Neural Comput & Applic (2009) 18:15–24
DOI 10.1007/s00521-008-0200-8

IJCNN 2007

Artificial neural networks for modeling in reaction process systems

Cristina Oliveira · Petia Georgieva · Fernando Rocha · Sebastião Feyo de Azevedo

Received: 9 March 2008 / Accepted: 14 August 2008 / Published online: 16 September 2008

Abstract This work is focused on developing a more efficient computational scheme for estimation of process reaction rates based on NN models. Two scenarios are considered: (1) the kinetics coefficients of the process are completely known and the process states are partly known (measured); (2) the kinetics coefficients and the states of the process are partly known. The contribution of the paper is twofold. From one side we formulate a hybrid (ANN and mechanistic) model that outperforms the traditional reaction rate estimation approaches. From other side, a new procedure for NN supervised training is proposed when target outputs are not available. The two scenarios are successfully tested for two benchmark problems, estimation of the precipitation rate of calcium phosphate and estimation of sugar crystallization growth rate.

Keywords Neural network computational models · Reaction rate estimation · State observer

1 Introduction

The dynamics of biochemical reaction processes are usually described by mass and energy balance differential equations. These equations combine two elements, the phenomena of conversion of one reaction component into another (i.e. the reaction kinetics) and the transport dynamics of the components through the reactor. The identification of such mathematical models from experimental input/output data is still a challenging issue due to the inherent nonlinearity and complexity of this class of processes. The most difficult problem is how to model the reaction kinetics and more particularly, the reaction rates. The traditional way is to estimate the reaction rates in the form of analytical expressions [1, 2]. First, the parameterized structure of the reaction rate is determined based on data obtained by specially designed experiments. Then the respective parameters of this structure are estimated. Reliable parameter estimation is only possible if the proposed model structure is correct and theoretically identifiable [3]. Therefore, the reaction rate analytical structure is usually determined after a huge number of expensive laboratory experiments. It is further assumed that the initial values of the identified parameters are close to the real process parameters [4], which is typically satisfied only for well known processes. The above considerations motivated a search for alternative estimation solutions based on computationally more attractive paradigms as are the neural networks (NN).

Over the last 20 years, the NNs became a well-established methodology as exemplified by their applications to identification and control of nonlinear systems [5, 6]. Their increasing use not only in modeling and control, but also in various fields including pattern recognition, feature extraction, classification, speech and vision, is in great part due to their universal approximation properties and generalization ability with respect to unknown data [7]. Moreover, the NNs can learn online, based on local information, can treat easily multi input multi output (MIMO) systems and their performance degrades gracefully due to the parallel, distributed processing structure and high degree of connectivity among the units. All these remarkable biologically inspired

C. Oliveira · F. Rocha · S. Feyo de Azevedo
Department of Chemical Engineering, Faculty of Engineering, University of Porto, 4200-465 Porto, Portugal

P. Georgieva (✉)
Department of Telecommunications Electronics and Informatics /IEETA, University of Aveiro, 3810-193 Aveiro, Portugal
e-mail: petia@ua.pt

properties made the NNs successful alternative for dynamic modeling and estimation.

In this paper, instead of an exhaustive search for the most appropriate parameterized reaction rate structure, a NN is applied to estimate the reaction rates. In the next section, a hybrid model of a general reaction process is introduced, with a NN as a reaction rate submodel in the framework of a nonlinear state space analytical process model. In Sect. 3, a systematic NN training procedure is formulated assuming that all kinetics coefficients are available but not all process states are measured. In Sect. 4, a similar procedure is developed assuming that the kinetics coefficients and the states of the process are partly known. The new approach is tested for two benchmark problems, estimation of the precipitation rate of calcium phosphate (in Sect. 5) and estimation of sugar crystallization growth rate (in Sect. 6). Conclusions are drawn in the last section.

2 Knowledge-based hybrid model (KBHM)

The generic class of reaction systems can be described by the following equations [1]

$$\frac{dX}{dt} = K\varphi(X,T) + DX + F_{in} - F_{out} \quad (1.1)$$

$$\frac{dT}{dt} = K_0\varphi(X,T) - D_0T + F_{in} \quad (1.2)$$

where, for $n, m \in N$, the constants and variables denote

$X = (x_1(t),\ldots, x_n(t))$	concentrations of n reaction components
$K = [k_1,\ldots, k_m] \in R^{n\times m}$	$n \times m$ kinetics coefficients
$\varphi = (\varphi_1, \ldots\varphi_m)^T$	reaction rates of m reactions
$F_{in} = (F_1^{in}(t), \ldots F_l^{in}(t))$	flow rates of l feeding flows
$F_{out} = (F_1^{out}(t), \ldots F_p^{out}(t))$	p effluent flow rates (final product, purified liquid, etc.)
T/K_0	reactor temperature/exothermic coefficient
D	dilution rate (i.e. influent flow rate)/(volume of medium in the reactor)
D_0	heat transfer rate

The nonlinear state-space model (1) proved to be the most suitable form of representing several industrial processes as crystallization and precipitation, polymerization reactors, distillation columns, biochemical fermentation and biological systems. The reaction kinetics or the rate of a reaction (φ_i) is the rate of mass consumption or production of that component per unit of time. It is usually time varying and dependent of the stage of the process. The reaction rates are the key factor for reliable description of the concentrations $x_i(t)$. In this work, a feedforward neural network (FFNN) is chosen to estimate the reaction rate vector. The NN submodel is incorporated in the general dynamical model (1) and the mixed structure is termed knowledge-based hybrid model (KBHM), see Fig. 1. A systematic procedure for NN-based estimation of reaction rates is discussed in the next section.

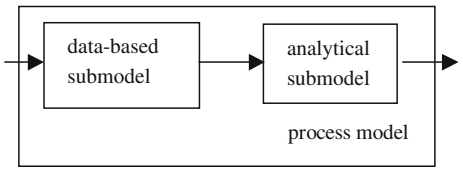


Fig. 1 Knowledge-based hybrid model (KBHM)

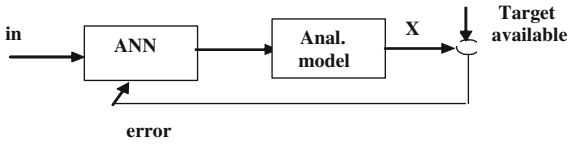


Fig. 2 Hybrid NN training structure

3 NN reaction rate estimation with known kinetics coefficients

The main challenge for the application of any data-based modeling technique is that the process reaction rates are not measured variables. The NN supervised learning paradigm is based on error correction principle and in order to update the network weights, an error signal between the network output and the corresponding target (reference output) is required. However, for the reaction rates, targets are not available. Therefore, an alternative NN training procedure is proposed in this work. The idea is to propagate the network output through the fixed analytical model until it comes to an output for which data are available (see Fig. 2). The proper choice of the partial analytical model and the formulation of the error signal for NN updating are discussed below. The procedure is based on the following assumptions:

(A1) Not all process states of model (1) are measured.

(A2) All kinetics coefficients are known, that is K_0 and all entries of matrix K are available.

For more convenience, the model (1) is reformulated based on the following augmented vectors

$$X_{aug} = \begin{bmatrix} X \\ T \end{bmatrix}, \quad X_{aug} \in R^{n+1}, \quad K_{aug} = \begin{bmatrix} K \\ K_0 \end{bmatrix},$$

$K_{aug} \in R^{(n+1)\times m}$.

Then (1) is rewritten as

$$\frac{dX_{aug}}{dt} = K_{aug}\varphi(X_{aug}) - DX_{aug} + F_{in} - F_{out}. \quad (2)$$

3.1 Step 1: state vector partition A

The general dynamical model (2) represents a particular class of nonlinear state-space models. The nonlinearity lies in the reaction rates $\varphi(X_{aug})$ that are nonlinear functions of the state variables X_{aug}. These functions enter the model in the form $K_{aug}\varphi(X_{aug})$, where K_{aug} is a constant matrix, which is a set of linear combinations of the same nonlinear functions $\varphi_1(X_{aug}),\ldots\varphi_m(X_{aug})$. This particular structure can be exploited to separate the nonlinear part from the linear part of the model by a suitable linear state transformation. More precisely, the following nonsingular partition is chosen [8]

$$LK_{aug} = \begin{bmatrix} K_a \\ K_b \end{bmatrix}, \quad \text{rank}(K_{aug}) = l, \quad (3.1)$$

where $L \in R^{n\times n}$ is a quadratic permutation matrix, K_a is a $l \times m$ full row rank sub-matrix of K_{aug} and $K_b \in R^{(n-l)\times m}$. The induced partitions of the vectors X_{aug}, F_{in} and F_{out} are

$$LX_{aug} = \begin{bmatrix} X_a \\ X_b \end{bmatrix}, \quad LF_{in} = \begin{bmatrix} F_{in\_a} \\ F_{in\_b} \end{bmatrix}, \quad LF_{out} = \begin{bmatrix} F_{out\_a} \\ F_{out\_b} \end{bmatrix}, \quad (3.2)$$

with $X_a \in R^l, F_{in\_a} \in R^l, F_{out\_a} \in R^l, X_b \in R^{n-l}, F_{in\_b} \in R^{n-l}$, $F_{out\_b} \in R^{n-l}$.

According to (3), model (2) is also partitioned into two submodels

$$\frac{dX_a}{dt} = K_a\varphi(X_a, X_b) - DX_a + F_{in\_a} - F_{out\_a} \quad (4)$$

$$\frac{dX_b}{dt} = K_b\varphi(X_a, X_b) - DX_b + F_{in\_b} - F_{out\_b} \quad (5)$$

Based on (3), a new vector $Z \in R^{n+1-l}$ is defined as a linear combination of the state variables

$$Z = A_0X_a + X_b, \quad (6)$$

where matrix $A_0 \in R^{(n+1-l)\times l}$ is the unique solution of

$$A_0K_a + K_b = 0,$$

that is

$$A_0 = -K_bK_a^{-1}, \quad (7)$$

Note that, a solution for A_0 exist if and only if K_a is not singular. Thus, a necessary and sufficient condition for the existence of a desired partition (3), is that K_a is a $p \times m$ full rank matrix, which was the initial assumption. Then, the first derivative of vector Z is

$$\begin{aligned}\frac{dZ}{dt} = A_0\frac{dX_a}{dt} + \frac{dX_b}{dt} &= A_0[K_a\varphi(X_a, X_b) \\ &\quad -DX_a + F_{in\_a} - F_{out\_a}] \\ &\quad + K_b\varphi(X_a, X_b) - DX_b + F_{in\_b} \\ &\quad - F_{out\_b} \\ &= (A_0K_a + K_b)\varphi(X_a, X_b) \\ &\quad - D(A_0X_a + X_b) \\ &\quad + A_0(F_{in\_a} - F_{out\_a}) \\ &\quad + (F_{in\_b} - F_{out\_b}) \quad (8)\end{aligned}$$

Since matrix A_0 is chosen such that (7) holds, the term in (8) related with φ is cancelled and we get

$$\frac{dZ}{dt} = -DZ + A_0(F_{in\_a} - F_{out\_a}) + (F_{in\_b} - F_{out\_b}) \quad (9)$$

The state partition A results in a vector Z whose dynamics, given by (9), is independent of the reaction rate vector φ. In general, (3) is not a unique partition and for any particular case a number of choices are possible.

3.2 Step 2: state vector partition B (measured and unmeasured states)

Now a new state partition is defined as sub-vectors of measured and unmeasured states X_1, X_2, respectively. The model (2) is also partitioned into two submodels

$$\frac{dX_1}{dt} = K_1\varphi(X_1, X_2) - DX_1 + F_{in\_1} - F_{out\_1} \quad (10.1)$$

$$\frac{dX_2}{dt} = K_2\varphi(X_1, X_2) - DX_2 + F_{in\_2} - F_{out\_2} \quad (10.2)$$

From state partitions A and B, vector Z can be represented in the following ways

$$Z = A_0X_a + X_b = A_1X_1 + A_2X_2. \quad (11)$$

The first representation is determined in (6), then applying linear algebra transformations A_1 and A_2 are computed to fit the equality (11). The purpose of the two state partitions A and B is to get estimates of the unmeasured states (vector X_2 independently of the reaction rates (vector φ). The procedure for recovery of X_2 is termed state observer and is discussed below.

3.3 Step 3: state observer

Based on (9) and (11), the unmeasured states observer is the following [1]

$$\frac{d\hat{Z}}{dt} = -D\hat{Z} + A_0(F_{in\_a} - F_{out\_a}) + (F_{in\_b} - F_{out\_b}) \quad (12)$$

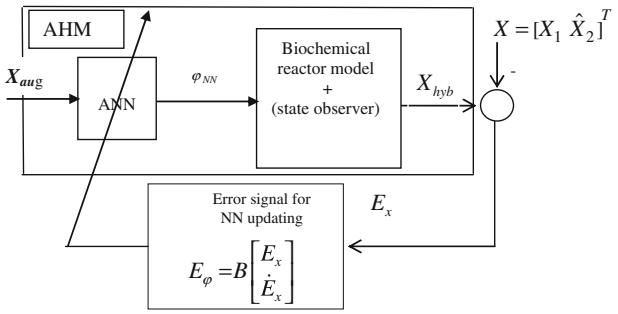


Fig. 3 Hybrid NN-based reaction rates identification structure

$$\hat{X}_2 = A_2^{-1}(\hat{Z} - A_1 X_1) \tag{13}$$

where $\hat{Z}$ and $\hat{X}_2$ denote estimates of Z and X_2, respectively. Starting with some initial conditions, Z is easily estimated by (12), since all other terms are known. Then the estimation of X_2 is straightforward by solving the algebraic equation (13). Note that, estimates $\hat{X}_2$ exist if and only if A_2 is not singular. Therefore, a necessary and sufficient condition for unmeasured states observability is that A_2 is a full rank matrix.

3.4 Step 4: error signal for NN training

The hybrid structure for NN training is shown in Fig. 3, where the adaptive hybrid model (AHM) is formulated as

$$\frac{\mathrm{d}X_{\mathrm{hyb}}}{\mathrm{d}t} = K_{\mathrm{aug}}\varphi_{\mathrm{NN}} - DX_{\mathrm{hyb}} + F_{\mathrm{in}} - F_{\mathrm{out}} + \Omega(X_{\mathrm{aug}} - X_{\mathrm{hyb}}) \tag{14}$$

The true (but unknown) process behavior is assumed to be represented by (2). Then the error dynamics is modeled as the difference between (2) and (14)

$$\frac{\mathrm{d}(X_{\mathrm{aug}} - X_{\mathrm{hyb}})}{\mathrm{d}t} = K_{\mathrm{aug}}(\varphi - \varphi_{\mathrm{NN}}) - D(X_{\mathrm{aug}} - X_{\mathrm{hyb}}) + \Omega(X_{\mathrm{aug}} - X_{\mathrm{hyb}}) \tag{15}$$

The following definitions take place:

$E_x = (X_{\mathrm{aug}} - X_{\mathrm{hyb}})$ is termed as the observation error, $E_\varphi = \varphi - \varphi_{NN}$ is the error signal for updating the NN parameters.

X_{aug} consists of the measured (X_1) and the estimated $(\hat{X}_2)$ states. Thus, (15) can be rearranged as follows

$$\frac{\mathrm{d}E_x}{\mathrm{d}t} = K_{\mathrm{aug}}E_\varphi - (D - \Omega)E_x \tag{16}$$

and from (16) the error signal for NN training is

$$E_\varphi = K_{\mathrm{aug}}^{-1}[D - \Omega \quad 1]\begin{bmatrix} E_x \\ \dot{E}_x \end{bmatrix} = B\begin{bmatrix} E_x \\ \dot{E}_x \end{bmatrix} \tag{17}$$

Ω is a design parameter which defines the speed of the observation error convergence. Note that the necessary identifiability condition for the reaction rate vector is the non-singularity of matrix K_{aug}.

The cost function to be minimized at each iteration of network training is the sum of squared errors, where N is the time instants over which the optimization is performed (batch mode of training)

$$J = \frac{1}{N}\sum_{k=1}^{N}\left[E_\varphi(k)\right]^2$$

The network parameters (W^{NN}) are updated applying a gradient method such that

$$W_{\mathrm{new}}^{\mathrm{NN}} = W_{\mathrm{old}}^{\mathrm{NN}} + f\left(\frac{\partial J}{\partial W_{\mathrm{old}}^{\mathrm{NN}}}\right) \tag{18}$$

There are a number of algorithms to determine $f\left(\frac{\partial J}{\partial W_{\mathrm{old}}^{\mathrm{NN}}}\right)$. In our study, the Levenberg–Marquardt back-propagation is the preferred algorithms [9], which is a version of Newton method. However, alternative methods can be also applied. Note that, the error signal for updating the network parameters is a function of the observation error (E_x) and the speed of the observation error $(\dot{E}_x)$. The intuition behind is that the network parameters are changed proportionally to their effect on the prediction of the process states and the prediction of their dynamics.

4 NN reaction rate estimation with partly unknown kinetics coefficients

In this scenario, the following assumptions hold true:

(A1) Not all process states of model (2) are measured.
(A3) Not all kinetics coefficients are known (not all entries of K_{aug} are available).

The states partition A, in the previous section, made possible estimation of the unmeasured states independently and prior to the network training. This enabled the batch mode of NN training providing parameter adaptation only after the complete data set has been processed and the respective errors computed. However, in the present case, when the kinetics coefficients (K_{aug}) are partly unknown, the unmeasured states (X_2) has to be estimated jointly with the estimation of K_{aug} and the estimation of the reaction rates (φ_{NN}). Therefore, the network training is performed simultaneously with the estimation of X_2 and K_{aug}. As a result, the sequential (online) mode of NN training is the only possible alternative. The procedure for NN reaction rate estimation can be summarized as follows:

1. Choose initial guess for the unknown kinetics coefficients $\hat{K}_{\mathrm{estim}}$: $\hat{K}_{\mathrm{aug}} = \begin{bmatrix} K_{\mathrm{known}} & \hat{K}_{\mathrm{estim}} \end{bmatrix}$
2. Choose an appropriate state partition A: $L\hat{K}_{\mathrm{aug}} = \begin{bmatrix} \hat{K}_a \\ \hat{K}_b \end{bmatrix}$, and compute $\hat{A}_0 = -\hat{K}_b\hat{K}_a^{-1}$

3. Define the state partition B: $Z = \hat{A}_0 X_a + X_b = \hat{A}_1 X_1 + \hat{A}_2 X_2$ and compute matrices $\hat{A}_1$and $\hat{A}_2$.
4. Estimate the unmeasured states (X_2) as

$$\frac{\mathrm{d}\hat{Z}}{\mathrm{d}t} = -D\hat{Z} + \hat{A}_0(F_{\mathrm{in\_}a} - F_{\mathrm{out\_}a}) + (F_{\mathrm{in\_}b} - F_{\mathrm{out\_}b}) \tag{19}$$

$$\hat{X}_2 = \hat{A}_2^{-1}(\hat{Z} - A_1 X_1) \tag{20}$$

5. Provide the NN with input data (the measured and the estimated states), compute the network output (φ_{NN}), the error signal for NN training (E_φ) and update the network parameters by (18).
6. Update the values of $\hat{K}_{\mathrm{aug}}$ by

$$\frac{\mathrm{d}\hat{K}_{\mathrm{estim}}}{\mathrm{d}t} = \Gamma(X_1 - \hat{X}_1), \tag{21}$$

where

$$\frac{\mathrm{d}\hat{X}_1}{\mathrm{d}t} = \hat{K}_1\varphi_{\mathrm{NN}} - D\hat{X}_1 + F_{\mathrm{in\_}1} - F_{\mathrm{out\_}1} - \Omega(X_1 - \hat{X}_1) \tag{22}$$

and go back to step 2 or stop if the quality criteria is satisfied.

Equation (22) controls the convergence of kinetics coefficients estimation. It is a copy of the model (10.1) with an extra term, proportional to the error between the measured (X_1) and the estimated state $\left(\hat{X}_1\right)$. In case of perfect estimation this term is equal to zero. Ω and Γ are design parameters that can significantly speed up the adaptation procedure.

4.1 Error signal for NN training

The adaptive hybrid model (AHM) is formulated as

$$\frac{\mathrm{d}X_{\mathrm{hyb}}}{\mathrm{d}t} = \hat{K}_{\mathrm{aug}}\varphi_{\mathrm{NN}} - DX_{\mathrm{hyb}} + F_{\mathrm{in}} - F_{\mathrm{out}} + \Omega(X_{\mathrm{aug}} - X_{\mathrm{hyb}}) \tag{24}$$

Subtracting (2) from (24), we obtain the error dynamics

$$\frac{\mathrm{d}E_x}{\mathrm{d}t} = K_{\mathrm{aug}}\varphi - \hat{K}_{\mathrm{aug}}\varphi_{\mathrm{NN}} + (\Omega - D)E_x. \tag{25}$$

Define an extended error signal for NN training ($E_{K\varphi}$) as follows:

$$E_{K\phi} = K_{\mathrm{aug}}\varphi - \hat{K}_{\mathrm{aug}}\varphi_{\mathrm{NN}} \tag{26}$$

Rearranging (15), we get the following error signal $E_{K\varphi}$ used for NN training

$$E_{K\varphi} = B\begin{bmatrix} E_x \\ \dot{E}_x \end{bmatrix}, \quad B = [\,D - \Omega \quad 1\,]. \tag{27}$$

The intuition behind (27) is that the network adaptation depends on the reliable estimation of $\hat{K}_{\mathrm{aug}}$. If $K_{\mathrm{aug}} = \hat{K}_{\mathrm{aug}}$the error reduces to (17)

$$E_\varphi = \varphi - \varphi_{\mathrm{NN}} = \tilde{B}\begin{bmatrix} E_x \\ \dot{E}_x \end{bmatrix}, \quad \tilde{B} = K_{\mathrm{aug}}^{-1}[\,D - \Omega \quad 1\,]$$

The procedures presented in Sects. 3 and 4 are further tested for two case studies and compared with alternative solutions.

Remark The structural identifiably of the kinetics coefficients is comprehensively studied in Chen and Bastin [8]. It is proved that the identifiability properties of matrix K_{aug} are invariant with respect to the choice of partition A. The authors define necessary condition according to which if the rank of K_{aug} is l at most $n - l$ coefficients in each reaction can be identified without the knowledge of the reaction rates.

5 Case study 1: estimation of the precipitation rate of calcium phosphate with completely known kinetics coefficients

The precipitation of calcium phosphate was studied by many authors under different conditions [10, 11]. Depending on the temperature, the level of supersaturation, pH and initial concentration of reagents, one can obtain different calcium phosphate phases. One of them is the dicalcium phosphate dehydrate (DCPD) known also as brushite. DCPD is recognized as an important product in the application of fertilizers to soil and is studied mainly for its role in the physiological formation of calcium phosphates.

For the present study, the precipitation of DCPD was performed in a batch laboratory crystallizer. The precipitation was carried out by mixing equimolar quantities of calcium hydroxide suspension and orthophosphoric acid solution. Five successive stages were identified during a number of experiments performed with different initial reagent concentrations [11]: (1) spontaneous precipitation of hydroxyapatite (HAP); (2) complete dissolution of calcium and HAP growth; (3) appearance of first nuclei of brushite; (4) coexistence of HAP and brushite and (5) transformation of HAP into brushite and growth of brushite. The last stage represents the main challenge with respect to the precipitation rate modeling because it has to take simultaneously into account two kinetics phenomena: brushite grows first, due to direct consumption of calcium in the solution and second, due to the transformation of HAP into brushite.

According to the general dynamical model (2), the state space representation of the last stage, namely the transformation of HAP into brushite and growth of brushite, is

$$\frac{\mathrm{d}M_c}{\mathrm{d}t} = -q_{m1}\varphi(M_{\mathrm{c}}, M_{\mathrm{B}}) \tag{28.1}$$

$$\frac{\mathrm{d}M_{\mathrm{HAP}}}{\mathrm{d}t} = -K_{\mathrm{HAP}}(M_{\mathrm{HAP}})^2 \tag{28.2}$$

$$\frac{dM_B}{dt} = \varphi(M_c, M_B) + 10q_{m2}K_{HAP}(M_{HAP})^2, \quad (28.3)$$

where M_c is the mass of calcium into solution, M_{HAP} is the mass of HAP and M_B is the mass of brushite. $\varphi(\cdot)$ is the precipitation rate, K_{HAP}, q_{m1}, q_{m2} are kinetics coefficients. Note that M_{HAP} is not a function of φ, therefore only M_c (the measured state) and M_B (the unmeasured state) explicitly depend on the precipitation rate. In this very simple case, the state partitions A and B are equal

$$X_a = X_1 = M_c, \quad X_b = X_2 = M_B, \quad (29)$$

with

$$K_a = -q_{m1}, \quad K_b = 1 \quad (30)$$

$$A_0 = A_1 = 1/q_{m1} \quad A_2 = 1. \quad (31)$$

According to (12–13), the state observer of M_B is the following

$$\frac{d\hat{M}_{HAP}}{dt} = -K_{HAP}(M_{HAP})^2 \quad (32)$$

$$\frac{d\hat{Z}}{dt} = 10q_{m2}K_{HAP}\left(\hat{M}_{HAP}\right)^2, \quad (33)$$

$$\hat{M}_B = \hat{Z} - \frac{1}{q_{m1}}M_c. \quad (34)$$

The observation error is

$$E_x = \begin{bmatrix} M_c - M_{chyb} \\ \hat{M}_B - M_{Bhyb} \end{bmatrix}, \quad (35)$$

with respective matrices

$$D = 0, \quad \Omega = \begin{bmatrix} -\omega_1 & 0 \\ 0 & -\omega_2 \end{bmatrix}. \quad (36)$$

For the current study, the value of $\omega_1 = \omega_2 = 0.5$ was chosen after trial and error. Finally, the NN error is

$$E_\varphi = \frac{1}{q_{m1}}\begin{bmatrix} \left(\dot{M}_c - \dot{M}_{chyb}\right) - \omega_1\left(M_c - M_{chyb}\right) \\ \left(\hat{M}_B - \dot{M}_{Bhyb}\right) - \omega_2\left(\hat{M}_B - M_{Bhyb}\right) \end{bmatrix}. \quad (37)$$

For the numerical realization, the error at each iteration is determined as follows

$$\begin{aligned} E_{\varphi(i)} = \frac{1}{q_{m1}} & \left[\frac{M_{c(i)} - M_{c(i-1)}}{T} - \frac{M_{c(i)hyb} - M_{c(i-1)hyb}}{T}\right. \\ & \left. -\omega_1\left(M_{c(i)} - M_{c(i)hyb}\right)\right] \\ & + \left[\frac{\hat{M}_{B(i)} - \hat{M}_{B(i-1)}}{T} - \frac{M_{B(i)hyb} - M_{B(i-1)hyb}}{T}\right. \\ & \left. -\omega_2\left(\hat{M}_{B(i)} - M_{B(i)hyb}\right)\right] \end{aligned} \quad (38)$$

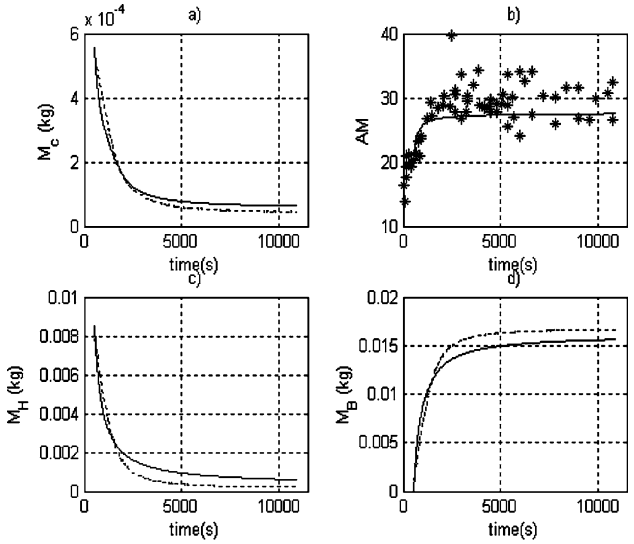
A feedforward NN with two inputs (M_c, M_B), 1 output (φ_{NN}), one hidden layer with seven sigmoid nodes was trained to minimize the error (38). The Levenberg–Marquardt (LM) algorithm was chosen as the training method due to its advantages in terms of execution time and robustness [9]. Since the LM algorithm requires a lot of memory, a powerful (in terms of memory) computer is the main condition for successful training. In order to solve the problem of several local minima, that is typical for all gradient based optimization algorithms (including the LM method), we have repeated several time the optimization specifying different starting points.

The hybrid model is evaluated for its ability to predict process behavior for various initial concentrations of reagents (0.05, 0.1, 0.2, 0.3, and 0.4 M). The results are summarized in Figs. 4 and 5. Data for the main system states (M_c measured, M_{HAP} and M_B estimated) are denoted by dashed line in all figures on subplots a, c and d, respectively. Data for average particle size (AM) are denoted by stars in all subplots b. The model time trajectories of M_c (subplots a) and AM (subplots b) are direct indicators of the model quality since measurements for them are available. The mass of HAP and brushite are not directly measured variables but they can be inferred by the available measurements. Thus, the plots of M_{HAP} (subplots c) and M_B (subplots d) are indirect indicators of the model reliability. Data from experiments with initial concentrations of reagents 0.2 and 0.3 M were used to train the NN, therefore, it is not surprisingly that all model variables (M_c, M_{HAP}, M_B, AM) closely match the data from these two types of experiments (see Fig. 4). However, more valuable are the results depicted in Fig. 5 where the model was tested on new 'unseen' validation data corresponding to experiments with different initial concentrations (0.05, 0.1 and 0.4 M). Data and model trajectories match quite well which confirms the ability of the hybrid NN to estimate the precipitation rate of calcium phosphate.

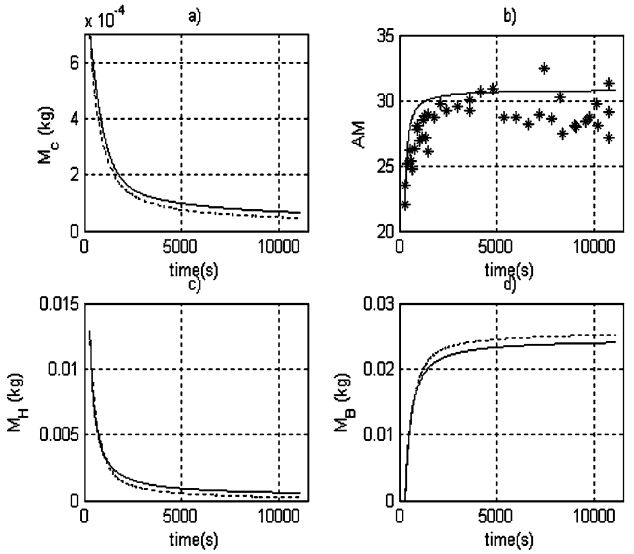
For the purpose of comparative analysis, three existing analytical models were also applied on the same task, see Table 1. The parameters of the fixed models were tuned with the same data obtained by experiments with (0.2 and 0.3 M) initial concentrations of reagents Here, we take the nondimensional error index (NDEI), which is defined as the root mean square error (RMSE) divided by the standard deviation of the target series.

6 Case study 2: estimation of sugar crystallization growth rate with partly unknown kinetics coefficients

Sugar crystallization occurs through mechanisms of nucleation, growth and agglomeration that are known to be affected by several not well-understood operating conditions. The search for efficient methods for process description is linked both to the scientific interest of understanding fundamental mechanisms of the crystallization



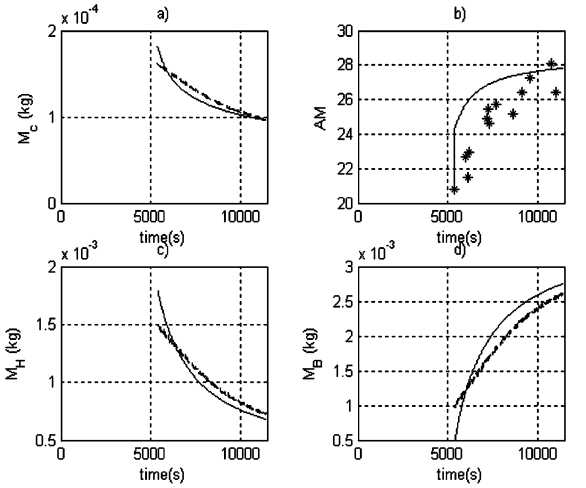
(a) Initial concentration of reagents 0.2 M



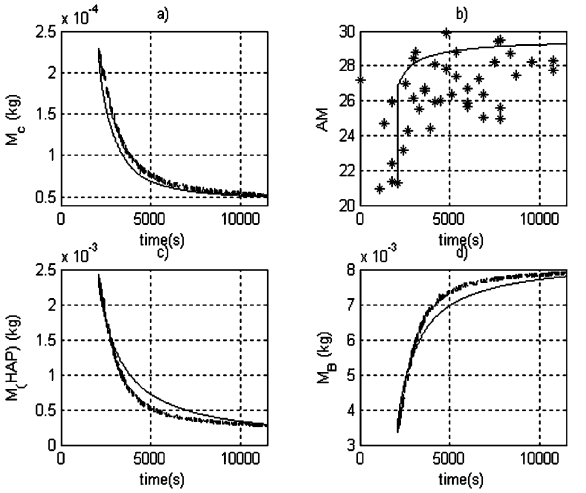
(b) Initial concentration of reagents 0.3 M

Fig. 4 Results with the training data. Data points (*doted lines* or *stars*) and the model estimations (*solid lines*) along time for (**a**) mass of calcium in solution, (**b**) the average (in mass) particle size AM (μm); (**c**) mass of HAP; (**d**) mass of brushite

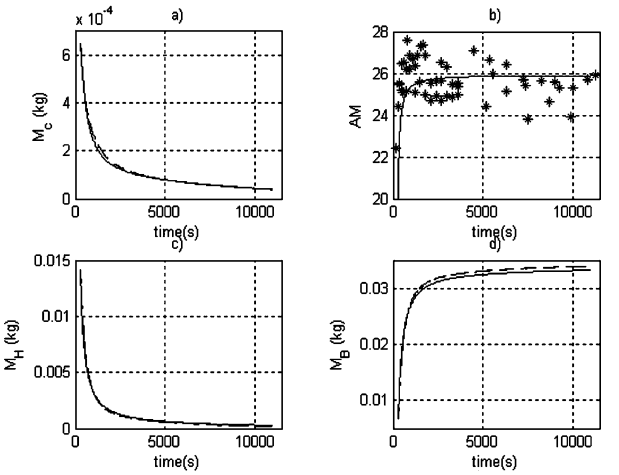
process and to the relevant practical interest of production requirements. The sugar production batch cycle is divided in several phases. During the first phase the pan is partially filled with a juice containing dissolved sucrose. The liquor is concentrated by evaporation, under vacuum, until the supersaturation reaches a predefined value. At this point seed crystals are introduced into the pan to induce the production of crystals (crystallization phase). As evaporation takes place further liquor or water is added to the pan. This maintains the level of supersaturation and increases the volume contents. The third phase consists of tightening



(a) Initial concentration of reagents 0.05 M



(b) Initial concentration of reagents 0.1 M



(c) Initial concentration of reagents 0.4 M

Fig. 5 Results with the validation data. Data points (*dashed line* or *stars*) and the model estimations (*solid lines*) along time for (**a**) mass of calcium in solution, (**b**) the average (in mass) particle size AM (μm); (**c**) mass of HAP; (**d**) mass of brushite

which is controlled by the evaporation capacity [12]. Since the objective of this paper is to illustrate the technique introduced in Sect. 4, the following assumptions are adopted:

1. Only the states that explicitly depend on the crystal growth rate are extracted from the comprehensive mass balance process model;
2. The population balance is expressed only in terms of number of crystals;
3. The agglomeration phenomenon is neglected.

The simplified process model is then

$$\frac{dM_s}{dt} = -k_1 G + F_f \rho_f B_f \mathrm{Pur}_f \tag{39.1}$$

$$\frac{dM_c}{dt} = k_1 G \tag{39.2}$$

$$\frac{dT_m}{dt} = k_2 G + bF_f + cJ_{vap} + d \tag{39.3}$$

$$\frac{dm_0}{dt} = k_3 G \tag{39.4}$$

where M_s is the mass of dissolved sucrose, M_c is the mass of crystals, T_m is the temperature of the massecuite, m_0 is the number of crystals. Pur_f and ρ_f are the purity (mass fraction of sucrose in the dissolved solids) and the density of the incoming feed. F_f is the feed flowrate, J_{vap} is the evaporation rate and b, c, d are parameters incorporating the enthalpy terms and specific heat capacities. They are derived as functions of physical and thermodynamic properties. Apart from k_3, all other kinetics coefficients are usually badly known. The full state vector is

$$X_{aug} = [\,M_s \quad M_c \quad T_m \quad m_0\,]^T, \quad \text{with}$$
$$K_{aug} = [\,-k_1 \quad k_1 \quad k_2 \quad k_3\,]^T.$$

The following nonsingular state partition A is chosen,

$$X_a = M_c, \quad X_b = [\,M_s \quad T_m \quad m_0\,]^T.$$

Initial values for the unknown coefficients are assigned $\left(\hat{k}_1 = \hat{k}_2 = 0.01\right)$. Matrix A_0 is then

$$A_0 = \begin{bmatrix} 1 & -\frac{\hat{k}_2}{\hat{k}_1} & -\frac{k_3}{\hat{k}_1} \end{bmatrix}^T \tag{40}$$

M_c and T_m are measured states, then the unique state partition B is

$$X_1 = [\,M_c \quad T_m\,]^T, \quad X_2 = [\,M_s \quad m_0\,]^T,$$

The matrices of the second representation of vector Z are

$$A_1 = \begin{bmatrix} 1 & -\frac{\hat{k}_2}{\hat{k}_1} & -\frac{k_3}{\hat{k}_1} \\ 0 & 1 & 0 \end{bmatrix}^T, \quad A_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}^T \tag{41}$$

For this case $D = 0$, then the estimations of the individual elements of Z are

$$\begin{aligned} \hat{Z}_1 &= M_c + \hat{M}_s \\ \hat{Z}_2 &= -\frac{\hat{k}_2}{\hat{k}_1} M_c + T_m \\ \hat{Z}_3 &= -\frac{k_3}{\hat{k}_1} M_c + \hat{m}_0 \end{aligned} \tag{42}$$

The estimations of the unmeasured states are

$$\begin{bmatrix} \hat{M}_s \\ \hat{m}_0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \left(\begin{bmatrix} \hat{Z}_1 \\ \hat{Z}_2 \\ \hat{Z}_3 \end{bmatrix} - \begin{bmatrix} 1 & 0 \\ -\frac{\hat{k}_2}{\hat{k}_1} & 1 \\ -\frac{k_3}{\hat{k}_1} & 0 \end{bmatrix} \begin{bmatrix} M_c \\ T_m \end{bmatrix} \right). \tag{43}$$

The unknown kinetic coefficients updating is

$$\begin{bmatrix} \frac{d\hat{k}_1}{dt} \\ \frac{d\hat{k}_2}{dt} \end{bmatrix} = \begin{bmatrix} \gamma_1 & 0 \\ 0 & \gamma_2 \end{bmatrix} \begin{bmatrix} M_c - \hat{M}_c \\ T_m - \hat{T}_m \end{bmatrix}. \tag{44}$$

$$\begin{bmatrix} \frac{d\hat{M}_c}{dt} \\ \frac{d\hat{T}_m}{dt} \end{bmatrix} = \begin{bmatrix} \hat{k}_1 \\ \hat{k}_2 \end{bmatrix} G_{NN} + \begin{bmatrix} 0 \\ bF_f + cJ_{vap} + d \end{bmatrix} - \begin{bmatrix} \omega_1 & 0 \\ 0 & \omega_2 \end{bmatrix} \begin{bmatrix} M_c - \hat{M}_c \\ T_m - \hat{T}_m \end{bmatrix} \tag{45}$$

Table 1 Estimation results for three analytical models and the hybrid model

Calcium phosphate precipitation rate models	NDEI for model validation data
Monod type model [2]	0.404
$\phi = \frac{\lambda_{HAP1} M_{HAP}}{\beta_{HAP1} + M_{HAP}} + \frac{\lambda_{c1} M_c}{\beta_{c1} + M_c}$	
λ_{HAP1}, λ_{c1}, β_{HAP1}, β_{c1} tuning parameters	
Contois type model [1]	0.096
$\varphi = \frac{\lambda_{HAP2} M_{HAP}}{\beta_{HAP2} M_B + M_{HAP}} + \frac{\lambda_{c2} M_c}{\beta_{c2} M_B + M_c}$	
λ_{HAP2}, λ_{c2}, β_{HAP2}, β_{c2} tuning parameters	
"Logistic" type model [1] $\varphi = \exp(-\lambda_3 M_B)$, λ_3 tuning parameter	0.062
Hybrid model (this paper)	0.023
FFNN with two inputs (M_c, M_B), one output (φ_{NN}), one hidden layer with seven sigmoid nodes	

For the current study, the values of $\omega_1 = \omega_2 = 0.5$ and $\gamma_1 = \gamma_2 = 1$ were selected by trial and error. However, improved results might be expected if the values would be more carefully selected. The observation error and the extended NN error signal are

$$E_x = \begin{bmatrix} M_c - M_{chyb} \\ T_m - T_{mhyb} \end{bmatrix},$$
$$E_{K\varphi} = \begin{bmatrix} \left(\dot{M}_c - \dot{M}_{chyb}\right) - \omega_1\left(M_c - M_{chyb}\right) \\ \left(\dot{T}_m - \dot{T}_{mhyb}\right) - \omega_2\left(T_m - T_{mhyb}\right) \end{bmatrix}, \quad (46)$$

In the numerical implementation the first derivative of the observation error is computed as the difference between the current $E_x(k)$ and the previous value $E_x(k-1)$ of the observation error divided by the integration step (Δt)

$$\dot{E}_x = \frac{E_x(k) - E_x(k-1)}{\Delta t}.$$

A feedforward NN with four inputs (M_c, T_m, m_0, M_s), one output (G_{NN}), one hidden layer with nine sigmoid nodes was trained to minimize the error. For two of the inputs (M_c, T_m) industrial data of six batches are provided, the rest of the inputs are estimated. As in the previous section, the Levenberg–Marquardt algorithm was chosen as the training method.

The hybrid model obtained is compared with an analytical model of the sugar crystallization, reported in Georgieva et al. [12]. The difference between the two models lays in the way how the growth rate is estimated. In the hybrid model G is estimated by a NN following the technique discussed in Sect. 4. In the analytical model G is given by the following empirical correlation

$$G = K_g \exp\left[-\frac{57000}{R(T_m + 273)}\right](S-1)^2 \exp[-13.863(1 - P_{sol})]\left(1 + 2\frac{V_c}{V_m}\right) \quad (47)$$

where S is the supersaturation, P_{sol} is the purity of the solution and V_c/V_m is the volume fraction of crystals. K_g is a constant, optimized following a classical nonlinear least-squares regression.

The performance of the two models is examined with respect to prediction quality of the crystal size distribution (CSD) at the end of the process which is quantified by two parameters—the final average (in mass) particle size (MA) and the final coefficient of particle variation (CV). In Table 1, the predictions given by the two models are compared with the experimental data for the CSD. The hybrid model outperforms the analytical model particularly with respect to predictions of CV. In this sense, the hybrid model is sufficiently reliable to be integrated in the structure of a model-based controller (Table 2).

Table 2 Final CSD—experimental data versus hybrid and analytical model predictions

Batch no.	Experimental data		Hybrid model (this paper)		Analytical model	
	AM (mm)	CV (%)	AM (mm)	CV (%)	AM (mm)	CV (%)
1	0.479	32.6	0.49	29.2	0.583	21.26
2	0.559	33.7	0.51	30.17	0.542	18.43
3	0.680	43.6	0.52	33.81	0.547	18.69
4	0.494	33.7	0.51	31.7	0.481	14.16
5	0.537	32.5	0.55	29.66	0.623	24.36
6	0.556	35.5	0.51	29.13	0.471	13.642
7	0.560	31.6	0.59	34.34	0.755	34.9
8	0.530	31.2	0.58	35.16	0.681	27.39
Average error			5.9%	10.2%	13.7%	36.1%

7 Conclusions

This work is focused on developing a more efficient computational scheme for estimation of process reaction rates based on NN models. Two scenarios are considered: (1) the kinetics coefficients of the process are completely known and the process states are partly known (measured). Such a scenario is very common in less complex reaction systems; (2) the kinetics coefficients and the states of the process are partly known. This is the most general case in process engineering.

The contribution of the paper is twofold. From one side we formulate a hybrid (ANN and first principles) model that outperforms the traditional reaction rate estimation approaches. From other side a new procedure for NN supervised training is proposed when target (reference) outputs are not available. The NN is embedded in the framework of a first principle process model and the error signal for updating the network weights is determined analytically. In the two scenarios considered, the unmeasured states are first estimated and then the NN is trained with both estimated and measured data. In case the kinetics parameters are all known (case 1) the state estimation can be performed independently of the reaction rates estimation. However, in case 2, both reaction rates and kinetics coefficients have to be estimated simultaneously, therefore the network training is computationally much more challenging.

Acknowledgments This work was financed by the Portuguese Foundation for Science and Technology within the activity of the Research Unit IEETA-Aveiro, which is gratefully acknowledged.

References

1. Bastin G, Dochain D (1990) On-line estimation and adaptive control of bioreactors. Elsevier Science Publishers, Amsterdam

2. Lubenova V, Rocha I, Ferreira EC (2003) Estimation of multiple biomass growth rates and biomass concentration in a class of bioprocesses. Bioprocess Biosyst Eng 25(6):395–406. doi:10.1007/s00449-003-0325-1
3. Walter E, Pronzato L (1997) Identification of parametric models from experimental data. Springer, UK
4. Noykove N, Muller TG, Gylenberg M, Timmer J (2002) Quantitative analysis of anaerobic wastewater treatment processes: identifiably and parameter estimation. Biotechnol Bioeng 78(1): 91–103
5. Galvanauskas V, Georgieva P, Feyo de Azevedo S (2006) Dynamic optimisation of industrial sugar crystallization process based on a hybrid (mechanistic + ANN) model. In: IEEE world congress on computational intelligence, Vancouver, 16–21 July 2006
6. Georgieva P, Oliveira C, Rocha F, Feyo de Azevedo S (2007) Process modelling strategy combining analytical and data based techniques—I. NN identification of reaction rates. In: International joint conference on neural networks (IJCNN), Orlando, 12–17 August 2007
7. Haykin S (1999) Neural networks: a comprehensive foundation. Prentice Hall, NJ
8. Chen L, Bastin G (1996) Structural identifiability of the yeals coefficients in bioprocess models when the reaction rates are unknown. Math Biosci 132:35–67. doi:10.1016/0025-5564(95)00048-8
9. Hagan Martin T, Mehnaj Mohammad B (1994) Training feed forward networks with the Marquardt algorithm. IEEE Trans Neural Netw 5(6):989–993. doi:10.1109/72.329697
10. Sorensen JS, Lundager Madsen HE (2000) The influence of magnetism on precipitation of calcium phosphate. J Crystal Growth 216:399–408. doi:10.1016/S0022-0248(00)00449-8
11. Ferreira A, Oliveira C, Rocha F (2003) The different phases in the precipitation of dicalcium phosphate dehydrate. J Crystal Growth 252:599. doi:10.1016/S0022-0248(03)00899-6
12. Georgieva P, Meireles MJ, Feyo de Azevedo S (2003) Knowledge based hybrid modeling of a batch crystallization when accounting for nucleation, growth and agglomeration phenomena. Chem Eng Science 58:3699–3707