As outlined there, the distinction between the two approaches is that inverse problems involve inferring uncertain input components from output observations, whereas MOR focuses on obtaining low-dimensional models that accurately represent the main features of the input–output relationship by approximating them within a reduced subspace.

Process engineering applications: In process engineering, inverse modeling aims to infer the unknown independent variables, or inputs, of a system from its observed outputs. For example, in materials' design, Venkatasubramanian (2019) use inverse problems for the discovery and design of new materials with desired properties. In process control, inverse modeling techniques enable the control and optimization of process variables, as demonstrated in the control of distillation plants using ANN controllers (Savkovic-Stevanovic, 1996). In addition, inverse modeling contributes to process understanding by quantifying the effects of time variations and production dynamics, allowing engineers to gain insight into the relationships between process variables and system behavior (Tomba et al., 2014). Moreover, inverse modeling is used in process optimization and parameter estimation in the food processing industry, where techniques such as model validation, optimization algorithms, and sensitivity analysis improve process efficiency and product quality (Reddy et al., 2022). The integration of data-driven models into a broader decision framework is facilitated by the OMLT software package, which provides the essential optimization equations (Ceccon et al., 2022). This concept is further extended to incorporate surrogate models into broader design or operations problems involving complex flowsheets.

4.2.3. Data assimilation

As illustrated in Fig. 11, data assimilation (DA) merges knowledge from a numerical model of a system with newly acquired observational data (Bocquet et al., 2015). The goal is to accurately predict the state of the system while accounting for uncertainties. The primary benefit of DA is to improve predictions, reduce model uncertainties, and adjust model parameters.

Two main categories of DA are typically prevalent, i.e., *variational* DA and *statistical* (or sequential) DA. *Variational* DA optimally combines model and data by minimizing a given criterion (typically a cost function). Key methods include 3D-Var and 4D-Var approaches as well as nudging techniques. The goal is to improve predictions by adjusting model parameters based on observations. *Statistical* DA uses a set of model trajectories or scenarios that are intermittently updated according to data. It infers the past, present, or future state of a system. Ensemble Kalman Filter approaches, Bayesian methods, and nonlinear filtering all fall into this category. Finally, hybrid methods combine both approaches, i.e., hybrid ensemble-variational (EnVar) DA, such as 3D-EnVar in Hamill and Snyder (2000), or 4DEnVar in Lorenc (2003), which combine 3D-Var and 4D-Var with ensemble, statistical approaches.

Process engineering applications: Originally developed for computational geosciences, DA has found applications in various fields, including process engineering. In this context, DA techniques integrate real-time measurements into model predictions, improving prediction accuracy and supporting real-time process monitoring, control, and optimization (Law et al., 2015). By addressing model deficiencies and uncertainties, DA improves both process understanding and decision-making capabilities (Law et al., 2015). Despite these benefits, further evaluation is needed to fully assess the effectiveness of DA methods in process engineering applications (Bocquet et al., 2015).

4.2.4. Uncertainty quantification analysis

Uncertainty quantification (UQ) provides a systematic approach to managing uncertainty from sources such as measurement error, model

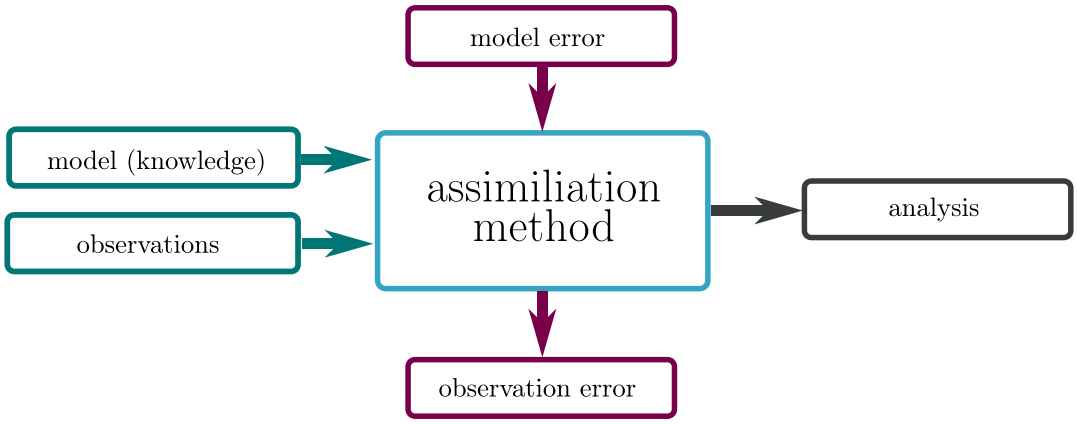

Fig. 11. A generic scheme for data assimilation (DA).

inaccuracy, and parameter variability. A key component of UQ is the assessment of uncertainty in the computational estimation of a quantity of interest (QoI), focusing on statistical information such as expected values, variances, and probability distributions. A comprehensive reference on UQ theory and practice is Smith (2013). UQ are divided into *forward propagation* and *inverse evaluation*.

Forward propagation: Forward propagation assesses how uncertainties propagate through a model to predict the overall uncertainty in the system response. Existing approaches to forward propagation include *probabilistic* and *non-probabilistic* approaches. *Probabilistic approaches* treat uncertainties as random variables and use probability distributions to represent them. *Non-probabilistic approaches* work with small sample sizes and are often used when the credibility of probabilistic analysis results is questioned due to small sample sizes.

Inverse evaluation: Inverse evaluation focuses on uncertainties associated with the model or its parameters. Two main approaches in inverse UQ are *Frequentist* and *Bayesian*. *Frequentist* methods use statistical techniques that treat model parameters as fixed but unknown. Uncertainties are estimated from experimental data.

In contrast, *Bayesian* methods quantify uncertainty through probability distributions. The key steps in the Bayesian approach are: (1) defining the prior distribution, (2) specifying the noise distribution, (3) computing the likelihood function, (4) applying Bayes' law to derive the posterior distribution, and (5) predicting the experimental response and discrepancy function. Numerical methods, such as those described in Table 4, are crucial for efficient Bayesian inference, especially when analytical solutions are intractable.

Gaussian processes (GP) techniques extend the Bayesian approach by modeling both the function and its uncertainty through probability distributions. This provides not only function estimates but also their associated uncertainties, which helps to manage constraints and ensure that they are satisfied with some probability (Roberts et al., 2013). For example, Raissi et al. (2017) show how modifying GP priors using differential operators allows parameter inference from limited or noisy data. Integrating GPs into the Bayesian framework improves the robustness and reliability of uncertainty quantification.

Process engineering applications: UQ plays a pivotal role in process engineering, providing a robust framework for managing and reducing uncertainties. For example, Thelen (2023) addresses the quantification of epistemic uncertainty, which arises from our limited knowledge, as opposed to aleatory uncertainty inherent in the system itself. Their approach is illustrated with a case study of optimizing the decommissioning of a lithium-ion battery cell. In Rafiei and Ricardez-Sandoval (2018), a MC sampling strategy for joint uncertainty propagation is applied to facilitate constraint relaxation in a water treatment plant scenario. Additionally, in Bradford et al. (2020), a stochastic MPC approach incorporating GPs is employed for constraint relaxation in a batch reactor. Similarly, Bradford et al. (2018) use GPs for optimization of lutein production from microalgae.

4.2.5. Control

As discussed in Section 2.1, the DTs modeling framework not only represents real-world systems but also actively interacts with and influences them. Control is a key aspect of this interaction, aimed at managing dynamic systems to achieve desired states while minimizing undesired behavior such as lags, overshoots, and deviations from steady state. Effective control within DTs facilitates real-time optimization, decision-making, and adaptability to system changes (Tao et al., 2018). An example of a control scheme is shown in Fig. 12. For a deeper understanding of control theory, the reader is referred to Franklin et al. (1998) and Ogata (2010).

To enable control, two considerations must be addressed: *Observability* and *controllability*. *Observability* focuses on the controller's ability to monitor the system state through output measurements, while *Controllability* evaluates the effectiveness of the control signal in steering the system toward the desired state.

A variety of non-model-based control strategies are available. Classical *controllers* (P, I, or PID) adjust control inputs based on the difference between a desired setpoint and the actual state of the system. *Fuzzy logic control*, which uses linguistic variables instead of precise numerical values, effectively handles uncertainty and imprecision. *Reinforcement learning*, a type of machine learning, allows an agent to optimize future actions based on rewards or penalties received from interactions with its environment. Importantly, model-based and non-model-based strategies are synergistically combined. For example, integrating a classical controller into an optimal control framework improves robustness to model inaccuracies and unanticipated disturbances.

When suitable models are available, two main techniques for model-based control can be applied: *Optimal control* and *MPC*. *Optimal control* determines the best control strategy for a dynamic system over time to optimize an objective. *MPC* uses a discrete-time model of the process to predict its future behavior based on input signals. This allows *MPC* to compute input signals that satisfy certain quality criteria and lead to optimal outputs (Allgöwer et al., 2004).

Process engineering applications: Control systems play a critical role in chemical engineering, ensuring that processes run efficiently, safely, and sustainably. Among the most important standards in this area are the National Electric Code (NFPA 70), which sets benchmarks for safe electrical design and installation, and the ISA-88 and ISA-95 standards, which provide frameworks for batch control and enterprise control system integration, respectively (Alford and Buckbee, 2020). While classical controllers are often used in industry, the academic literature frequently explores MPC, with notable studies including Allgöwer et al.

Table 4
Methods for numerical integration and approximation within the Bayesian method.

Method	Description
Numerical quadrature schemes	Provide a straightforward approach for integral evaluation but may be impractical for high-dimensional problems. Examples include Riemann sums, the trapezoid rule, Gauss–Legendre, Gauss–Kronrod, Newton-Cotes, and Simpson rules.
Laplace approximations	Used to approximate integrals involving exponential functions. It approximates the posterior distribution as a Gaussian centered around the maximum of the log-posterior. Effective for unimodal distributions but may not capture multimodal distributions accurately (Shun and McCullagh, 1995).
MC integration	A general method for numerical integration using random sampling. Offers flexibility and simplicity but can suffer from convergence issues, especially in high dimensions (Hammersley and Handscomb, 1964).
MCMC methods	Combines MC integration with Markov chain theory to compute posterior distributions. Methods such as Metropolis–Hastings and Gibbs sampling are commonly used for sampling from the target distribution (Hastings, 1970).

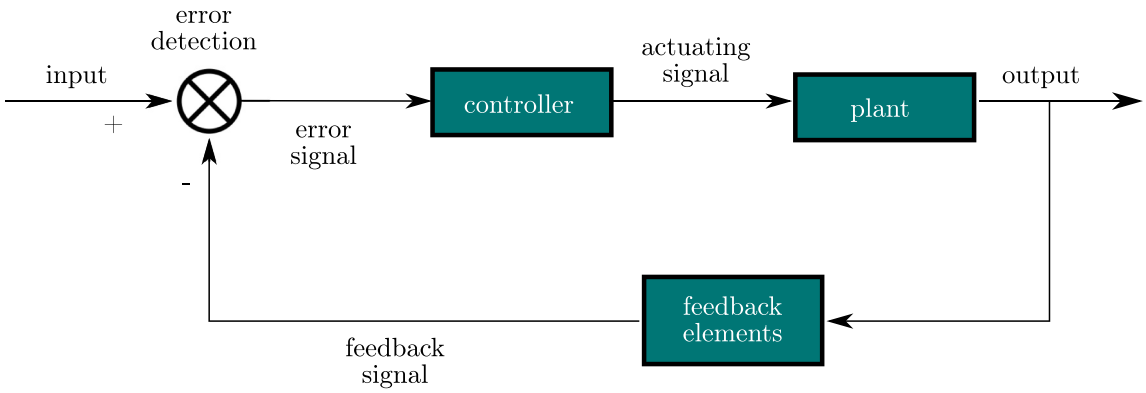

Fig. 12. A generic closed-loop control scheme.

(2004), Morari and Lee (1999) and Schwenzer et al. (2021). A recent development in process engineering is the integration of machine learning into control systems (Himmel et al., 2024; Wu et al., 2019b; Zhang et al., 2019b), which enables adaptive and self-learning processes for more precise control under dynamic conditions. Comprehensive reviews of control in chemical engineering are provided by Bequette (2003), Prett and García (2013) and Stephanopoulos (1984).

5. Conclusions and outlook

DTs offer a transformative potential for process engineering, enabling real-time monitoring, future state prediction, and process optimization. They improve safety and reduce environmental impact by facilitating early risk detection. Despite this promise, the widespread use of DTs remains limited. The primary challenge lies not in the individual components themselves, but in the interdisciplinary integration of these diverse subfields. Our work focuses on addressing this challenge by advancing a key component: the computational tools and numerical methods essential for DTs development.

In this review, we employ the hierarchical loop concept as an organizational framework to structure our discussion of computational tools and methods into three interrelated loops: the physical problem-solving loop, the optimization loop, and the decision loop. The physical problem-solving loop includes numerical simulations, including mechanistic and data-driven approaches, with an emphasis on surrogate and reduced-order models. The optimization loop includes optimization techniques, UQ, co-simulation, control, and inverse problem-solving. Finally, the decision loop ensures the reliability of DTs based on the results of the first two loops. Throughout this review, we have provided practical examples from process engineering that illustrate the application of these tools.

Further advances in fast, robust modeling techniques will accelerate the full realization of DTs. As technologies continue to evolve, building comprehensive DTs for process engineering will become more feasible, offering significant potential for improved industrial operations.

List of acronyms

AAA adaptive Antoulas-Anderson

ANN artificial neural network

AI artificial intelligence

BT balanced truncation

CSTR continuous stirred tank reactor

DA data assimilation

DAE differential–algebraic equation

DEIM discrete empirical interpolation method

DMD dynamic mode decomposition

DMDc DMD with control

DOE Design of experiments

DT digital twin

EPF elementary process function

FDM finite difference method

FEM finite element method

FOM full order model

FVM finite volume method

GP Gaussian processes

IoT internet of things

MC Monte Carlo

MM moment matching

MOR model-order reduction

MPC model predictive control

ODE ordinary differential equation

OpInf operator inference

PCA principle components analysis

PEM polymer electrolyte membrane

PDE partial differential equation

PINN physics-informed neural network

POD proper orthogonal decomposition

PSA pressure swing adsorption

QoI quantity of interest

ROM reduced-order model

RB reduced basis

SciML scientific machine learning

SGD stochastic gradient descent

SI sparse identification

SSI subspace identification

SVM support vector machine

SI system identification

UQ uncertainty quantification

CRediT authorship contribution statement

Luisa Peterson: Writing – original draft, Visualization, Investigation, Formal analysis, Conceptualization. **Ion Victor Gosea:** Writing – original draft, Visualization, Supervision, Investigation, Formal analysis, Conceptualization. **Peter Benner:** Writing – review & editing, Validation, Supervision, Funding acquisition, Conceptualization. **Kai Sundmacher:** Writing – original draft, Validation, Supervision, Funding acquisition, Conceptualization.

Declaration of competing interest

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

Acknowledgments

This work is part of the research initiative "SmartProSys: Intelligent Process Systems for the Sustainable Production of Chemicals" funded by the Ministry for Science, Energy, Climate Protection and the Environment of the State of Saxony-Anhalt. Additionally, this work is partially funded by the Bundesministerium für Bildung und Forschung (BMBF), Germany and Project Management Jülich (PtJ) under grant 03HY302R.

Data availability

No data was used for the research described in the article.

References

Abdullah, F., Wu, Z., Christofides, P.D., 2021. Data-based reduced-order modeling of nonlinear two-time-scale processes. Chem. Eng. Res. Des. 166, 1–9. http://dx.doi.org/10.1016/j.cherd.2020.11.009.

Abramovici, M., Göbel, J.C., Savarino, P., 2017. Reconfiguration of smart products during their use phase based on virtual product twins. CIRP Ann 66 (1), 165–168. http://dx.doi.org/10.1016/j.cirp.2017.04.042.

Agarwal, A., Biegler, L.T., Zitney, S.E., 2009. Simulation and optimization of pressure swing adsorption systems using reduced-order modeling. Ind. Eng. Chem. Res. 48 (5), 2327–2343. http://dx.doi.org/10.1021/ie071416p.

Alford, J., Buckbee, G., 2020. Industrial process control systems: a new approach to education. Chem. Eng. Prog. 74 (12), 35–42, https://www.aiche.org/resources/publications/cep/2020/december/industrial-process-control-systems-new-approach-education.

Allgöwer, F., Findeisen, R., Nagy, Z.K., 2004. Nonlinear model predictive control: From theory to application. J. Chin. Inst. Chem. Eng. 35 (3), 299–316, https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=47f3da67e18def14952a66e0bf0c7594e5043b18.

Ammar, A., Ben Saada, M., Cueto, E., Chinesta, F., 2024. Casting hybrid twin: physics-based reduced order models enriched with data-driven models enabling the highest accuracy in real-time. Int. J. Mater. Form. 17 (2), 16. http://dx.doi.org/10.1007/s12289-024-01812-4.

Antoulas, A.C., 2005. Approximation of Large-Scale Dynamical Systems. In: Advances in Design and Control, SIAM, Philadelphia, PA, USA, http://dx.doi.org/10.1137/1.9780898718713.

Antoulas, A.C., Beattie, C.A., Gugercin, S., 2020. Interpolatory Methods for Model Reduction. In: Computational Science & Engineering, SIAM, Philadelphia, PA, USA, http://dx.doi.org/10.1137/1.9781611976083.

Asch, M., 2022. A Toolbox for Digital Twins: From Model-Based to Data-Driven. In: Mathematics in Industry, SIAM, http://dx.doi.org/10.1137/1.9781611976977.

Ascher, U.M., Petzold, L.R., 1998a. Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations, vol. 61, SIAM, http://dx.doi.org/10.1137/1.9781611971392.

Ascher, U.M., Petzold, L.R., 1998b. Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations. SIAM, Philadelphia, http://dx.doi.org/10.1137/1.9781611971392.

Aspentech Digital Twin, 2023. Digital twin technology (aspentech). https://www.aspentech.com/en/apm-resources/digital-twin-technology. (Accessed 31 January 2023).

Asprion, N., Böttcher, R., Pack, R., Stavrou, M.-E., Höller, J., Schwientek, J., Bortz, M., 2019. Gray-box modeling for the optimization of chemical processes. Chem. Ing. Tech. 91 (3), 305–313. http://dx.doi.org/10.1002/cite.201800086.

Aversano, G., Bellemans, A., Li, Z., Coussement, A., Gicquel, O., Parente, A., 2019. Application of reduced-order models based on PCA & Kriging for the development of digital twins of reacting flow applications. Comput. Chem. Eng. 121, 422–441. http://dx.doi.org/10.1016/j.compchemeng.2018.09.022.

AVEVA Digital Twin, 2023. A strategic approach to digital twin (aveva group plc). https://engage.aveva.com/strategic-approach-digital-twin. (Accessed 31 January 2023).

Balta, E.C., Pease, M., Moyne, J., Barton, K., Tilbury, D.M., 2023. Digital twin-based cyber-attack detection framework for cyber-physical manufacturing systems. IEEE Trans. Autom. Sci. Eng. http://dx.doi.org/10.1109/TASE.2023.3243147.

Bamberg, A., Urbas, L., Bröcker, S., Bortz, M., Kockmann, N., 2021. The digital twin – your ingenious companion for process engineering and smart production. Chem. Eng. Technol. 44 (6), 954–961. http://dx.doi.org/10.1002/ceat.202000562.

Bangi, M.S.F., Kwon, J.S.-I., 2020. Deep hybrid modeling of chemical process: Application to hydraulic fracturing. Comput. Chem. Eng. 134, 106696. http://dx.doi.org/10.1016/j.compchemeng.2019.106696.

Baur, U., Benner, P., Feng, L., 2014. Model order reduction for linear and nonlinear systems: A system-theoretic perspective. Arch. Comput. Methods Eng. 21 (4), 331–358. http://dx.doi.org/10.1007/s11831-014-9111-2.

Benner, P., Goyal, P., Heiland, J., Pontes Duff, I., 2022. Operator inference and physics-informed learning of low-dimensional models for incompressible flows. Electron. Trans. Numer. Anal. 56, 28–51. http://dx.doi.org/10.1553/etna_vol56s28.

Benner, P., Goyal, P., Kramer, B., Peherstorfer, B., Willcox, K., 2020. Operator inference for non-intrusive model reduction of systems with non-polynomial nonlinear terms. Comput. Methods Appl. Mech. Engrg. 372, http://dx.doi.org/10.1016/j.cma.2020.113433.

Benner, P., Gugercin, S., Willcox, K., 2015. A survey of projection-based model reduction methods for parametric dynamical systems. SIAM Rev. 57 (4), 483–531. http://dx.doi.org/10.1137/130932715.

Benner, P., Ohlberger, M., Cohen, A., Willcox, K., 2017. Model Reduction and Approximation. In: Computational Science & Engineering, Society for Industrial and Applied Mathematics, Philadelphia, PA, http://dx.doi.org/10.1137/1.9781611974829.

Benner, P., Schilders, W., Grivet-Talocia, S., Quarteroni, A., Rozza, G., Silveira, L.M., 2021. Model Order Reduction. Vol. I-III, De Gruyter, Berlin, Boston, https://www.degruyter.com/serial/mor-b/html.

Benner, P., Stykel, T., 2017. Model order reduction for differential-algebraic equations: A survey. In: Ilchmann, A., Reis, T. (Eds.), Surveys in Differential-Algebraic Equations IV. In: Differential-Algebraic Equations Forum, Springer International Publishing, Cham, pp. 107–160. http://dx.doi.org/10.1007/978-3-319-46618-7_3.

Bequette, B.W., 2003. Process Control: Modeling, Design, and Simulation. Prentice Hall Professional, ISBN: 978-0134033754.

Bequette, B.W., 2019. 110Th anniversary: commentary: the smart human in smart manufacturing. Ind. Eng. Chem. Res. 58 (42), 19317–19321. http://dx.doi.org/10.1021/acs.iecr.9b03544.

Bhadriraju, B., Bangi, M.S.F., Narasingam, A., Kwon, J.S.-I., 2020. Operable adaptive sparse identification of systems: Application to chemical processes. AIChE J. 66 (11), e16980. http://dx.doi.org/10.1002/aic.16980.

Bhadriraju, B., Narasingam, A., Kwon, J.S.-I., 2019. Machine learning-based adaptive model identification of systems: Application to a chemical process. Chem. Eng. Res. Des. 152, 372–383. http://dx.doi.org/10.1016/j.cherd.2019.09.009.

Bhutani, N., Rangaiah, G., Ray, A., 2006. First-principles, data-based, and hybrid modeling and optimization of an industrial hydrocracking unit. Ind. Eng. Chem. Res. 45 (23), 7807–7816. http://dx.doi.org/10.1021/ie060247q.

Biegler, L.T., 2010. Nonlinear Programming: Concepts, Algorithms, and Applications to Chemical Processes. In: MOS-SIAM Series on Optimization, SIAM, http://dx.doi.org/10.1137/1.9780898719383.

Bizon, K., 2023. A journey from mechanistic to data-driven models in process engineering: dimensionality reduction, surrogate and hybrid approaches, and digital twins. Chem. Process Eng.: New Front. http://dx.doi.org/10.24425/cpe.2023.146726.

Bocquet, M., Elbern, H., Eskes, H., Hirtl, M., Žabkar, R., Carmichael, G., Flemming, J., Inness, A., Pagowski, M., Pérez Camaño, J., 2015. Data assimilation in atmospheric chemistry models: current status and future prospects for coupled chemistry meteorology models. Atmos. Chem. Phys. 15 (10), 5325–5358. http://dx.doi.org/10.5194/acp-15-5325-2015.

Borjas, S.D.M., Garcia, C., 2011. Subspace identification for industrial processes. TEMA Tend. Mat. Apl. Comput. 12, 183–194, doi:tema.2011.012.03.0183.

Both, G.-J., Choudhury, S., Sens, P., Kusters, R., 2021. DeepMoD: Deep learning for model discovery in noisy data. J. Comput. Phys. 428, 109985. http://dx.doi.org/10.1016/j.jcp.2020.109985.

Bradford, E., Imsland, L., Zhang, D., del Rio Chanona, E.A., 2020. Stochastic data-driven model predictive control using Gaussian processes. Comput. Chem. Eng. 139, 106844. http://dx.doi.org/10.1016/j.compchemeng.2020.106844.

Bradford, E., Schweidtmann, A.M., Zhang, D., Jing, K., del Rio-Chanona, E.A., 2018. Dynamic modeling and optimization of sustainable algal production with uncertainty using multivariate Gaussian processes. Comput. Chem. Eng. 118, 143–158. http://dx.doi.org/10.1016/j.compchemeng.2018.07.015.

Bremer, J., Goyal, P., Feng, L., Benner, P., Sundmacher, K., 2016. Nonlinear model order reduction for catalytic tubular reactors. In: Computer Aided Chemical Engineering. Vol. 38, Elsevier, pp. 2373–2378. http://dx.doi.org/10.1016/B978-0-444-63428-3.50400-8.

Bremer, J., Goyal, P., Feng, L., Benner, P., Sundmacher, K., 2017. POD-DEIM for efficient reduction of a dynamic 2D catalytic reactor model. Comput. Chem. Eng. 106, 777–784. http://dx.doi.org/10.1016/j.compchemeng.2017.02.032.

Ceccon, F., Jalving, J., Haddad, J., Thebelt, A., Tsay, C., Laird, C.D., Misener, R., 2022. OMLT: Optimization & machine learning toolkit. J. Mach. Learn. Res. 23 (1), https://www.jmlr.org/papers/volume23/22-0277/22-0277.pdf.

Chaturantabut, S., Sorensen, D.C., 2010. Nonlinear model reduction via discrete empirical interpolation. SIAM J. Sci. Comput. 32 (5), 2737–2764. http://dx.doi.org/10.1137/090766498.

Chen, R.T., Rubanova, Y., Bettencourt, J., Duvenaud, D.K., 2018. Neural ordinary differential equations. Adv. Neural Inf. Process. Syst. 31, https://proceedings.neurips.cc/paper_files/paper/2018/file/69386f6bb1dfed68692a24c8686939b9-Paper.pdf.

Chen, X., Yan, X., 2012. Using improved self-organizing map for fault diagnosis in chemical industry process. Chem. Eng. Res. Des. 90 (12), 2262–2277. http://dx.doi.org/10.1016/j.cherd.2012.06.004.

Chen, Y., Yang, O., Sampat, C., Bhalode, P., Ramachandran, R., Ierapetritou, M., 2020. Digital twins in pharmaceutical and biopharmaceutical manufacturing: a literature review. Processes 8 (9), 1088. http://dx.doi.org/10.3390/pr8091088.

Chinesta, F., Cueto, E., Abisset-Chavanne, E., Duval, J.L., Khaldi, F.E., 2020. Virtual, digital and hybrid twins: a new paradigm in data-based engineering and engineered data. Arch. Comput. Methods Eng. 27 (1), 105–134. http://dx.doi.org/10.1007/s11831-018-9301-4.

Cohen, T., Welling, M., 2016. Group equivariant convolutional networks. In: International Conference on Machine Learning. PMLR, pp. 2990–2999, https://proceedings.mlr.press/v48/cohenc16.pdf.

Cuomo, S., Di Cola, V.S., Giampaolo, F., Rozza, G., Raissi, M., Piccialli, F., 2022. Scientific machine learning through physics–informed neural networks: Where we are and what's next. J. Sci. Comput. 92 (3), 88. http://dx.doi.org/10.1007/s10915-022-01939-z.

Data Digital Twin, 2023. Supercomputing superpowers: NVIDIA brings digital twin simulation to HPC data center operators. https://blogs.nvidia.com/blog/2022/11/14/omniverse-digital-twin-data-center/. (Accessed 17 March 2023).

de Beer, J., Depew, C., 2021. The role of process engineering in the digital transformation. Comput. Chem. Eng. 154, 107423. http://dx.doi.org/10.1016/B978-0-12-823377-1.50121-X.

de Carvalho Servia, M.Á., Sandoval, I.O., Zhang, D., Hellgardt, K., Hii, K.K.M., del Rio Chanona, E.A., 2023. Automated kinetic model discovery-a methodological framework. In: Computer Aided Chemical Engineering. Vol. 52, Elsevier, pp. 33–38. http://dx.doi.org/10.1016/B978-0-443-15274-0.50006-8.

del Rio-Chanona, E.A., Manirafasha, E., Zhang, D., Yue, Q., Jing, K., 2016. Dynamic modeling and optimization of cyanobacterial C-phycocyanin production process by artificial neural network. Algal Res. 13, 7–15. http://dx.doi.org/10.1016/j.algal.2015.11.004.

Deringer, V.L., Bartók, A.P., Bernstein, N., Wilkins, D.M., Ceriotti, M., Csányi, G., 2021. Gaussian process regression for materials and molecules. Chem. Rev. 121 (16), 10073–10141. http://dx.doi.org/10.1021/acs.chemrev.1c00022.

Dittler, D., Häbig, P., Möß ner, N., Demirel, G., Müller, T., Jazdi, N., Hufendieck, K., Weyrich, M., 2022. Digitaler zwilling für eine modulare offshore-plattform: Effizienzsteigerung grüner power-to-X-Produktionsprozesse. ATP Mag. 64 (6–7), 72–80. http://dx.doi.org/10.17560/atp.v63i6-7.2606.

Dones, I., Skogestad, S., Preisig, H.A., 2011. Application of balanced truncation to nonlinear systems. Ind. Eng. Chem. Res. 50 (17), 10093–10101. http://dx.doi.org/10.1021/ie200706d.

Eason, J.P., Biegler, L.T., 2021. Model reduction in chemical process optimization. In: Benner, P., Grivet-Talocia, S., Quarteroni, A., Rozza, G., Schilders, W., Silveira, L.M. (Eds.), Volume 3 Applications. De Gruyter, Berlin, Boston, pp. 1–32. http://dx.doi.org/10.1515/9783110499001-001.

Edington, L., Dervilis, N., Ben Abdessalem, A., Wagg, D., 2023. A time-evolving digital twin tool for engineering dynamics applications. Mech. Syst. Signal Process. 188, 109971. http://dx.doi.org/10.1016/j.ymssp.2022.109971.

Eppinger, T., Longwell, G., Mas, P., Goodheart, K., Badiali, U., Aglave, R., 2021. Increase food production efficiency using the executable digital twin (xDT). Chem. Eng. Trans. 87, 37–42. http://dx.doi.org/10.3303/CET2187007.

Essential Tools of Machine Learning, 2023. The essential tools of scientific machine learning (scientific ML). https://www.stochasticlifestyle.com/the-essential-tools-of-scientific-machine-learning-scientific-ml/. (Accessed 21 June 2023).

Eyke, N.S., Green, W.H., Jensen, K.F., 2020. Iterative experimental design based on active machine learning reduces the experimental burden associated with reaction screening. React. Chem. Eng. 5 (10), 1963–1972. http://dx.doi.org/10.1039/D0RE00232A.

Favoreel, W., De Moor, B., van Overschee, P., 1999. Subspace identification of bilinear systems subject to white inputs. IEEE Trans. Autom. Control 44 (6), 1157–1165. http://dx.doi.org/10.1109/9.769370.

Favoreel, W., De Moor, B., van Overschee, P., 2000. Subspace state space system identification for industrial processes. J. Process Control 10 (2–3), 149–155. http://dx.doi.org/10.1016/S0959-1524(99)00030-X.

Ferrari, A., Willcox, K., 2024. Digital twins in mechanical and aerospace engineering. Nat. Comput. Sci. 4 (3), 178–183. http://dx.doi.org/10.1038/s43588-024-00613-8.

Forootani, A., Benner, P., 2024. GN-SINDy: Greedy sampling neural network in sparse identification of nonlinear partial differential equations. arXiv:2405.08613.

Franklin, G.F., Powell, J.D., Workman, M.L., et al., 1998. Digital Control of Dynamic Systems, vol. 3, Addison-Wesley, Menlo Park, CA, ISBN: 0-9791226-3-5, available at https://www.researchgate.net/publication/31849881_Digital_Control_of_Dynamic_Systems-Third_Edition.

Freund, H., Sundmacher, K., 2008. Towards a methodology for the systematic analysis and design of efficient chemical processes: Part 1. From unit operations to elementary process functions. Chem. Eng. Process.: Process Intensif. 47 (12), 2051–2060. http://dx.doi.org/10.1016/j.cep.2008.07.011.

Freund, H., Sundmacher, K., 2011. Process intensification, 1. Fundamentals and molecular level. In: Ullmann's Encyclopedia of Industrial Chemistry. John Wiley & Sons, Ltd, http://dx.doi.org/10.1002/14356007.o22_o02, (Chapter Process Intensification, 1. Fundamentals and Molecular Level).

Garmatter, D., Maggi, A., Wenzel, M., Monem, S., Hahn, M., Stoll, M., Sager, S., Benner, P., Sundmacher, K., 2021. Power-to-chemicals: A superstructure problem for sustainable syngas production. In: Mathematical Modeling, Simulation and Optimization for Power Engineering and Management. Springer, http://dx.doi.org/10.1007/978-3-030-62732-4_7.

Gevers, M., 2006. System identification without lennart ljung: what would have been different? Stud. AB Norrtalje 2, 61–85, In: T. Glad and G. Hendeby, Eds., Forever Ljung in System Identification, ISBN: 91-44-02051-1.

Ghattas, O., Willcox, K., 2021. Learning physics-based models from data: perspectives from inverse problems and model reduction. Acta Numer. 30, 445–554. http://dx.doi.org/10.1017/S0962492921000064.

Glaessgen, E., Stargel, D., 2012. The digital twin paradigm for future NASA and US air force vehicles. In: 53rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference 20th AIAA/ASME/AHS Adaptive Structures Conference 14th AIAA. p. 1818, https://ntrs.nasa.gov/citations/20120008178.

Golub, G.H., van Loan, C.F., 2013. Matrix Computations, fourth ed. Johns Hopkins University Press, http://dx.doi.org/10.56021/9781421407944.

Gosea, I.V., Gugercin, S., Beattie, C., 2022. Data-driven balancing of linear dynamical systems. SIAM J. Sci. Comput. 44 (1), A554–A582. http://dx.doi.org/10.1137/21M1411081.

Gosea, I.V., Peterson, L., Goyal, P., Bremer, J., Sundmacher, K., Benner, P., 2024. Learning reduced-order Quadratic-Linear models in process engineering using Operator Inference. arXiv preprint arXiv:2402.17698, https://doi.org/10.48550/arXiv.2402.17698.

Gosea, I.V., Živković, L.A., Karachalios, D.S., Vidaković-Koch, T., Antoulas, A.C., 2023. A data-driven nonlinear frequency response approach based on the loewner framework: preliminary analysis. IFAC-PapersOnLine 56 (1), 234–239. http://dx.doi.org/10.1016/j.ifacol.2023.02.040, 12th IFAC Symposium on Nonlinear Control Systems NOLCOS 2022.

Goswami, S., Jagtap, A.D., Babaee, H., Susi, B.T., Karniadakis, G.E., 2024. Learning stiff chemical kinetics using extended deep neural operators. Comput. Methods Appl. Mech. Engrg. 419, 116674. http://dx.doi.org/10.1016/j.cma.2023.116674.

Goyal, P., Benner, P., 2022. Discovery of nonlinear dynamical systems using a Runge–Kutta inspired dictionary-based sparse regression approach. Proc. R. Soc. A 478 (2262), 20210883. http://dx.doi.org/10.1098/rspa.2021.0883.

Greppi, P., 2010. Process simulation as a domain-specific OPC unified architecture information model. In: Computer Aided Chemical Engineering. Vol. 28, Elsevier, pp. 667–672. http://dx.doi.org/10.1016/S1570-7946(10)28112-9.

Grimme, E.J., 1997. Krylov Projection Methods for Model Reduction. University of Illinois at Urbana-Champaign, https://perso.uclouvain.be/paul.vandooren/ThesisGrimme.pdf.

Guo, Y., Klink, A., Bartolo, P., Guo, W.G., 2023. Digital twins for electro-physical, chemical, and photonic processes. CIRP Ann http://dx.doi.org/10.1016/j.cirp.2023.05.007.

Gustavsen, B., Semlyen, A., 1999. Rational approximation of frequency domain responses by vector fitting. IEEE Trans. Power Deliv. 14 (3), 1052–1061. http://dx.doi.org/10.1109/61.772353.

Hachicha, S., Kharrat, M., Chaari, A., 2014. N4SID and MOESP algorithms to highlight the ill-conditioning into subspace identification. Int. J. Autom. Comput. 11 (1), 30–38. http://dx.doi.org/10.1007/s11633-014-0763-z.

Hahn, J., Edgar, T.F., 2002. An improved method for nonlinear model reduction using balancing of empirical gramians. Comput. Chem. Eng. 26 (10), 1379–1397. http://dx.doi.org/10.1016/S0098-1354(02)00120-5.

Hamill, T.M., Snyder, C., 2000. A hybrid ensemble Kalman filter–3D variational analysis scheme. Mon. Weather Rev. 128 (8), 2905–2919. http://dx.doi.org/10.1175/1520-0493(2000)128<2905:AHEKFV>2.0.CO;2.

Hammersley, J.M., Handscomb, D.C., 1964. Monte Carlo Methods. In: Monographs on Statistics and Applied Probability, Springer, http://dx.doi.org/10.1007/978-94-009-5819-7.

Hartmann, D., van der Auweraer, H., 2022. The executable digital twin: merging the digital and the physics worlds. arXiv:2210.17402.

Hassan, M., Montague, G., Iqbal, M.Z., Fahey, J., 2024. Virtual reality-based bioreactor digital twin for operator training. Digit. Chem. Eng. 11, 100147. http://dx.doi.org/10.1016/j.dche.2024.100147.

Hastings, W.K., 1970. Monte Carlo sampling methods using Markov chains and their applications. Biometrika 57 (1), 97–109. http://dx.doi.org/10.1093/biomet/57.1.97.

He, R., Chen, G., Dong, C., Sun, S., Shen, X., 2019. Data-driven digital twin technology for optimized control in process systems. ISA Trans. 95, 221–234. http://dx.doi.org/10.1016/j.isatra.2019.05.011.

Helgers, H., Hengelbrock, A., Rosengarten, J.F., Stitz, J., Schmidt, A., Strube, J., 2022. Towards autonomous process control—Digital twin for HIV-gag VLP production in HEK293 cells using a dynamic metabolic model. Processes 10 (10), 2015. http://dx.doi.org/10.3390/pr10102015.

Herwig, C., Pörtner, R., Möller, J. (Eds.), 2021a. Digital Twins: Applications to the Design and Optimization of Bioprocesses. Springer International Publishing, http://dx.doi.org/10.1007/978-3-030-71660-8.

Herwig, C., Pörtner, R., Möller, J. (Eds.), 2021b. Digital Twins: Tools and Concepts for Smart Biomanufacturing. Springer International Publishing, http://dx.doi.org/10.1007/978-3-030-71660-8.

Himmel, A., Matschek, J., Kok, R., Morabito, B., Nguyen, H.H., Findeisen, R., 2024. Machine learning for control of (bio) chemical manufacturing systems. In: Artificial Intelligence in Manufacturing. Elsevier, pp. 181–240. http://dx.doi.org/10.1016/B978-0-323-99134-6.00009-8.

History Digital Twin, 2023. The history and creation of the digital twin concept. https://www.challenge.org/insights/digital-twin-history/. (Accessed 12 February 2023).

Hu, X., Chaudhari, L., Lin, S., Stanton, S., Asgari, S., Lian, W., 2012. A state space thermal model for HEV/EV battery using vector fitting. In: 2012 IEEE Transportation Electrification Conference and Expo. ITEC, IEEE, pp. 1–8. http://dx.doi.org/10.1109/ITEC.2012.6243450.

Hubbs, C.D., Li, C., Sahinidis, N.V., Grossmann, I.E., Wassick, J.M., 2020. A deep reinforcement learning approach for chemical production scheduling. Comput. Chem. Eng. 141, 106982. http://dx.doi.org/10.1016/j.compchemeng.2020.106982.

Huynh, D.B.P., Knezevic, D.J., Patera, A.T., 2013. A static condensation reduced basis element method: approximation and a posteriori error estimation. ESAIM Math. Model. Numer. Anal. 47 (1), 213–251. http://dx.doi.org/10.1051/m2an/2012022.

Industrie 4.0 Glossar, 2023. Industrie 4.0 glossar. http://www.plattformi40.de/I40/Navigation/DE/Service/Glossar/glossar.html. (Accessed 31 May 2023).

Ivanciuc, O., 2007. Applications of support vector machines in chemistry. Rev. Comput. Chem. 23, 291, http://dx.doi.org/10.1002/9780470116449.ch6.

Ji, W., Qiu, W., Shi, Z., Pan, S., Deng, S., 2021. Stiff-PINN: Physics-informed neural network for stiff chemical kinetics. J. Phys. Chem. A 125 (36), 8098–8106. http://dx.doi.org/10.1021/acs.jpca.1c05102.

Jones, D., Snider, C., Nassehi, A., Yon, J., Hicks, B., 2020. Characterising the digital twin: A systematic literature review. CIRP J. Manuf. Sci. Technol. 29, 36–52. http://dx.doi.org/10.1016/j.cirpj.2020.02.002.

Kaba, O., Ravanbakhsh, S., 2022. Equivariant networks for crystal structures. Adv. Neural Inf. Process. Syst. 35, 4150–4164.

Kang, J.-L., Wang, C.-C., Wong, D.S.-H., Jang, S.-S., Wang, C.-H., 2021. Digital twin model and dynamic operation for a plant-scale solid oxide fuel cell system. J. Taiwan Inst. Chem. Eng. 118, 60–67. http://dx.doi.org/10.1016/j.jtice.2021.01.001.

Kapteyn, M.G., Knezevic, D.J., Huynh, D., Tran, M., Willcox, K.E., 2022. Data-driven physics-based digital twins via a library of component-based reduced-order models. Internat. J. Numer. Methods Engrg. 123 (13), 2986–3003. http://dx.doi.org/10.1002/nme.6423.

Kapteyn, M.G., Knezevic, D.J., Willcox, K., 2020. Toward predictive digital twins via component-based reduced-order models and interpretable machine learning. In: AIAA Scitech 2020 Forum. p. 0418. http://dx.doi.org/10.2514/6.2020-0418.

Kapteyn, M.G., Pretorius, J.V., Willcox, K.E., 2021. A probabilistic graphical model foundation for enabling predictive digital twins at scale. Nat. Comput. Sci. 1 (5), 337–347. http://dx.doi.org/10.1038/s43588-021-00069-0.

Kapteyn, M.G., Willcox, K.E., 2022. Design of digital twin sensing strategies via predictive modeling and interpretable machine learning. J. Mech. Des. 144 (9), 091710. http://dx.doi.org/10.1115/1.4054907.

Karniadakis, G.E., Kevrekidis, I.G., Lu, L., Perdikaris, P., Wang, S., Yang, L., 2021. Physics-informed machine learning. Nat. Rev. Phys. 3 (6), 422–440. http://dx.doi.org/10.1038/s42254-021-00314-5.

Karpatne, A., Atluri, G., Faghmous, J.H., Steinbach, M., Banerjee, A., Ganguly, A., Shekhar, S., Samatova, N., Kumar, V., 2017. Theory-guided data science: A new paradigm for scientific discovery from data. IEEE Trans. Knowl. Data Eng. 29 (10), 2318–2331. http://dx.doi.org/10.1109/TKDE.2017.2720168.

Kender, R., Kaufmann, F., ler, F.R., Wunderlich, B., Golubev, D., Thomas, I., Ecker, A.-M., Rehfeldt, S., Klein, H., 2021. Development of a digital twin for a flexible air separation unit using a pressure-driven simulation approach. Comput. Chem. Eng. 151, 107349. http://dx.doi.org/10.1016/j.compchemeng.2021.107349.

King, R.P., 1974. Applications of stochastic differential equations to chemical-engineering problems -an introductory review. Chem. Eng. Commun. 1 (5), 221–237. http://dx.doi.org/10.1080/00986447408960433.

Kingma, D.P., Ba, J., 2014. Adam: A method for stochastic optimization. arXiv:1412.6980.

Kirsch, A., 2011. An Introduction to the Mathematical Theory of Inverse Problems, vol. 120, Springer, NY, USA, http://dx.doi.org/10.1007/978-1-4419-8474-6.

Kockmann, N., 2018. 100% digital process industry–Impressions and results from the tutzing symposium 2018. Chem. Ing. Tech. 90 (11), 1621–1627. http://dx.doi.org/10.1002/cite.201800135.

König, J., Freitag, M., 2022. Time-limited balanced truncation for data assimilation problems. arXiv:2212.07719.

Koopman, B.O., von Neumann, J., 1932. Dynamical systems of continuous spectra. Proc. Natl. Acad. Sci. 18 (3), 255–263. http://dx.doi.org/10.1073/pnas.18.3.255.

Kramer, B., Peherstorfer, B., Willcox, K.E., 2024. Learning nonlinear reduced models from data with operator inference. Annu. Rev. Fluid Mech. 56, http://dx.doi.org/10.1146/annurev-fluid-121021-025220.

Kritzinger, W., Karner, M., Traar, G., Henjes, J., Sihn, W., 2018. Digital twin in manufacturing: A categorical literature review and classification. IFAC-PapersOnLine 51 (11), 1016–1022. http://dx.doi.org/10.1016/j.ifacol.2018.08.474.

Kumar, A., Baldea, M., Edgar, T.F., 2016. Real-time optimization of an industrial steam-methane reformer under distributed sensing. Control Eng. Pract. 54, 140–153. http://dx.doi.org/10.1016/j.conengprac.2016.05.010.

Kumari, P., Bhadriraju, B., Wang, Q., Kwon, J.S.-I., 2021. Development of parametric reduced-order model for consequence estimation of rare events. Chem. Eng. Res. Des. 169, 142–152. http://dx.doi.org/10.1016/j.cherd.2021.02.006.

Kunkel, P., Mehrmann, V., 2006. Differential-Algebraic Equations: Analysis and Numerical Solution, vol. 2, European Mathematical Society (EMS) Press, http://dx.doi.org/10.4171/017.

Kutz, J.N., Brunton, S.L., Brunton, B.W., Proctor, J.L., 2016. Dynamic Mode Decomposition: Data-Driven Modeling of Complex Systems. SIAM, http://dx.doi.org/10.1137/1.9781611974508.

Langiu, M., Shu, D.Y., Baader, F.J., Hering, D., Bau, U., Xhonneux, A., Müller, D., Bardow, A., Mitsos, A., Dahmen, M., 2021. COMANDO: A next-generation open-source framework for energy systems optimization. Comput. Chem. Eng. 152, 107366. http://dx.doi.org/10.1016/j.compchemeng.2021.107366.

Law, K., Stuart, A., Zygalakis, K., 2015. Data Assimilation A Mathematical Introduction. In: Texts in Applied Mathematics (TAM), vol. 62, Springer, http://dx.doi.org/10.1007/978-3-319-20325-6.

Lawless, A., Nichols, N., Boess, C., Bunse-Gerstner, A., 2008. Using model reduction methods within incremental four-dimensional variational data assimilation. Mon. Weather Rev. 136 (4), 1511–1522. http://dx.doi.org/10.1175/2007MWR2103.1.

Lee, A., Ghouse, J.H., Eslick, J.C., Laird, C.D., Siirola, J.D., Zamarripa, M.A., Gunter, D., Shinn, J.H., Dowling, A.W., Bhattacharyya, D., et al., 2021. The IDAES process modeling framework and model library—Flexibility for process simulation and optimization. J. Adv. Manuf. Process. 3 (3), e10095. http://dx.doi.org/10.1002/amp2.10095.

Li, Z., Kovachki, N., Azizzadenesheli, K., Liu, B., Bhattacharya, K., Stuart, A., Anandkumar, A., 2020. Neural operator: Graph kernel network for partial differential equations. arXiv:2003.03485.

Li, S., Yue, Y., Feng, L., Benner, P., Seidel-Morgenstern, A., 2014. Model reduction for linear simulated moving bed chromatography systems using Krylov-subspace methods. AIChE J. 60 (11), 3773–3783. http://dx.doi.org/10.1002/aic.14561.

Liesche, G., Schack, D., Sundmacher, K., 2019. The FluxMax approach for simultaneous process synthesis and heat integration: production of hydrogen cyanide. AIChE J. 65 (7), e16554. http://dx.doi.org/10.1002/aic.16554.

Lima, P.V., Saraiva, P.M., Group, G.-P., 2007. A semi-mechanistic model building framework based on selective and localized model extensions. Comput. Chem. Eng. 31 (4), 361–373. http://dx.doi.org/10.1016/j.compchemeng.2006.07.006.

Lin, W., Qian, Y., Li, X., 2000. Nonlinear dynamic principal component analysis for on-line process monitoring and diagnosis. Comput. Chem. Eng. 24 (2–7), 423–429. http://dx.doi.org/10.1016/S0098-1354(00)00433-6.

Ljung, L., 1999. System Identification: Theory for the User. In: Information and System Sciences Series, Prentice Hall, Subsequent edition, ISBN: 978-0136566953.

Locatelli, M., Schoen, F., 2013. Global Optimization: Theory, Algorithms, and Applications. In: MOS-SIAM Series on Optimization, SIAM, http://dx.doi.org/10.1137/1.9781611972672.

Lopez, P.C., Udugama, I.A., Thomsen, S.T., Roslander, C., Junicke, H., Mauricio-Iglesias, M., Gernaey, K.V., 2020. Towards a digital twin: a hybrid data-driven and mechanistic digital shadow to forecast the evolution of lignocellulosic fermentation. Biofuels, Bioprod. Biorefin. 14 (5), 1046–1060. http://dx.doi.org/10.1002/bbb.2108.

Lorenc, A.C., 2003. The potential of the ensemble Kalman filter for NWP—A comparison with 4D-var. Q. J. R. Meteorol. Soc. 129 (595), 3183–3203. http://dx.doi.org/10.1256/qj.02.132.

Lydia, A., Francis, S., 2019. Adagrad—an optimizer for stochastic gradient descent. Int. J. Inf. Comput. Sci. 6 (5), 566–568, https://www.researchgate.net/publication/334736907_Adagrad_-_An_Optimizer_for_Stochastic_Gradient_Descent.

Ma, C.Y., Wang, X.Z., 2009. Inductive data mining based on genetic programming: Automatic generation of decision trees from data for process historical data analysis. Comput. Chem. Eng. 33 (10), 1602–1616. http://dx.doi.org/10.1016/j.compchemeng.2009.04.005.

Mabkhot, M.M., Al-Ahmari, A.M., Salah, B., Alkhalefah, H., 2018. Requirements of the smart factory system: A survey and perspective. Machines 6 (2), 23. http://dx.doi.org/10.3390/machines6020023.

Mayo, A.J., Antoulas, A.C., 2007. A framework for the solution of the generalized realization problem. Linear Algebra Appl. 425 (2–3), 634–662. http://dx.doi.org/10.1016/j.laa.2007.03.008.

McBride, K., Sundmacher, K., 2019. Overview of surrogate modeling in chemical process engineering. Chem. Ing. Tech. 91 (3), 228–239. http://dx.doi.org/10.1002/cite.201800091.

McQuarrie, S.A., Huang, C., Willcox, K.E., 2021. Data-driven reduced-order models via regularised operator inference for a single-injector combustion process. J. R. Soc. New Zealand 51 (2), 194–211. http://dx.doi.org/10.1080/03036758.2020.1863237.

Md Nor, N., Che Hassan, C.R., Hussain, M.A., 2020. A review of data-driven fault detection and diagnosis methods: Applications in chemical process systems. Rev. Chem. Eng. 36 (4), 513–553. http://dx.doi.org/10.1515/revce-2017-0069.

Min, Q., Lu, Y., Liu, Z., Su, C., Wang, B., 2019. Machine learning based digital twin framework for production optimization in petrochemical industry. Int. J. Inf. Manage. 49, 502–519. http://dx.doi.org/10.1016/j.ijinfomgt.2019.05.020.

Mitchell, M., 1998. An Introduction to Genetic Algorithms. The MIT Press, https://mitpress.mit.edu/9780262631853/an-introduction-to-genetic-algorithms/.

Morari, M., Lee, J.H., 1999. Model predictive control: past, present and future. Comput. Chem. Eng. 23 (4–5), 667–682. http://dx.doi.org/10.1016/S0098-1354(98)00301-9.

Nakatsukasa, Y., Sete, O., Trefethen, L.N., 2018. The AAA algorithm for rational approximation. SIAM J. Sci. Comput. 40 (3), A1494–A1522. http://dx.doi.org/10.1137/16M1106122.

Narasingam, A., Kwon, J.S.-I., 2017. Development of local dynamic mode decomposition with control: Application to model predictive control of hydraulic fracturing. Comput. Chem. Eng. 106, 501–511. http://dx.doi.org/10.1016/j.compchemeng.2017.07.002.

National Academy of Engineering and National Academies of Sciences, Engineering, and Medicine, 2023. Foundational Research Gaps and Future Directions for Digital Twins. The National Academies Press, Washington, DC, http://dx.doi.org/10.17226/26894, https://nap.nationalacademies.org/catalog/26894/foundational-research-gaps-and-future-directions-for-digital-twins.

Natori, Y., O'Young, L., 1996. Vision of 21st century's plant and how to get there. Comput. Chem. Eng. 20, S1469–S1479. http://dx.doi.org/10.1016/0098-1354(96)00251-7.

Newrzella, S.R., Franklin, D.W., Haider, S., 2021. 5-dimension cross-industry digital twin applications model and analysis of digital twin classification terms and models. IEEE Access 9, 131306–131321. http://dx.doi.org/10.1109/ACCESS.2021.3115055.

Niederer, S.A., Sacks, M.S., Girolami, M., Willcox, K., 2021. Scaling digital twins from the artisanal to the industrial. Nat. Comput. Sci. 1 (5), 313–320. http://dx.doi.org/10.1038/s43588-021-00072-5.

Nocedal, J., Wright, S.J., 1999. Numerical Optimization. Springer, http://dx.doi.org/10.1007/978-0-387-40065-5.

Ogata, K., 2010. Modern Control Engineering, fifth ed. Pearson, https://www.pearson.com/en-us/subject-catalog/p/modern-control-engineering/P200000003521/9780137551064.

Ohlberger, M., Rave, S., Schindler, F., 2016. Model reduction for multiscale lithium-ion battery simulation. In: Karasözen, B., Manguoğlu, M., Tezer-Sezgin, M., Göktepe, S., Uğur, O. (Eds.), Numerical Mathematics and Advanced Applications ENUMATH 2015. In: LNCSE, vol. 112, Springer, pp. 317–331. http://dx.doi.org/10.1007/978-3-319-39929-4_31.

Örs, E., Schmidt, R., Mighani, M., Shalaby, M., 2020. A conceptual framework for AI-based operational digital twin in chemical process engineering. In: 2020 IEEE International Conference on Engineering, Technology and Innovation. ICE/ITMC, IEEE, pp. 1–8. http://dx.doi.org/10.1109/ICE/ITMC49519.2020.9198575.

Owoyele, O., Pal, P., 2022. ChemNODE: A neural ordinary differential equations framework for efficient chemical kinetic solvers. Energy AI 7, 100118. http://dx.doi.org/10.1016/j.egyai.2021.100118.

Pan, I., Mason, L.R., Matar, O.K., 2022. Data-centric engineering: Integrating simulation, machine learning and statistics. Challenges and opportunities. Chem. Eng. Sci. 249, 117271. http://dx.doi.org/10.1016/j.ces.2021.117271.

Pantelides, C.C., Renfro, J.G., 2013. The online use of first-principles models in process operations: Review, current status and future needs. Comput. Chem. Eng. 51, 136–148. http://dx.doi.org/10.1016/j.compchemeng.2012.07.008.

Park, K.T., Yang, J., Noh, S.D., 2021. VREDI: virtual representation for a digital twin application in a work-center-level asset administration shell. J. Intell. Manuf. 32, 501–544. http://dx.doi.org/10.1007/2Fs10845-020-01586-x.

Partopour, B., Paffenroth, R.C., Dixon, A.G., 2018. Random forests for mapping and analysis of microkinetics models. Comput. Chem. Eng. 115, 286–294. http://dx.doi.org/10.1016/j.compchemeng.2018.04.019.

Peherstorfer, B., Willcox, K., 2016. Data-driven operator inference for nonintrusive projection-based model reduction. Comput. Methods Appl. Mech. Engrg. 306, 196–215. http://dx.doi.org/10.1016/j.cma.2016.03.025.

Peschel, A., Freund, H., Sundmacher, K., 2010. Methodology for the design of optimal chemical reactors based on the concept of elementary process functions. Ind. Eng. Chem. Res. 49 (21), 10535–10548. http://dx.doi.org/10.1021/ie100476q.

Peterson, L., Bremer, J., Sundmacher, K., 2024a. Challenges in data-based reactor modeling: A critical analysis of purely data-driven and hybrid models for a CSTR case study. Comput. Chem. Eng. 184, 108643. http://dx.doi.org/10.1016/j.compchemeng.2024.108643.

Peterson, L., Forootani, A., Medina, E.I.S., Gosea, I.V., Sundmacher, K., Benner, P., 2024b. Towards digital twins for power-to-x: Comparing surrogate models for a catalytic CO_2 methanation reactor. Authorea Prepr. http://dx.doi.org/10.36227/techrxiv.172263007.76668955/v1.

Peterson, L., Goyal, P., Gosea, I.V., Bremer, J., Benner, P., Sundmacher, K., 2024c. Learning reduced-order models for dynamic CO_2 methanation using operator inference. In: Computer Aided Chemical Engineering. Vol. 53, Elsevier, pp. 3319–3324. http://dx.doi.org/10.1016/B978-0-443-28824-1.50554-8.

Pineda, M., Stamatakis, M., 2018. On the stochastic modelling of surface reactions through reflected chemical Langevin equations. Comput. Chem. Eng. 117, 145–158. http://dx.doi.org/10.1016/j.compchemeng.2018.05.003.

Pintelon, R., Schoukens, J., 2012. System Identification: a Frequency Domain Approach. John Wiley & Sons, https://ieeexplore.ieee.org/servlet/opac?bknumber=6198969. Online ISBN: 9781118287422.

Pirdashti, M., Curteanu, S., Kamangar, M.H., Hassim, M.H., Khatami, M.A., 2013. Artificial neural networks: applications in chemical engineering. Rev. Chem. Eng. 29 (4), 205–239. http://dx.doi.org/10.1007/978-3-030-87660-9_6.

Pistikopoulos, E.N., Barbosa-Povoa, A., Lee, J.H., Misener, R., Mitsos, A., Reklaitis, G.V., Venkatasubramanian, V., You, F., Gani, R., 2021. Process systems engineering–the generation next? Comput. Chem. Eng. 147, 107252. http://dx.doi.org/10.1016/j.compchemeng.2021.107252.

Pontes Duff, I., Goyal, P., Benner, P., 2024. Stability-certified learning of control systems with quadratic nonlinearities. arXiv:2403.00646.
Prett, D.M., García, C.E., 2013. Fundamental Process Control: Butterworths Series in Chemical Engineering. Butterworth-Heinemann.
Prévôt, C., Röckner, M., 2007. A Concise Course on Stochastic Partial Differential Equations, vol. 1905, Springer, http://dx.doi.org/10.1007/978-3-540-70781-3.
Proctor, J., Brunton, S.L., Kutz, J.N., 2016. Dynamic mode decomposition with control. SIAM J. Appl. Dyn. Syst. 15 (1), 142–161. http://dx.doi.org/10.1137/15M1013857.
PSE Digital Twin, 2023. Green reactor solutions. https://www.psenterprise.com/solutions/greactor. (Accessed 31 January 2023).
Qian, E., Farcas, I.-G., Willcox, K., 2022a. Reduced operator inference for nonlinear partial differential equations. SIAM J. Sci. Comput. 44 (4), A1934–A1959. http://dx.doi.org/10.1137/21M139397.
Qian, E., Tabeart, J.M., Beattie, C., Gugercin, S., Jiang, J., Kramer, P.R., Narayan, A., 2022b. Model reduction of linear dynamical systems via balancing for Bayesian inference. J. Sci. Comput. 91 (1), 29. http://dx.doi.org/10.1007/2Fs10915-022-01798-8.
Qiao, Z., Christensen, A.S., Welborn, M., Manby, F.R., Anandkumar, A., Miller, III, T.F., 2021. Unite: Unitary n-body tensor equivariant network with applications to quantum chemistry. arXiv:2105.14655.
Qin, S.J., 2006. An overview of subspace identification. Comput. Chem. Eng. 30 (10–12), 1502–1513. http://dx.doi.org/10.1016/j.compchemeng.2006.05.045.
Quarteroni, A., Manzoni, A., Negri, F., 2015. Reduced Basis Methods for Partial Differential Equations: An Introduction. In: La Matematica per il 3+2 (UNITEXTMAT), vol. 92, Springer, http://dx.doi.org/10.1007/978-3-319-15431-2.
Raccuglia, P., Elbert, K.C., Adler, P.D., Falk, C., Wenny, M.B., Mollo, A., Zeller, M., Friedler, S.A., Schrier, J., Norquist, A.J., 2016. Machine-learning-assisted materials discovery using failed experiments. Nature 533 (7601), 73–76. http://dx.doi.org/10.1038/nature17439.
Rafiei, M., Ricardez-Sandoval, L.A., 2018. Stochastic back-off approach for integration of design and control under uncertainty. Ind. Eng. Chem. Res. 57 (12), 4351–4365. http://dx.doi.org/10.1021/acs.iecr.7b03935.
Raissi, M., Perdikaris, P., Karniadakis, G.E., 2017. Machine learning of linear differential equations using Gaussian processes. J. Comput. Phys. 348, 683–693. http://dx.doi.org/10.1016/j.jcp.2017.07.050.
Raissi, M., Perdikaris, P., Karniadakis, G.E., 2019. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. J. Comput. Phys. 378, 686–707. http://dx.doi.org/10.1016/j.jcp.2018.10.045.
Raschka, S., Mirjalili, V., 2019. Python Machine Learning: Machine Learning and Deep Learning with Python, Scikit-Learn, and TensorFlow 2. Packt Publishing Ltd, ISBN: 978-1787125933.
Reddy, R.S., Arepally, D., Datta, A.K., 2022. Inverse problems in food engineering: A review. J. Food Eng. 319, 110909. http://dx.doi.org/10.1016/j.jfoodeng.2021.110909.
Redeker, M., Weskamp, J.N., Rössl, B., Pethig, F., 2021. Towards a digital twin platform for industrie 4.0. In: 2021 4th IEEE International Conference on Industrial Cyber-Physical Systems. ICPS, IEEE, pp. 39–46. http://dx.doi.org/10.1109/ICPS49255.2021.9468204.
Rentschler, P., Klahn, C., Dittmeyer, R., 2023. The need for dynamic process simulation: A review of offshore power-to-x systems. Chem. Ing. Tech. http://dx.doi.org/10.1002/cite.202300156.
Rheinboldt, W.C., 1998. Methods for Solving Systems of Nonlinear Equations. SIAM, http://dx.doi.org/10.1137/1.9781611970012.
Ripley, B.D., 2009. Stochastic Simulation. John Wiley & Sons, http://dx.doi.org/10.1002/9780470316726.
Roberts, S., Osborne, M., Ebden, M., Reece, S., Gibson, N., Aigrain, S., 2013. Gaussian processes for time-series modelling. Phil. Trans. R. Soc. A 371 (1984), 20110550. http://dx.doi.org/10.1098/rsta.2011.0550.
Roussel, M.R., 1996. The use of delay differential equations in chemical kinetics. J. Phys. Chem. 100 (20), 8323–8330. http://dx.doi.org/10.1021/jp9600672.
Rüther, T., Gosea, I.V., Jahn, L., Antoulas, A.C., Danzer, M.A., 2023. Introducing the loewner method as a data-driven and regularization-free approach for the distribution of relaxation times analysis of lithium-ion batteries. Batteries 9 (2), 132. http://dx.doi.org/10.3390/batteries9020132.
Sancho, A., Ribeiro, J., Reis, M.S., Martins, F.G., 2022. Cluster analysis of crude oils with k-means based on their physicochemical properties. Comput. Chem. Eng. 157, 107633. http://dx.doi.org/10.1016/j.compchemeng.2021.107633.
Sansana, J., Joswiak, M.N., Castillo, I., Wang, Z., Rendall, R., Chiang, L.H., Reis, M.S., 2021. Recent trends on hybrid modeling for industry 4.0. Comput. Chem. Eng. 151, 107365. http://dx.doi.org/10.1016/j.compchemeng.2021.107365.
Savkovic-Stevanovic, J., 1996. Neural net controller by inverse modeling for a distillation plant. Comput. Chem. Eng. 20, S925–S930. http://dx.doi.org/10.1016/0098-1354(96)00162-7.
Schack, D., Liesche, G., Sundmacher, K., 2020. The FluxMax approach: Simultaneous flux optimization and heat integration by discretization of thermodynamic state space illustrated on methanol synthesis process. Chem. Eng. Sci. 215, 115382. http://dx.doi.org/10.1016/j.ces.2019.115382.
Schmid, P.J., 2010. Dynamic mode decomposition of numerical and experimental data. J. Fluid Mech. 656, 5–28. http://dx.doi.org/10.1017/S0022112010001217.
Schmid, P.J., 2022. Dynamic mode decomposition and its variants. Annu. Rev. Fluid Mech. 54, 225–254. http://dx.doi.org/10.1146/annurev-fluid-030121-015835.
Schubert, J., Simutis, R., Dors, M., Havlik, I., Lübbert, A., 1994. Bioprocess optimization and control: Application of hybrid modelling. J. Biotechnol. 35 (1), 51–68. http://dx.doi.org/10.1016/0168-1656(94)90189-9.
Schweidtmann, A.M., Esche, E., Fischer, A., Kloft, M., Repke, J.-U., Sager, S., Mitsos, A., 2021. Machine learning in chemical engineering: A perspective. Chem. Ing. Tech. 93 (12), 2029–2039. http://dx.doi.org/10.1002/cite.202100083.
Schwenzer, M., Ay, M., Bergs, T., Abel, D., 2021. Review on model predictive control: An engineering perspective. Int. J. Adv. Manuf. Technol. 117 (5), 1327–1349. http://dx.doi.org/10.1007/s00170-021-07682-3.
Scientific machine learning, 2023. Scientific machine learning. https://oden.utexas.edu/research/crosscutting-research-areas/scientific-machine-learning. (Accessed 13 September 2023).
Serra, J.M., Corma, A., Chica, A., Argente, E., Botti, V., 2003. Can artificial neural networks help the experimentation in catalysis? Catal. Today 81 (3), 393–403. http://dx.doi.org/10.1016/S0920-5861(03)00137-8.
Shang, C., You, F., 2019. Data analytics and machine learning for smart process manufacturing: Recent advances and perspectives in the big data era. Engineering 5 (6), 1010–1016. http://dx.doi.org/10.1016/j.eng.2019.01.019.
Sharma, N., Liu, Y., 2022. A hybrid science-guided machine learning approach for modeling chemical processes: A review. AIChE J. 68 (5), e17609. http://dx.doi.org/10.1002/aic.17609.
Shih, Y.-P., Shieh, C.-S., 1978. Model reduction of continuous and discrete multivariable systems by moments matching. Comput. Chem. Eng. 2 (4), 127–132. http://dx.doi.org/10.1016/0098-1354(78)80019-2.
Shin, J., Badgwell, T.A., Liu, K.-H., Lee, J.H., 2019. Reinforcement learning–overview of recent progress and implications for process control. Comput. Chem. Eng. 127, 282–294. http://dx.doi.org/10.1016/j.compchemeng.2019.05.029.
Shun, Z., McCullagh, P., 1995. Laplace approximation of high dimensional integrals. J. R. Stat. Soc. Ser. B Stat. Methodol. 57 (4), 749–760, https://www.jstor.org/stable/2345941.
Sirovich, L., 1987. Turbulence and the dynamics of coherent structures part I: Coherent structures. Quart. Appl. Math. 45 (3), 561–571, https://www.jstor.org/stable/43637457.
Smith, R.C., 2013. Uncertainty Quantification: Theory, Implementation, and Applications. Computational Science and Engineering Series, vol. 12, SIAM, https://my.siam.org/Store/Product/viewproduct/?ProductId=24973024.
Soesanto, J.F., Maciszewski, B., Mirmontazeri, L., Romero, S., Michonski, M., Milne, A., Huang, B., 2024. Digital twin and control of an industrial-scale bitumen extraction process. Comput. Chem. Eng. 185, 108671. http://dx.doi.org/10.1016/j.compchemeng.2024.108671.
Sorrentino, A., Patel, B., Gosea, I.V., Antoulas, A.C., Vidaković-Koch, T., 2023. Determination of the distribution of relaxation times through loewner framework: A direct and versatile approach. J. Power Sources 585, 233575. http://dx.doi.org/10.1016/j.jpowsour.2023.233575.
Stephanopoulos, G., 1984. Chemical Process Control, vol. 2, Prentice Hall, Englewood Cliffs, NJ.
von Stosch, M., Oliveira, R., Peres, J., de Azevedo, S.F., 2014. Hybrid semi-parametric modeling in process systems engineering: Past, present and future. Comput. Chem. Eng. 60, 86–101. http://dx.doi.org/10.1016/j.compchemeng.2013.08.008.
Svitnič, T., Sundmacher, K., 2022. Renewable methanol production: Optimization-based design, scheduling and waste-heat utilization with the FluxMax approach. Appl. Energy 326, 120017. http://dx.doi.org/10.1016/j.apenergy.2022.120017.
Talkhestani, B.A., Jung, T., Lindemann, B., Sahlab, N., Jazdi, N., Schloegl, W., Weyrich, M., 2019. An architecture of an intelligent digital twin in a cyber-physical production system. AT - Automatisierungstechnik 67 (9), 762–782. http://dx.doi.org/10.1515/auto-2019-0039.
Tao, F., Cheng, J., Qi, Q., Zhang, M., Zhang, H., Sui, F., 2018. Digital twin-driven product design, manufacturing and service with big data. Int. J. Adv. Manuf. Technol. 94 (9), 3563–3576. http://dx.doi.org/10.1007/s00170-017-0233-1.
Teixeira, A.P., Clemente, J.J., Cunha, A.E., Carrondo, M.J., Oliveira, R., 2006. Bioprocess iterative batch-to-batch optimization based on hybrid parametric/nonparametric models. Biotechnol. Prog. 22 (1), 247–258. http://dx.doi.org/10.1021/bp0502328.
Thelen, A., Zhang, X., Fink, O., Lu, Y., Ghosh, S., Youn, B.D., Todd, M.D., Mahadevan, S., Hu, C., Hu, Z., 2022. A comprehensive review of digital twin—part 1: modeling and twinning enabling technologies. Struct. Multidiscip. Optim. 65 (12), 354. http://dx.doi.org/10.1007/s00158-022-03425-4.
Thelen, A., Zhang, X., Fink, O., Lu, Y., Ghosh, S., Youn, B.D., Todd, M.D., Mahadevan, S., Hu, C., Hu, Z., 2023. A comprehensive review of digital twin—part 2: roles of uncertainty quantification and optimization, a battery digital twin, and perspectives. Struct. Multidiscip. Optim. 66 (1), 1. http://dx.doi.org/10.1007/s00158-022-03410-x.
Thiyagalingam, J., Shankar, M., Fox, G., Hey, T., 2022. Scientific machine learning benchmarks. Nat. Rev. Phys. 4 (6), 413–420. http://dx.doi.org/10.1038/s42254-022-00441-7.
Tomba, E., Barolo, M., García-Muñoz, S., 2014. In-silico product formulation design through latent variable model inversion. Chem. Eng. Res. Des. 92 (3), 534–544. http://dx.doi.org/10.1016/j.cherd.2013.08.027.

Uebbing, J., Biegler, L.T., Rihko-Struckmann, L., Sager, S., Sundmacher, K., 2021. Optimization of pressure swing adsorption via a trust-region filter algorithm and equilibrium theory. Comput. Chem. Eng. 151, 107340. http://dx.doi.org/10.1016/j.compchemeng.2021.107340.

Uebbing, J., Rihko-Struckmann, L., Sager, S., Sundmacher, K., 2020. CO_2 methanation process synthesis by superstructure optimization. J. CO2 Util. 40, 101228. http://dx.doi.org/10.1016/j.jcou.2020.101228.

Van Overschee, P., De Moor, B., 1994. N4SID: Subspace algorithms for the identification of combined deterministic-stochastic systems. Automatica 30 (1), 75–93. http://dx.doi.org/10.1016/0005-1098(94)90230-5.

Vassiliadis, V.S., Mappas, V., Espaas, T.A., Dorneanu, B., Isafiade, A., Möller, K., Arellano-Garcia, H., 2024. Reloading process systems engineering within chemical engineering. Chem. Eng. Res. Des. http://dx.doi.org/10.1016/j.cherd.2024.07.066.

Velegar, M., Keller, C., Kutz, J.N., 2024. Optimized dynamic mode decomposition for reconstruction and forecasting of atmospheric chemistry data. arXiv:2404.12396.

Venkatasubramanian, V., 2019. The promise of artificial intelligence in chemical engineering: Is it here, finally? AIChE J. 65 (2), 466–478. http://dx.doi.org/10.1002/aic.16489.

Venkatasubramanian, V., Chan, K., Caruthers, J.M., 1994. Computer-aided molecular design using genetic algorithms. Comput. Chem. Eng. 18 (9), 833–844. http://dx.doi.org/10.1016/0098-1354(93)E0023-3.

Verhaegen, M., Dewilde, P., 1992. Subspace model identification part 2. Analysis of the elementary output-error state-space model identification algorithm. Int. J. Control 56 (5), 1211–1241. http://dx.doi.org/10.1080/00207179208934364.

Verhaegen, M., Verdult, V., 2007. Filtering and System Identification: a Least Squares Approach. Cambridge University Press, http://dx.doi.org/10.1017/CBO9780511618888.

Wagner, C., Grothoff, J., Epple, U., Drath, R., Malakuti, S., Grüner, S., Hoffmeister, M., Zimermann, P., 2017. The role of the industry 4.0 asset administration shell and the digital twin during the life cycle of a plant. In: 2017 22nd IEEE International Conference on Emerging Technologies and Factory Automation. ETFA, IEEE, pp. 1–8. http://dx.doi.org/10.1109/ETFA.2017.8247583.

Walmsley, T.G., Patros, P., Yu, W., Young, B.R., Burroughs, S., Apperley, M., Carson, J.K., Udugama, I.A., Aeowjaroenlap, H., Atkins, M.J., et al., 2024. Adaptive digital twins for energy-intensive industries and their local communities. Digit. Chem. Eng. 10, 100139. http://dx.doi.org/10.1016/j.dche.2024.100139.

Wang, J., Moreira, J., Cao, Y., Gopaluni, R.B., 2023a. Simultaneous digital twin identification and signal-noise decomposition through modified generalized sparse identification of nonlinear dynamics. Comput. Chem. Eng. 177, 108294. http://dx.doi.org/10.1016/j.compchemeng.2023.108294.

Wang, S.-H., Pillai, H.S., Wang, S., Achenie, L.E., Xin, H., 2021. Infusing theory into deep learning for interpretable reactivity prediction. Nature Commun. 12 (1), 5288. http://dx.doi.org/10.1038/s41467-021-25639-8.

Wang, Y., Tao, F., Zuo, Y., Zhang, M., Qi, Q., 2023b. Digital-twin-enhanced quality prediction for the composite materials. Engineering 22, 23–33. http://dx.doi.org/10.1016/j.eng.2022.08.019.

Willard, J., Jia, X., Xu, S., Steinbach, M., Kumar, V., 2020. Integrating physics-based modeling with machine learning: A survey. arXiv:2003.04919.

Willcox, K., Segundo, B., 2024. The role of computational science in digital twins. Nat. Comput. Sci. 4 (3), 147–149. http://dx.doi.org/10.1038/s43588-024-00609-4.

Williams, M.O., Kevrekidis, I.G., Rowley, C.W., 2015. A data–driven approximation of the koopman operator: Extending dynamic mode decomposition. J. Nonlinear Sci. 25, 1307–1346. http://dx.doi.org/10.1007/s00332-015-9258-5.

Willis, M., Di Massimo, C., Montague, G., Tham, M., Morris, A., 1991. Artificial neural networks in process engineering. In: IEE Proceedings D (Control Theory and Applications). Vol. 138, IET, pp. 256–266. http://dx.doi.org/10.1049/ip-d.1991.0036.

Wright, L., Davidson, S., 2020. How to tell the difference between a model and a digital twin. Adv. Model. Simul. Eng. Sci. 7 (1), 1–13. http://dx.doi.org/10.1186/s40323-020-00147-4.

Wu, Z., Rincon, D., Christofides, P.D., 2019a. Real-time adaptive machine-learning-based predictive control of nonlinear processes. Ind. Eng. Chem. Res. 59 (6), 2275–2290. http://dx.doi.org/10.1021/acs.iecr.9b03055.

Wu, Z., Tran, A., Rincon, D., Christofides, P.D., 2019b. Machine learning-based predictive control of nonlinear processes. Part I: theory. AIChE J. 65 (11), e16729. http://dx.doi.org/10.1002/aic.16729.

Wu, B., Widanage, W.D., Yang, S., Liu, X., 2020. Battery digital twins: Perspectives on the fusion of models, data and artificial intelligence for smart battery management systems. Energy AI 1, 100016. http://dx.doi.org/10.1016/j.egyai.2020.100016.

Yang, S., Navarathna, P., Ghosh, S., Bequette, B.W., 2020. Hybrid modeling in the era of smart manufacturing. Comput. Chem. Eng. 140, 106874. http://dx.doi.org/10.1016/j.compchemeng.2020.106874.

Yue, Y., Feng, L., Seidel-Morgenstern, A., Benner, P., 2014. Efficient model reduction of SMB chromatography by krylov-subspace method with application to uncertainty quantification. In: 24 European Symposium on Computer Aided Process Engineering. In: Computer Aided Chemical Engineering, vol. 33, Budapest, Ungarn, pp. 925–930. http://dx.doi.org/10.1016/B978-0-444-63456-6.50155-1.

Zalai, D., Dietzsch, C., Herwig, C., 2013. Risk-based process development of biosimilars as part of the quality by design paradigm. PDA J. Pharmaceut. Sci. Technol. 67 (6), 569–580. http://dx.doi.org/10.5731/pdajpst.2013.00943.

Zhang, D., Del Rio-Chanona, E.A., Petsagkourakis, P., Wagner, J., 2019a. Hybrid physics-based and data-driven modeling for bioprocess online simulation and optimization. Biotechnol. Bioeng. 116 (11), 2919–2930. http://dx.doi.org/10.1002/bit.27120.

Zhang, Z., Wu, Z., Rincon, D., Christofides, P.D., 2019b. Real-time optimization and control of nonlinear processes using machine learning. Mathematics 7 (10), 890. http://dx.doi.org/10.3390/math7100890.