Kinetics-Constrained Neural Ordinary Differential Equations: Artificial Neural Network Models tailored for Small Data to boost Kinetic Model Development

Aleksandr Fedorov*, Anna Perechodjuk, David Linke*

Leibniz-Institut für Katalyse e.V., Albert-Einstein-Str. 29a, 18059 Rostock, Germany

*Corresponding author.

E-mail: aleksandr.fedorov@catalysis.de (A. Fedorov)

david.linke@catalysis.de (D. Linke)

Abstract

Artificial neural networks (ANNs) are powerful tools for solving a wide range of tasks in fundamental and applied science. However, training and building reliable ANN models requires a lot of data which so far hinders their wider application in kinetic modelling where typically only small (experimental) datasets are available. In the present work we propose a method to design ANN models for kinetic modelling that can be trained even with small data sets as are typically available. The key idea is to constrain the architecture of the ANN models by integrating kinetic and thermodynamic knowledge leading to what we call Kinetics-Constrained Neural Ordinary Differential Equations (KCNODE). The feasibility and effectiveness of the approach is first demonstrated in a numerical experiment using the catalytic hydrogenation of CO_2 to methane as example. Next, we demonstrate the approach for real experimental data of a more complex reaction, the hydrogenation of CO_2 to higher hydrocarbons (CO_2-FT). Finally, the ANN trained for CO_2-FT is used to derive an improved mechanistic model for the reverse water gas shift reaction which is a key reaction in the CO_2-FT reaction network. This last step exemplifies how the opportunity to obtain reliable ANN models from small data opens new ways to approach kinetic model development.

Keywords: kinetic modelling, machine learning, artificial neural networks, CO_2 hydrogenation, Kinetics-constrained Neural Ordinary Differential Equations.

Introduction

The use of ordinary differential equations (ODEs) for describing dynamic systems plays a key role in different fields of applied science and engineering. One of these fields is kinetic modelling of heterogeneous catalytic reactions. Another is the development of reliable models for describing catalytic process for scale-up or optimizing industrial plants in chemical engineering [1]. Traditionally, the development of such models builds upon a microkinetic analysis that requires a deep knowledge about the reaction mechanism. Postulating rate-determining steps and equilibrium stages or a most abundant reaction intermediate are general approaches to simplify the model resulting in Langmuir-Hinshelwood-Hougen-Watson (LHHW) expressions [2]. Moreover, the parameter estimation from experimental data is still an untrivial calculation problem [3].

In recent years, there has been a rapid rise in the use of machine learning (ML) and data science methods for solving many complex tasks in the different fields of science, industry, and business. These methods were successfully applied for smart discovery of high-performance and functional materials [4], revealing the latent knowledge from the material science literature [5], modelling complex physicochemical processes [6, 7] etc. [8-12]. The modern approaches of ML are also applied for accelerating the development of kinetic models [13, 14].

Artificial neural networks (ANNs) as universal approximators [15] are a frequently applied machine learning instrument. Their popularity is related to their flexibility to be used in various types of ANN architectures [16-20]. In the field of dynamic system modelling, Maziar Raissi et al. [7] introduced physics informed neural networks (PINN) as a kind of neural networks for solving forward and inverse problems involving nonlinear partial differential equations. They embed the knowledge about physical laws into the loss function and, thus, the trained ANN models satisfy the experimental data as well as the governing equations. Weiqi Ji et al. successfully applied PINN for solving stiff chemical problem [21]. Gabriel S.Gusmão et al. [22] generalized PINN for any chemical kinetics that was called kinetics-informed neural networks. Despite of the effectivity of the suggested approach, it still requires knowledge about the reaction mechanism and reaction rate expressions.

In 2018, Ricky T. Q. Chen et al. introduced a new family of deep neural network models that are Neural ODEs [23]. They suggested parametrizing the derivative of the hidden state using a neural network:

1. $\frac{dx}{dt} = ANN(x, t, \theta)$

Neural ODE is an attractive approach for data-driven modelling of dynamic systems including kinetic models [24, 25]. In fact, ANN can be used for approximating reaction rate expressions that are usually unknown. Weiqi Ji and Sili Deng [26] suggested a new type of neural ODE – chemical reaction neural network (CRNN) for autonomous identifying reaction pathways. The architecture of CRNN is based on the law of mass action and the Arrhenius law. Despite of successfully demonstration for numerical examples [26], in the case of complex dynamic system of biomass pyrolysis [27], it is difficult to interpret the developed CRNN model. This is caused by the fact that many chemical processes including heterogeneous and catalytic reactions cannot be described by simple power-law-like equations.

In summary, ANNs are powerful instrument of ML for developing quantitative regression models and can be applied for solving a lot of tasks. The main advantage is that only data are required for building models, and, thus, it obviates the need to suggest (guess) reaction mechanism, rate-determining stage etc. in the case of kinetic modelling. However, one of the known disadvantages in applying neural networks is the requirement of a lot of data for training [28]. This so far limits the usage of ANNs in the case of small data sets which are typically available for experiment-driven development of kinetic models. Therefore, a methodology to obtain reliable ANN-based models from small data sets would be extremely valuable because it could significantly expand the scope of applying ANN models. In the present work we investigated new approaches to apply neural ODEs for kinetic modelling of CO_2 hydrogenation to hydrocarbons using small data sets. To solve the small data problem and to improve the predicting ability of the models, the architecture of ANN was constrained by integrating the kinetic and thermodynamic knowledge. The efficiency of these new ANN models was validated by carrying out numerical experiments using the LHHW-type model of CO_2 hydrogenation to methane from literature [2]. The suggested approach was finally applied to a more challenging problem: the kinetics of CO_2 hydrogenation into hydrocarbons was modelled based on real measured experimental data.

Experimental part

Numerical data generation

To develop and validate the ANN models, the data from a typical fixed bed reactor are needed. Thus, we performed the steady-state ideal plug-flow reactor simulation using the kinetic model developed by Lucas Brübach et al. [2]. This model is based on two reactions:

2. $CO_2 + H_2 = CO + H_2O$ (RWGS)
3. $CO + 2H_2 = (CH_2) + H_2O$ (CO-FT)

They used C_4-species as a pseudo product for representing the hydrocarbons as product of CO-FT reaction. In the present work, to simplify the calculation, we considered only methane as a product of CO-FT reaction. Based on elementary reaction steps, they suggested the following empirical LHHW expressions for describing the rates r_i of the RWGS and CO-FT reactions:

4. $$r_{RWGS} = \frac{k_{RWGS}\left(p_{CO_2}\sqrt{p_{H_2}} - \frac{p_{CO}p_{H_2O}}{K_{eq}\sqrt{p_{H_2}}}\right)}{\left(1 + a_{RWGS}\frac{p_{H_2O}}{p_{H_2}}\right)^2}$$

5. $$r_{FT} = \frac{k_{FT}p_{CO}p_{H_2}}{\left(1 + a_{FT}\frac{p_{H_2O}}{p_{H_2}} + b_{FT}p_{CO}\right)^2}$$

The temperature dependency of the rate constants is given by re-parameterized Arrhenius equation:

6. $$k_i = k_{i,ref}\exp\left(-\frac{E_{a,i}}{R}\left(\frac{1}{T} - \frac{1}{T_{ref}}\right)\right)$$

For calculation of the equilibrium constant K_e of the RWGS reaction at different temperature the following expression was used:

7. $$\log K_e = 3.933 - \frac{4076}{T/K - 39.64}$$

The values of all parameters used in the simulations can be found in the Table S1. As reactor model, we selected the steady-state ideal plug-flow reactor model. It was assumed that temperature and total pressure are constant along the reactor length; axial dispersion and wall

effect are negligible; internal and external mass transfer limitations are also negligible. Based on this, the data was generated by the solving the following system of ordinary differential equations:

$$8.\quad \frac{dF_j}{dV} = \rho_c \sum_i \nu_{i,j}\, r_i$$

For solving this stiff system of ordinary differential equations, the numerical Adams/BDF method with automatic stiffness detection and switching was used [29]. The developed reactor model was used for generating training and test data sets. A small training data set mimicking a typical set of catalytic experiments that could be used for creating the kinetic models in conventional way was used for training ANN models. The reaction conditions used for generating the training data set are presented in the Table 1. The catalyst volume V was varied in a range from 10^{-4} to 10^{-1} cm^{-3}. For each reaction conditions, the 7 points of V were uniformly spaced in a logarithmic scale across the chosen range. In the case of training the baseline model (see the baseline model), the data with zero molar flows of CO_2 + CO or zero flow of H_2 were additionally added. In these cases, the molar flows are constant along the reactor length and equal to the molar flows at the reactor inlet.

Table 1. Reaction conditions for generating the training data set.

#	Temperature, °C	Pressure, bar	Molar flow, $mol \cdot s^{-1}$			
			CO_2	CO	H_2	N_2
1	250	10	1.0	0	3.0	1.0
2	300	10	1.0	0	3.0	1.0
3	350	10	1.0	0	3.0	1.0
4	300	15	1.0	0	3.0	1.0
5	300	20	1.0	0	3.0	1.0
6	300	10	1.0	0	2.0	1.0
7	300	10	1.0	0	6.0	1.0
8	300	10	0.5	0.5	3.0	1.0

To make the modelling task more realistic, we added Gauss noise to the data to imitate an experimental error. Four training sets with 0.0, 2.5%, 5.0%, and 10% noise were generated. To validate the performance of the obtained neural ODE models, another larger test data set was generated by a full factorial design [30] using the following reaction conditions:

- Temperature: 255, 285, 315, 345 °C.
- Pressure: 12, 14, 16, 18 bar.
- CO_2 molar flow: 0.0, 0.5, 1.2 $mol \cdot s^{-1}$.
- H_2 molar flow: 2.5, 4.0, 5.5 $mol \cdot s^{-1}$.

- CO molar flow: 0.0, 0.3 mol·s^{-1}.
- CH_4 molar flow: 0.0, 0.3 mol·s^{-1}.
- H_2O molar flow: 0.0, 0.3 mol·s^{-1}.

The molar flow of N_2 was set to 1.0 mol·s^{-1}. Thus, the total amount of different reaction conditions was 1152 (compared to 8 in the training data set Table 1). The catalyst volume V was varied in a range from 10^{-4} to 1 cm^{-3}. For each reaction conditions, the 20 points of V were uniformly spaced in a logarithmic scale across the chosen range.

Baseline neural ODE

As a basic model, the following neural ODE model was used in the numerical experiment:

$$\frac{dF}{dV} = ANN(\bar{p}, T, \theta) \quad (9)$$

where F are molar flows of compounds, the reaction temperature T, and the vector of the partial pressure $\bar{p}$. In such notation the inputs of neural network have partial pressure of components, and, thus, the partial pressure was calculated by the following formula:

$$p_i = \frac{F_i \cdot P_\Sigma}{\sum F_j} \quad (10)$$

where F_i are molar flows of i-compound; P_Σ is the total pressure. It is worth mentioning that there are parameters like T or P_Σ which are independent of V and should be constant along the reactor length. Parameterized neural ODE (PNODE) [31] suggested by Kookjin Lee and Eric J.Parish is an extension of neural ODE and was developed for cases when such independent parameters have to be taken into account. The main idea is to use additional encoder-decoder neural layers. Despite of authors demonstrated the effectiveness of PNODEs on benchmark problems from computational physics, the suggested approach still has a disadvantage. It is related to the complication of the architecture of neural ODEs and an increase in number of learnable parameters that require more data for model training. In our work we suggest a simpler approach for the problem of independent parameters that does not require to use additional ANN learnable variables. Our approach is to return zeros values as outputs of ANN for such parameters. This makes them to be constant and independent of V.

The neural network consists of 3 layers with 8, 8, and 5 nodes per layer, and with activation functions: hyperbolic tangent, exponential and linear, respectively. The first layer consists of 6 inputs: partial pressure of compounds (for CO_2, CO, H_2, H_2O, CH_4) and temperature. The output of the linear layer is expanded by concatenating with 3D zeros vector (for temperature, total pressure and N_2 molar flow). Before training, the value of total pressure and temperature were scaled by dividing by the maximum values.

Kinetics-constrained neural ODE

CO_2 hydrogenation to methane

In this part, the approach for kinetic modelling of catalytic reactions using neural ODE models constrained with the information about kinetic and thermodynamic knowledge is described. As an example, the architecture of an ANN for kinetic modeling of CO_2 hydrogenation into methane is described. Firstly, instead of using ANN for approximating molar flows of each component as in the baseline model, we suggest using ANN for approximating the reaction rates which are the feature of most interest to reaction engineers but also the basis for a detailed, mechanism-based kinetic modelling. In the case of the CO_2 hydrogenation to methane, the considered reactions are RWGS and CO-FT. It is worth mentioning that reactions pathways can be obtained by analyzing selectivity-conversion dependencies for any reaction networks [32-35]. So, the rates of formation/consumption of each compound can be calculated by using the stoichiometric matrix M_ν of the corresponding reactions:

$$11.\ \overline{r_c} = M_\nu \overline{r_R},$$

where $\overline{r_c}$ and $\overline{r_R}$ are the vectors of the compound and reaction rates. Secondly, the ANN model of the rate expression is split into two parts. The first part is a feedforward neural network where inputs are partial pressure of components, and the second one is the Arrhenius-type expression for temperature dependence of rates. The second part represents a neural network layer with exponential activation function that is applied in the CRNN [26]. 'Arrhenius' temperature T_m as input for this layer was used and is defined as:

$$12.\ T_m = \frac{1}{R}\left(\frac{1}{T} - \frac{1}{T_{ref}}\right),$$

where R – the gas constant, T – temperature, T_{ref} = 573.15 K. Thus, weights and biases for this layer represent the activation energies and the logarithm of rate constants, respectively. The next step is to integrate the information about thermodynamic of the RWGS reaction. At

equilibrium of partial pressure of components, the rate of RWGS reaction must be equal to 0 and if the reaction product Π is more than K_{eq} the rate must be negative. Finally, the trivial information that the rates must be equal to 0 at the absence of reagents is added (the rates should be zero if the partial pressure of involved reagents is zero). As summary, the rates can be presented as:

$$13.\ r_{RWGS} = k_{RWGS}^{ref} \exp\left(-\frac{E_a^{RWGS}}{R}\left(\frac{1}{T}-\frac{1}{T_{ref}}\right)\right) p_{CO_2} p_{H_2} \left(1-\frac{\Pi}{K_{eq}}\right) ANN(\bar{p},\theta)$$

$$14.\ r_{FT} = k_{FT}^{ref} \exp\left(-\frac{E_a^{FT}}{R}\left(\frac{1}{T}-\frac{1}{T_{ref}}\right)\right) p_{CO} p_{H_2} ANN(\bar{p},\theta)$$

where rate constants are defined by expression (3); Π – the reaction product of the RWGS that can be defined as:

$$15.\ \Pi = \frac{p_{CO} p_{H_2O}}{p_{CO_2} p_{H_2}}$$

We refer to such type of neural ODE a 'kinetics-constrained neural ODE' (KCNODE). The general idea and the architecture of KCNODE for kinetic modelling of CO_2 hydrogenation to methane are presented in the Figure 1.

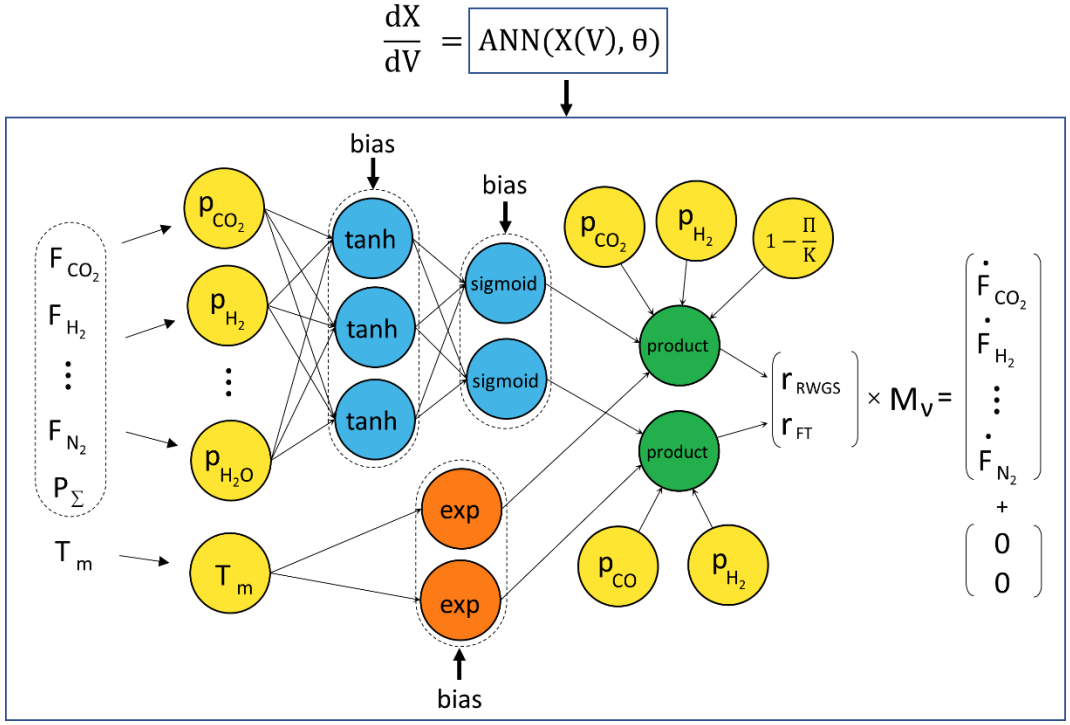

Figure 1. Topology of kinetics-constrained neural ordinary differential equation. *tanh*, *sigmoid* and *exp* are neural layers with hyperbolic tangent, sigmoid and exponential activation functions, respectively; *product* is a layer which returns the multiplication of all the inputs; the sign '×' expresses matrix multiplication and the sign '+' is the concatenation. F_i – molar

flow of i-compound; P_Σ – the total pressure; P_i – the partial pressure of i-compound; Π – the reaction product; K – the equilibrium constant.

It is important to emphasize that for any values of neural network parameters the following constraints are met:

- Material balance.
- The rate of RWGS reaction is 0 at equilibrium. At $\Pi < K_{eq}$ the rate > 0.
- Arrhenius dependency of the rates on temperature.
- The rates are equal to 0 in absence of reagents.

It is worth also noting the key difference between our approach and PINN models. In PINN, the knowledge about physical laws is embedded at training of ANN by using corresponding loss functions. In our case, the architecture of ANN is already intrinsically constrained by kinetic and thermodynamic knowledge. The hidden layer of the neural network with a hyperbolic tangent activation function consists of 8 nodes. To investigate the performance of KCNODE models, an analogue model with 20 nodes in the hidden layer was also constructed. Before training the values of pressure was scaled by dividing by the maximum total pressure.

CO_2 hydrogenation to hydrocarbons

The suggested approach was applied for kinetic modeling of the process of CO_2 hydrogenation into hydrocarbons using real experimental data. It is generally accepted [36, 37] that the process of CO_2-FT synthesis proceeds in two stages: the RWGS reaction followed by CO hydrogenation via FT mechanism. Besides, an additional pathway of direct CO_2 hydrogenation to methane, also known as the Sabatier reaction, occurs over Fe-based catalysts that was suggested in the works [32, 35, 38, 39]. Thus, the following reaction set was used for building the model:

16. $CO_2 + H_2 \rightleftarrows CO + H_2O$
17. $CO + \left(2 + \frac{1}{n}\right) H_2 \rightarrow \frac{1}{n} C_n H_{2n+2} + H_2O$
18. $CO + 2H_2 \rightarrow \frac{1}{n} C_n H_{2n} + H_2O$
19. $CO + 2H_2 \rightarrow \frac{1}{n} C_n H_{2n+1} OH + \left(1 - \frac{1}{n}\right) H_2O$
20. $CO_2 + 4H_2 \rightarrow CH_4 + 2H_2O$

The reaction of CO hydrogenation via FT mechanism has a kinetic limitation. The mass fraction W_n (n – carbon number) of hydrocarbons (or alcohols) in the products can be described by the Anderson-Schulz-Flory (ASF) distribution [40-42]:

$$21.\ W_n = n(1-\alpha)^2\alpha^{n-1},$$

where n is carbon number, α is the chain growth probability that is defined as ratio between the rate of the chain prolongation per the sum of the chain prolongation and termination rates [43]. The equation allows describing the formation of each product observed using only a few parameters. Thus, assuming the validity of the ASF the formation rate of compound with carbon number n can be defined as:

$$22.\ r_n = n\alpha^{n-1}r_1$$

The expression (18) was also used to describe the formation of olefins and alcohols because they correspond to ASF except for CH_3OH and C_2H_4 (see Figure 3, D and E). To consider these deviations from ASF, additional dimensionless empirical coefficients (δ_{en} and δ_{OH}) were used and the rates of methanol and ethylene formation are defined as:

$$23.\ r_{C_2H_4} = 2 \cdot \delta_{en} \cdot \alpha_{en} \cdot r_1^{en}$$

$$24.\ r_{CH_3OH} = \delta_{OH} \cdot r_1^{OH}$$

Here, it is worth mentioning that r_1^{en} means the rate of formation of 'hypothetical' alkene with carbon number 1 which are not considered as a product of CO_2 hydrogenation but used only to calculate the rates of alkene formation. In the case of paraffins, a non-linear dependence of hydrocarbon distribution in the corresponding ASF coordinates is observed (see Figure 3, C). Such non-linear behavior can be related to the presence of the two different types of active sites taking part in processes of the chain growth that was discussed in the work.[44] Thus, the rate of paraffin formation with carbon number n can be described as:

$$25.\ r_n = \mu \cdot n \cdot \alpha_{an_1}^{n-1} \cdot r_1^{an} + (1-\mu) \cdot n \cdot \alpha_{an_2}^{n-1} \cdot r_1^{an}$$

where μ and $1-\mu$ are the fraction of paraffin products having molecular weight distribution parameters α_{an_1} and α_{an_2} respectively. It is worth noting that the suggested parameters like the chain growth probability α_{an_1}, α_{an_2}, α_{en}, and α_{OH}; empirical coefficients δ_{en} and δ_{OH}; fraction μ also depend on reaction conditions (partial pressure of compounds, temperature), and that the ANN model was used for predicting the values of these parameters.

The inputs for the hidden layer of ANN in the case of modeling of the CO_2-FT process are partial pressure of compounds (CO_2, CO, H_2, H_2O), temperature, and time on stream. Time on stream was used because an activation-deactivation process was observed (see Figure S9) during the measurement of the catalyst activity in CO_2 hydrogenation. The ANN was used for predicting the reaction rates (11) – (15) as well as the suggested parameters

(α_{an_1}, α_{an_2}, α_{en}, α_{OH}, δ_{en}, δ_{OH}, and μ). Thus, the number of outputs for each a node of the hidden layer was 12. For the hidden layer of the ANN with a hyperbolic tangent activation function 8 nodes were used. The model takes into consideration the formation of alkanes and alkenes (up to 15 carbon number), and alcohols (up to 7). Before training the value of total pressure was scaled by dividing by the maximum total pressure and time-on-stream was scaled by dividing by the maximum value of time-on-stream in the experimental data set.

Training neural network models

The Runge-Kutta 3/8 Method (*RK4)* and Runge-Kutta of order 5 of Dormand-Prince-Shampine method (*DOPRI5*) method were used for the integration of neural ODE [45]. Training neural network models was carried out by minimization of the loss functions using ADAM [46] optimizer with a learning rate of 0.005. The training ended when the loss function flatlined. In the case of the numerical experiment, the following loss function was used:

$$26.\ loss = MSE(F^{exp}, F^{pred})$$

In the case of training KCNODE using experimental data the following loss function was used:

$$27.\ loss = 5 \cdot MSE_{i=(CO_2,CO,CH_4)}\left(\frac{F_i^{exp}}{F_i^{scale}}, \frac{F_i^{pred}}{F_i^{scale}}\right) + \\ + \frac{1}{3}\left(MSE_{i=alkane}\left(\frac{F_i^{exp}}{F_i^{scale}}, \frac{F_i^{pred}}{F_i^{scale}}\right) + MSE_{i=alkene}\left(\frac{F_i^{exp}}{F_i^{scale}}, \frac{F_i^{pred}}{F_i^{scale}}\right) + \right. \\ \left. + MSE_{i=alcohol}\left(\frac{F_i^{exp}}{F_i^{scale}}, \frac{F_i^{pred}}{F_i^{scale}}\right)\right)$$

where F_i^{scale} – a characteristic scale for compound i that was calculated as a maximum value for the molar flow of i-compound in the dataset. In the work [47] Suyong Kim et al. suggested using such the scaling to mitigate stiffness of ODE. Another, reason for splitting the loss function in this way is to make the contribution of each hydrocarbon (or alcohol) in the loss function smaller than the contribution of CO_2, CO, and CH_4 because only few parameters are needed to describe hydrocarbon (and alcohol) distribution. L_2-regularization was used for mitigating stiffness of neural ODE at training. The parameter of the regularization was set to 10^{-6}.

Catalyst preparation, characterization, and testing

For measuring data for kinetic modeling of CO_2 hydrogenation to hydrocarbons, a typical Fe-based catalyst doped by K and Cu was prepared by incipient wetness impregnation method using Al_2O_3 support. The full description of preparation method and catalyst characterization can be found in the supplementary material. The catalysts were tested in CO_2-FT reaction at the following reaction conditions:

- Temperature in a range 250-310 °C.
- Pressure in a range 10-20 bar.
- Modified residence time in a range 52.4-5327 $kg \cdot s \cdot m^{-3}$.
- CO_2:H_2 ratio in a range 1.5-6.0.
- CO:CO_2 ratio in a range 0.0-0.52.

Time on stream was at least 14 hours for each reaction condition. The detailed description of catalytic experiments and experimental setup can be found in the supplementary material. It is worth noting that the catalysts were periodically tested at the same condition (baseline) during catalytic measurements. This allowed to follow activation-deactivation processes occurring under CO_2 reaction conditions (see Figure S9).

Software and hardware specifications

Python programming language (version 3.9.12-64bit, Windows 10 Pro) was used for calculation [48]. Scientific libraries NumPy [49] (version 1.23.0), SciPy [50] (version 1.8.1), Pandas [51] (version 1.23.0), Scikit-learn [52] (version 1.1.1) were used for data analysis and evaluation. Pytorch [53] (version 1.12.0) and Torchdyn [45] (version 1.0.3) were used for building and training neural networks models. Plotly [54] (version 5.9.0) was used to visualize the results. All calculations were performed by using 11th Gen Intel Core i-7 11700F (no GPU was used). The code and data will be made available after a retention period at https://github.com/LIKAT-Rostock/kcnode-paper.

Results and discussion

Numerical experiment

In this part we investigated the generalization of the neural ODE models trained by using the small, simulated data set. The process of training the neural ODE models represented as dependency between loss function and epoch number is summarized in the Figures S1. The

generalization of the obtained models in predicting training and test data sets are presented in Figure 2.

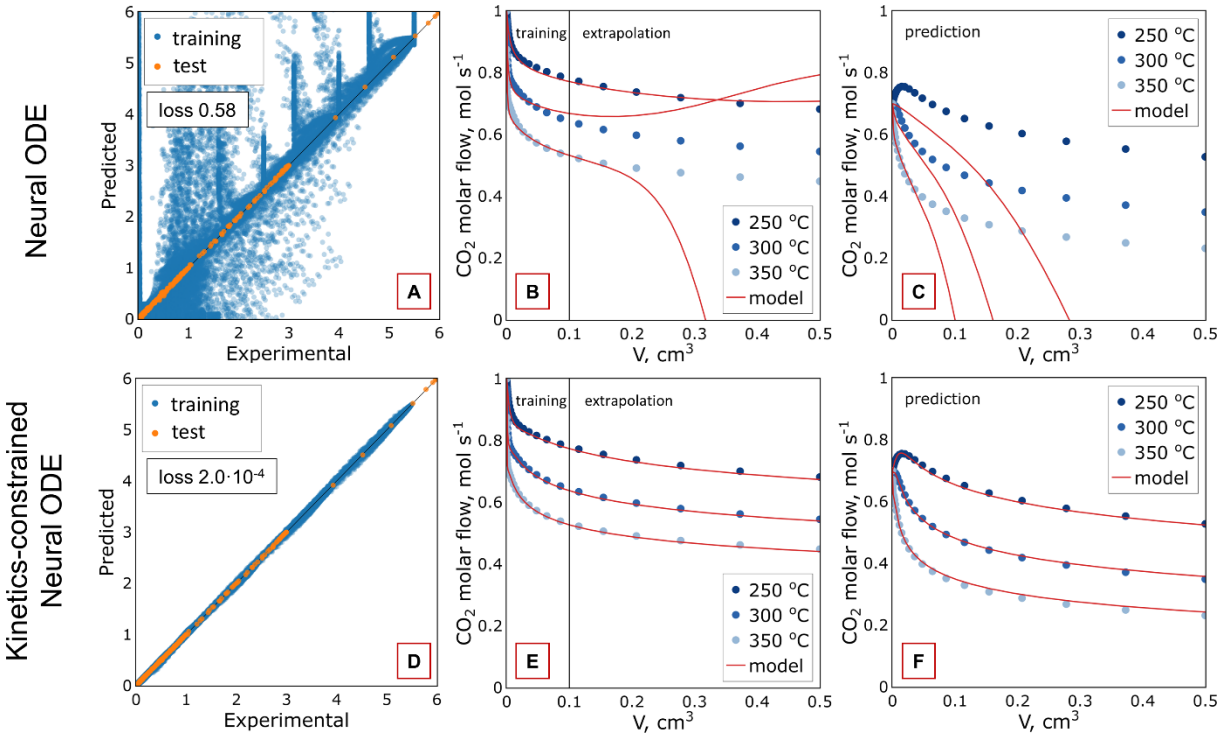

Figure 2. Comparison of the generalization capability between baseline neural ODE (top) and kinetics-constrained neural ODE (bottom) models. **A** and **D**: Parity plots for molar flows of reagents and products. **B** and **E**: Observed (dots) and fitting (line) data of dependencies between the molar flow of CO_2 and the catalyst volume V for training data and their extrapolation (reaction condition: CO_2:H_2:N_2 – 1:3:1, pressure – 10 bar). **C** and **F**: Observed (dots) and fitting (line) data of dependencies between the molar flow of CO_2 and the catalyst volume V for selected test data (reaction condition: CO_2:CO:H_2:H_2O:N_2 – 0.7:0.3:4:0.2:0.5, pressure – 12 bar).

One can see that the baseline model fits the training data correctly which corresponds to the low value of loss function (see Table 2). However, this model performs purely when it was used for predicting and describing data outside of the training range. This is clearly visible in Figure 2 (B-C) where the baseline model was used for extrapolating and fitting the test data set. Accordingly, a high value for the loss function was obtained for the test data test (Table 2). ANN models are known to require a lot of data for training, and the obtained result proves this. Therefore, we can conclude that the standard approach for chemical kinetics modelling using neural ODE cannot be applied when ANN models are trained based on small data set.

To improve the neural ODE models, the kinetic and thermodynamic information about CO_2 hydrogenation to methane was added into the architecture of neural ODE model. From the lower row of plots in Figure 2 one can see that the suggested approach improves the predicting ability of ANN model significantly. With this model, the value of the loss function for the test data set has decreased around 2900 times compared to the baseline model (Table 2). Moreover, the KCNODE model can describe the data outside of the training range correctly that is seen from the presented results. It is also worth noting that the loss function for the training data set in the case of the KCNODE model is lower than in the case of the baseline model (see Table 2). Wherein, the numbers of learnable parameters for the baseline and KCNODE models are 177 and 70, respectively. Thus, an increased number of parameters of the baseline model is not enough to achieve the same low value of the loss function for training data set as in the case of the KCNODE model. It is obvious that an increase in number of parameters for the baseline model will lead only to a further increase in the loss function value of the test data set.

Table 2. Values of loss function of the obtained models for training and test data sets.

	Loss function	
	training data set	test data set
Baseline model	$5.2 \cdot 10^{-4}$	0.58
KCNODE 0 %	$0.4 \cdot 10^{-4}$	$2.0 \cdot 10^{-4}$
KCNODE 2.5 %	$1.6 \cdot 10^{-4}$	$4.6 \cdot 10^{-4}$
KCNODE 5.0 %	$3.1 \cdot 10^{-4}$	$4.0 \cdot 10^{-4}$
KCNODE 10 %	$1.6 \cdot 10^{-3}$	$1.1 \cdot 10^{-3}$
KCNODE 20N*	$0.4 \cdot 10^{-4}$	$2.1 \cdot 10^{-4}$

% means the value of Gauss noise in the training data set; *Kinetics-constrained neural ODE model with 20 nodes in the hidden layer and was trained by ‘no-noise’ data.

It is well-known that real data has experimental errors that can influence the performance of ANN models. This can be especially critical in the case of small data sets. For investigating this influence, KCNODE models were obtained by using the training data sets with added Gauss noise (0-10 %). The loss function values for the models trained on data of various noise levels are presented in the Table 2 (training of the models in Figure S2, parity plots can be found in Figure S3). One can see an increase in the values of the loss function for training and test data sets with increasing noise in the training data, but the generalization of the obtained models is still good (see Figure S3). It is interesting to mention that the obtained

KCNODE models are quite resistant to overfitting. This cannot at all be said about the baseline model. To investigate the resistance of KCNODE model to overfitting in more detail, the model with an increased number of learnable parameters (the number of nodes in the hidden layer was increased from 8 to 20) was trained by using the same training data set (see Table 2). The obtained model has also high predicting ability. Wherein, the number of parameters of the obtained ANN model is 166 and even higher than the number of independent experimental data points (the number of reaction conditions × the number of points of V × the number of reactions = 8·7·2 = 112 < 166). It proves that the suggested architecture of ANN is resistant to overfitting.

Thus, we demonstrated that neural ODE models additionally constrained by adding kinetic and thermodynamic knowledge can be used to successfully model the kinetics of CO_2 hydrogenation to methane using a small data set even in the presence of noise in the data. They described experimental data correctly, wherein, the obtained models do not show a tendency for overfitting during the training. The presence of chemistry knowledge in the architecture of KCNODE model is the reason for the resistance to overfitting. It is obvious that the suggested approach can be also used for kinetic modeling of other chemical reactions. The next part of work is devoted to applying this approach for kinetic modeling CO_2 hydrogenation to hydrocarbons using real experimental data.

Modelling of CO_2 hydrogenation to hydrocarbons

The KCNODE model was applied for kinetic modeling of CO_2 hydrogenation to hydrocarbons. To validate the generalization of KCNODE model, 10-fold cross validation was used. The training runs of the model are presented in Figure S4. One can see that the values of loss function decrease for training and test data sets with increasing the number of epochs and become constant after around 10^4 iterations. The absence of an increase in values of the loss function for the test data sets indicates that the ANN models are not overfitting during training. The mean values of the loss function are $2.5 \cdot 10^{-3}$ and $1.6 \cdot 10^{-2}$ for training and test data sets, respectively. A parity plot comparing experimental and predicted values of molar flows in the test data is presented in Figure 3A. The final KCNODE model was trained by using all the experimental data because the ANN model was shown to be not prone to overfitting.

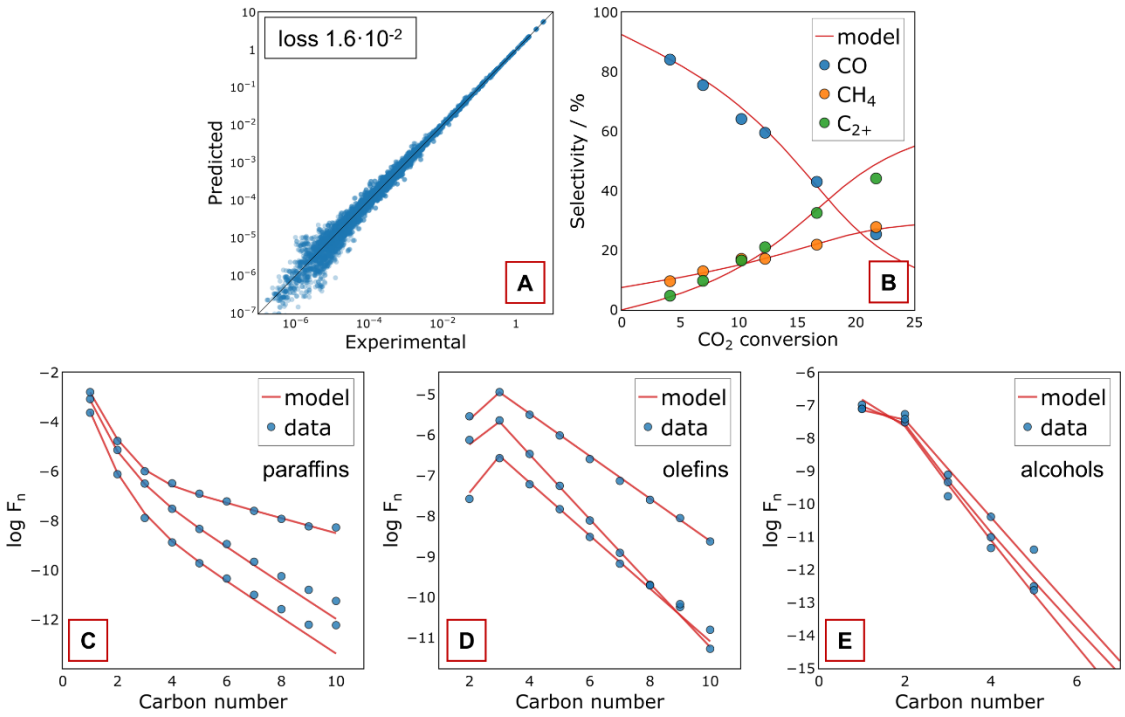

Figure 3. **A**: Parity plot of the KCNODE model for describing CO_2 hydrogenation to hydrocarbons using 10-fold cross validation. **B**: Product selectivity versus CO_2 conversion at reaction condition: CO_2:H_2:N_2 – 1:3:1, pressure – 15 bar temperature – 270 °C. **C**, **D**, **E**: Experimental and fitting data representing products distribution for alkanes, alkenes, and alcohols, respectively.

As seen in Figure 3, the obtained model correctly describes the conversion-selectivity profiles of CO_2 hydrogenation as well as the distribution of the reaction products (alkanes, alkenes, and alcohols). The profiles of the reaction products and reagents at other reaction conditions are also correctly described (see the supplementary material, Figures S5-S8). Thus, the KCNODE was successfully applied for describing the kinetics of CO_2 hydrogenation to hydrocarbons over Fe-based catalysts and was able to predict reactant consumption and products formation as well as the hydrocarbon (and alcohol) distribution.

Kinetic modeling of CO_2 hydrogenation over Fe-containing catalysts is usually based on empirical LHHW models. Riedel et al.[55] were the first who investigated the kinetics of CO_2 hydrogenation and suggested a LHHW model based on RWGS, CO-FT, and CO_2-FT reactions, where only propane was considered as a product of the FT synthesis. In the work[2], Brübach et al. suggested a new LHHW type kinetics expressions for modeling CO_2 hydrogenation, but they also considered only one organic compound (C_4) for describing reaction products. It is noted that the development of models predicting hydrocarbon distribution is a difficult task for kinetic modeling of CO_2 hydrogenation. Panzone et al.[56]

developed a semiempirical macrokinetic model to describe products of CO_2 hydrogenation including hydrocarbon distribution using the ASF distribution. Recently, a microkinetic approach was applied that allowed obtaining the models successfully describing reagent consumption and products formation as well as hydrocarbon distribution of CO_2 hydrogenation that were demonstrated in works.[57, 58] From the presented works one can see the complexity of kinetic modeling of CO_2 hydrogenation to hydrocarbons. In our work we demonstrated that neural KCNODE could be successfully applied for describing the kinetics of this complicated reaction.

It is worth mentioning that the calculational and time costs for building ANN models are significantly lower compared to the development of a kinetics model in the traditional way which requires deep knowledge about mechanism of reactions and great expertise in making appropriate simplifying assumptions, and for the solving inverse kinetic task (Figure 4). In contrast, an easily obtained KCNODE model can already be used for further development, e.g. scale-up related modelling in chemical engineering. However, the KCNODE model can be also used for gaining insights into the underlying chemical processes by generating virtual data. It could be considered as an intermediate model helping to develop a kinetic model based on fundamental mechanistic principles and to elucidate the reaction mechanism.

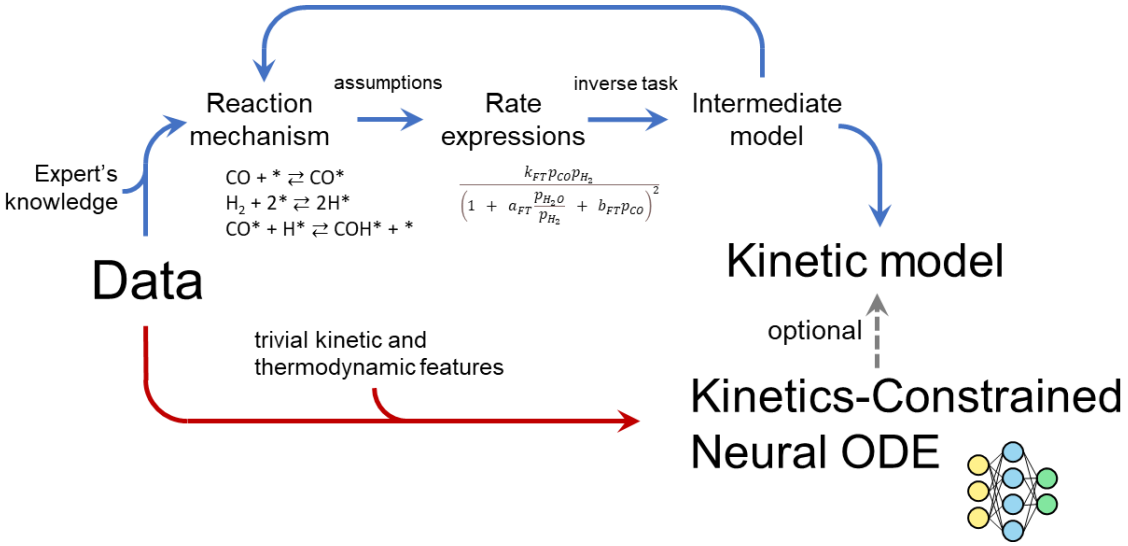

Figure 4. Comparison of traditional and KCNODE-based kinetic modeling.

To demonstrate this procedure of using the KCNEODE as intermediate model, we screened the literature for kinetic models to find the kinetic expression that is best matching the RWGS rate obtained by the developed KCNODE model (see the supplementary material). Based on the results, we suggested the following LHHW expression for describing the rate of RWGS reaction:

1. $$r_{RWGS} = \frac{k\left(p_{CO_2}p_{H_2} - \frac{p_{CO}p_{H_2O}}{K_{eq}}\right)}{\sqrt{p_{H_2}}\left(1 + a\cdot p_{CO} + b\cdot\frac{p_{H_2O}}{\sqrt{p_{H_2}}} + c\cdot p_{CO_2}\right)^2}$$

which is a composition of the models suggested in works.[2, 59] Thus, the KCNODE model was useful as an intermediate model to identify a suitable rate equation for the RWGS reaction. In a similar way, the KCNODE model can be used to develop and test other rate equations that may then be based on fundamental principles. Applying this procedure for other parameters (like r_{FT}, α, etc.) as well as the analysis of the catalyst activation/deactivation of CO_2-FTS was out of scope for the current investigation but would be great of interest. It worth mentioning that the suggested approach can be easily expanded and applied in the kinetic modelling of continuous stirred-tank reactor.

Conclusions

In summary, we have introduced the KCNODE (kinetics-constrained neural ordinary differential equation) approach for describing kinetics of heterogeneous catalytic reactions. The approach is based on constraining the architecture of neural ODE by integrating general knowledge about kinetics and thermodynamics of a catalytic process. A numerical experiment demonstrated that KCNODE model can be used for kinetic modeling of CO_2 hydrogenation to methane using a small data set. The method does work even in presence of noise in the data. The developed approach could also be successfully applied for modeling the kinetics of CO_2 hydrogenation to hydrocarbons based on experimental data. The obtained KCNODE model was able to describe reagent consumption and product formation as well as their distribution correctly. Because the KCNODE models can be trained on small data sets and are resistant to overfitting, the approach can be recommended for the fast development of kinetic models of any catalytic reaction; the KCNODE model can also help in developing kinetic models based on fundamental principles and for elucidating reaction mechanism. Training ANN models on small data does not incur high computational costs which makes the development such ANN approachable for many researchers. The KCNODE significantly expands the scope of usage of neural ODE for describing, analyzing, and developing models based on ordinary differential equations.

Acknowledgements

Financial support from BMBF through the project InnoSyn (FKZ: 03SF0616B) and from German Research Foundation (DFG) through the project NFDI4Cat (DFG no. 441926934) is gratefully acknowledged. The authors thank Dr. Henrik Lund and Dr. Armin Springer for the XRD analysis and recording SEM images as well as Anja Simmula for ICP-OES measurement. The authors also thank Reinhard Eckelt and Dr. Hanan Atia for BET and TPR-H_2 measurement.

CRediT authorship contribution statement

Aleksandr Fedorov: Conceptualization, Investigation, Methodology, Data curation, Writing - original draft, Writing - review & editing. **A. Perechodjuk**: Investigation. **David Linke**: Conceptualization, Supervision, Project administration, Funding acquisition, Writing - review & editing.

Literature

[1] G. Yablonsky, V. Bykov, A. Gorban, V. Elokhin, Kinetic Models of Catalytic Reactions, Elsevier1991.

[2] L. Brübach, D. Hodonj, P. Pfeifer, Kinetic Analysis of CO_2 Hydrogenation to Long-Chain Hydrocarbons on a Supported Iron Catalyst, Industrial & Engineering Chemistry Research 61(4) (2022) 1644-1654. https://doi.org/10.1021/acs.iecr.1c04018.

[3] S. Matera, W.F. Schneider, A. Heyden, A. Savara, Progress in Accurate Chemical Kinetic Modeling, Simulations, and Parameter Estimation for Heterogeneous Catalysis, ACS Catalysis 9(8) (2019) 6624-6647. https://doi.org/10.1021/acscatal.9b01234.

[4] P. Priya, N.R. Aluru, Accelerated design and discovery of perovskites with high conductivity for energy applications through machine learning, npj Computational Materials 7(1) (2021) 90. https://doi.org/10.1038/s41524-021-00551-3.

[5] V. Tshitoyan, J. Dagdelen, L. Weston, A. Dunn, Z. Rong, O. Kononova, K.A. Persson, G. Ceder, A. Jain, Unsupervised word embeddings capture latent knowledge from materials science literature, Nature 571(7763) (2019) 95-98. https://doi.org/10.1038/s41586-019-1335-8.

[6] G.E. Karniadakis, I.G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, L. Yang, Physics-informed machine learning, Nature Reviews Physics 3(6) (2021) 422-440. https://doi.org/10.1038/s42254-021-00314-5.

[7] M. Raissi, P. Perdikaris, 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 (2019) 686-707. https://doi.org/https://doi.org/10.1016/j.jcp.2018.10.045.

[8] M. Bogojeski, L. Vogt-Maranto, M.E. Tuckerman, K.-R. Müller, K. Burke, Quantum chemical accuracy from density functional approximations via machine learning, Nature Communications 11(1) (2020) 5223. https://doi.org/10.1038/s41467-020-19093-1.

[9] H. Gerdes, P. Casado, A. Dokal, M. Hijazi, N. Akhtar, R. Osuntola, V. Rajeeve, J. Fitzgibbon, J. Travers, D. Britton, S. Khorsandi, P.R. Cutillas, Drug ranking using machine learning systematically predicts the efficacy of anti-cancer drugs, Nature Communications 12(1) (2021) 1850. https://doi.org/10.1038/s41467-021-22170-8.

[10] D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton, Y. Chen, T. Lillicrap, F. Hui, L. Sifre, G. van den Driessche, T. Graepel, D. Hassabis, Mastering the game of Go without human knowledge, Nature 550(7676) (2017) 354-359. https://doi.org/10.1038/nature24270.

[11] F. Ashtiani, A.J. Geers, F. Aflatouni, An on-chip photonic deep neural network for image classification, Nature 606(7914) (2022) 501-506. https://doi.org/10.1038/s41586-022-04714-0.

[12] A.V. Fedorov, I.V. Shamanaev, Crystal Structure Representation for Neural Networks using Topological Approach, Molecular Informatics 36(8) (2017) 1600162. https://doi.org/https://doi.org/10.1002/minf.201600162.

[13] J. Burés, I. Larrosa, Organic reaction mechanism classification using machine learning, Nature 613(7945) (2023) 689-695. https://doi.org/10.1038/s41586-022-05639-4.

[14] J.T. Margraf, H. Jung, C. Scheurer, K. Reuter, Exploring catalytic reaction networks with machine learning, Nature Catalysis (2023). https://doi.org/10.1038/s41929-022-00896-y.

[15] K. Hornik, M. Stinchcombe, H. White, Multilayer feedforward networks are universal approximators, Neural Networks 2(5) (1989) 359-366. https://doi.org/https://doi.org/10.1016/0893-6080(89)90020-8.

[16] G. Bebis, M. Georgiopoulos, Feed-forward neural networks, IEEE Potentials 13(4) (1994) 27-31. https://doi.org/10.1109/45.329294.

[17] W. Rawat, Z. Wang, Deep Convolutional Neural Networks for Image Classification: A Comprehensive Review, Neural Computation 29(9) (2017) 2352-2449. https://doi.org/10.1162/neco_a_00990.

[18] S.M. Anwar, M. Majid, A. Qayyum, M. Awais, M. Alnowami, M.K. Khan, Medical Image Analysis using Convolutional Neural Networks: A Review, Journal of Medical Systems 42(11) (2018) 226. https://doi.org/10.1007/s10916-018-1088-1.

[19] J. Zhou, G. Cui, S. Hu, Z. Zhang, C. Yang, Z. Liu, L. Wang, C. Li, M. Sun, Graph neural networks: A review of methods and applications, AI Open 1 (2020) 57-81. https://doi.org/https://doi.org/10.1016/j.aiopen.2021.01.001.

[20] Y. Yu, X. Si, C. Hu, J. Zhang, A Review of Recurrent Neural Networks: LSTM Cells and Network Architectures, Neural Computation 31(7) (2019) 1235-1270. https://doi.org/10.1162/neco_a_01199.

[21] W. Ji, W. Qiu, Z. Shi, S. Pan, S. Deng, Stiff-PINN: Physics-Informed Neural Network for Stiff Chemical Kinetics, The Journal of Physical Chemistry A 125(36) (2021) 8098-8106. https://doi.org/10.1021/acs.jpca.1c05102.

[22] G.S. Gusmão, A.P. Retnanto, S.C.d. Cunha, A.J. Medford, Kinetics-informed neural networks, Catalysis Today (2022). https://doi.org/https://doi.org/10.1016/j.cattod.2022.04.002.

[23] R.T.Q. Chen, Y. Rubanova, J. Bettencourt, D. Duvenaud, Neural ordinary differential equations, (2018). https://doi.org/10.48550/ARXIV.1806.07366.

[24] S. Kim, W. Ji, S. Deng, Y. Ma, C. Rackauckas, Stiff neural ordinary differential equations, Chaos 31(9) (2021) 093122. https://doi.org/10.1063/5.0060697.

[25] O. Owoyele, P. Pal, ChemNODE: A neural ordinary differential equations framework for efficient chemical kinetic solvers, Energy and AI 7 (2022) 100118. https://doi.org/https://doi.org/10.1016/j.egyai.2021.100118.

[26] W. Ji, S. Deng, Autonomous Discovery of Unknown Reaction Pathways from Data by Chemical Reaction Neural Network, The Journal of Physical Chemistry A 125(4) (2021) 1082-1092. https://doi.org/10.1021/acs.jpca.0c09316.

[27] W. Ji, F. Richter, M.J. Gollner, S. Deng, Autonomous kinetic modeling of biomass pyrolysis using chemical reaction neural networks, Combustion and Flame 240 (2022) 111992. https://doi.org/https://doi.org/10.1016/j.combustflame.2022.111992.

[28] W. Yang, T.T. Fidelis, W.-H. Sun, Machine Learning in Catalysis, From Proposal to Practicing, ACS Omega 5(1) (2020) 83-88. https://doi.org/10.1021/acsomega.9b03673.

[29] L. Petzold, Automatic Selection of Methods for Solving Stiff and Nonstiff Systems of Ordinary Differential Equations, SIAM Journal on Scientific and Statistical Computing 4(1) (1983) 136-148. https://doi.org/10.1137/0904010.

[30] J. Antony, 6 - Full Factorial Designs, in: J. Antony (Ed.), Design of Experiments for Engineers and Scientists (Second Edition), Elsevier, Oxford, 2014, pp. 63-85. https://doi.org/https://doi.org/10.1016/B978-0-08-099417-8.00006-7.

[31] K. Lee, E.J. Parish, Parameterized neural ordinary differential equations: applications to computational physics problems, Proc. Math. Phys. Eng. Sci. 477(2253) (2021) 20210162. https://doi.org/10.1098/rspa.2021.0162.

[32] M. Albrecht, U. Rodemerck, M. Schneider, M. Bröring, D. Baabe, E.V. Kondratenko, Unexpectedly efficient CO_2 hydrogenation to higher hydrocarbons over non-doped Fe_2O_3, Applied Catalysis B: Environmental 204 (2017) 119-126. https://doi.org/https://doi.org/10.1016/j.apcatb.2016.11.017.

[33] F. Rahman, K.F. Loughlin, M.A. Al-Saleh, M.R. Saeed, N.M. Tukur, M.M. Hossain, K. Karim, A. Mamedov, Kinetics and mechanism of partial oxidation of ethane to ethylene and acetic acid over MoV type catalysts, Applied Catalysis A: General 375(1) (2010) 17-25. https://doi.org/https://doi.org/10.1016/j.apcata.2009.11.026.

[34] T.P. Otroshchenko, U. Rodemerck, D. Linke, E.V. Kondratenko, Synergy effect between Zr and Cr active sites in binary $CrZrO_x$ or supported $CrO_x/LaZrO_x$: Consequences for catalyst activity, selectivity and durability in non-oxidative propane dehydrogenation, Journal of Catalysis 356 (2017) 197-205. https://doi.org/https://doi.org/10.1016/j.jcat.2017.10.012.

[35] A. Fedorov, H. Lund, V.A. Kondratenko, E.V. Kondratenko, D. Linke, Elucidating reaction pathways occurring in CO_2 hydrogenation over Fe-based catalysts, Applied Catalysis B: Environmental (2023) 122505. https://doi.org/https://doi.org/10.1016/j.apcatb.2023.122505.

[36] B. Yao, T. Xiao, O.A. Makgae, X. Jie, S. Gonzalez-Cortes, S. Guan, A.I. Kirkland, J.R. Dilworth, H.A. Al-Megren, S.M. Alshihri, P.J. Dobson, G.P. Owen, J.M. Thomas, P.P. Edwards, Transforming carbon dioxide into jet fuel using an organic combustion-synthesized Fe-Mn-K catalyst, Nature Communications 11(1) (2020) 6395. https://doi.org/10.1038/s41467-020-20214-z.

[37] J. Wei, Q. Ge, R. Yao, Z. Wen, C. Fang, L. Guo, H. Xu, J. Sun, Directly converting CO_2 into a gasoline fuel, Nature Communications 8(1) (2017) 15174. https://doi.org/10.1038/ncomms15174.

[38] A.S. Skrypnik, Q. Yang, A.A. Matvienko, V.Y. Bychkov, Y.P. Tulenin, H. Lund, S.A. Petrov, R. Kraehnert, A. Arinchtein, J. Weiss, A. Brueckner, E.V. Kondratenko, Understanding reaction-induced restructuring of well-defined $Fe_xO_yC_z$ compositions and

its effect on CO_2 hydrogenation, Applied Catalysis B: Environmental 291 (2021) 120121. https://doi.org/https://doi.org/10.1016/j.apcatb.2021.120121.

[39] J. Zhu, G. Zhang, W. Li, X. Zhang, F. Ding, C. Song, X. Guo, Deconvolution of the Particle Size Effect on CO_2 Hydrogenation over Iron-Based Catalysts, ACS Catalysis 10(13) (2020) 7424-7433. https://doi.org/10.1021/acscatal.0c01526.

[40] G.V. Schulz, Über die Beziehung zwischen Reaktionsgeschwindigkeit und Zusammensetzung des Reaktionsproduktes bei Makropolymerisationsvorgängen, Zeitschrift für Physikalische Chemie 30B(1) (1935) 379-398. https://doi.org/doi:10.1515/zpch-1935-3027.

[41] P.J. Flory, Molecular Size Distribution in Linear Condensation Polymers1, Journal of the American Chemical Society 58(10) (1936) 1877-1885. https://doi.org/10.1021/ja01301a016.

[42] R.B. Anderson, R.A. Friedel, H.H. Storch, Fischer-Tropsch Reaction Mechanism Involving Stepwise Growth of Carbon Chain, The Journal of Chemical Physics 19(3) (1951) 313-319. https://doi.org/10.1063/1.1748201.

[43] G. Henrici-Olivé, S. Olivé, The Fischer-Tropsch Synthesis: Molecular Weight Distribution of Primary Products and Reaction Mechanism, Angewandte Chemie International Edition in English 15(3) (1976) 136-141. https://doi.org/https://doi.org/10.1002/anie.197601361.

[44] L.S. Glebov, G.A. Kliger, The molecular weight distribution of the products of the Fischer-Tropsch synthesis, Russian Chemical Reviews 63(2) (1994) 185-195. https://doi.org/10.1070/rc1994v063n02abeh000079.

[45] M. Poli, S. Massaroli, A. Yamashita, H. Asama, J. Park, TorchDyn: A neural differential equations library, (2020). https://doi.org/10.48550/ARXIV.2009.09346.

[46] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, (2014). https://doi.org/10.48550/ARXIV.1412.6980.

[47] S. Kim, W. Ji, S. Deng, Y. Ma, C. Rackauckas, Stiff neural ordinary differential equations, Chaos: An Interdisciplinary Journal of Nonlinear Science 31(9) (2021) 093122. https://doi.org/10.1063/5.0060697.

[48] G. Van Rossum, F.L. Drake, Python 3 Reference Manual, CreateSpace.

[49] C.R. Harris, K.J. Millman, S.J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N.J. Smith, R. Kern, M. Picus, S. Hoyer, M.H. van Kerkwijk, M. Brett, A. Haldane, J.F. del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, T.E. Oliphant, Array

programming with NumPy, Nature 585(7825) (2020) 357-362. https://doi.org/10.1038/s41586-020-2649-2.

[50] P. Virtanen, R. Gommers, T.E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S.J. van der Walt, M. Brett, J. Wilson, K.J. Millman, N. Mayorov, A.R.J. Nelson, E. Jones, R. Kern, E. Larson, C.J. Carey, İ. Polat, Y. Feng, E.W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E.A. Quintero, C.R. Harris, A.M. Archibald, A.H. Ribeiro, F. Pedregosa, P. van Mulbregt, A. Vijaykumar, A.P. Bardelli, A. Rothberg, A. Hilboll, A. Kloeckner, A. Scopatz, A. Lee, A. Rokem, C.N. Woods, C. Fulton, C. Masson, C. Häggström, C. Fitzgerald, D.A. Nicholson, D.R. Hagen, D.V. Pasechnik, E. Olivetti, E. Martin, E. Wieser, F. Silva, F. Lenders, F. Wilhelm, G. Young, G.A. Price, G.-L. Ingold, G.E. Allen, G.R. Lee, H. Audren, I. Probst, J.P. Dietrich, J. Silterra, J.T. Webber, J. Slavič, J. Nothman, J. Buchner, J. Kulick, J.L. Schönberger, J.V. de Miranda Cardoso, J. Reimer, J. Harrington, J.L.C. Rodríguez, J. Nunez-Iglesias, J. Kuczynski, K. Tritz, M. Thoma, M. Newville, M. Kümmerer, M. Bolingbroke, M. Tartre, M. Pak, N.J. Smith, N. Nowaczyk, N. Shebanov, O. Pavlyk, P.A. Brodtkorb, P. Lee, R.T. McGibbon, R. Feldbauer, S. Lewis, S. Tygier, S. Sievert, S. Vigna, S. Peterson, S. More, T. Pudlik, T. Oshima, T.J. Pingel, T.P. Robitaille, T. Spura, T.R. Jones, T. Cera, T. Leslie, T. Zito, T. Krauss, U. Upadhyay, Y.O. Halchenko, Y. Vázquez-Baeza, C. SciPy, SciPy 1.0: fundamental algorithms for scientific computing in Python, Nature Methods 17(3) (2020) 261-272. https://doi.org/10.1038/s41592-019-0686-2.

[51] W. McKinney, others, Data structures for statistical computing in python, Austin, TX, pp. 51-56.

[52] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: Machine Learning in Python, Journal of Machine Learning Research 12 2825-2830.

[53] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, S. Chintala, PyTorch: An Imperative Style, High-Performance Deep Learning Library, Advances in Neural Information Processing Systems 32, Curran Associates Inc.2019, pp. 8024--8035.

[54] P.T. Inc., Collaborative data science, (2015).

[55] T. Riedel, G. Schaub, K.-W. Jun, K.-W. Lee, Kinetics of CO_2 Hydrogenation on a K-Promoted Fe Catalyst, Industrial & Engineering Chemistry Research 40(5) (2001) 1355-1363. https://doi.org/10.1021/ie000084k.

[56] C. Panzone, R. Philippe, C. Nikitine, L. Vanoye, A. Bengaouer, A. Chappaz, P. Fongarland, Catalytic and Kinetic Study of the CO_2 Hydrogenation Reaction over a Fe–K/Al_2O_3 Catalyst toward Liquid and Gaseous Hydrocarbon Production, Industrial & Engineering Chemistry Research 60(46) (2021) 16635-16652. https://doi.org/10.1021/acs.iecr.1c02542.

[57] C. Panzone, R. Philippe, C. Nikitine, A. Bengaouer, A. Chappaz, P. Fongarland, Development and Validation of a Detailed Microkinetic Model for the CO_2 Hydrogenation Reaction toward Hydrocarbons over an Fe–K/Al_2O_3 Catalyst, Industrial & Engineering Chemistry Research 61(13) (2022) 4514-4533. https://doi.org/10.1021/acs.iecr.1c04672.

[58] L. Brübach, D. Hodonj, L. Biffar, P. Pfeifer, Detailed kinetic modeling of CO_2-based Fischer–Tropsch synthesis, Catalysts 12(6) (2022) 630. https://doi.org/10.3390/catal12060630.

[59] A.A. Hakeem, M. Li, R.J. Berger, F. Kapteijn, M. Makkee, Kinetics of the high temperature water–gas shift over Fe_2O_3/ZrO_2, Rh/ZrO_2 and Rh/Fe_2O_3/ZrO_2, Chemical Engineering Journal 263 (2015) 427-434. https://doi.org/https://doi.org/10.1016/j.cej.2014.10.104.