KINETICS-INFORMED NEURAL NETWORKS

Gabriel S. Gusmão, Adhika P. Retnanto[1], Shashwati C. da Cunha[1], Andrew J. Medford
School of Chemical & Biomolecular Engineering
Georgia Institute of Technology
Atlanta, GA 30332
{gusmaogabriels, aretnanto6, shashwatidc, ajm}@gatech.edu

[1]These authors contributed equally to this work.

December 1, 2020

ABSTRACT

Chemical kinetics consists of the phenomenological framework for the disentanglement of reaction mechanisms, optimization of reaction performance and the rational design of chemical processes. Here, we utilize feed-forward artificial neural networks as basis functions for the construction of surrogate models to solve ordinary differential equations (ODEs) that describe microkinetic models (MKMs). We present an algebraic framework for the mathematical description and classification of reaction networks, types of elementary reaction, and chemical species. Under this framework, we demonstrate that the simultaneous training of neural nets and kinetic model parameters in a regularized multiobjective optimization setting leads to the solution of the inverse problem through the estimation of kinetic parameters from synthetic experimental data. We probe the limits at which kinetic parameters can be retrieved as a function of knowledge about the chemical system states over time, and assess the robustness of the methodology with respect to statistical noise. This surrogate approach to inverse kinetic ODEs can assist in the elucidation of reaction mechanisms based on transient data.

Keywords surrogate model, regularization, neural network, catalysis, transient, *chemical kinetics*

1 Introduction

In recent decades, considerable attention has been devoted to chemical process intensification through the optimal design of unit operations based on rigorous modeling of thermodynamics, heat, mass and momentum transport, and, most importantly, chemical reactions [1]. Selectivity and yield of a chemical transformation, which are key factors in economic evaluation and design of associated unit operations, is determined by the intrinsic kinetics, transport resistances and operating conditions under the envolpe of thermodynamics limits. The detailed description of chemical kinetics, especially in heterogeneous catalysis, is hence of utmost importance in the design and operation of chemical processes. Kinetic models provide the phenomenological framework for the elucidation of reaction mechanism, optimization of reaction performance, and process design. Structurally, the detailed description of a chemical reaction network consists of the set of elementary steps involving bond breaking and/or formation from reactants to products. Chemical kinetics can be described in terms of the transition-state barriers between thermodynamically stable intermediate states. Despite the high complexity of chemical kinetics network, only few elementary steps typically control catalytic activity [2, 3].

The mathematical description of process models has departed from simplifying assumptions and become more rigorous [4]. The analysis and elucidation of reaction mechanism have benefited from the increasing accuracy, vis-à-vis the computational cost, of computational chemistry methods such as density functional theory (DFT). In the bottom-up approach, DFT-derived state energies from proposed reaction mechanisms give rise to potential energy diagrams. The ensuing microkinetic models (MKMs) have provided insights into rate-controlling steps [5] and trends in catalytic activity in terms of reaction conditions and catalyst compositions [6, 7]. Top-down or empirically driven efforts have relied on surface-science analytical chemistry methods, e.g. operando techniques [8, 9]: in-situ spectroscopy,

photoelectron (XPS, LEED, TEM), infrared (FT-IR), and temperature programmed reduction (TRP) and desorption (TPD), temporal, among other material science characterization techniques, such as NMR, XAFS and XRD [10, 11].

The mathematical modelling of complex reaction networks, especially those involving catalysis, requires in-depth understanding of the involved active sites as well as the thermodynamically stable intermediates and favorable elementary reactions. For nearly uniform chemical systems phases, as on metallic catalysts surfaces, the mean-field approximation (MFA) is typically invoked as a surrogate to supplant the need for a spatial description of catalytic active domains [12]. Transport limitations are then regarded as negligible or included as averages across the available catalyst surface. This requires adsorbate-adsorbate interactions to be either neglected or included as coverage-dependent properties. For highly heterogeneous and non-ideal systems, rigorous kinetic Monte Carlo methods can be used to build coarse-grained representations of catalyst active sites, for which diffusive processes and surface interactions can be explicitly accounted [13]. However, in the absence of diffusion limitations and adsorbate-adsorbate interaction, studies by Hoffmann and Bliggard [14] and by Andersen et. al. [15] have shown that results derived from mean-field models approximate those of complex kinetic Monte Carlo schemes. These results suggest that the mean-field framework is promising for the accurate description of uniform phases and approximate description of complex catalytically active domains.

Despite advances in first-principles and empirical mechanism elucidation, the accurate description of catalyst states over time-on-stream under operating conditions remains inherently challenging, while highly dependent on the chemical system under study. This is because changes in catalyst state under different operating conditions may translate into shifts in the predominant chemical pathway and, consequently, in selectivity and yield [16]. For instance, the carbide-promoted Fischer-Tropsch reaction is a complex mechanism where the presence of carbide-, oxygen- and metallic-rich states exhibit strong dependence on reaction conditions [17, 18]. Furthermore, catalytic systems often encompass multiple active-site domains with different prevalent chemical pathways in which diffusive processes may become relevant, as exemplified by Nørskov et. al. on the synthesis gas conversion to higher alcohols over different rhodium-catalyst facets [19]. Adsorbate-adsorbate interactions may also play a significant role in determining the preferential chemical pathway and the related rate-controlling steps [20, 21]. These nuances pose significant challenges in devising either ab-initio or empirically driven models.

In the top-down approach to chemical kinetics, the acquisition of steady-state data has traditionally been the foremost empirical procedure for the analysis and validation of reaction mechanism hypotheses [6]. However, acquiring intrinsic kinetic steady-state data is inherently laborious [22]. The design of experiments must guarantee the nonexistence of mass- and heat-transfer limitations to detach transport processes from kinetics. More importantly, steady-state data only allows the probing of rate-limiting steps, overshadowing the detailed description of other involved elementary steps. In contrast, dynamic or transient techniques to acquire kinetic data have been discussed for nearly half a century, but have recently received considerable interest due to advances in numerical methods and computational processing capabilities. Transient kinetics methods such as temporal analysis of products [23, 24, 25] (TAP), step-response experiments, steady-state isotopic transient kinetic analysis [26, 27, 28] (SSITKA) and tapered element oscillating microbalance (TEOM) [29] are so information-dense that a single series of states, gathered at their respective time points from one experiment, conveys kinetic information that would require numerous steady-state experiments. The temporal evolution of chemical system states can be mathematically represented as sets of coupled ordinary or partial differential equations encompassing the various state-changing steps, such as the elementary reactions and diffusive processes, involved in a chemical reaction. While in past decades intensive effort has been put into constructing ab-initio microkinetic models, attention has only recently been put into the related issue of catalyst optimization, parameter estimation or model selection based on experimental data [30, 31, 32, 33, 34, 35, 36, 37]. Such techniques allow the probing of multiple elementary steps, potentially enabling more robust top-down models to be developed from less experimental data.

To balance model accuracy with the computational cost of successive evaluation of high-fidelity models, surrogate models (SMs), also referred to as reduced-order, meta- or black-box models, have become an alternative functional mapping between model inputs and outputs. SMs are models of models for which, given data, parameters are estimated from the minimization of some error metric or merit function. The construction of SMs necessarily relies on experimental data or data generated by mathematically complex phenomenological models. SMs also require proper choice of sampling strategies for the input space, since the computational cost of randomly sampling high dimensional spaces increases exponentially due to the "curse of dimensionality" [38]. Uniform, random, Latin-hypercube, central composite design and quasi-random low-discrepancy are the most commonly adopted strategies for input space exploration, although more advanced strategies have recently been devised [39]. In flexible model architectures, such as neural nets (NNs), model fitting generally depends on the ability to calculate gradients across the multiple layers through the chain-rule in the back-propagation algorithm. However, to avoid the time-consuming process of estimating gradients or building the back-propagation structure, procedural algorithms for automatic or algorithmic differentiation (AD) are used to evaluate gradients across NN layers.

Several authors have proposed methods that rely on NNs for the solution of scientific problems involving ODEs and PDEs. Seminal work regarding the utilization of NNs for the solution of differential equations was introduced by Lagaris et. al. [40, 41] in the context of physics-informed neural networks, PINNs. Such endeavors showed that particular NN structures, such as residual networks (ResNets), resemble Euler-type discretization through successive time-steps [42, 43, 44].Subsequently, Neural ODEs (NODEs) have emerged from the analogy between the discrete time-domain in ResNet ODEs at their depth limits, in a continuous application of NN to ODE, which does not require definition of step-sizes for integration in advance [45]. Following works have proved that ResNet-type deep NNs converge to Neural ODEs in the deep limit under stochastic gradient descent minimization [46] . The fundamental difference between PINNs and NODEs lies in their respective mappings. In PINNs, NNs constitute the mapping between the independent variables and the states, with the explicit physical model providing the connection between states and their respective gradients. In NODEs, NNs are instead used as mapping from states to their gradients. While NODEs can replace unknown parts of a differential equation, PINNs are a basis set for the solution of known differential equations.

In this work, we focus on the utilization of PINNs for the solution of microkinetic models (MKMs) under the MFA, which are fully represented by the chemical system's stoichiometric matrix [47, 48]. The MFA acts as a general surrogate for detailed kinetics, which could include not only the backbone elementary reactions but also diffusive steps. MKMs consist of a set of elementary or irreducible reactions. The elementary reactions involve the interaction of at most two individual species and are represented by power-law kinetics, in which the probability of occurrence is proportional to their concentrations, and whose probability of success is given by the Arrhenius law. We use this general mathematical representation to approach the solution of known MKMs and address the issue of parameter fitting for kinetically relevant steps in complete MKMs.

By relying on PINNs, we resort to the universal approximation theorem for artificial neural networks (NNs) [49, 50, 51, 52] to propose strategies towards the design of SMs that can be utilized for solving kinetics ODEs. We introduce a framework for the classification of reaction network types according to the nature of involved elementary reactions, e.g. homogeneous systems, ad/desorption, reactions between surface intermediates. We further create reaction-type examples and apply our PINNs-based approach, demonstrating the need for semi-quantitative information on intermediates composition to fully recover ground-truth information on the MKM parameters.

2 Methods

2.1 Mean-Field Chemical Kinetics Framework

Power-law kinetics is summarized as a linear combination of rare events, whose sampling frequency scales linearly with the concentration of each participant, and whose probability of success is given by the Arrhenius law. The resulting set of intertwined molecular events is represented as a stack of ODEs, assigning the rate of change of each species to the power-law kinetic expression arising from stoichiometry. Let $\mathbf{c}$ represent an array of concentrations or concentration-related state variables like partial pressures, concentrations and coverage fractions, for species that are unbound (like gases) or bound (adsorbed molecules or radicals), at time t. Then their rates of change, $\dot{\mathbf{c}}$, or the MKM, is written as (1.1):

$$\dot{\mathbf{c}} = \mathbf{M}\,\mathbf{r}(\mathbf{c}, \theta) = \mathbf{M}\,(\mathbf{k}(\theta) \odot f(\mathbf{c})) \tag{1.1}$$

Where $\mathbf{M} \in \mathbb{Z}^{n \times m}$ is the corresponding stoichiometry matrix and $f(\cdot) : \mathbb{R}_+^n \to \mathbb{R}_+^m$ maps the concentrations $\mathbf{c} := \{\mathbf{c} \,|\, \mathbf{c} \in \mathbb{R}_+^n\}$ to the concentration-based terms of power-law kinetics. $\mathbf{k} := \{\mathbf{k}(\theta) \in \mathbb{R}_+^m, \theta \in \mathbb{R}_+\}$ is the temperature- and binding-energy-dependent, Arrhenius-like rate constant term.

2.2 Complex Reaction Networks - *gdacs* representation

We denote the subset of $\mathbf{c}$ for gas-phase species as $\mathbf{c_g} := \{c_i \,|\, i \in \mathcal{C}_g\}$, for bounded gas-counterpart coverage fractions as $\mathbf{c_a} := \{c_i \,|\, i \in \mathcal{C}_a\}$, and for bounded intermediates/radicals on the catalyst surface as $\mathbf{c_s} := \{c_i \,|\, i \in \mathcal{C}_s\}$. Thus $\cup_{i \in \{\mathbf{g},\mathbf{a},\mathbf{s}\}} \mathcal{C}_i = \{1, 2, ..., m \,|\, m \in \mathbb{N}\}$ and $\cap_{i \in \{\mathbf{g},\mathbf{a},\mathbf{s}\}} \mathcal{C}_i = \emptyset$.

The state-dependent reaction rate vector, denoted as $\mathbf{r}(\mathbf{c}, \theta) := \{\mathbf{r} \in \mathbb{R}^m \,|\, \mathbf{r} = k(\theta) \odot f(\mathbf{c})\}$, comprises of rates of reaction (frequencies) associated with different kinds of elementary reaction:

- reactions in homogeneous phase, $\mathbf{r_g} := \{r_i \,|\, i \in \mathcal{G}_g\}$,
- those involving adsorption-desorption, $\mathbf{r_d} := \{r_i \,|\, i \in \mathcal{G}_d\}$,
- reactions between adsorbed molecules, $\mathbf{r_a} := \{r_i \,|\, i \in \mathcal{G}_a\}$,
- reactions involving adsorbed molecules and radicals/intermediates on the surface, $\mathbf{r_c} := \{r_i \,|\, i \in \mathcal{G}_c\}$, and

- reactions between intermediates on the surface, $\mathbf{r_s} := \{r_i \,|\, i \in \mathcal{G}_s\}$,

such that $\cup_{i\in\{\mathbf{g},\mathbf{d},\mathbf{a},\mathbf{c},\mathbf{s}\}}\mathcal{G}_i = \{1, 2, ..., n \,|\, n \in \mathbb{N}\}$ and $\cap_{i\in\{\mathbf{g},\mathbf{d},\mathbf{a},\mathbf{c},\mathbf{s}\}}\mathcal{G}_i = \emptyset$.

The stoichiometry matrix is then a composition of reaction-type submatrices, as follows in (2.1).

$$\begin{bmatrix}\dot{\mathbf{c}}_{\mathbf{g}}\\ \dot{\mathbf{c}}_{\mathbf{a}}\\ \dot{\mathbf{c}}_{\mathbf{s}}\end{bmatrix} = \begin{bmatrix} \mathbf{M_{gg}} & \mathbf{M_{gd}} & \mathbf{0} & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{M_{ad}} & \mathbf{M_{aa}} & \mathbf{M_{ac}} & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{M_{sc}} & \mathbf{M_{ss}}\end{bmatrix} \mathbf{r}(\mathbf{c}, \theta) \tag{2.1}$$

The full stoichiometry matrix in (2.1) embodies different types of reaction-type submatrices, where $\{\mathbf{i}\} \subset \{\mathbf{g},\mathbf{d},\mathbf{a},\mathbf{c},\mathbf{s}\}$ are the corresponding reaction types $\mathbf{i}$ as columns of $\mathbf{M}$. We can, hence, further classify the submatrices in the full stoichiometry matrix in terms of reaction- and species-types:

- homogeneous reactions, $\mathbf{M_{gg}} \neq \mathbf{0}$,
- adsorption/desorption, $\mathbf{M_{\{ga\}d}} \neq \mathbf{0}$,
- reactions between adsorbed molecules, $\mathbf{M_{aa}} \neq \mathbf{0}$,
- reaction between adsorbed molecules and surface intermediates, $\mathbf{M_{\{as\}c}} \neq \mathbf{0}$, and
- reactions between surface intermediates, $\mathbf{M_{ss}} \neq \mathbf{0}$

Complex reaction paths with multiple reacting surface intermediates or the case of lack of in-depth understanding on the reaction mechanism may yield matrices of type **gdacs**, encompassing all possible reaction types. These networks may be seen under circumstances where alternate reaction pathways are possible depending on operating conditions or when multiple possible pathways are being simultaneously considered in the solution of an inverse kinetics problem. Hence, multiple potential mechanisms may be regarded as feasible, and their parameters estimated together for the elucidation of the prevailing pathways. For instance, combining the *g*, *da* and *dcs* reaction types listed above yields the matrix in Fig. 1 (inverse reactions are suppressed). The *gdacs* reaction network thus represents the maximum variety and depth in complexity, involving all types of species, either directly measurable through analytical methods or mathematically inferred from underlying material balances.

Reactions Types
(inverse reactions suppresed)

State Types		g	d			a	c			s		Reaction	Type
$\dot{\mathbf{c}}_{\mathbf{g}}$	A	-1	-1	0	0	0	0	0	0	0	g	$A + B \rightarrow C$	Homogeneous Reaction
	B	-1	0	-1	0	0	0	0	0	0		$A + * \rightarrow A*$	
	C	1	0	0	1	0	0	0	0	0	d	$B + * \rightarrow B*$	Gas ad/desorption / Surface-gas coupling
$\dot{\mathbf{c}}_{\mathbf{a}}$	A*	0	1	0	0	-1	-1	0	0	0		$C + * \rightarrow C*$	
	B*	0	0	1	0	-1	0	-1	0	0	a	$A* + B* \rightarrow C* + *$	Reaction Between Adsorbed Molecules
	C*	0	0	0	-1	1	0	0	1	0		$A* + * \rightarrow 2D*$	
$\dot{\mathbf{c}}_{\mathbf{s}}$	D*	0	0	0	0	0	2	0	0	-1	c	$B* + * \rightarrow 2E*$	Reaction Between Adsorbed Molecules and Intermediate Species
	E*	0	0	0	0	0	0	2	-1	-1		$E* + F* \rightarrow C* + *$	
	F*	0	0	0	0	0	0	0	-1	1	s	$D* + E* \rightarrow F* + *$	Reaction Between Intermediate Species
	*	0	-1	-1	1	1	-1	-1	1	1			

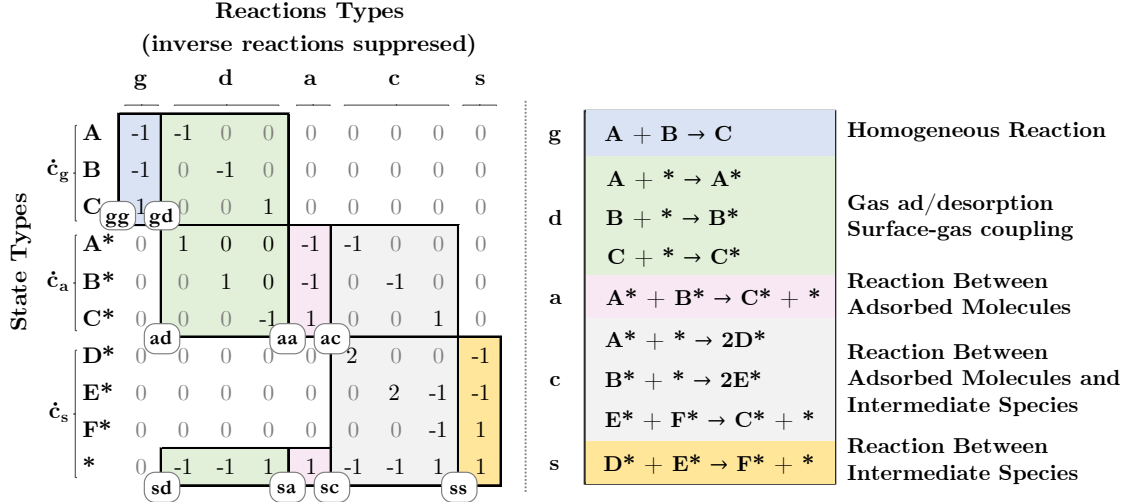(submatrix labels: gg, gd, ad, aa, ac, sd, sa, sc, ss)

Figure 1: Complex Reaction Network Stoichiometry Matrix Example.

This framework allows for classification of types of elementary reactions based solely on the chemical reaction stoichiometry matrix. In section 3, we build anecdotal examples of reaction network subtypes to guide the discussion on the need for operando analyses for the data-driven estimation of kinetic parameters, and serve as scaffolds for the construction of chemical kinetics PINNs.

2.3 Physics-Informed Neural Networks as Surrogate Models

The universal approximation theorem confers to feed-forward (FF) non-polynomial NNs the ability to approximate any continuous function given an appropriate number of neurons [49, 50, 51, 52]. The FFNN structures have proven to

serve as appropriate basis functions for ODEs, which may also be extended to PDEs [53, 54, 41]. FFNNs can hence be used to approximate ODE initial value problems associated with chemical kinetics. We further show their use in a collocation framework to solve properly scaled inverse problems, under optimal choice of activation function and NN architecture necessary for the solution of stiff ODEs.

Let $\boldsymbol{\omega_s} = \{\mathbf{W}^k \,|\, \mathbf{W}^k \in \mathbb{R}^{n_k \times m_k}, \, k \in \{1, 2, ..., n_l + 1\}\}$ denote a parameter tensor for an n_l-layer feed-forward neural network (FFNN). Let $\mathbf{x} \in \mathbb{R}^n$ be a state vector and $\mathbf{t} \in \mathbb{R}^p$ be collocation points, and $\phi_k(\cdot) : \mathbb{R}^m \to \mathbb{R}^m$ be the activation function associated with layer k. The input-output structure for the FF-NN can be represented by the recurrence relation in (3.1).

$$
\begin{aligned}
\mathbf{z}_k &= \left\{ \phi_k \left(\mathbf{W}^k \mathbf{z}_{k-1} \right) \,\big|\, k \in \{1, 2, ..., n_l\}, \, z_0 = t \right\} \\
\mathbf{x}(t, \boldsymbol{\omega_s}) &= \mathbf{z}_{n_l+1} = \mathbf{W}^{n_l+1} \mathbf{z}_{n_l}
\end{aligned}
\tag{3.1}
$$

We refer to the general use of neural-networks methods to solve kinetic ODEs as kINNs, kinetics-informed neural networks, using PINNs as a scaffold with suitable modifications to enforce structural boundary conditions (BCs) or other phenomenological constraints. The general first-order non-homogeneous coupled ODE implicit representation of an MKM in terms of the kINN is shown in (3.2), which can be generalized to p^{th} order if $\mathbf{x}$ is of class $\mathcal{C}^p$ with respect to t.

$$
g[t, \mathbf{x}, \dot{\mathbf{x}}] = g(t, \mathbf{x}(t, \boldsymbol{\omega_s}), \dot{\mathbf{x}}(t, \boldsymbol{\omega_s})) = g(t, \boldsymbol{\omega_s}) = 0
\tag{3.2}
$$

Where $\mathbf{x}$ is the output of the SM encompassing one or more NNs and suitable output transformation layer, and $\dot{\mathbf{x}}$ is evaluated through AD of the former with respect to the single-input input layer. The degrees of freedom in the algebraic equation are all contained in the SM parameters, $\boldsymbol{\omega_s}$, which therefore determines the solution of (3.2) given proper BCs. The time derivative of the SM, $\dot{\mathbf{x}}$, is evaluated with respect to the single-input input layer through AD coupled with just-in-time (JIT) compilation through the JAX library [55, 56]. The objective function of the neural network is trained with the *Adam* optimization algorithm. The MKM implicit ODE can be represented in terms of (3.2) as show in (3.3).

$$
g(t, \boldsymbol{\omega_s}) = \dot{\mathbf{x}}(t, \boldsymbol{\omega_s}) - \mathbf{M} \left(\mathbf{k}(\theta) \odot f(\mathbf{x}(t, \boldsymbol{\omega_s})) \right) = 0
\tag{3.3}
$$

The implicit form, g, is loosely utilized in the next section to denote residuals of the SM-based formulation to be minimized over collocation point in integration domains.

2.3.1 Structural Boundary Conditions

Dirichlet Structural BCs are implemented by designing operators that impose certain behavior onto the NN in regions or points in the integration domain. In general, for kinetics ODEs in particular, an operator $\mathbf{C}_{\mathbf{bc}}^{\mathbf{D}}[\cdot]$ must be devised to enforce *Dirichlet*-type boundary conditions, i.e. $\mathbf{x}(\boldsymbol{\omega_s}, t_0) = \mathbf{x}_0 \,\forall\, \boldsymbol{\omega_s}$. A natural choice to satisfy *Dirichlet* BC is the hyperbolic tangent function $\phi(t) = \tanh(\kappa t) \in \mathcal{C}^1$, such that $\tanh(0) = 0$ and $\partial_t \tanh(0) = 1$. Here $\kappa \in \mathbb{R}_+$ is a characteristic time constant that must be determined for each chemical system or also modelled in terms of independent variables, as shown in section 2.3.2. Here we utilize $\hat{\mathbf{x}}$ and $\mathbf{x}[\hat{\mathbf{x}}]$ as the NN and SM outputs, respectively, with parameters $\boldsymbol{\omega_s}$ and independent variable t. The BC operator is defined as $\mathbf{C}_{\mathbf{bc}}^{\mathbf{D}}[\hat{\mathbf{x}}(\boldsymbol{\omega_s}, t), \mathbf{x}_0] = \hat{\mathbf{x}}(\boldsymbol{\omega_s}, t)\phi(\kappa(t - t_0)) + \mathbf{x}_0 (1 - \phi(\kappa(t - t_0)))$.

From simple inspection, the BC is universally enforced in (3.4) and gradients depend are a function of SM parameter in (3.5).

$$
\hat{\mathbf{x}}(\boldsymbol{\omega_s}, t)\phi(\kappa(t - t_0))|_{t=t_0} + \mathbf{x}_0 (1 - \phi(\kappa(t - t_0))) |_{t=t_0} = \hat{\mathbf{x}}(\boldsymbol{\omega_s}, t_0)\phi(0) + \mathbf{x}_0 (1 - \phi(0)) = \mathbf{x}_0
\tag{3.4}
$$

$$
\begin{aligned}
\partial_t \, \hat{\mathbf{x}}(\boldsymbol{\omega_s}, t)\phi(\kappa(t - t_0))|_{t=t_0} + \partial_t \, \mathbf{x}_0 (1 - \phi(\kappa(t - t_0))) |_{t=t_0} &= \\
\dot{\hat{\mathbf{x}}}(\boldsymbol{\omega_s}, t_0)\phi(0) + \mathbf{x}(\boldsymbol{\omega_s}, t_0)\partial_t \dot{\phi}(0) - \mathbf{x}_0 \partial_t \dot{\phi}(0) &\\
= \hat{\mathbf{x}}(\boldsymbol{\omega_s}, t_0) - \mathbf{x}_0 &
\end{aligned}
\tag{3.5}
$$

Such that the SM vanishes at $t = t_0$, (3.4), with the continuous time-derivative of $\mathbf{x}$ equal to the NN output, $\hat{\mathbf{x}}$, (3.5). Therefore, any *Dirichlet* BC can be automatically satisfied in this form for kINNs. The same approach can be extended to *Neumann* BCs with proper choice of ϕ function. The application of the operator $\mathbf{C}_{\mathbf{bc}}^{\mathbf{D}}$ on MKMs follows naturally by making $\mathbf{x} \sim \mathbf{c}$ in (1.1) as depicted by (3.3).

2.3.2 Characteristic Time

In the case of *tanh* as choice of ϕ for $\mathbf{C}_{\mathbf{bc}}^{\mathbf{D}}$, a proper choice for characteristic time constant can be circumvented by replacing κ by an ancillary shallow NN that can be learned along with the underlying SM NNs, with a monotonically increasing output layer. In this work we adopted an exponential form for κ whenever solving the forward ODEs, as in (3.6).

$$\kappa = e^{\mathbf{u}(\boldsymbol{\omega_t}, t-t_0)} \tag{3.6}$$

Where $\mathbf{u}$ is a shallow FFNN with parameters $\boldsymbol{\omega_t}$. The BC operator on ϕ and κ, $C_{\mathbf{bc}}^{\mathbf{D}}$, can be thus be represented in terms of ϕ and $\mathbf{u}$ as follows in (3.7), where parameter dependencies were suppressed. For the sake of simplicity, we further assume κ is embedded into the SM structure $\mathbf{x}$ so its weights, $\boldsymbol{\omega_t}$, are included in $\boldsymbol{\omega_s}$.

$$\mathbf{C}_{\mathbf{bc}}^{\mathbf{D}}[\hat{\mathbf{x}}(t), \mathbf{x}_0] = \hat{\mathbf{x}}(t) \tanh\left(e^{\mathbf{u}(t-t_0)}(t-t_0)\right) + \mathbf{x}_0 \left(1 - \tanh\left(e^{\mathbf{u}(t-t_0)}(t-t_0)\right)\right) \tag{3.7}$$

2.3.3 Normalization Constraints

MKMs typically treat the concentration of adsorbed and intermediate species in terms of fractions of the total number of available active sites. It is hence desirable for SMs to structurally enforce normalization, eliminating a degree of freedom. We propose the projection of radius-one hypersphere onto the natural basis to enforce inherent normalization.

Let $\mathbf{x}(\boldsymbol{\omega_s}, t) \in \mathbb{R}^p$ be the output values of the constrained surrogate related to the neural network $\hat{x}(\boldsymbol{\omega_s}, t) \in \mathbb{R}^{p-1}$, such that

$$x_i(\boldsymbol{\omega_s}, t) = \left(1 - \sin^2\left(\hat{x}_i(\boldsymbol{\omega_s}, t)\right)\right) \prod_{j<i} \sin^2\left(\hat{x}_j(\boldsymbol{\omega_s}, t)\right) \ \forall \ i < p; \ i, j \in \mathbb{N}$$
$$x_p(\boldsymbol{\omega_s}, t) = \prod_{j<p} \sin^2\left(\hat{x}_j(\boldsymbol{\omega_s}, t)\right) \tag{3.8}$$

Such trigonometric transformation enforces bounds to $\mathbf{x}$, such that $0 \leq \mathbf{x} \leq 1$ and $\sum \mathbf{x} = 1 \ \forall \ \hat{\mathbf{x}} \in \mathbb{R}^{p-1}$. The SM output can hence be denoted as the result of the normalization-constraint operator C_N over the NN output $\hat{\mathbf{x}}$, i.e. $\mathbf{x} = C_N[\hat{\mathbf{x}}]$.

2.4 kINN Forward Problem

The surrogate approach to solving the forward problem involves training the neural network to satisfy the BCs, as well as the differential form specifying the reaction kinetics in (3.2), over a defined time grid. We constrain the analysis to non-spatially dependent systems, formulating the problem for an ideal, perfectly stirred, uniform, pseudohomogeneous batch reactor model, where the underlying ODE describing the state-variables evolution over time is analogous to (1.1). The general solution for the forward problem, i.e. solving the ODE, consists of minimizing the implicit form in (3.2) given consistent BCs for the states, where the real-valued function g corresponds to (1.1) in standard form given fixed model parameters, $\boldsymbol{\omega_m}$, which is outlined by (3.3). The initial value is represented by $\mathbf{x_0}$, and the residual is evaluated over the collocation points, $\mathbf{t}$, as shown in (4.1). This represents the minimization of the residual j_m, under some norm γ, between the Jacobians of the states from the derivative-based rate laws and the surrogate model. AD provides the Jacobians, i.e. $\dot{\mathbf{x}}$ and $\partial_{\boldsymbol{\omega_s}} j_m(t, \boldsymbol{\omega_s})$, for the objective function. The *Adam* optimization algorithm is used to minimize the residual, j_m. All case studies in this work have been developed under $\gamma = 2$.

$$\min_{\boldsymbol{\omega_s}} \quad j_m(\boldsymbol{\omega_s}) = ||g(\mathbf{t}, \boldsymbol{\omega_s})||_{\gamma>0}$$
$$\text{s.t.} \quad \mathbf{x}(t_0, \boldsymbol{\omega_s}) = \mathbf{x_0} \tag{4.1}$$

Importantly, in the case of MKMs for which surface coverage fractions are in play, the normalization of elements of g can be structurally enforced as shown in 2.3.3, by having $g(\mathbf{x}(\hat{\mathbf{x}}))$ as in (3.8). Furthermore, when dealing with mixed normalized (coverage) and non-normalized (bulk-phase) chemical species, the problem can be split into two separate underlying NNs which are concatenated in the SM output layer: a non-normalized one representing the mapping between time and gas-phase concentrations and a normalized one mapping between time and surface coverages. The implicit representation of MKM kINN is given in (3.3). When structural BCs are applied and normalization constraints are included, the surrogate model consists of the following combinations of underlying NNs and operators, (4.2), using the same nomenclature as defined in 2.1.

$$\mathbf{x} = C_{\mathbf{bc}}^{\mathbf{D}} \begin{bmatrix} \hat{\mathbf{x}}_{\mathbf{g}} \\ C_{\mathbf{N}}[\hat{\mathbf{x}}_{\mathbf{as}}] \end{bmatrix} \tag{4.2}$$

Where $\hat{\mathbf{x}}_{\mathbf{g}}$ corresponds to the output layer of the NN associated with gas (unbound) species and $\hat{\mathbf{x}}_{\mathbf{as}}$ encompasses the output layer of subjacent NN that conveys the $p-1$ angular components that, once mapped through the operator $C_{\mathbf{N}}$ outputs p normalized coverage fractions. Finally, the initial-value BC operator $C_{\mathbf{bc}}^{\mathbf{D}}$ is applied onto the combined (concatenated) NNs.

Algorithm 1 Forward kINN

1: **Input:** $\mathbf{t}_0, \mathbf{x}_0, \boldsymbol{\omega_m}$, maximum iterations M, tolerance T
2: Initialize NN, $\hat{\mathbf{x}}_i$, SM parameters $\boldsymbol{\omega_s}$, residual $j_\mathrm{m}(\boldsymbol{\omega_s})$
3: **while** iter $< M$ **or** $j_\mathrm{m}(\boldsymbol{\omega_s}) < T$ **do**
4:     Predict states $\mathbf{x}$ from SM over $\mathbf{t}$
5:     Predict states derivatives $\dot{\mathbf{x}}$ from SM over $\mathbf{t}$
6:     Compute objective function $j_\mathrm{m}(\boldsymbol{\omega_s})$ over $\mathbf{t}$
7:     Compute gradients of $j_\mathrm{m}(\boldsymbol{\omega_s})$ with AD; $\partial_{\boldsymbol{\omega_s}} j_\mathrm{m}(\boldsymbol{\omega_s})$
8:     Update SM parameters $\boldsymbol{\omega_s}$ using *Adam*
9: **Output:** Concentration profile $\mathbf{x}(t, \boldsymbol{\omega_s})$

2.5 kINN Inverse Problem

Under the same model assumptions from the forward MKM problem, we extend the PINN-based kINN approach to MKM inverse ODE problems by letting $\tilde{\mathbf{x}} \in \mathbb{R}_+^{n \times d}$ represent observed (measured) concentrations of n chemical species over the respective d time-points $\mathbf{t} \in \mathbb{R}_+^d$, and including it in the objective function. As in the forward problem, state variables are represented by surrogates from separate NNs and applied operators, as in (4.2), i.e. $\mathbf{x}(t, \boldsymbol{\omega_s})$. The inverse problem consists of the forward kINN model with the inclusion of interpolation of observed datapoints, $\tilde{\mathbf{x}}$ as a regularization term in the objective function. This interpolation is supported by the universal approximation theorem. The error to be minimized becomes the combination of the residual between the SM and the observed data (interpolation), j_d, and that between SM and the kinetic model describing the behavior (physics regularization), j_m, in a similar approach to Raissi *et al.* [57]. As in the forward problem, the combined error is minimized using the *Adam* optimization algorithm.

$$j_\mathrm{d}(\boldsymbol{\omega_s}) = ||\mathbf{x}(\mathbf{t}, \boldsymbol{\omega_s}) - \tilde{\mathbf{x}}||_{\gamma>0}$$
$$j_\mathrm{m}(\boldsymbol{\omega_s}, \boldsymbol{\omega_m}) = ||\partial_t \mathbf{x}(\mathbf{t}, \boldsymbol{\omega_s}) - \mathbf{M}\left(\mathbf{k}(\boldsymbol{\omega_m}) \odot f(\mathbf{x}(\mathbf{t}, \boldsymbol{\omega_s}))\right)||_{\gamma>0}$$

Finally, let $\boldsymbol{\omega} = \{\boldsymbol{\omega_s}, \boldsymbol{\omega_m}\}$ combine the residual from the forward-kINN as regularization term with weight α into the inverse ODE.

$$\min_{\boldsymbol{\omega}} \quad j_\mathrm{t}(\boldsymbol{\omega}) = j_\mathrm{m}(\boldsymbol{\omega_s}, \boldsymbol{\omega_m}) + \alpha j_\mathrm{d}(\boldsymbol{\omega_s})$$
$$\text{s.t.} \quad \boldsymbol{\omega_m} \in \mathbb{R}^{\dim(\boldsymbol{\omega_m})}, \ \boldsymbol{\omega_s} \in \mathbb{R}^{\dim(\boldsymbol{\omega_s})}; \alpha \in \mathbb{R}_+ \tag{5.1}$$

The regularization hyperparameter α needs to be optimized given an MKM postulate and the observed data. Furthermore, the regularization intensity plays a decisive role in the case of noisy data, since initial training steps with excessive regularization potentially lead to local convergence far from the global optimum in the model parameter space. A conservative strategy implemented in this work is to perform initial training steps at low values α and then proceed with increments in α until an inflection point between $\log(j_\mathrm{d})$ and $\log(j_\mathrm{m})$ is observed. In section 3.3, a sensitivity analysis is performed over α values for an anecdotal MKM's related inverse problem. Also, the learning of model parameters must take place in a similar domain and scale as the SM-associated NN weights. As kinetic parameters have non-negative values and may substantially vary in order of magnitude, we choose to train an exponential mapping, i.e. set $\mathbf{k} = \exp(\boldsymbol{\omega_m})$. Furthermore, for the *tanh* activation function, a saturation occurs at arguments of ± 5, and for *swish* with a similar pattern as *exp*, such that $\boldsymbol{\omega_m}$ and $\boldsymbol{\omega_s}$ are of the same order of magnitude. This allows that they be grouped as a lumped variable and trained simultaneously.

Algorithm 2 Inverse kINN

   Input: $\mathbf{t}, \tilde{\mathbf{x}}$, maximum iterations M, tolerance T
2: Initialize NN, $\hat{\mathbf{x}}_i$, SM parameters $\boldsymbol{\omega_s}$, kinetic model parameters $\boldsymbol{\omega_m}$, residual $j_\mathrm{t}(\boldsymbol{\omega})$
   while iter $< M$ **or** $j_\mathrm{t}(\boldsymbol{\omega}) < T$ **do**
4:     Predict states $\mathbf{x}$ from SM over $\mathbf{t}(t, \boldsymbol{\omega_s})$
       Predict states derivatives $\dot{\mathbf{x}}$ from SM over $\mathbf{t}(t, \boldsymbol{\omega_s})$
6:     Compute objective function $j_\mathrm{t}(\boldsymbol{\omega})$ over $\mathbf{t}$
       Compute gradients of $j_\mathrm{t}(\boldsymbol{\omega})$ with AD; $\partial_\omega j_\mathrm{t}(\boldsymbol{\omega_s}, \boldsymbol{\omega_m})$
8:     Update SM parameters $\boldsymbol{\omega_s}$ and kinetic model parameters $\boldsymbol{\omega_m}$ using *Adam*
   Output: Kinetic model parameters $\boldsymbol{\omega_m}$

---

3 Results

Different non-stiff anecdotal kinetic models are solved in the forward sense to justify the use of FFNNs as bases for the solution of the inverse problem. We increase reaction network complexity, moving through (i) an example of a simple homogeneous case (type *g*), (ii) one involving reaction between adsorbed molecular species (type *da*), (iii) one including reactions between adsorbed molecules and reaction intermediates (type *dc*) and (iv) one including a reaction between intermediates (type *dcs*). Abstract species are used in the following examples, which may suit a myriad of actual chemical reactions.

The scheme for solving the forward kINNs problem, and the inverse problem in three different scenarios, is shown in Fig. 2. The scenarios, which vary in the completeness of knowledge about the chemical systems and the presence of noise representing actual measurements, are further detailed in section 3.3. To obtain the ground truth set for the comparison of the forward kINN results, the ensuing ODEs of all combinations of MKMs and BCs were solved numerically with the stiff-nonstiff algorithm LSODA from the FORTRAN ODEPACK [58, 59]. Similarly, the generation of abstract observed data and further per-scenario processing was carried out using the same numerical method, as represented by the (A) region. For the solution of forward kINNs, the MKM stoichiometry matrix, the kinetic parameters, ω_m, and BCs is provided (D). Conversely, for the inverse problem, ω_m is adjusted and abstract data is generated and provided as if experimentally assessed (B). The kINN SM-based solution flow diagram is portrayed in (C), exhibiting the interplay between the SM and the physical models, and the final cost function, j_m.
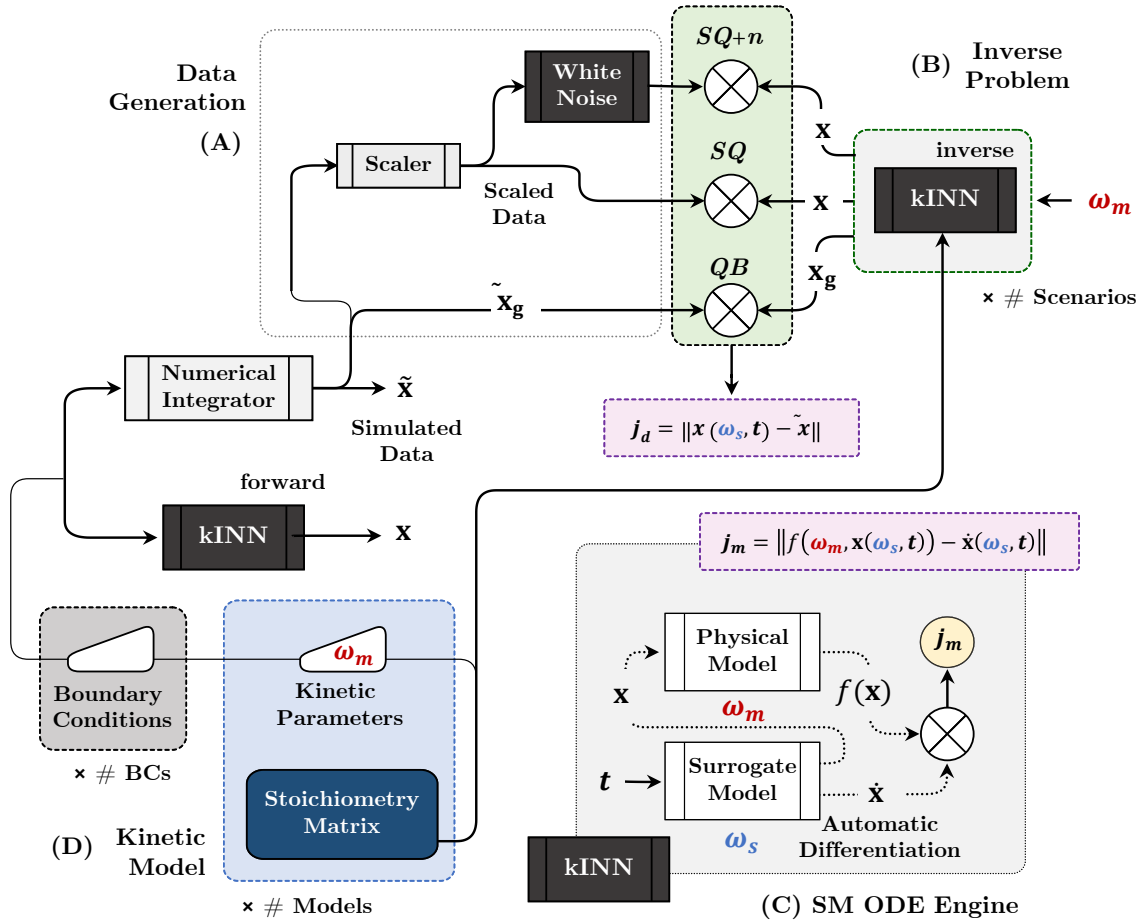
Figure 2: Data generation (A), forward and inverse (B) kINNs scheme, kINNs formulation (C) and kinetic model (D).

3.1 Reaction Network Examples

3.1.1 Reaction Network Type *g*

In the simplest homogeneous case conveyed in **g**.1, species A and B reversibly react to form C in gas phase or in solution. To simplify the problem solution, the rate constants **k** are considered state- and spatially-independent, which implies an isothermal concentrated-parameter system. Various chemical reactions can be represented in terms of **g**.1, such as dissociation, dimerization, and lumped-kinetics surrogates derived from complex kinetics data. A simple lumped-kinetics that could be represented as in **g**.1 would be ethylene dimerization to 1-butene, where A and B would represent ethylene, C_2H_4 and C would denote 1-butene, C_4H_8.

$$A + B \underset{k_{-1}^d}{\overset{k_1^d}{\rightleftharpoons}} C \tag{g.1}$$

3.1.2 Reaction Network Type *da*

For the *da* example, we transpose the reactive step from the homogeneous phase in **g**.1 to the heterogeneous phase. by including the adsorption and desorption elementary steps, **d**.(1-3). and a surface reaction between adsorbed molecules, **a**.1. The set of elementary reactions may be a simple representation of catalyzed dimerization reaction, in the same spirit of section 3.1.1, except for the reaction would instead occur in a single step on a catalyst surface.

$$A + * \underset{k_{-1}^d}{\overset{k_1^d}{\rightleftharpoons}} A* \tag{d.1}$$

$$B + * \underset{k_{-2}^d}{\overset{k_2^d}{\rightleftharpoons}} B* \tag{d.2}$$

$$C + * \underset{k_{-3}^d}{\overset{k_3^d}{\rightleftharpoons}} C* \tag{d.3}$$

$$A* + B* \underset{k_{-1}^a}{\overset{k_1^a}{\rightleftharpoons}} C* \tag{a.1}$$

3.1.3 Reaction Network Type *dc*

The *a*-type reaction is replaced by two *d*-type reactions, i.e. reaction involving one intermediate species ($D*$), which does not have a corresponding gas phase species. The *d*-type adsorption and desorption steps, **d**.(1-3), are suppressed, i.e. implicitly included in the set of elementary reactions comprising the *dc*-example below. An example of a reaction that could be represented overall in such terms would be ethylene epoxidation, where A would represent ethylene, B molecular oxygen, O_2, D atomic oxygen, O, and C would represent ethylene epoxide.

$$B* + * \underset{k_{-1}^c}{\overset{k_1^c}{\rightleftharpoons}} 2D* \tag{c.1}$$

$$A* + D* \underset{k_{-2}^c}{\overset{k_2^c}{\rightleftharpoons}} C* + * \tag{c.2}$$

3.1.4 Reaction Network Type *dcs*

Elementary reactions between surface intermediates exclusively are embodied by *s*-type reactions. In the following example, $D*$, $E*$ and $F*$ are reaction intermediates with no stable desorbed counterpart. In terms of heterogeneous kinetics complexity depth, *s*-type reactions entail intermediates that cannot be directly measured or inferred from ordinary gas/liquid phase effluent/heads-space analytical chemistry techniques, such as gas/liquid chromatography and mass spectroscopy. Butene cracking followed by hydrogenation to ethane could be represented by such a reaction mechanism, where A would entail butene, D ethylene, B molecular hydrogen, H_2, E atomic hydrogen, H, F is the ethyl intermediate, and C ethane.

$$A * + * \underset{k_{-3}^c}{\overset{k_3^c}{\rightleftharpoons}} 2D* \tag{c.3}$$

$$B * + * \underset{k_{-4}^c}{\overset{k_4^c}{\rightleftharpoons}} 2E* \tag{c.4}$$

$$D * + E* \underset{k_{-1}^s}{\overset{k_1^s}{\rightleftharpoons}} F * + * \tag{s.1}$$

$$F * + E* \underset{k_{-5}^c}{\overset{k_5^c}{\rightleftharpoons}} C * + * \tag{c.5}$$

3.2 Forward kINNs

Three-hidden-layer NNs with *tanh*, *swish* and *tanh* activation functions respectively were utilized as basis for the solution of forward kINNs. For the overall reversible reaction $A \rightleftharpoons B + C$ represented by each reaction network type, several anecdotal examples for the forward problem were solved given a set of diverse *Dirichlet* BCs in terms of x_A, x_B and x_C at $t = 0$, and sets of rate constants chosen to portray different degrees of stiffness, i.e. faster rates for surface reactions, as in Table 1. For all reaction types, hyperbolic tangent *Dirichlet*-type boundary-condition operators, C_{bc}^D, were applied to SMs, as in section 2.3.1. Ancillary single-hidden-layer FFNNs with three (*a*- and *da*-type) or six (*dc* and *dcs*-type) neurons and *swish* activation functions were trained in parallel for each forward problem to allow for a time-dependent characteristic time or gain for the BC operator, which provides a learning gain-layer in the case of varying stiffness, as defined in section 2.3.2. The forward kINNs are defined in terms of their associated reaction network (stoichiometry matrix), BCs and kinetic parameters. Normalization constraints are not included in the forward kINNs since they bring additional non-linearity that is unnecessary given that the problems are well-defined by their ODEs and BCs.

Table 1: Rate Constants and NN architecture

Type	Rate Constants											Architecture (layers)	
	$\frac{k_1^g}{k_{-1}^g}$	$\frac{k_1^d}{k_{-1}^d}$	$\frac{k_2^d}{k_{-2}^d}$	$\frac{k_3^d}{k_{-3}^d}$	$\frac{k_1^a}{k_{-1}^a}$	$\frac{k_1^c}{k_{-1}^c}$	$\frac{k_2^c}{k_{-2}^c}$	$\frac{k_3^c}{k_{-3}^c}$	$\frac{k_4^c}{k_{-4}^c}$	$\frac{k_5^c}{k_{-5}^c}$	$\frac{k_1^s}{k_{-1}^s}$	Hidden [*tanh,swish,tanh*]	Output [linear,$C_N[\cdot]$]
g	$\frac{10}{1}$	-	-	-	-	-	-	-	-	-	-	[5, 5, 5]	[3,-]
da	-	$\frac{10}{4}$	$\frac{40}{60}$	$\frac{200}{40}$	$\frac{100}{80}$	-	-	-	-	-	-	[12, 12, 12]	[3,4]
dc	-	$\frac{20}{8}$	$\frac{16}{4}$	$\frac{12}{8}$	-	$\frac{1200}{400}$	$\frac{2000}{1600}$	-	-	-	-	[16, 16, 16]	[3,5]
dcs	-	$\frac{20}{8}$	$\frac{24}{12}$	$\frac{16}{40}$	-	-	-	$\frac{640}{960}$	$\frac{160}{80}$	$\frac{560}{160}$	$\frac{640}{240}$	[20, 20, 20]	[3,7]

The cost function to be minimized for learning the solution of the ODEs using kINNs was defined in terms of mean-square error, MSE. The JAX library was utilized for the forward AD, allowing the obtainment of the gradients with respect to the SMs' parameters, which were optimized with the *Adam* algorithm [60]. Since BCs are structurally enforced by C_{bc}^D, the ODE solutions are found by minimizing j_d, i.e. finding ω_s such that the state derivatives from the physical model given SM state estimates, $\mathbf{x}$, match those estimated by forward AD of the SMs with respect to their single value time input.

Table 2 compares the numerical results with the SM derived for the forward kINNs. The set of BCs was defined so as to represent the two scenarios: one in which there are similar concentrations of both reactants, (BC 1, $\{x_A, x_B, x_C, x_*\}_{t_0} = \{0.6, 0.4, 0.0, 1.0\}$), and another to represent the case where considerable amount of product is present, (BC 2, $\{x_A, x_B, x_C, x_*\}_{t_0} = \{0.2, 0.3, 0.5, 1.0\}$). The two conditions were chosen to represent chemical transformations that predominantly run in the forward or reverse reaction, and thus inform further attempts to solve the related inverse problems. The performance metrics are defined as a function of the collocation points about which the SM is trained, which is exemplified in Fig. 3 by the *dcs*-reaction network type solution for BC 1. For all scenarios, about 100 collocation points were sampled in a logarithm space in the time domain, such that there is higher probability that points be placed in regions of higher derivatives near the BC. To refine the solutions, SMs were trained in three stages with varying training step size from 10^{-4} to 10^{-6}, over a total of 10^3 epochs, and 10^2 iterations per epoch. The convergence criteria was that either $j_t < 10^{-12}$ or that the maximum number of epochs was reached. Evidently, absolute error metrics such as MAE and MSE cannot be compared between different MKMs and BCs, since different ODE solutions may have different stiffness and therefore larger or smaller derivative values in different scales as compared to actual state values. Irrespective of MKM and BCs, the coefficient of determination for all examples is unity
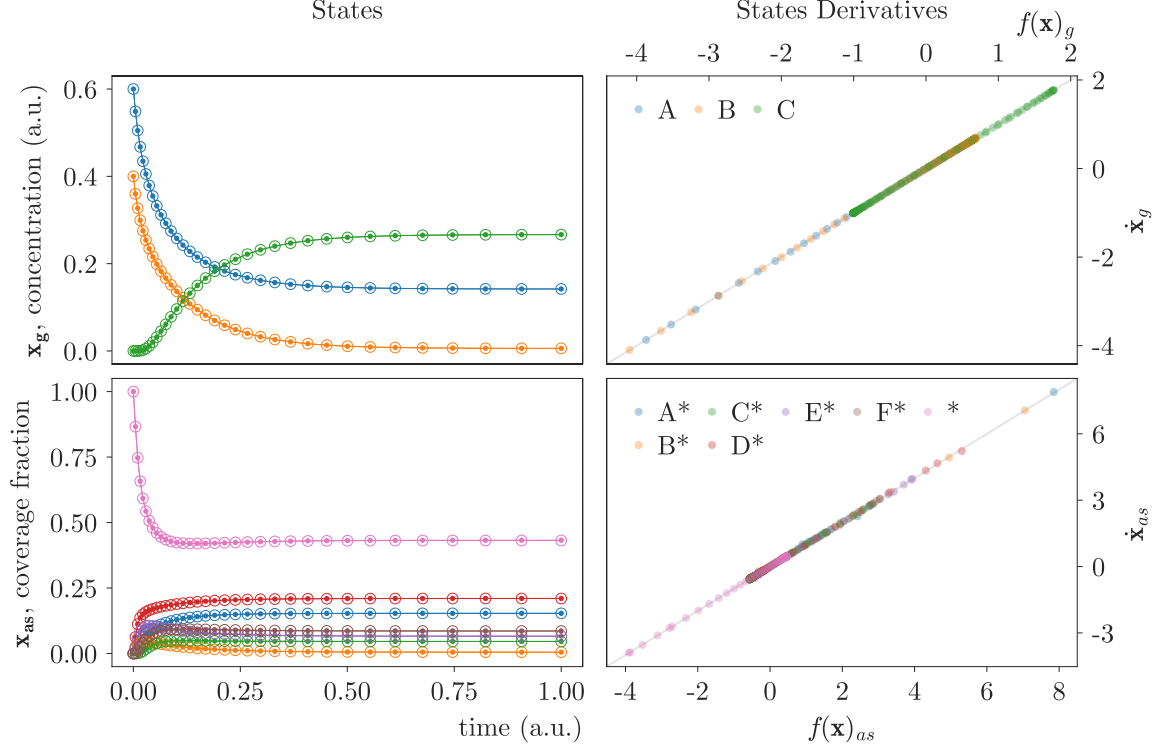
Figure 3: Reaction network type *dcs* forward solution with BC of $\{x_A, x_B\}_{t_0} = \{0.6, 0.4\}$ and clean surface (BC 1). States solution (top) (SM, open circles; numerical solution, closed markers) under minimization of (4.1) (bottom) for observable variables (unbound species, top) and latent variables (coverages, bottom).

within round-off error. Such results support the hypothesis that NNs can be utilized as general bases for the solution of forward kINNs, which is the underpinning assumption for their utilization in the solution of inverse kINNs.

In Fig. 3, the final solution for the type-*dcs* MKM under the reactant-only BC (BC 1) is shown. The predicted states (open circles) line up over the numerical solution (solid circles) for both the observable concentrations $\mathbf{x_g}$ (top) and coverage fractions $\mathbf{x_{as}}$ (bottom). The minimization of the underlying forward kINNs cost function j_m can be inferred by the parity plot between physical (kinetic) model derivatives from estimated states, and their derivative estimates from forward AD. For the remaining cases, the standardized derivative parity plots for the observed and latent states can be compared with the coefficient of determination reported in Table 2, for which coefficients of determinations are computed as the average squared correlation coefficient over different states.

Table 2: Forward kINNs Results Summary - Performance Metrics (vs Ground Truth)

Type	BC	Observed States Derivatives			Latent States Derivatives		
		r^2	10^2 MAE	10^4 MSE	r2	10^2 MAE	10^3 MSE
g	1	1.00	5.50×10^{-2}	3.92×10^{-3}	-	-	-
	2	1.00	1.79×10^{-3}	4.27×10^{-6}	-	-	-
da	1	1.00	0.251	0.112	1.00	0.297	1.84×10^{-2}
	2	1.00	0.893	2.49	1.00	1.73	1.32
dc	1	1.00	0.330	0.209	0.999	1.26	0.470
	2	1.00	0.278	0.168	1.00	0.738	0.142
dcs	1	1.00	0.218	9.87×10^{-2}	1.00	0.713	0.400
	2	1.00	6.89×10^{-2}	8.45×10^{-3}	1.00	0.333	3.58×10^{-2}

11

3.3 Inverse kINNs

The forward numerical solution data for the different reaction network types were utilized as the ground truth data for the inverse kINN studies. We define three scenarios or categories for the inverse kINNs involving heterogeneous models, i.e. reaction networks other than type-g only, as portrayed by scheme (A) in Fig. 2.

- **Quantitative Unbound Species (QB):** no information about chemical intermediates and adsorbed molecules; the only measured states are related to unbound chemical species; e.g. gas, liquid species that can be measured downstream (flow-type reactors) or sampled from the reactor headspace (batch) with standard analytical equipment: gas/liquid chromatography, mass spectroscopy. For the QB scenarios, only data generated for the observable variables ie. unbound chemical species are fed to the training loop, such that j_{d} has the length of the observable variables. Meanwhile, j_{m}, in (5.1), encompasses derivatives for all present (or assumed present) chemical species and intermediates that are measured and unmeasured.

- **QB + Semiquantitative Bound Species (SQ):** it is assumed that deconvolved SQ data can be obtained, e.g. with operando analytical techniques, for all adsorbed molecules and stable reaction intermediates, and that the SQs are linearly related to the concentration of such species. For the SQ scenario, the surface coverages generated from the numerical solution of the forward problem are rescaled by dividing the time series values by their standard deviation. In the inverse kINNs problem, the scaling factors between fractional coverages and the SQintensities are also included as parameters to be learned in $\boldsymbol{\omega_{\mathrm{s}}}$.

- **SQ + noise (SQ+n):** a normally distributed (white noise) homoscedastic error is added to the scaled data for the SQ scenarios. This setup intends to represent a more realistic scenario and the role of the regularization parameter α is paramount in determining whether the noise is being fitted or the underlying physics is properly represented.

Unlike the forward kINNs, whose solutions for the two BCs are independent of each other, the inverse kINNs are interdependent since they share, and thus act on, the same kinetic model and its kinetic parameters encoded in $\boldsymbol{\omega_{\mathrm{m}}}$. The solution of inverse kINNs can involve multiple experimental (observed) data to which independent SMs are trained under a single underlying physical model. In this section, all inverse kINN estimated model parameters were obtained by the simultaneous interpolation of BC 1 and 2 abstract forward data from the numerical solution of the MKM ODEs.

The normalization constraint operator, $C_{\mathbf{N}}$, plays a major role in the learning of scaling factors for the SQ scenarios, since it constrains the scaling factors to describe normalized (fractional) coverages. Importantly, the regularization parameter, α, (5.1). must be determined for each specific reaction network type and associated data as in any method where hyperparameters need be optimized. The optimal α range can be inferred from the relative change in the residual terms with respect to data, j_{d}, and to the physical model, j_{m}. We hence undertook sensitivity analyses on the residual terms for the anecdotal dcs-type reaction network scenarios to estimate their Pareto frontiers. We successively increased α from 10^{-6} to 10^6 and further decreased it back to 10^{-6} to assess the non-dominated point in the estimated Pareto sets for the SQ and SQ+n scenarios. For each value of α, the same SM learning strategy in terms of convergence criteria, number of epochs, iteration per epoch and solution refinement as defined in section 3.2 was adopted.

Fig. 4 (B) shows the learned estimated Pareto sets as α is increased (tightened). The SM begins by learning the underlying ODE structure (low α, low j_{m}, high j_{d}). As α increases, j_{d} monotonically decreases with a counterpart increase in j_{m}, resulting in a forward concave set with an inflection point for j_{m} around $\log(j_{\mathrm{m}}) \approx -4$ (open markers) for both the SQ (black) and SQ+n (red) scenarios. In (A), the steepest improvement in the correlation between the ground-truth and learned model parameter is located in the same region of the concave tightening-Pareto inflection point α. When relaxing α with the same SMs trained in the tightening direction, the Pareto set becomes convex, as expected for a multiobjective minimization problem between two cost functions. Furthermore, it is possible to estimate the region where the relaxation-Pareto inflection point should lie from the gradient of the tightening-Pareto about the inflection point, as shown by dashed lines connecting ω^*'s in (B). A lower bound for the relaxation-Pareto inflection j_{m} can be inferred from the intersection between the curve defined by the tightening-Pareto inflection gradient and that defined by the final value of j_{d} in the tightening-Pareto set.

This tightening-relaxation strategy and the observed Pareto-set behavior allow the proper estimation of α values that provide an equilibrated trade-off between satisfying the MKM ODE and interpolating the observed data. Specifically in the case of SQ+n, further tightening α beyond the inflection point leads to little improvement (plateau in the red curve, B) in the inverse problem solution, instead further fitting the SM to data noise (overfitting). This can be inferred from the trend in the correlation between learned and ground truth model parameters over successive learning steps in Fig. 4 (A). Additionally, if α is further increased, a second inflection point is expected, where there is a detachment of the SM from the underlying MKM ODE and a steep reduction in j_{d} due to overfitting. In the case of SQ, the basis slack region in (B) portrays the inability of the specific choice of SM architecture (i.e. number of layers, activation function) to satisfy both the MKM ODE (physical model) and interpolate the ground-truth data. Fig. 4 also illustrates, in (C), that
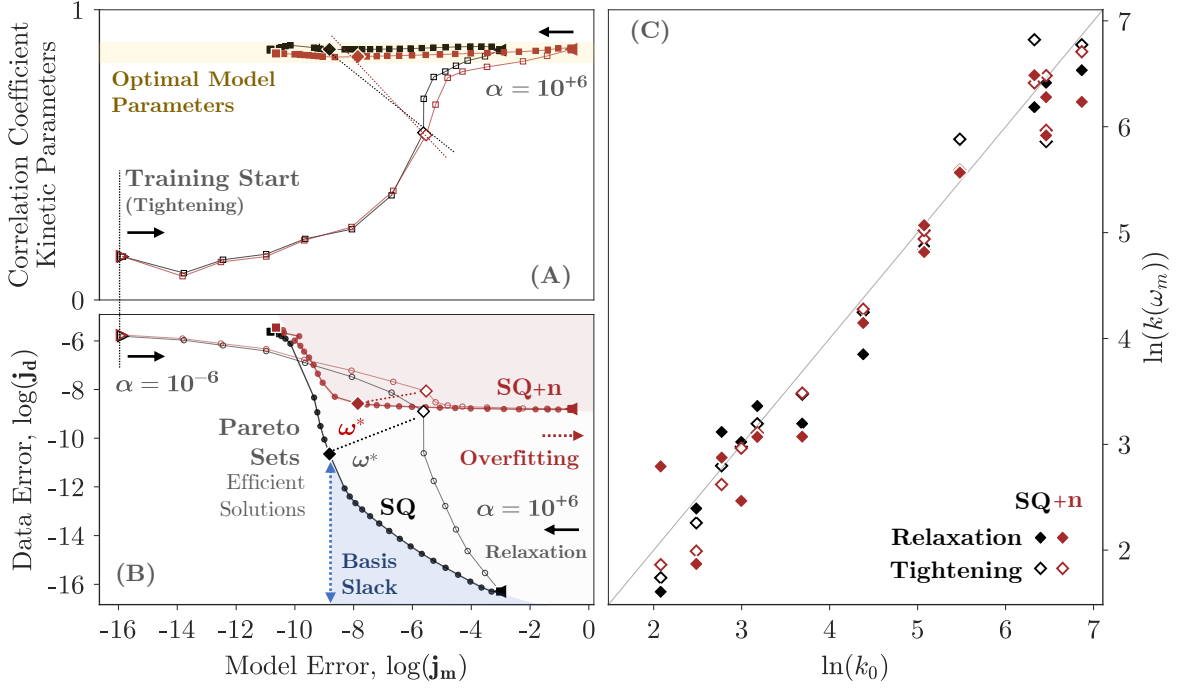
Figure 4: Inverse kINN - Regularization Sensitivity Analysis: Pareto Set Estimate.

the learned physical model parameters at either the tightening or relaxation inflection points are highly correlated with the ground-truth parameters irrespective of noise.

Instead of refining α for each inverse kINN, to accelerate the solution of the inverse problem, independently of scenario or reaction network type, we defined reaching the inflection in the tightening direction as a stopping criterion. To refine the solution, an additional training cycle was performed. The tightening steps started with $\alpha = 10^{-6}$, with 10-fold α increment between each training cycle. Table S1 in the Supplementary Material compiles the performance metrics for the observable variables states and their derivatives for the four different reaction network types, two BCs and three inverse scenarios. Exact match between observed states within noise limits and highly correlated derivatives are observed for all cases. In terms of latent variables, the performance metrics in Table 3 show the agreement between simulated and fitted latent states and their derivatives, irrespective of reaction network type, for scenarios SQ and SQ+n . As for the QB scenarios, even with lack of direct knowledge on surface coverages, the surface dynamics information can be indirectly retrieved for model da. Hence, the latent (bound) state profile can be inferred from unbound data only. With the increase in depth, the correlation between predicted states starts to fade, since species coverages are not directly measured in the QB type c and s, and there exist multiple solutions for the combination of their evolution over time and the kinetic parameters associated with their reactions. The temporal profiles for the final solutions for observed and latent states as well as their derivative (SM vs. physics model) are reported in the Supplementary Material, Figs. S2.1 to S2.6, S3.1 to S3.6 and S4.1 to S4.6.

To showcase the robustness of SM-based inverse kINNs, we consider the final refined solution profiles for the SQ+n scenario for the dcs-type reaction network with BC 1 in Fig. 5. In the SQ+n scenarios, not only are kinetic parameters learned, but also the scaling factor that connects intermediate species abundance as a signal or intensity to fractional coverages. For the optimized regularization, which is dependent on the α value, the underlying SM interpolation suppresses the effect of the added homoscedastic noise. Larger α values would lead to predicted points (interpolation, open markers) moving further towards the fed data (closed circle markers), resulting in SM overfit with respect to data. In contrast to the forward kINNs, BCs are not explicitly provided, so all observed data act as a composite set of BCs whose weight is conveyed by j_d. As depicted by Eq. (5.1), the efficient solutions are given by non-dominated points that minimize the multiobjective optimization function j_t, involving j_d and j_m. The latter, which represents the error in satisfying the MKM ODE, has similar structure as the one for forward kINNs except that ω_m becomes a variable to be solved.

In inverse kINNs, as the SM is trained to minimize Eq. (5.1), the kinetic parameters are learned from the underlying physical model (MKM). For all studied cases, kinetic parameters, ω_m, were randomly uniformly initialized at scale

Table 3: Inverse kINNs Results Summary - Latent Variables - Performance Metrics

Type	BC	Mode	Latent States Derivatives			Latent State (ground truth)		
			r^2	10^1 MAE	10^3 MSE	r^2	10^1 MAE	10^3 MSE
da	1	QB	1.00	9.59×10^{-2}	0.299	0.972	0.317	1.40
		SQ	0.944	1.31	3.25×10^2	0.999	1.09×10^{-2}	2.30×10^{-3}
		SQ+n	0.941	0.762	1.90×10^1	0.863	0.156	0.479
	2	QB	1.00	7.59×10^{-2}	0.169	0.998	0.348	1.69
		SQ	0.998	0.782	4.47×10^1	1.00	1.26×10^{-2}	2.23×10^{-3}
		SQ+n	1.00	0.447	3.37	0.948	0.179	0.415
dc	1	QB	1.00	0.191	0.592	0.965	1.18	2.32×10^1
		SQ	0.999	0.263	3.50	1.00	6.39×10^{-3}	6.21×10^{-4}
		SQ+n	0.983	0.596	5.93	0.895	0.267	1.29
	2	QB	0.996	0.104	0.154	0.940	1.51	3.68×10^1
		SQ	1.00	0.113	0.460	1.00	6.64×10^{-3}	6.96×10^{-4}
		SQ+n	0.988	0.408	3.70	0.988	0.324	2.05
dcs	1	QB	0.975	0.187	0.811	0.374	1.46	3.84×10^1
		SQ	1.00	5.53×10^{-2}	8.53×10^{-2}	1.00	2.66×10^{-2}	1.28×10^{-2}
		SQ+n	1.00	0.128	0.324	0.996	0.147	0.389
	2	QB	0.736	0.181	0.985	0.236	1.81	5.76×10^1
		SQ	1.00	9.30×10^{-2}	0.251	1.00	2.68×10^{-2}	1.30×10^{-2}
		SQ+n	0.998	0.211	1.44	0.989	0.106	0.145
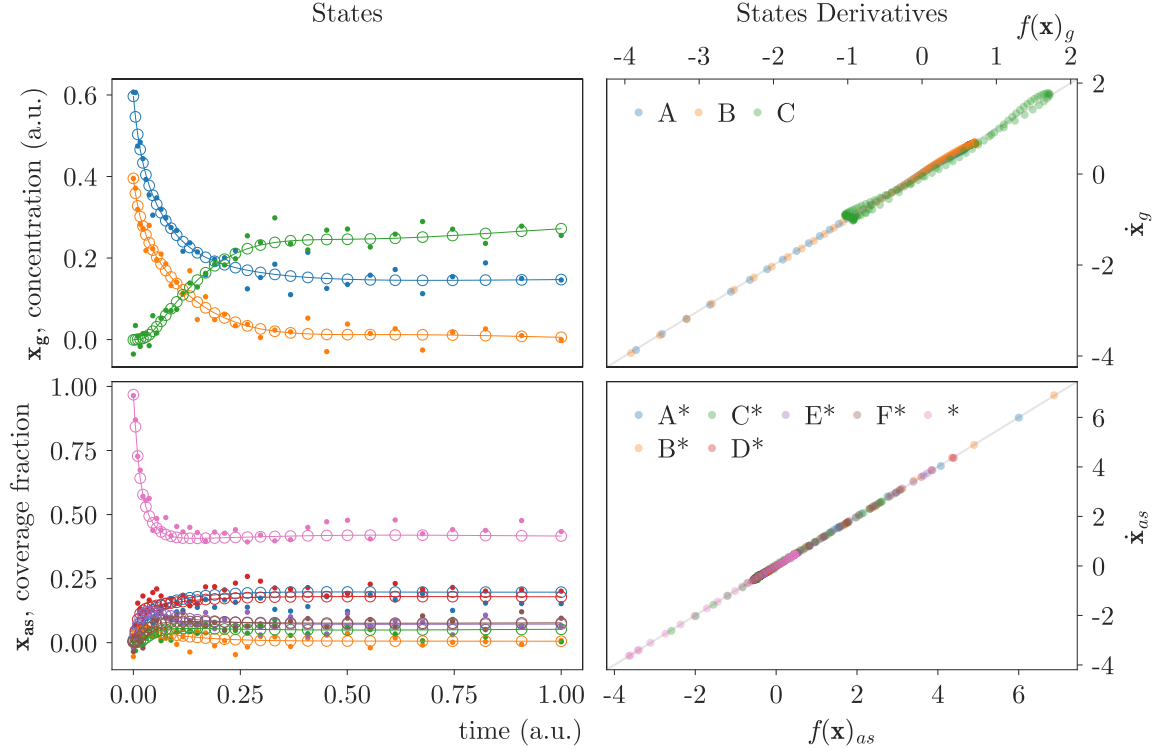
Figure 5: Reaction network type *dcs* inverse solution. States solution (left) (SM, open circles; noisy observed data, closed markers) under minimization of (5.1) (right) for observable variables (unbound species, top) and latent variables (coverages, bottom).

of 10^{-2}, whereas ground-truth values would range approximately from $0 \leq \ln(k_0) \leq 8$, as can be inferred from the natural logarithm of the values presented in Table 1. The final refined solution for the *SQ*+n scenario of *dcs*-type reaction network, with BC 1 and 2 solved (interpolated) in parallel is shown in Fig. 6. These results are illustrative of cases in which knowledge on surface coverages over time, despite the presence of noise, can provide a description of the system dynamics from which kinetic constants can be retrieved.
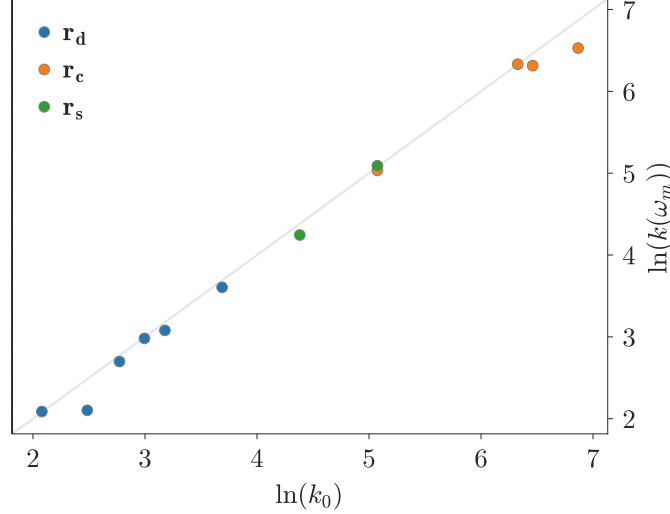
Figure 6: Reaction network type *dcs* inverse solution. Natural logarithms of ground truth parameter $\ln(k_0)$ vs. regressed parameters, $\ln(k(\boldsymbol{\omega_m})) = \boldsymbol{\omega_m}$.

A summary of final refined regression results for all inverse kINNs scenarios is given by Table 4. All scenarios comprise the simultaneous training of SM weights and the natural logarithm of the true kinetic model parameters for BCs 1 and 2. For the purely homogeneous *g* reaction networks, the kinetic parameters are easily retrieved with MAE below 10^{-3} for the *QB* scenario (in the absence of noise). The inverse kINNs for reaction networks of types *g*, *d* and *da* may also be solved, since they involve reactions between unbound species and corresponding bound molecules, but not reactions between intermediate species, as can be seen from the results for reaction network types *g* and *da*. As for reaction networks that include *c* and *g*-type reactions, the results support the need for surface composition information over time, as depicted by the detached correlation coefficients and high MAE for type *c* reactions, which may be acquired with operando techniques. In the availability of surface composition information (i.e. signal, intensity), forward and reverse kinetic parameters can be regressed even for the highest-depth reactions, those between surface intermediates (*s*-type).

Table 4: Inverse kINNs Results Summary - Regression Performance Metrics

Type	Mode	(10^3) g	d	MAE - $\log(k_0)$ (10^1) a	c	(10^1) s	total ρ
g	*QB*	0.990	-	-	-	-	1.00
	QB+n	2.29×10^1	-	-	-	-	1.00
da	*QB*	-	0.161	0.976	-	-	0.991
	SQ	-	5.95×10^{-2}	0.116	-	-	0.999
	SQ+n	-	0.150	0.811	-	-	0.999
dc	*QB*	-	0.520	-	1.80×10^1	-	-0.523
	SQ	-	1.70×10^{-2}	-	7.32×10^{-2}	-	1.00
	SQ+n	-	0.236	-	0.420	-	0.996
dcs	*QB*	-	5.06	-	1.02×10^1	1.90×10^1	0.205
	SQ	-	4.10×10^{-2}	-	0.116	0.661	0.999
	SQ+n	-	0.110	-	0.114	0.962	0.997

15

4 Discussion

Architecture Three-layer NNs have proven satisfactory to accommodate the different domains between the latent and observable variables in the inverse and forward problems in this study. Deeper NNs are particularly important when trigonometric BCs are enforced onto the output of the latent variables in inverse kINNs. Other optimized NN architectures cold have produced similar or better results; NN architecture optimization is beyond the scope of this work. For non-normalized non-stiff problems, a single *swish* hidden layer suffices to solve the MKM ODEs. For solutions that exhibit stiff behavior in certain regions and flat response in others, the characteristic time can be modelled as a function of the independent variable, which was implemented for all forward kINNs in this work in accordance with the mathematical structure proposed section 2.3.2.

Training Given the high number of weights to be optimized from the SM and underlying NNs, and the possibility of non-convexity especially in cases with large number of reaction intermediates (species of type *s*), we opt for the current standard algorithm for deep NN training, *Adam*, which combines a stochastic gradient descent optimizer with adaptive momentum. Although reliant on a noisy average estimation, SGD has been demonstrated to provide fast convergence and associated parameters, even compared to more elaborate optimizers [61]. The RMSProp built onto SGD in *Adam* is known to result in improved performance in on-line settings. Considering these properties and the documented success of *Adam* in NN settings [60], it was used as an optimizer in both forward and inverse settings.

Sampling & Regularization As regards the sampling procedure, other strategies could have been utilized instead of sampling evenly spaced, fixed points in the logarithm space. Cross-validation routines in inverse kINN schemes could prevent overfitting for α values beyond the tightening inflection point in Fig. 4. Furthermore, pre-training the SM with low α is paramount, especially in the case of noisy data, since it allows the SM to first learn the MKM ODE, which is further presented the observed data to which to interpolate. Such a procedure is also important when there are too few observed data points and large NNs, to prevent the physical model parameters from rapidly squashing at the beginning of training cycle and to avoid the issue of vanishing gradients. Proper scaling of the physical model parameters is also beneficial to accelerate the learning procedure in our current inverse kINNs setup, since it provides the learning (optimization) algorithm *Adam* with parameters of similar order of magnitude.

Inverse Problem In this work, we combine the optimization of the SM and kinetic model parameters into the NN training algorithm. We suggest that the solution for this combined approach may serve as initial guess to accelerate the solution of methods with mathematically rigorous formulations from the standpoint of optimization, such as the adjoint state method [62] that enforces Karush-Kuhn-Tucker, KKT [63, 64] optimality conditions. Such methods may vastly benefit from pre-optimized initial guesses, since their deterministic behavior, especially for the adjoint formulation, often leads to local convergence when deterministic gradient-based optimization algorithms are utilized.

MKMs & kINNs Analysis of the inverse solutions for the different reaction network types examples of varying complexity or depth (section 2.1) highlights the need for operando techniques to allow for information related reactions related to species of type *c* and *s* to be retrieved. The *SQ* and *SQ*+npose similar scenarios with respect to operando techniques. Coverage fractions time series obtained from the forward numerical solution of the initial value problems are rescaled with their multiplication by a random positive scalar, emulating deconvolved spectroscopic signals. Inverse kINNs therefore entail learning the kinetic parameters associated with the MKM as well as the scaling factor that maps signal intensities to coverages of intermediate species. The accurate SM-based parameter estimation for the complex *dcs* mechanism is a supporting evidence that inverse kINNs, seen as either physics-regularized interpolation of data or data-regularized SM-based solution of ODEs, are a robust method for the estimation of kinetic model parameters.

5 Conclusion

In this work, we use artificial neural networks (NNs) as basis functions for microkinetic models (MKMs), proposing strategies to create surrogate models (SMs) for the solution of the kinetic ODEs. We present a general classification and notation for catalytic reaction networks based on reaction and species types and the phases of species involved. This notation provides an immediate understanding of the extent of dissociation between the observable or measurable states and the underlying intermediates, and hence the complexity of attempting mechanism elucidation. The suitability of NNs as basis functions for the solution of ordinary differential equations is demonstrated by their ability to solve kinetic forward problems. Structural BCs are imposed on the NN to enforce ODE BCs as in initial value problems. Furthermore, we also propose the inclusion of ancillary NNs to learn the characteristic timescale of the ODEs, which is important in the case of stiffness about the BC. Normalization of the coverage fractions of intermediates is enforced by separating bulk and surface species into two independent neural networks.

Using a lumped approach for the simultaneous training of NNs weights and kinetic model parameters in a multiobjective optimization framework, we demonstrate the ability to resort to NNs as basis functions and NNs training algorithms to

retrieve kinetic parameters from data generated for anecdotal MKM ODEs. The optimization problem is framed in terms of a regularization parameter that can be used to guide the training cycles by (1) low regularization, allowing the SMs to learn or adapt to the physical model ODE, slowly transitioning to (2) high regularization to force the SM to interpolate observed data while parameters of the physical model are simultaneously learnt. This framework can be utilized to estimate Pareto sets from which training maturity can be inferred and thus propose stopping criteria for training cycles.

Our SM-based approach for the solution of inverse problem was tested against different types of reaction networks. We show that for homogeneous reactions, when the concentration of species over time are known, or for reactions involving non-dissociated adsorbed chemical species, the approach can be utilized to readily retrieve parameters associated with the underlying kinetic model. For more complex scenarios, where there is incomplete knowledge about concentration of intermediate species on the surface of a catalyst, we assessed the possibility of using deconvolved operando spectroscopic signals for the adsorbed species to extend regression capabilities. Scaling or calibration factors are included as variables in the training process, and we prove that our approach can retrieve such factors as well as the kinetic model parameters for these complex systems. Although our analyses are limited to cases where data can be represented as derived from concentrated-parameter models, we suggest that SM-based formulation of inverse kinetic ODEs can be applied to transient techniques for the acquisition of kinetic parameters from experimental data, such as temporal analysis of products (TAP) or step-response experiments.

Acknowledgements

Conflict of Interest

The authors declare the nonexistence of conflicts of interest.

References

[1] Yong-ha Kim, Lydia K. Park, Sotira Yiacoumi, and Costas Tsouris. Modular Chemical Process Intensification: A Review. *Annual Review of Chemical and Biomolecular Engineering*, 8(1):359–380, 6 2017.

[2] G Jones, T Bligaard, F Abild-Pedersen, and J K Nørskov. Using scaling relations to understand trends in the catalytic activity of transition metals. *Journal of Physics: Condensed Matter*, 20(6):064239, 2 2008.

[3] Zachary W. Ulissi, Andrew J. Medford, Thomas Bligaard, and Jens K. Nørskov. To address surface reaction network complexity using scaling relations machine learning and DFT calculations. *Nature Communications*, 8(1):14621, 4 2017.

[4] Kevin McBride and Kai Sundmacher. Overview of Surrogate Modeling in Chemical Process Engineering. *Chemie Ingenieur Technik*, 91(3):228–239, 3 2019.

[5] Carsten Stegelmann, Anders Andreasen, and Charles T. Campbell. Degree of Rate Control: How Much the Energies of Intermediates and Transition States Control Rates. *Journal of the American Chemical Society*, 131(23):8077–8082, 6 2009.

[6] J. A. Dumesic. The Microkinetics of heterogeneous catalysis. *ACS professional reference book*, 1993.

[7] Andrew J Medford, Chuan Shi, Max J Hoffmann, Adam C Lausche, Sean R Fitzgibbon, Thomas Bligaard, and Jens K Nørskov. CatMAP: A Software Package for Descriptor-Based Microkinetic Mapping of Catalytic Trends. *Catalysis Letters*, 145(3):794–807, 3 2015.

[8] Henrik Topsøe. Developments in operando studies and in situ characterization of heterogeneous catalysts. *Journal of Catalysis*, 216(1-2):155–164, 5 2003.

[9] Andrew J. Medford, M. Ross Kunz, Sarah M. Ewing, Tammie Borders, and Rebecca Fushimi. Extracting Knowledge from Data through Catalysis Informatics. *ACS Catalysis*, 8(8):7403–7429, 8 2018.

[10] Thomas Bligaard, R. Morris Bullock, Charles T. Campbell, Jingguang G. Chen, Bruce C. Gates, Raymond J. Gorte, Christopher W. Jones, William D. Jones, John R. Kitchin, and Susannah L. Scott. Toward Benchmarking in Catalysis Science: Best Practices, Challenges, and Opportunities. *ACS Catalysis*, 6(4):2590–2602, 4 2016.

[11] Takashi Toyao, Zen Maeno, Satoru Takakusagi, Takashi Kamachi, Ichigaku Takigawa, and Ken-ichi Ichi Shimizu. Machine Learning for Catalysis Informatics: Recent Applications and Prospects. *ACS Catalysis*, 10(3):2260–2297, 2 2020.

[12] James A. Dumesic, George W. Huber, and Michel Boudart. Principles of Heterogeneous Catalysis. In *Handbook of Heterogeneous Catalysis*, pages 29–55. Wiley-VCH Verlag GmbH & Co. KGaA, Weinheim, Germany, 3 2008.

[13] Abhijit Chatterjee and Dionisios G. Vlachos. An overview of spatial microscopic and accelerated kinetic Monte Carlo methods. *Journal of Computer-Aided Materials Design*, 14(2):253–308, 3 2007.

[14] Max J. Hoffmann and Thomas Bligaard. A Lattice Kinetic Monte Carlo Solver for First-Principles Microkinetic Trend Studies. *Journal of Chemical Theory and Computation*, 14(3):1583–1593, 3 2018.

[15] Mie Andersen, Craig P. Plaisance, and Karsten Reuter. Assessment of mean-field microkinetic models for CO methanation on stepped metal surfaces using accelerated kinetic Monte Carlo. *Journal of Chemical Physics*, 147(15):152705, 10 2017.

[16] Karsten Reuter. Ab Initio Thermodynamics and First-Principles Microkinetics for Surface Catalysis. *Catalysis Letters*, 146(3):541–563, 3 2016.

[17] Yuekang Jin, Guanghui Sun, Zhengming Wang, Haibin Pan, Lingshun Xu, Hong Xu, and Weixin Huang. Elementary surface reactions on Co(0001) under Fischer-Tropsch synthesis conditions. *Journal of Physical Chemistry C*, 121(39):21535–21540, 2017.

[18] Pei Pei Chen, Jin Xun Liu, and Wei Xue Li. Carbon Monoxide Activation on Cobalt Carbide for Fischer-Tropsch Synthesis from First-Principles Theory. *ACS Catalysis*, pages 8093–8103, 2019.

[19] Nuoya Yang, Andrew J. Medford, Xinyan Liu, Felix Studt, Thomas Bligaard, Stacey F. Bent, and Jens K. Nørskov. Intrinsic Selectivity and Structure Sensitivity of Rhodium Catalysts for C2+ Oxygenate Production. *Journal of the American Chemical Society*, 138(11):3705–3714, 3 2016.

[20] A. B. Mhadeshwar, J. R. Kitchin, M. A. Barteau, and D. G. Vlachos. The role of adsorbate-adsorbate interactions in the rate controlling step and the most abundant reaction intermediate of NH3 decomposition on Ru. *Catalysis Letters*, 96(1-2):13–22, 7 2004.

[21] Lars C. Grabow, Britt Hvolbæk, and Jens K. Nørskov. Understanding Trends in Catalytic Activity: The Effect of Adsorbate–Adsorbate Interactions for CO Oxidation Over Transition Metals. *Topics in Catalysis*, 53(5-6):298–310, 5 2010.

[22] Javier Pérez-Ramírez, Rob J. Berger, Guido Mul, Freek Kapteijn, and Jacob A. Moulijn. Six-flow reactor technology a review on fast catalyst screening and kinetic studies. *Catalysis Today*, 60(1):93–109, 7 2000.

[23] J. T. Gleaves, J. R. Ebner, and T. C. Kuechler. Temporal Analysis of Products (TAP) — A Unique Catalyst Evaluation System with Submillisecond Time Resolution. *Catalysis Reviews*, 30(1):49–116, 2 1988.

[24] John T. Gleaves, Gregory Yablonsky, Xiaolin Zheng, Rebecca Fushimi, and Patrick L. Mills. Temporal analysis of products (TAP)-Recent advances in technology for kinetic analysis of multi-component catalysts. *Journal of Molecular Catalysis A: Chemical*, 315(2):108–134, 1 2010.

[25] K. Morgan, N. Maguire, R. Fushimi, J. T. Gleaves, A. Goguet, M. P. Harold, E. V. Kondratenko, U. Menon, Y. Schuurman, and G. S. Yablonsky. Forty years of temporal analysis of products, 2017.

[26] J. Happel. Transient tracing. *Chemical Engineering Science*, 33(11):1567, 1978.

[27] Cristian Ledesma, Jia Yang, De Chen, and Anders Holmen. Recent approaches in mechanistic and kinetic studies of catalytic reactions using SSITKA technique, 12 2014.

[28] Schohn L. Shannon and James G. Goodwin. Characterization of Catalytic Surfaces by Isotopic-Transient Kinetics during Steady-State Reaction. *Chemical Reviews*, 95(3):677–695, 1995.

[29] Rob J. Berger, Freek Kapteijn, Jacob A. Moulijn, Guy B. Marin, Juray De Wilde, Maria Olea, De Chen, Anders Holmen, Luca Lietti, Enrico Tronconi, and Yves Schuurman. Dynamic methods for catalytic kinetics. *Applied Catalysis A: General*, 342(1-2):3–28, 6 2008.

[30] Preeti Aghalayam, Young K. Park, and Dionisios G. Vlachos. Construction and optimization of complex surface-reaction mechanisms. *AIChE Journal*, 46(10):2017–2029, 2000.

[31] P. Aghalayam, Y. K. Park, and D. G. Vlachos. A detailed surface reaction mechanism for CO oxidation on Pt. *Proceedings of the Combustion Institute*, 28(1):1331–1339, 1 2000.

[32] Patricia Rubert-Nason, Manos Mavrikakis, Christos T. Maravelias, Lars C. Grabow, and Lorenz T. Biegler. Advanced solution methods for microkinetic models of catalytic reactions: A methanol synthesis case study. *AIChE Journal*, 60(4):1336–1346, 4 2014.

[33] J. M. Caruthers, J. A. Lauterbach, K. T. Thomson, V. Venkatasubramanian, C. M. Snively, A. Bhan, S. Katare, and G. Oskarsdottir. Catalyst design: Knowledge extraction from high-throughput experimentation. *Journal of Catalysis*, 216(1-2):98–109, 5 2003.

[34] Jonas Sjöblom and Derek Creaser. New approach for microkinetic mean-field modelling using latent variables. *Computers & Chemical Engineering*, 31(4):307–317, 2 2007.

[35] A. B. Mhadeshwar, H. Wang, and D. G. Vlachos. Thermodynamic Consistency in Microkinetic Development of Surface Reaction Mechanisms. *Journal of Physical Chemistry B*, 107(46):12721–12733, 11 2003.

[36] Adam Yonge, M. Ross Kunz, Rakesh Batchu, Zongtang Fang, Tobin Issac, Rebacca Fushimi, and Andrew J. Medford. Tapsolver: A Python package for the simulation and analysis of TAP reactor experiments. *arXiv*, 8 2020.

[37] M. Ross Kunz, Adam Yonge, Zongtang Fang, Andrew J. Medford, Denis Constales, Gregory Yablonsky, and Rebecca Fushimi. Data Driven Reaction Mechanism Estimation via Transient Kinetics and Machine Learning. *arXiv*, 11 2020.

[38] Alexander I.J. Forrester and Andy J. Keane. Recent advances in surrogate-based optimization. *Progress in Aerospace Sciences*, 45(1-3):50–79, 1 2009.

[39] Sushant Suhas Garud, I.A. Karimi, and Markus Kraft. Smart Sampling Algorithm for Surrogate Model Development. *Computers & Chemical Engineering*, 96:103–114, 1 2017.

[40] I. E. Lagaris, A. Likas, and D. I. Fotiadis. Artificial Neural Networks for Solving Ordinary and Partial Differential Equations. *IEEE Transactions on Neural Networks*, 9(5):987–1000, 5 1997.

[41] I.E. Lagaris, Aristidis Likas, and D.I. Fotiadis. Artificial neural networks for solving ordinary and partial differential equations. *IEEE Transactions on Neural Networks*, 9(5):987–1000, 5 1998.

[42] Yiping Lu, Aoxiao Zhong, Quanzheng Li, and Bin Dong. Beyond Finite Layer Neural Networks: Bridging Deep Architectures and Numerical Differential Equations. *35th International Conference on Machine Learning, ICML 2018*, 7:5181–5190, 10 2017.

[43] Eldad Haber and Lars Ruthotto. Stable Architectures for Deep Neural Networks. *Inverse Problems*, 34(1), 5 2017.

[44] Lars Ruthotto and Eldad Haber. Deep Neural Networks Motivated by Partial Differential Equations. *Journal of Mathematical Imaging and Vision*, 62(3):352–364, 4 2018.

[45] Ricky T. Q. Chen, Yulia Rubanova, Jesse Bettencourt, and David Duvenaud. Neural Ordinary Differential Equations. *NIPs*, 109(NeurIPS):31–60, 6 2018.

[46] Benny Avelin and Kaj Nyström. Neural ODEs as the deep limit of ResNets with constant weights. *Analysis and Applications*, 2020.

[47] Per Stoltze. Microkinetic simulation of catalytic reactions. *Progress in Surface Science*, 65(3-4):65–150, 10 2000.

[48] Gabriel S. Gusmão and Phillip Christopher. A general and robust approach for defining and solving microkinetic catalytic systems. *AIChE Journal*, 61(1):188–199, 1 2015.

[49] G. Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals, and Systems*, 2(4):303–314, 12 1989.

[50] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Universal approximation of an unknown mapping and its derivatives using multilayer feedforward networks. *Neural Networks*, 3(5):551–560, 1 1990.

[51] Kurt Hornik. Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4(2):251–257, 1991.

[52] Moshe Leshno, Vladimir Ya Lin, Allan Pinkus, and Shimon Schocken. Multilayer feedforward networks with a nonpolynomial activation function can approximate any function. *Neural Networks*, 6(6):861–867, 1 1993.

[53] A.J. Meade and A.A. Fernandez. The numerical solution of linear ordinary differential equations by feedforward neural networks. *Mathematical and Computer Modelling*, 19(12):1–25, 6 1994.

[54] A.J. Meade and A.A. Fernandez. Solution of nonlinear ordinary differential equations by feedforward neural networks. *Mathematical and Computer Modelling*, 20(9):19–44, 11 1994.

[55] Roy Frostig, Matthew Johnson, and Chris Leary. Compiling machine learning programs via high-level tracing. In *SysML*, 3 2018.

[56] James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, and Skye Wanderman-Milne. {JAX}: composable transformations of {P}ython+{N}um{P}y programs. *http://github.com/google/jax*, 2018.

[57] M. Raissi, P. Perdikaris, and G. E. Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019.

[58] Alan C. Hindmarsh. LSODE and LSODI, two new initial value ordinary differential equation solvers. *ACM SIGNUM Newsletter*, 15(4):10–11, 12 1980.

[59] Linda Petzold. Automatic Selection of Methods for Solving Stiff and Nonstiff Systems of Ordinary Differential Equations. *SIAM Journal on Scientific and Statistical Computing*, 4(1):136–148, 3 1983.

[60] Diederik P. Kingma and Jimmy Lei Ba. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*. International Conference on Learning Representations, ICLR, 12 2015.

[61] Yann Lecun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 5 2015.

[62] Ibrahim Ayed, Emmanuel De Bézenac, Arthur Pajot, Julien Brajard, and Patrick Gallinari. Learning Dynamical Systems from Partial Observations, 2019.

[63] W Karush. *Minima of Functions of Several Variables with Inequalities as Side Conditions*. University of Chicago, Department of Mathematics, 1939.

[64] H W Kuhn and A W Tucker. Nonlinear Programming. In *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability*, pages 481–492, Berkeley, Calif., 1951. University of California Press.

KINETICS-INFORMED NEURAL NETWORKS

SUPPLEMENTARY MATERIAL

Gabriel S. Gusmão, Adhika P. Retnanto[1], Shashwati C. da Cunha[1], Andrew J. Medford
School of Chemical & Biomolecular Engineering
Georgia Institute of Technology
Atlanta, GA 30332
{gusmaogabriels, aretnanto6, shashwatidc, ajm}@gatech.edu

[1]These authors contributed equally to this work.

December 1, 2020

MKM ODEs Solutions

This section summarizes the performance metrics for inverse problem solutions of each type of reaction network (*g*, *da*, *dc*, *dcs*) discussed in Results. For each example, results are then shown for quantitative bound species (*QB*), semiquantitative bound species (*SQ*), and semiquantitative bound species with white noise (*SQ+n*) under two boundary conditions (BC1, BC2). For these problems, we list the parameters used for data generation, stoichiometry matrices (M), ground truth versus NN predictions for concentration and coverage profiles, and regressed versus true kinetic parameters. The shape of the stoichiometry matrices reflect the elementary reactions of the system such that the rows are associated with the mass balance of each species, and the columns are associated with reaction rates, therefore having the same size as kinetic parameters k.

Table S1: Inverse kINNs Results Summary - Observable Variables - Performance Metrics

Type BC	Mode		Observed States Derivatives				Observed States	
			r^2	10^1 MAE	10^2 MSE	r^2	10^4 MAE	10^8 MSE
g	1	*QB*	1.00	1.24×10^{-2}	4.36×10^{-4}	1.00	0.235	7.13×10^{-2}
		SQ	1.00	1.24×10^{-2}	4.36×10^{-4}	1.00	0.235	7.13×10^{-2}
		SQ+n	0.998	0.262	0.112	0.999	4.01×10^1	2.77×10^3
	2	*QB*	1.00	1.35×10^{-3}	4.13×10^{-6}	1.00	2.95×10^{-2}	1.14×10^{-3}
		SQ	1.00	1.35×10^{-3}	4.13×10^{-6}	1.00	2.95×10^{-2}	1.14×10^{-3}
		SQ+n	0.133	0.411	0.330	0.398	4.36×10^1	3.24×10^3
da	1	*QB*	0.960	0.151	3.76×10^{-2}	1.00	7.72×10^{-2}	1.42×10^{-2}
		SQ	0.889	0.618	3.74	1.00	1.91	6.81
		SQ+n	0.820	0.736	1.09	0.927	5.35×10^1	7.89×10^3
	2	*QB*	1.00	0.198	0.151	1.00	7.78×10^{-2}	1.62×10^{-2}
		SQ	0.998	0.529	1.42	1.00	0.537	0.587
		SQ+n	0.988	0.551	0.524	0.983	4.33×10^1	3.86×10^3
dc	1	*QB*	0.987	0.492	0.365	1.00	4.66×10^{-2}	6.69×10^{-3}
		SQ	1.00	0.104	7.12×10^{-2}	1.00	0.235	9.80×10^{-2}
		SQ+n	0.998	0.534	0.530	0.998	4.25×10^1	3.89×10^3
	2	*QB*	0.999	0.275	0.139	1.00	2.70×10^{-2}	1.93×10^{-3}
		SQ	1.00	7.08×10^{-2}	9.40×10^{-3}	1.00	4.45×10^{-2}	3.32×10^{-3}
		SQ+n	0.996	0.652	0.605	0.947	4.47×10^1	3.10×10^3
dcs	1	*QB*	0.997	0.730	1.08	1.00	0.688	1.40
		SQ	1.00	6.78×10^{-2}	1.12×10^{-2}	1.00	0.304	0.412
		SQ+n	0.997	0.396	0.229	0.999	3.16×10^1	2.01×10^3
	2	*QB*	0.995	0.768	1.03	1.00	0.877	3.16
		SQ	1.00	4.81×10^{-2}	4.08×10^{-3}	1.00	0.475	0.651
		SQ+n	0.999	0.364	0.208	0.996	3.47×10^1	2.10×10^3

Reaction Network Examples

1   Reaction Network Type *g*

Reaction Network Type *g* conveys the simplest homogeneous case for the reversible reaction of species A and B into C in gas phase or in solution. To simplify the problem solution, the rate constants $\mathbf{k}$ are suppressed to its bulk states and can be represented in terms of **g**.1, such as: dissociation and dimerization reaction, and lumped-kinetics surrogates derived from complex kinetics data.

$$A + B \underset{k_2}{\overset{k_1}{\rightleftharpoons}} C \tag{g.1}$$

$$\ln(\mathbf{k}_0) = [2.30 \; 0.00]$$

$$M = \begin{bmatrix} -1 & 1 \\ -1 & 1 \\ 1 & -1 \end{bmatrix} \tag{S1.1}$$

$$\mathbf{x}^T = [x_A \; x_B \; x_C]$$
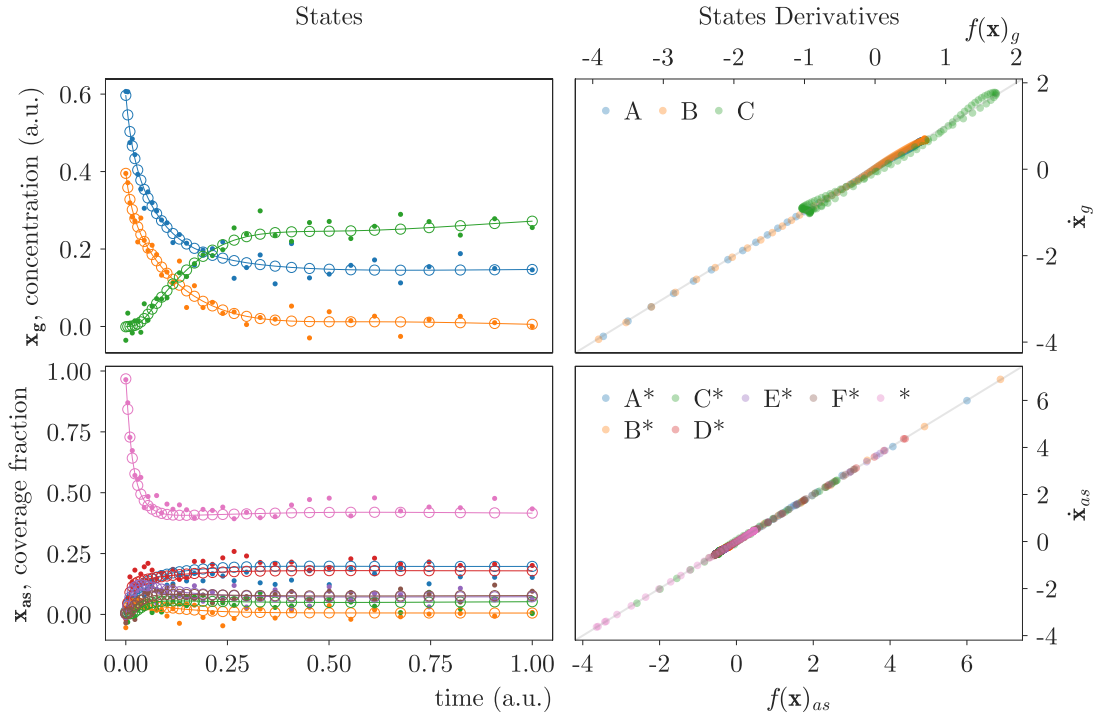
Figure S1.1: Reaction network type *g* inverse solution for QB data using BC1. States solution (left) (SM, open circles; observed data, closed markers) under minimization of (5.1) (right) for observable variables (unbound species, top).
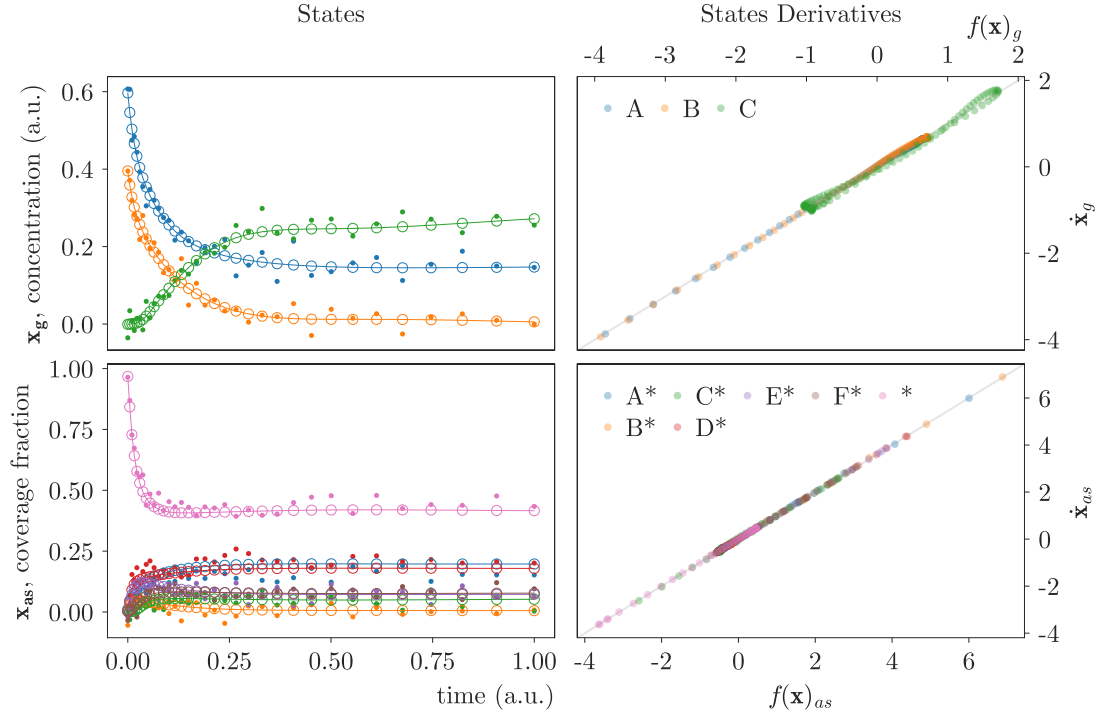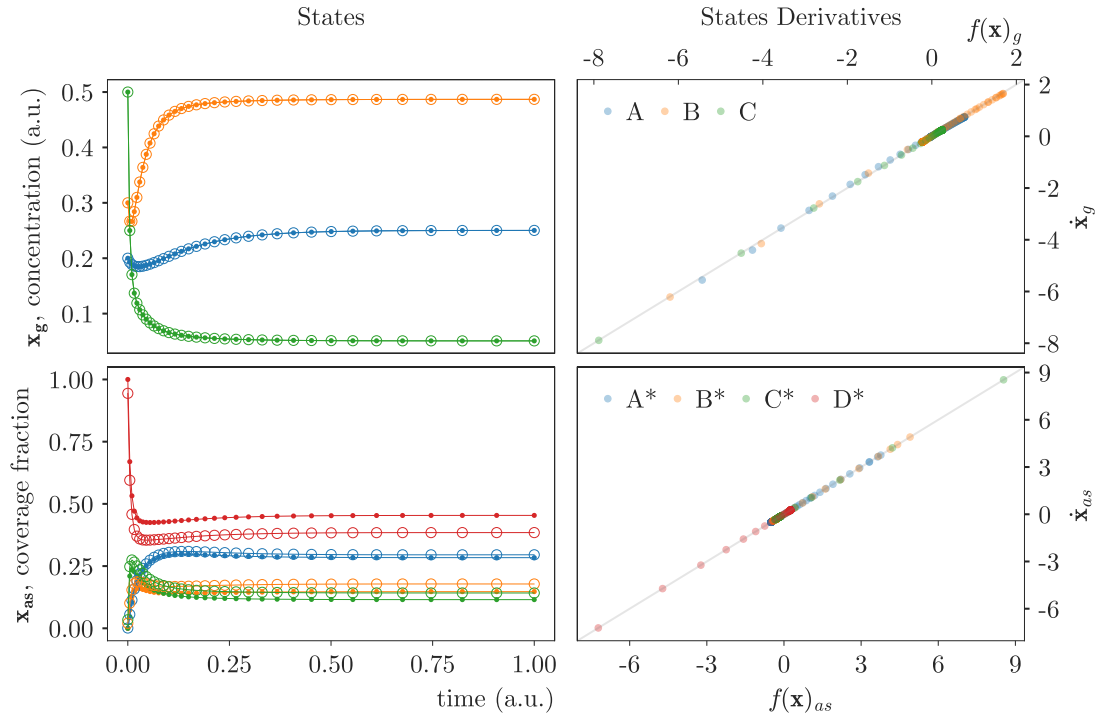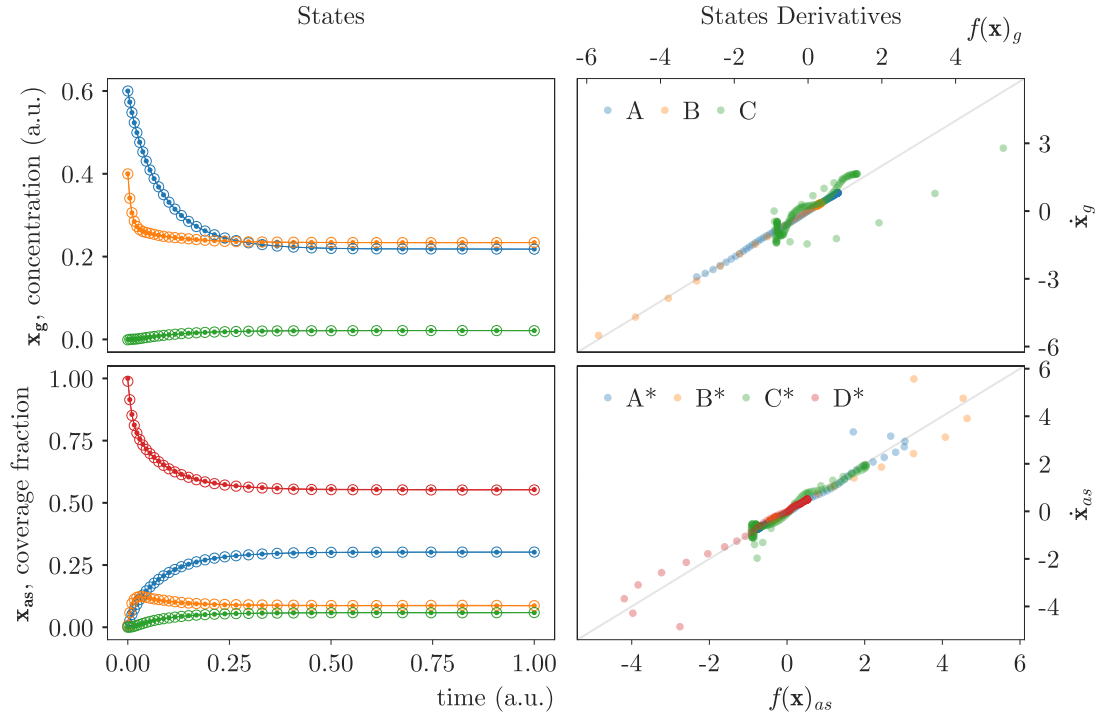
Figure S1.2: Reaction network type *g* inverse solution for QB data using BC2. States solution (left) (SM, open circles; observed data, closed markers) under minimization of (5.1) (right) for observable variables.
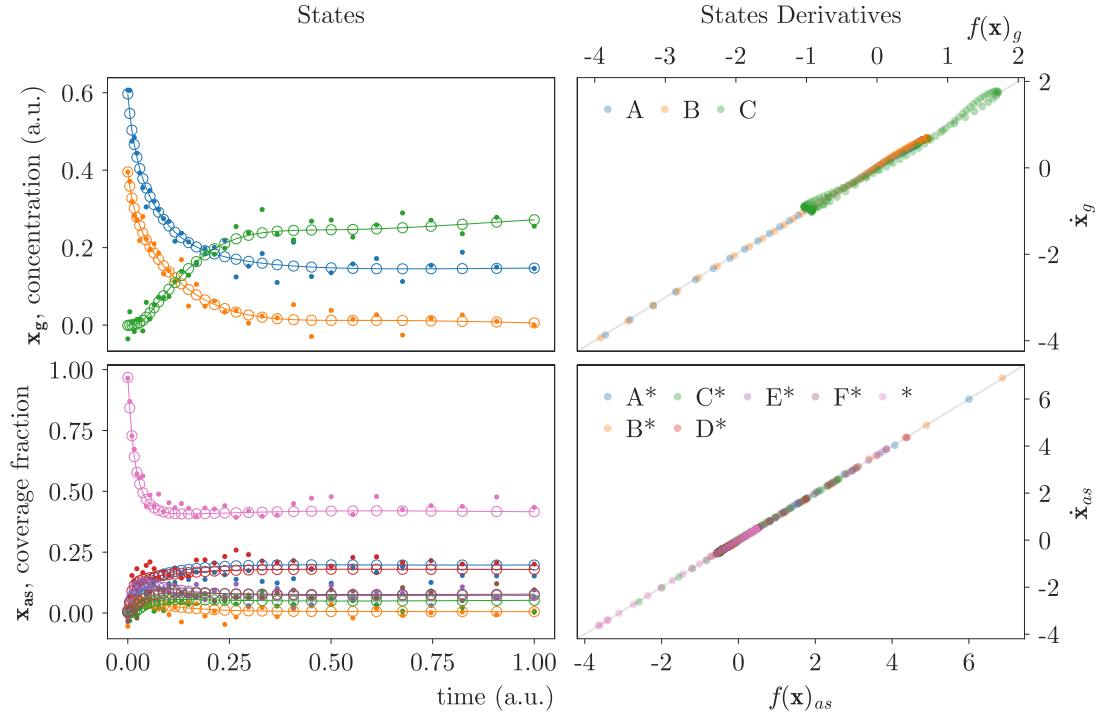
Figure S1.3: Reaction network type *g* inverse solution for data with noise using BC1. States solution (left) (SM, open circles; QB+n data, closed markers) under minimization of (5.1) (right) for observable variables (unbound species, top).
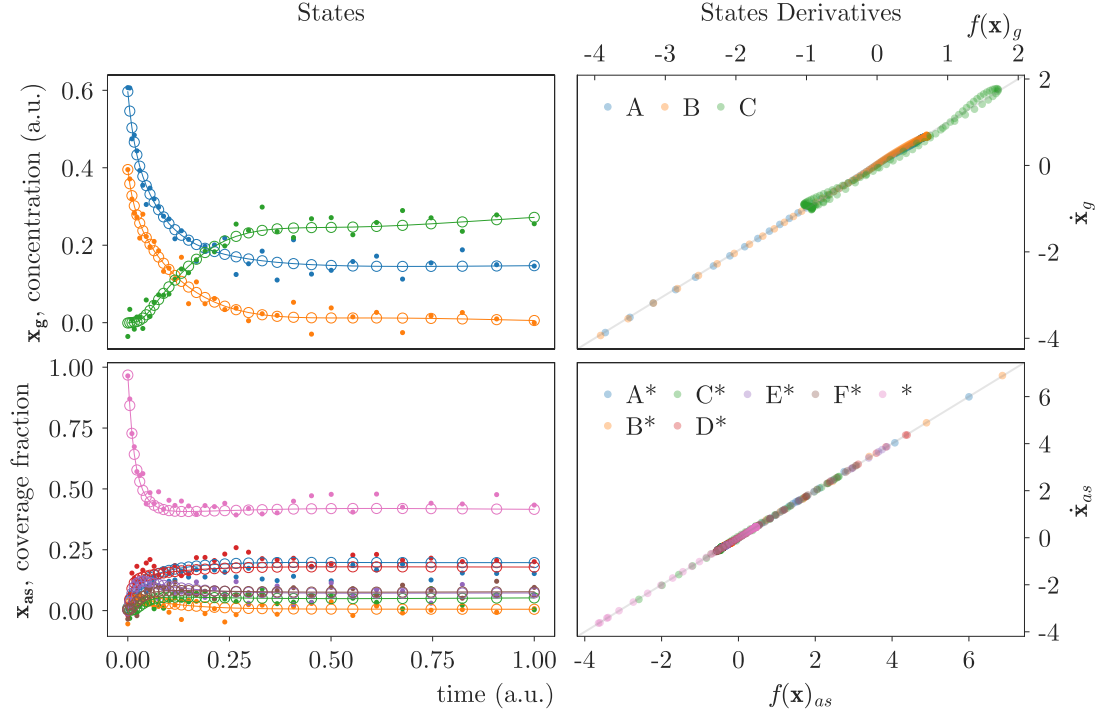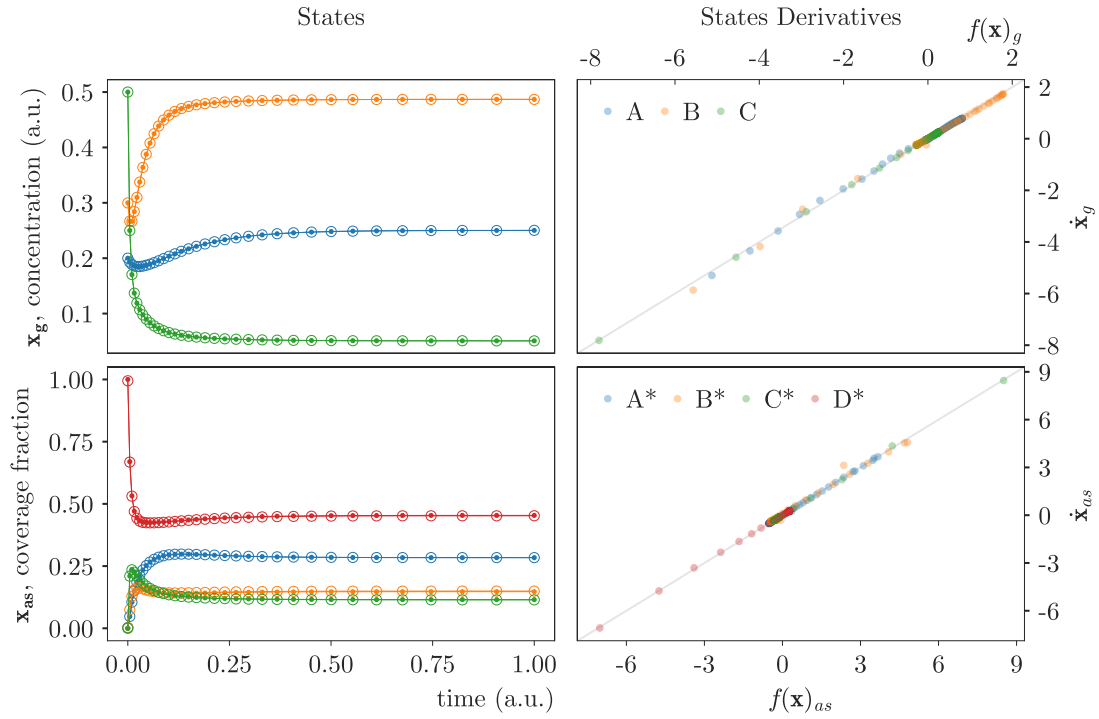
Figure S1.4: Reaction network type *g* inverse solution for data with noise using BC2. States solution (left) (SM, open circles; QB+n data, closed markers) under minimization of (5.1) (right) for observable variables.
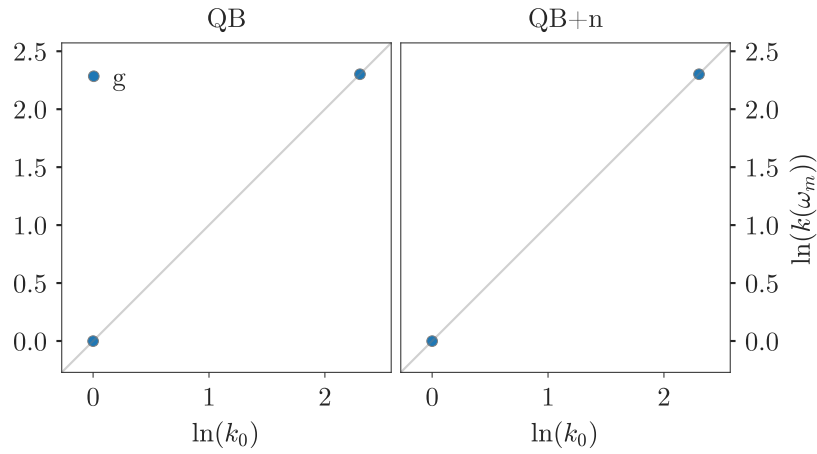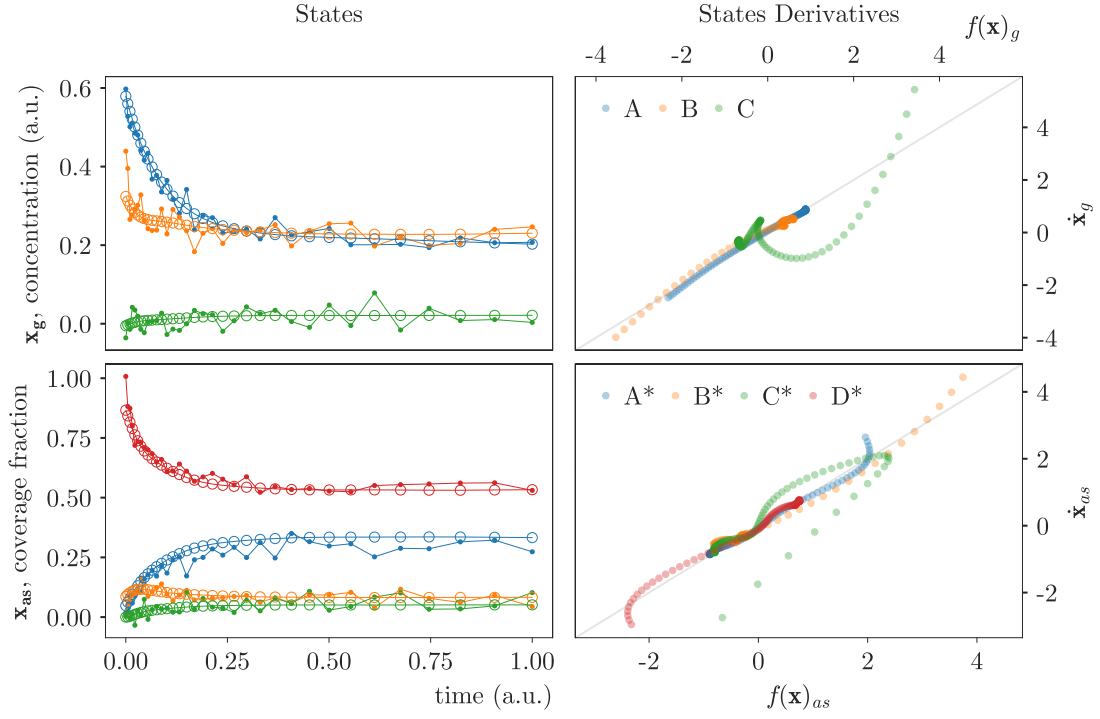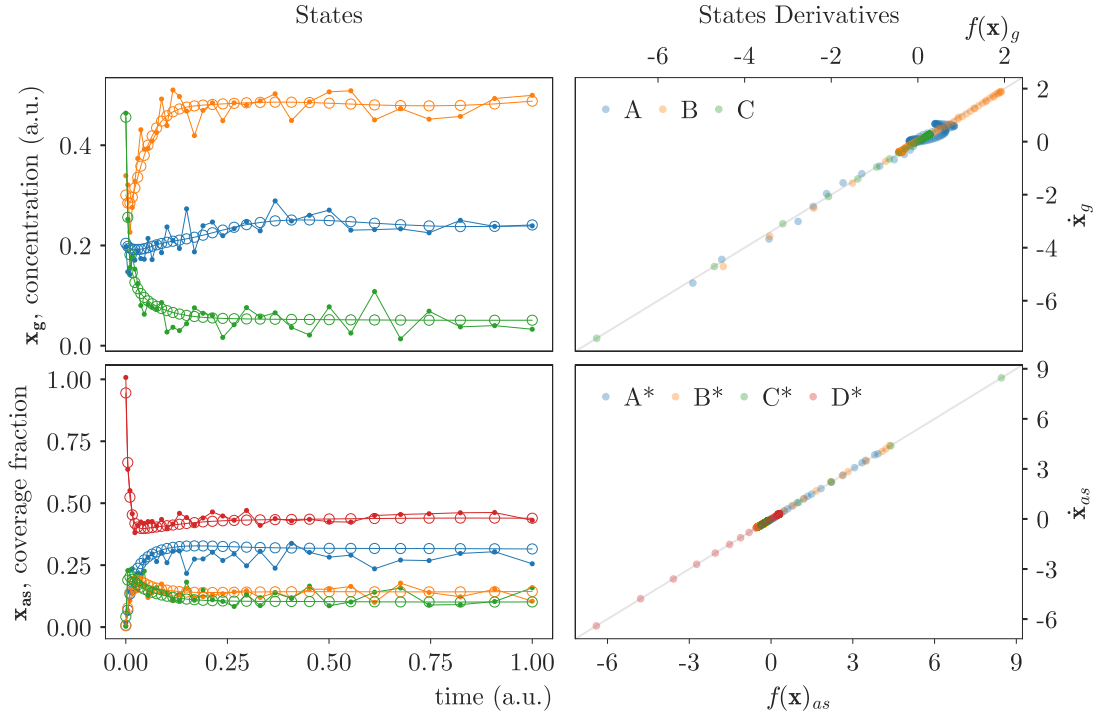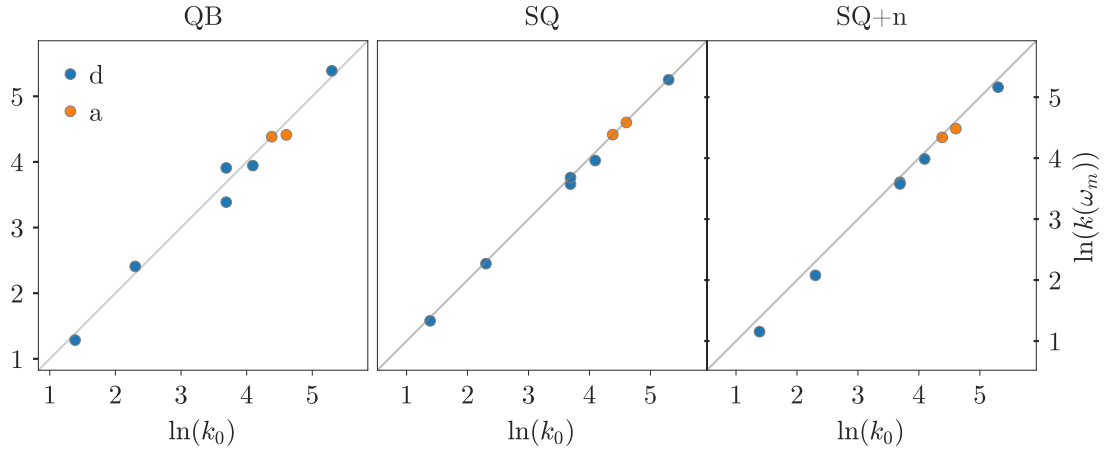

Figure S1.5: Reaction network type *g* inverse solution. Natural logarithms of ground truth parameter $\ln(k_0)$ vs. regressed parameters, $\ln(k(\boldsymbol{\omega}_{\mathbf{m}})) = \boldsymbol{\omega}_{\mathbf{m}}$, with and without noise.

2   Reaction Network Type *da*

Here we transpose the reactive step from the homogeneous phase in **g**.1 to the heterogeneous phase by including the adsorption and desorption elementary steps, **d**.(1-3). and a surface reaction between adsorbed molecules, **a**.1.

$$A + * \underset{k_4}{\overset{k_3}{\rightleftharpoons}} A* \tag{d.1}$$

$$B + * \underset{k_6}{\overset{k_5}{\rightleftharpoons}} B* \tag{d.2}$$

$$C + * \underset{k_8}{\overset{k_7}{\rightleftharpoons}} C* \tag{d.3}$$

$$A* + B* \underset{k_{10}}{\overset{k_9}{\rightleftharpoons}} C* \tag{a.1}$$

$$\ln(\mathbf{k}_0) = [2.30 \ 1.39 \ 3.69 \ 4.09 \ 5.30 \ 3.69 \ 4.61 \ 4.38]$$

$$M = \begin{bmatrix} -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 & -1 & 1 \\ 0 & 0 & 1 & -1 & 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 0 & 1 & -1 & 1 & -1 \\ -1 & 1 & -1 & 1 & -1 & 1 & 1 & -1 \end{bmatrix} \tag{S2.1}$$

$$\mathbf{x}^T = [x_A \ x_B \ x_C \ x_{A*} \ x_{B*} \ x_{C*} \ x_*]$$
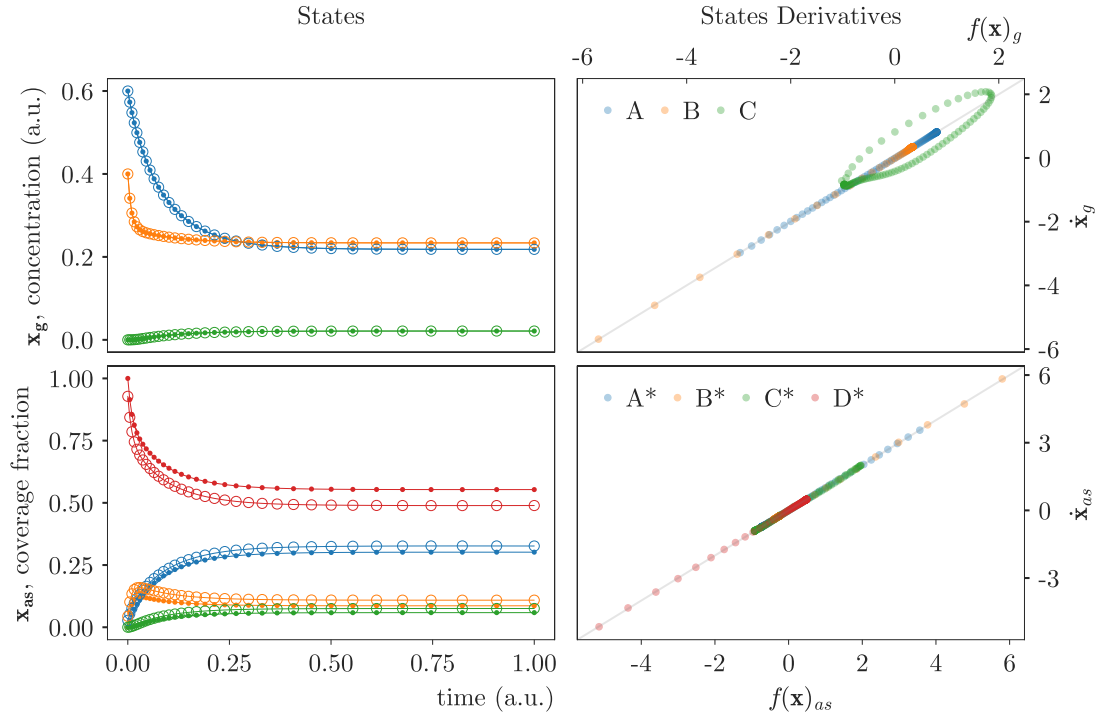
Figure S2.1: Reaction network type *da* inverse solution for QB data using BC1. States solution (left) (SM, open circles; observed data, closed markers) under minimization of (5.1) (right) for observable variables (unbound species, top).
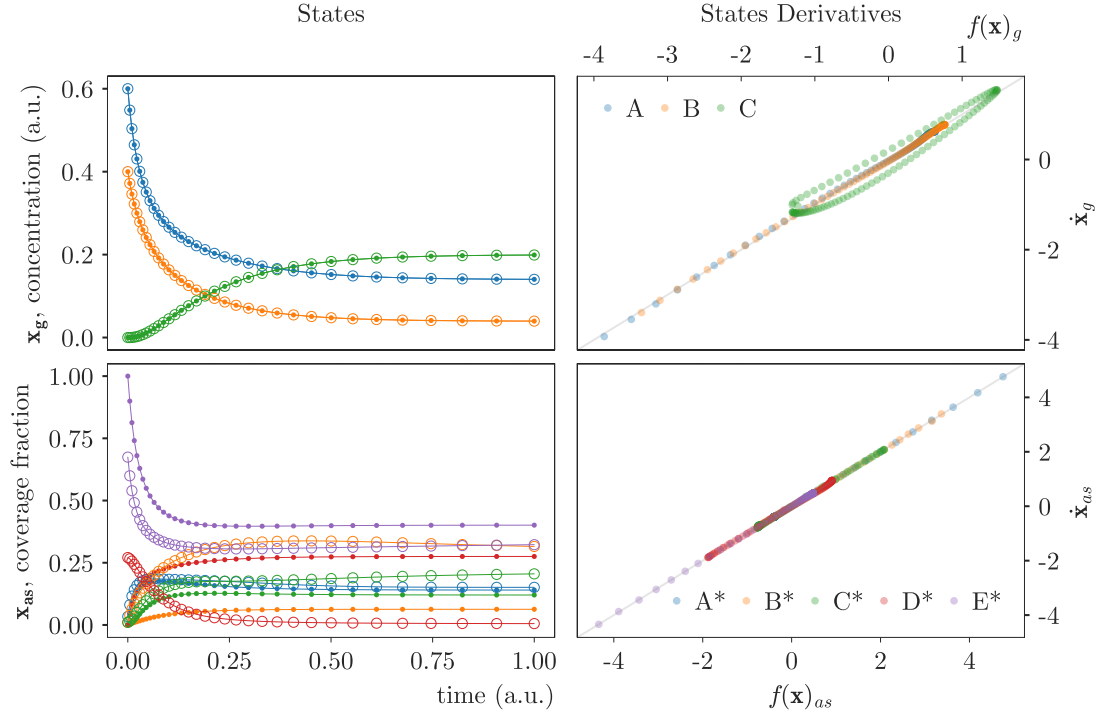

Figure S2.2: Reaction network type *da* inverse solution for QB data using BC2. States solution (left) (SM, open circles; observed data, closed markers) under minimization of (5.1) (right) for observable variables (unbound species, top).
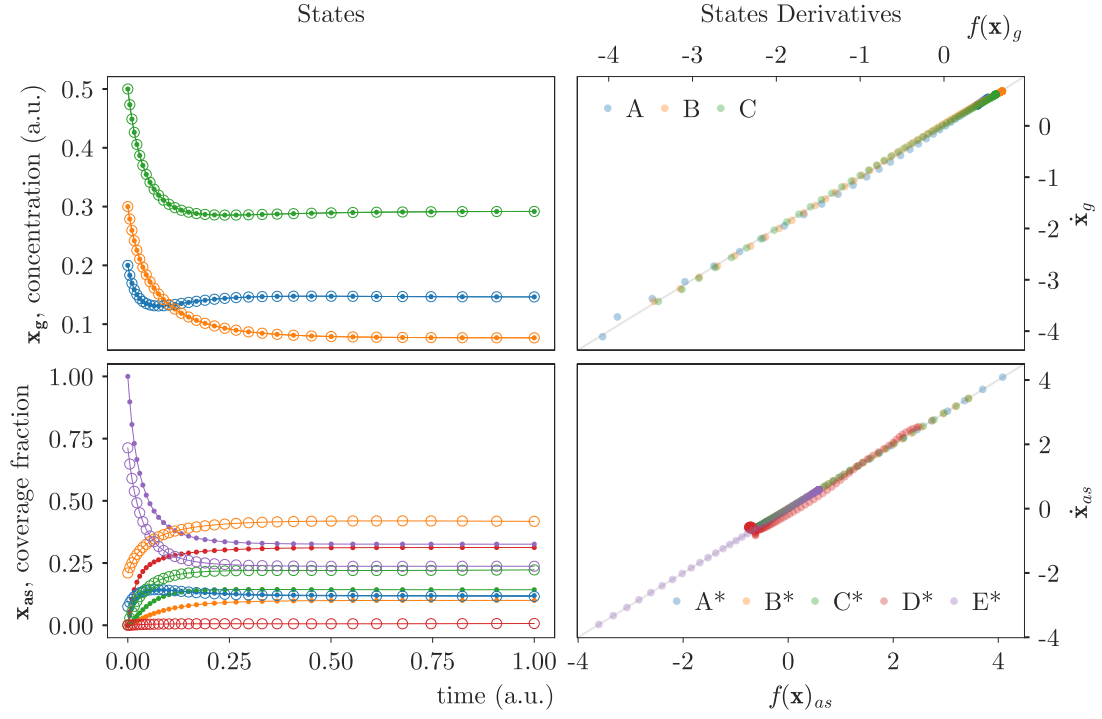
Figure S2.3: Reaction network type *da* inverse solution for data with noise using BC1. States solution (left) (SM, open circles; SQ data, closed markers) under minimization of (5.1) (right) for observable variables.

Figure S2.4: Reaction network type *da* inverse solution for data with noise using BC2. States solution (left) (SM, open circles; SQ data, closed markers) under minimization of (5.1) (right) for observable variables.

Figure S2.5: Reaction network type *da* inverse solution for data with noise using BC1. States solution (left) (SM, open circles; SQ+n data, closed markers) under minimization of (5.1) (right) for observable variables.
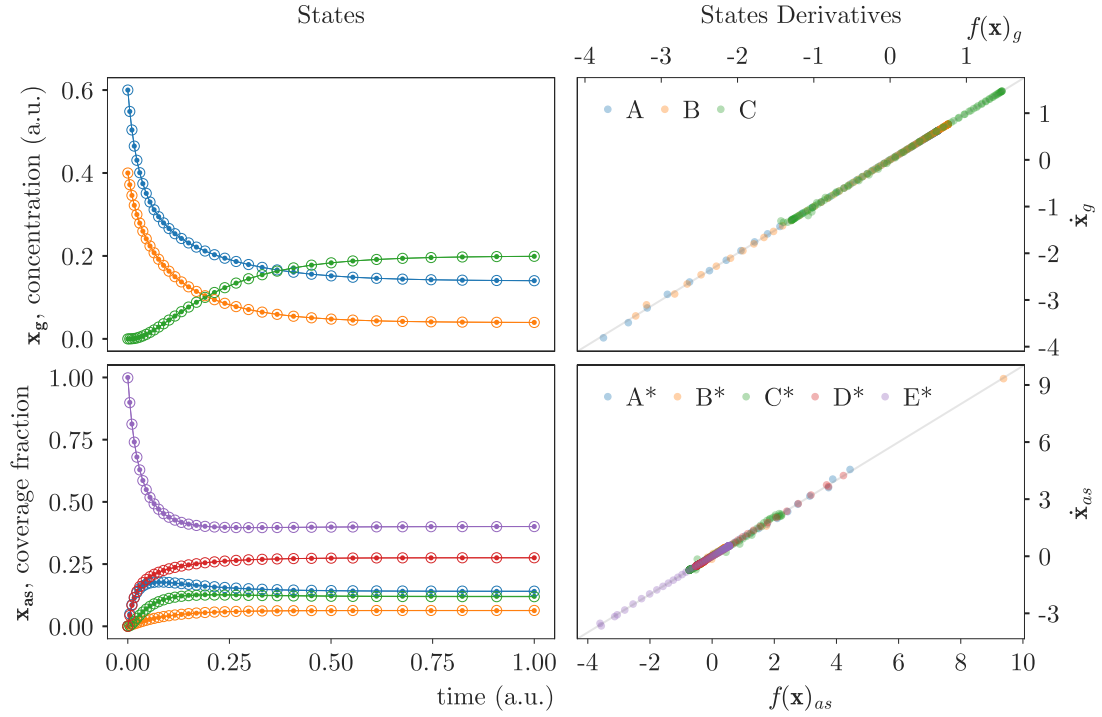

Figure S2.6: Reaction network type *da* inverse solution for data with noise using BC2. States solution (left) (SM, open circles; SQ+n data, closed markers) under minimization of (5.1) (right) for observable variables.

Figure S2.7: Reaction network type *da* inverse solution. Natural logarithms of ground truth parameter $\ln(k_0)$ vs. regressed parameters, $\ln(k(\boldsymbol{\omega_m})) = \boldsymbol{\omega_m}$, for QB, SQ and SQ+n data.

3   Reaction Network Type *dc*

The *a*-type reaction is replaced by two *d*-type reactions, i.e. reaction involving one intermediate species ($D*$), which does not have a corresponding gas phase species. The *d*-type are suppressed in the following set of elementary reactions.

$$B * + * \underset{k_{12}}{\overset{k_{11}}{\rightleftharpoons}} 2D* \tag{c.1}$$

$$A * + D* \underset{k_{14}}{\overset{k_{13}}{\rightleftharpoons}} C * + * \tag{c.2}$$

$$\ln(\mathbf{k}_0) = [3.00 \ 2.08 \ 2.77 \ 1.39 \ 2.48 \ 2.08 \ 7.09 \ 5.99 \ 7.60 \ 7.38]$$

$$M = \begin{bmatrix} -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 \\ 0 & 0 & 1 & -1 & 0 & 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & -2 & -1 & 1 \\ -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & 1 & -1 \end{bmatrix} \tag{S3.1}$$

$$\mathbf{x}^T = [x_A \ x_B \ x_C \ x_{A*} \ x_{B*} \ x_{C*} \ x_{D*} \ x_*]$$

Figure S3.1: Reaction network type *dc* inverse solution for QB data using BC1. States solution (left) (SM, open circles; observed data, closed markers) under minimization of (5.1) (right) for observable variables (unbound species, top).
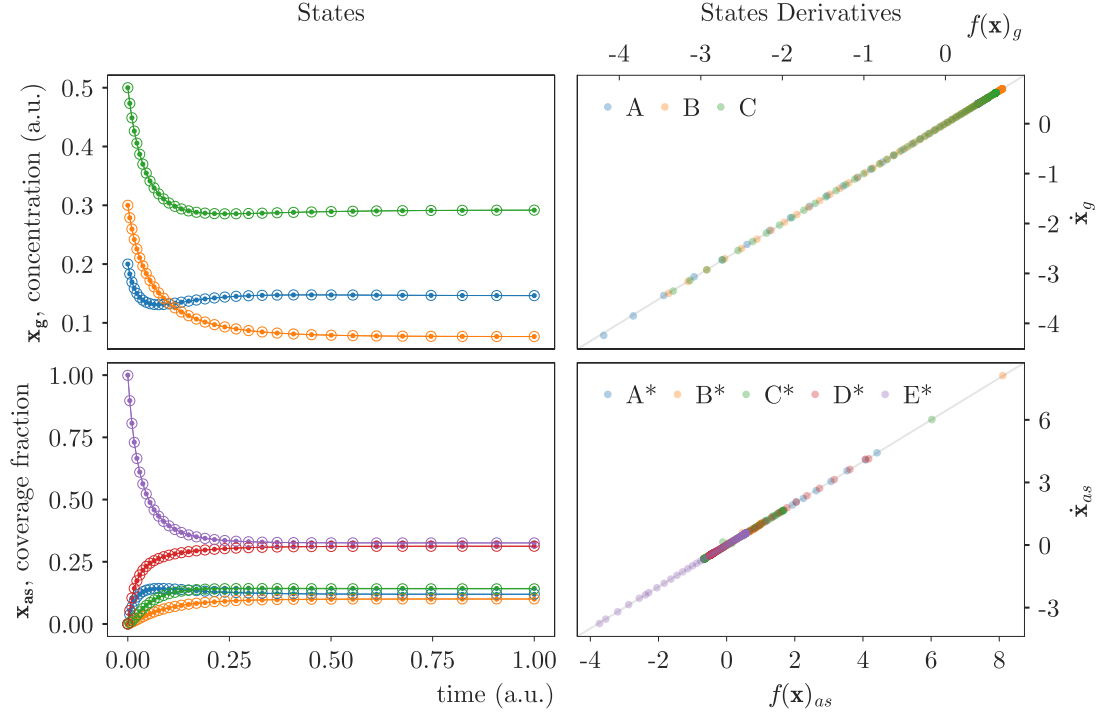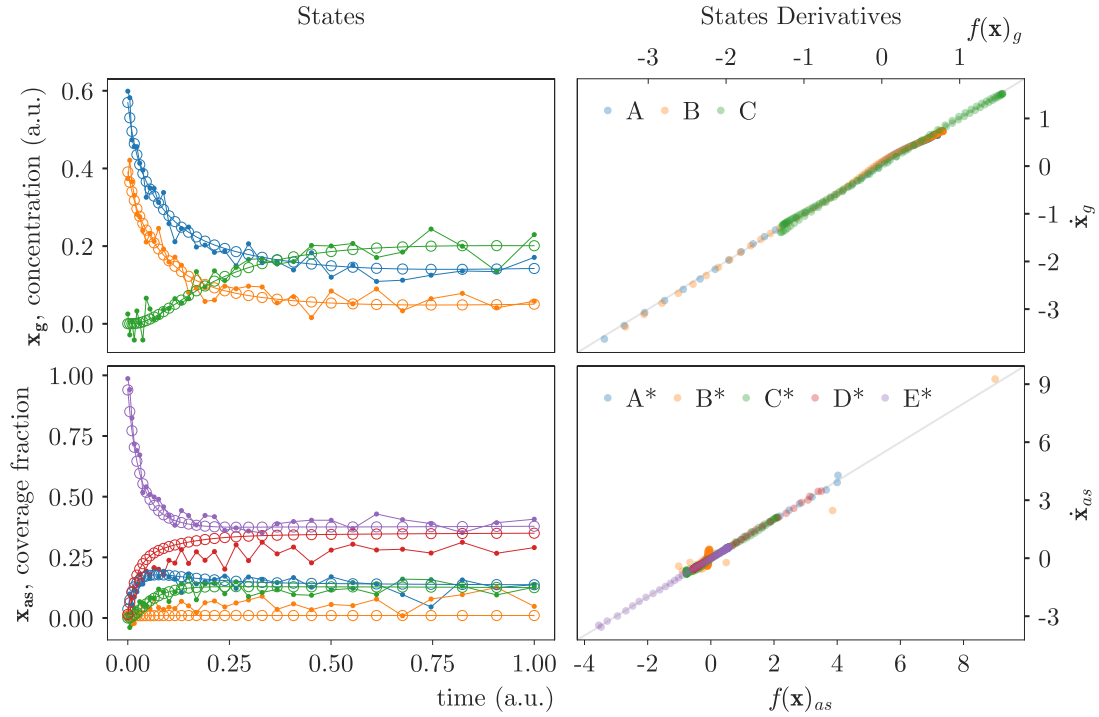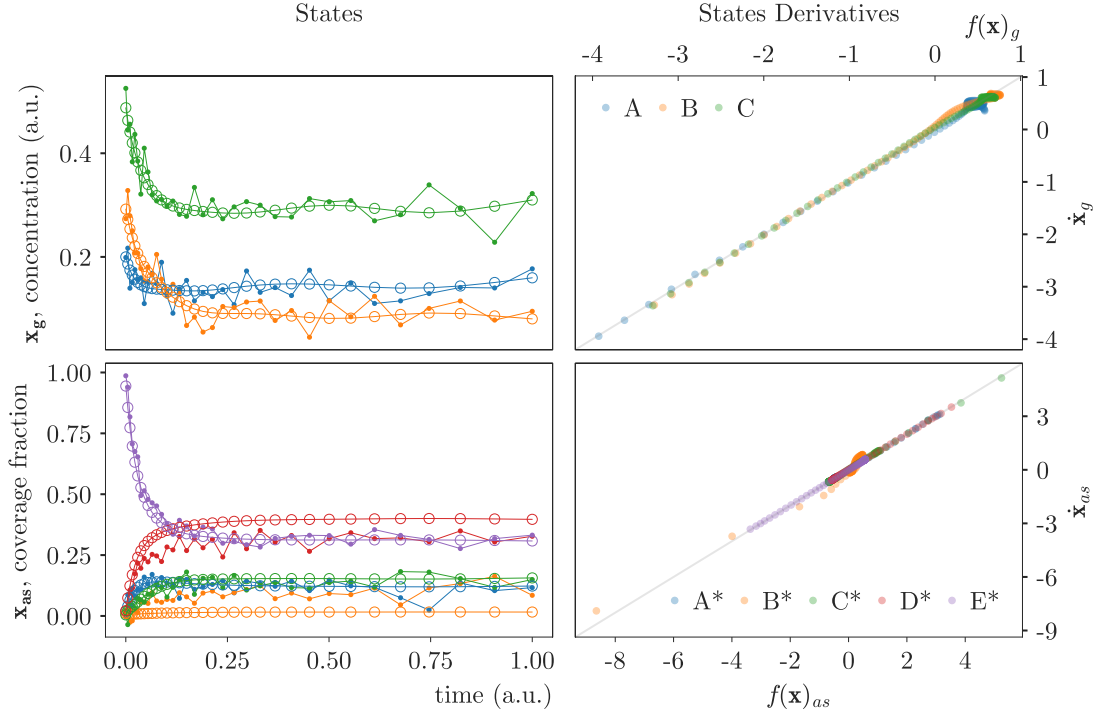
Figure S3.2: Reaction network type *dc* inverse solution for QB data using BC2. States solution (left) (SM, open circles; observed data, closed markers) under minimization of (5.1) (right) for observable variables (unbound species, top).
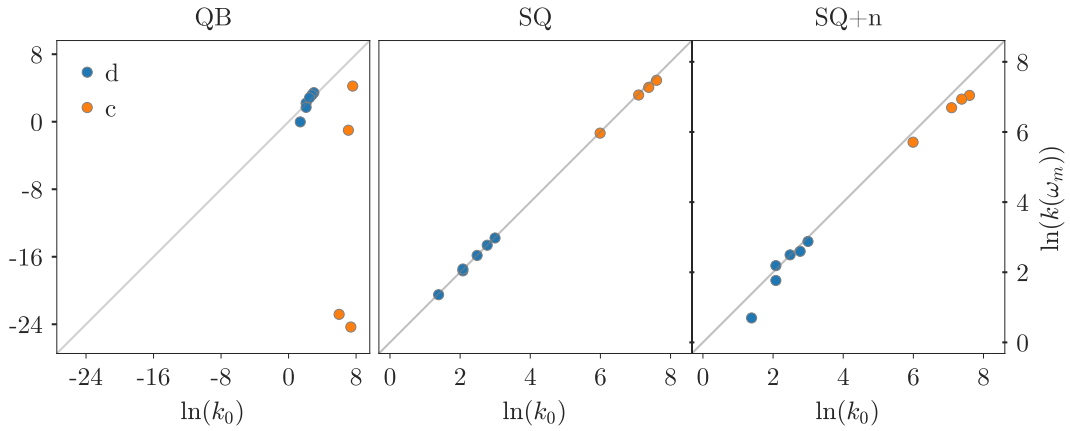

Figure S3.3: Reaction network type *dc* inverse solution for SQ data using BC1. States solution (left) (SM, open circles; observed data, closed markers) under minimization of (5.1) (right) for observable variables (unbound species, top).
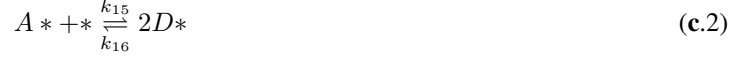
Figure S3.4: Reaction network type *dc* inverse solution for SQ data using BC2. States solution (left) (SM, open circles; observed data, closed markers) under minimization of (5.1) (right) for observable variables (unbound species, top).

Figure S3.5: Reaction network type *dc* inverse solution for SQ+n data using BC1. States solution (left) (SM, open circles; observed data, closed markers) under minimization of (5.1) (right) for observable variables (unbound species, top).

Figure S3.6: Reaction network type *dc* inverse solution for SQ+n data using BC2. States solution (left) (SM, open circles; observed data, closed markers) under minimization of (5.1) (right) for observable variables (unbound species, top).

Figure S3.7: Reaction network type *dc* inverse solution. Natural logarithms of ground truth parameter $\ln(k_0)$ vs. regressed parameters, $\ln(k(\boldsymbol{\omega_m})) = \boldsymbol{\omega_m}$, for QB, SQ and SQ+n data.

4 Reaction Network Type *dcs*

The *s*-type elementary steps embody elementary reactions between surface intermediates exclusively. In the following example, $D*$, $E*$ and $F*$ are reaction intermediates for which there are no stable desorbed counterparts, which cannot be directly measured or inferred from ordinary analytical chemistry techniques.

$$A * + * \underset{k_{16}}{\overset{k_{15}}{\rightleftharpoons}} 2D* \tag{c.2}$$

$$B * + * \underset{k_{18}}{\overset{k_{17}}{\rightleftharpoons}} 2E* \tag{c.3}$$

$$D * + E* \underset{k_{20}}{\overset{k_{19}}{\rightleftharpoons}} F * + * \tag{s.1}$$

$$F * + E* \underset{k_{22}}{\overset{k_{21}}{\rightleftharpoons}} C * + * \tag{c.4}$$

$$\ln(\mathbf{k}_0)^T = [3.00\ \ 2.08\ \ 3.18\ \ 2.48\ \ 2.77\ \ 3.69\ \ 6.46\ \ 6.87\ \ 5.08\ \ 4.38\ \ 6.46\ \ 5.48\ \ 6.33\ \ 5.08]$$

$$M = \begin{bmatrix}
-1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & -1 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \\
0 & 0 & 0 & 0 & 0 & 0 & 2 & -2 & 0 & 0 & -1 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & -2 & -1 & 1 & -1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & 1 \\
-1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & 1 & -1 & 1 & -1
\end{bmatrix} \tag{S4.1}$$

$$\mathbf{x}^T = [x_A\ \ x_B\ \ x_C\ \ x_{A*}\ \ x_{B*}\ \ x_{C*}\ \ x_{D*}\ \ x_{E*}\ \ x_{F*}\ \ x_*]$$
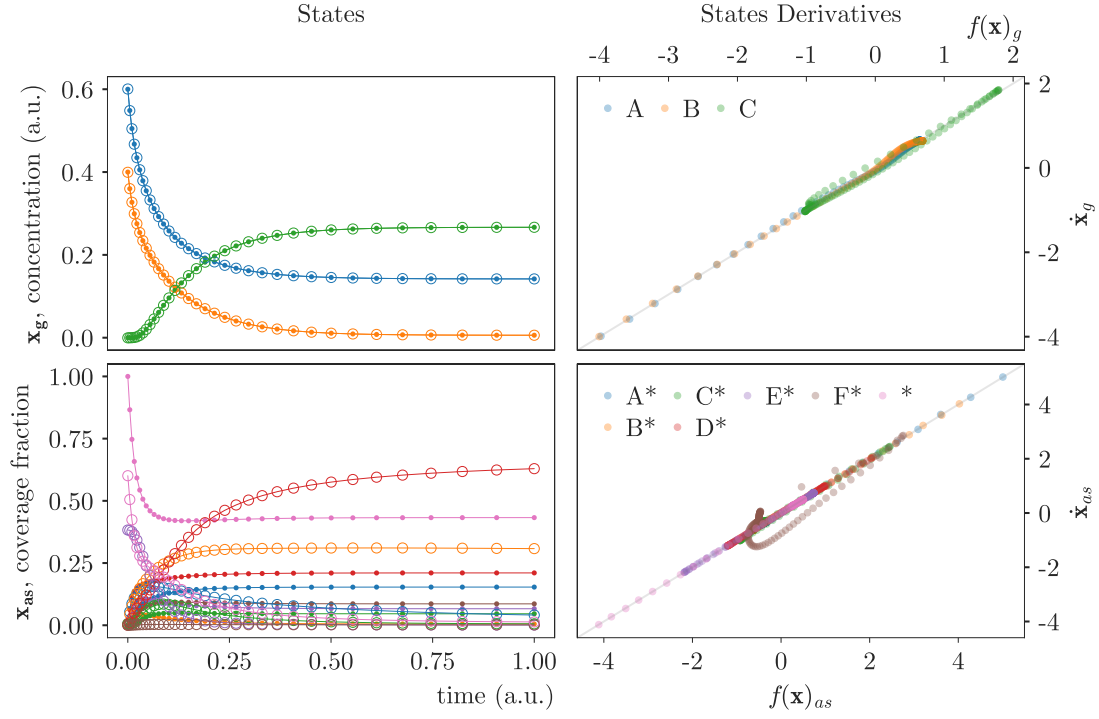
Figure S4.1: Reaction network type *dcs* inverse solution for QB data using BC1. States solution (left) (SM, open circles; observed data, closed markers) under minimization of (5.1) (right) for observable variables (unbound species, top).
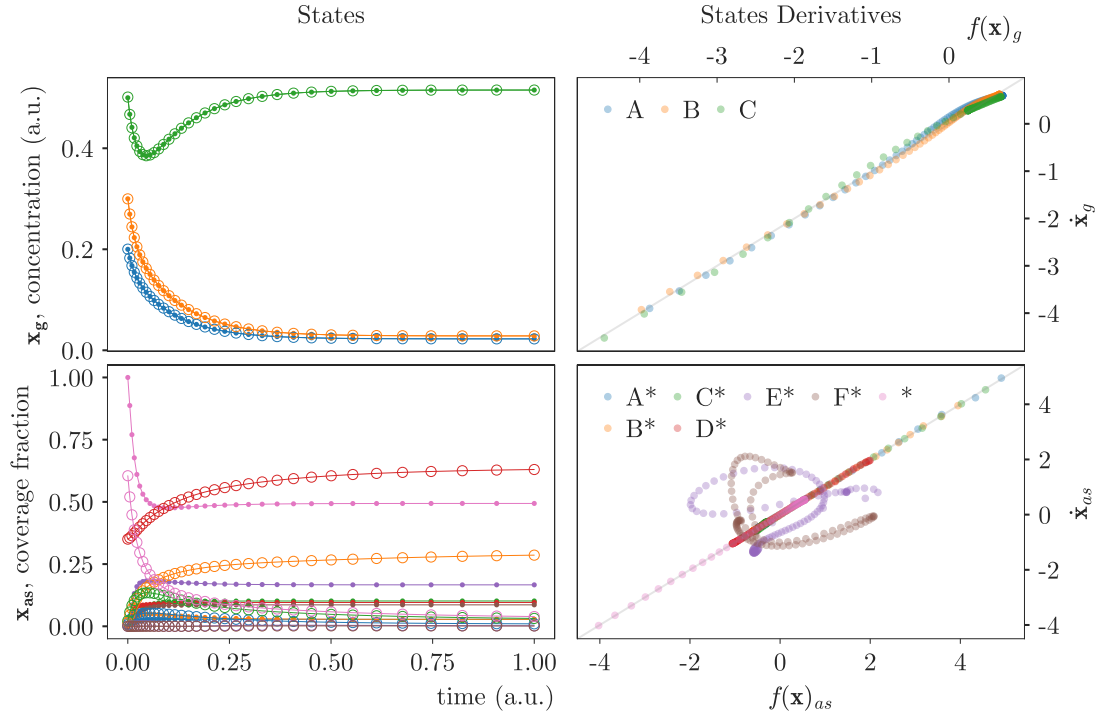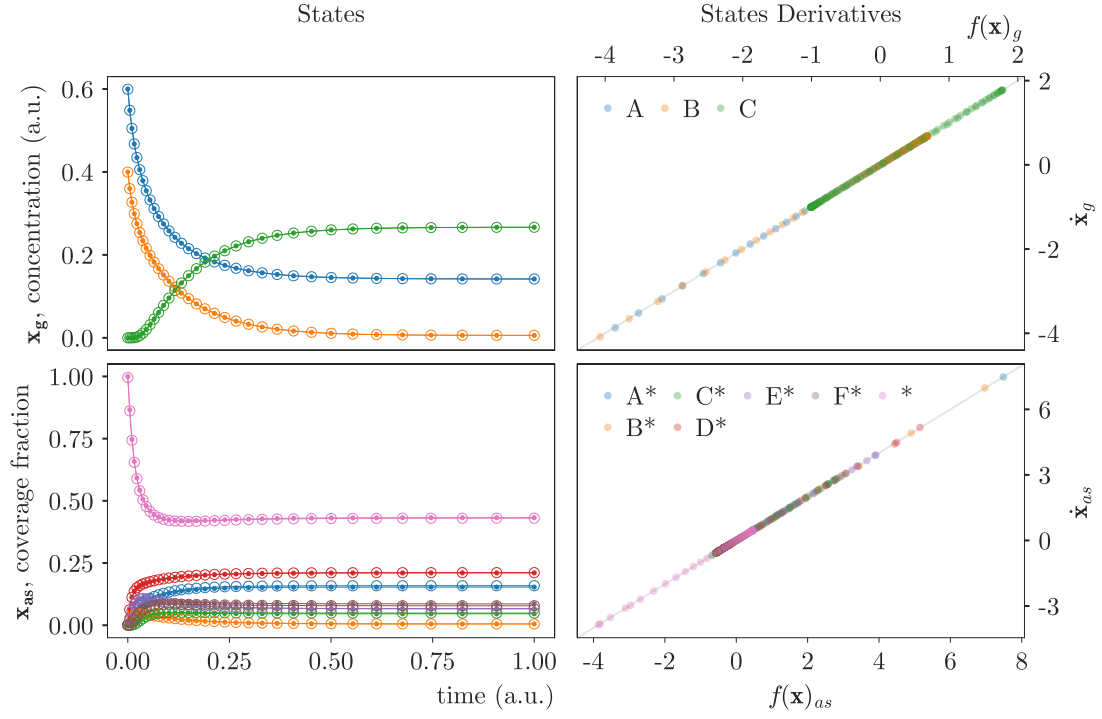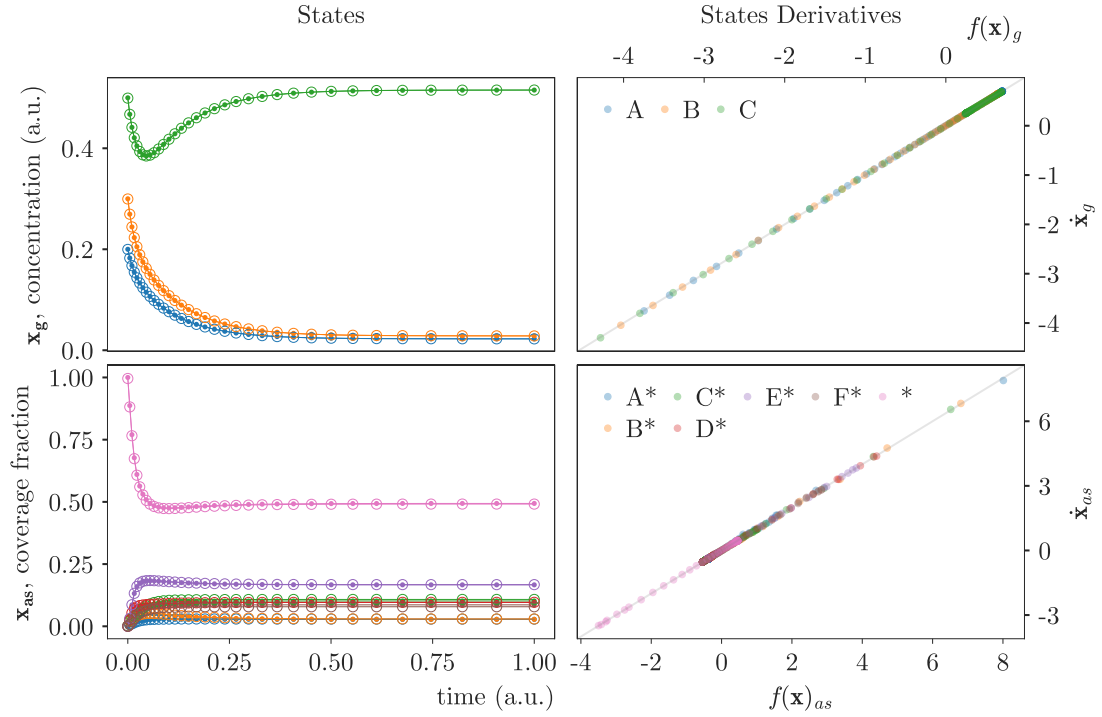

Figure S4.2: Reaction network type *dcs* inverse solution for QB data using BC2. States solution (left) (SM, open circles; observed data, closed markers) under minimization of (5.1) (right) for observable variables (unbound species, top).

Figure S4.3: Reaction network type *dcs* inverse solution for SQ data using BC1. States solution (left) (SM, open circles; observed data, closed markers) under minimization of (5.1) (right) for observable variables (unbound species, top).
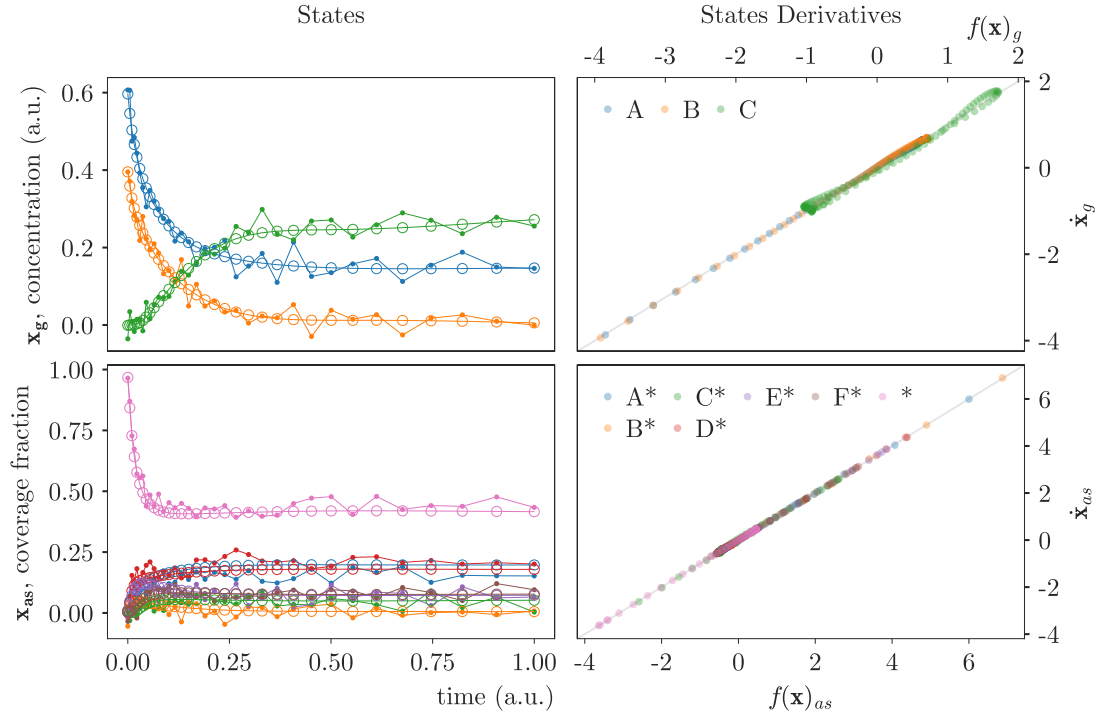

Figure S4.4: Reaction network type *dcs* inverse solution for SQ data using BC2. States solution (left) (SM, open circles; observed data, closed markers) under minimization of (5.1) (right) for observable variables (unbound species, top).

Figure S4.5: Reaction network type *dcs* inverse solution for SQ+n data using BC1. States solution (left) (SM, open circles; observed data, closed markers) under minimization of (5.1) (right) for observable variables (unbound species, top).
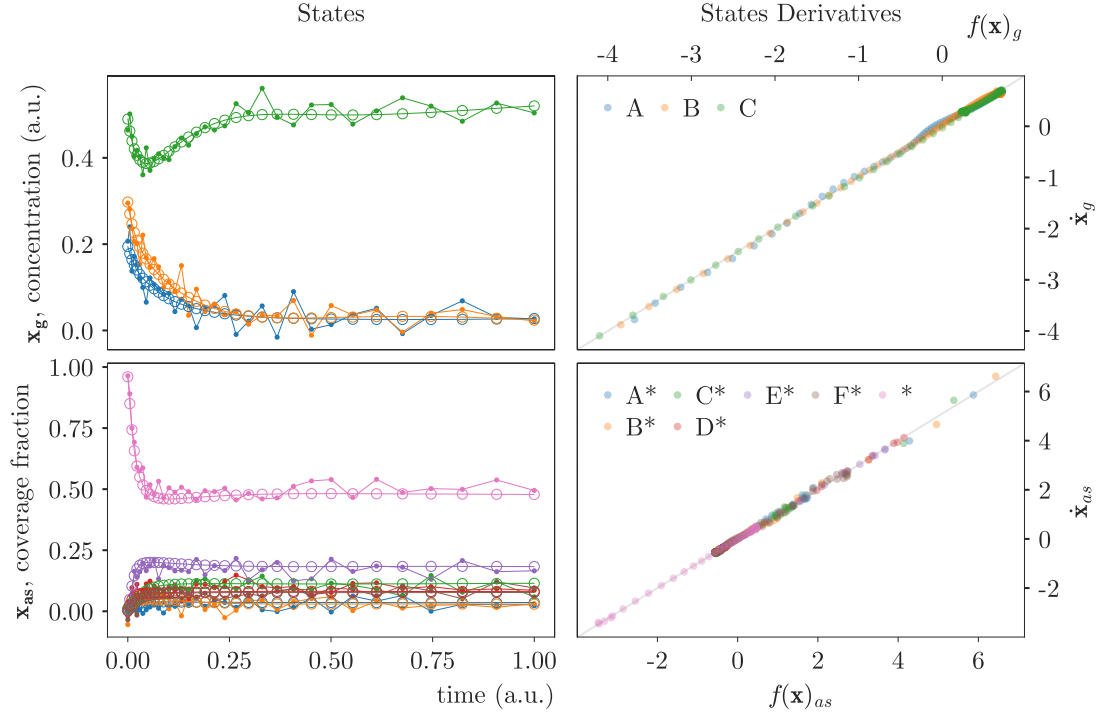
Figure S4.6: Reaction network type *dcs* inverse solution for SQ+n data using BC2. States solution (left) (SM, open circles; observed data, closed markers) under minimization of (5.1) (right) for observable variables (unbound species, top).
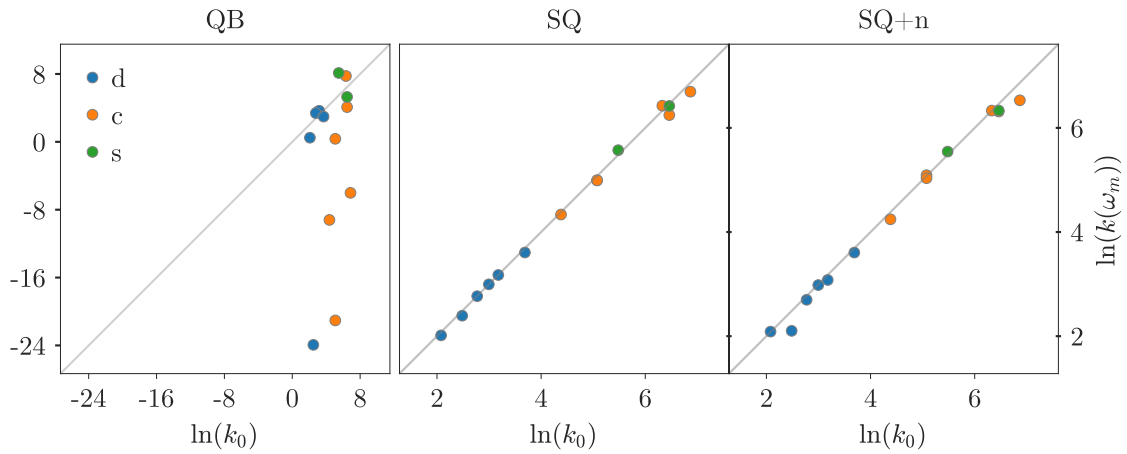

Figure S4.7: Reaction network type *dcs* inverse solution. Natural logarithms of ground truth parameter $\ln(k_0)$ vs. regressed parameters, $\ln(k(\boldsymbol{\omega_m})) = \boldsymbol{\omega_m}$, for QB, SQ and SQ+n data.