Deep Learning Based Model Predictive Control for a Reverse Osmosis Desalination Plant

Divas Karimanzira, Thomas Rauschenbach

Department of Surface Water and Maritime Systems, Fraunhofer IOSB-AST, Ilmenau, Germany
Email: divas.karimanzira@iosb-ast.fraunhofer.de

Abstract

Reverse Osmosis (RO) desalination plants are highly nonlinear multi-input-multioutput systems that are affected by uncertainties, constraints and some physical phenomena such as membrane fouling that are mathematically difficult to describe. Such systems require effective control strategies that take these effects into account. Such a control strategy is the nonlinear model predictive (NMPC) controller. However, an NMPC depends very much on the accuracy of the internal model used for prediction in order to maintain feasible operating conditions of the RO desalination plant. Recurrent Neural Networks (RNNs), especially the Long-Short-Term Memory (LSTM) can capture complex nonlinear dynamic behavior and provide long-range predictions even in the presence of disturbances. Therefore, in this paper an NMPC for a RO desalination plant that utilizes an LSTM as the predictive model will be presented. It will be tested to maintain a given permeate flow rate and keep the permeate concentration under a certain limit by manipulating the feed pressure. Results show a good performance of the system.

Keywords

1. Introduction

Recently, there have been an increased interest and commercialization of desalination systems due to significant improvement in technology and the advantageous developments in membrane technology. The dynamics of an RO desalination system are highly nonlinear, constrained and subject to uncertainties such as membrane fouling and varying feed water quality. Therefore, the design of a suitable controller for the RO desalination system is a very challenging task.

There have been several approaches for controlling nonlinear systems in general such as the linear quadratic regulator (LQR) [1], proportional integral derivative controller (PID) [2], backstepping control [3] and sliding mode control (SMC) [4]. Nevertheless, all these techniques usually do not take into account the actual constraints of the process and just consider the control effects. Furthermore, the parameters of the controllers are chosen aimlessly, hence the optimality of the system cannot be guaranteed.

Model predictive control has been applied to control RO desalination processes [5] [6] [7] [8] [9]. It is obvious that the performance of the model predictive controller largely depends on the quality of the predictive model used, especially if the system is complex and highly nonlinear. Several techniques have been used for system identification for the MPC, e.g., Kalman filtering [10], maximum likelihood estimation [11] [12]. However, it is known that the Kalman filter requires knowledge of the mathematics behind the system, which we know is very difficult to obtain for highly complex processes such as the RO desalination system with several unknown disturbances, and the physical phenomena such as membrane fouling. Artificial Neural Networks (ANNs) have proven to be very good function approximators and do not need any mathematical model, but the input-output data of the system [13]. There have been applications of ANNs for the MPC control [14] [15] [16], especially the Multilayer Perceptron (MLP). The MLP has some limitations to time variant systems, because the learned results are static input-output maps. Furthermore, the prediction steps of the MLP are limited [17].

In [18] [19], Recurrent Neural Networks (RNNs) were introduced into the structure of the MPC, because they can capture the system dynamics and provide long-range predictions [20]. It is well-known that RNNs have issues with vanishing and exploding gradients, which makes their training difficult sometimes, therefore we propose to use a special form of RNN, *i.e.*, the Long Short Term Memory (LSTM). LSTM is a special version of RNNs structure that was designed to model chronological sequences and their long-range dependencies more precisely than conventional RNNs [18].

Even though, it is not new to combine MPC with recurrent neural networks [21] [22] [23], the application of LSTM as the predictive model for the MPC for desalination processes is hardly found in literature. This fact motivated us to put our focus on system identification using LSTM with a view towards closed-loop control with MPC for control of a RO desalination plant. The new contributions of this paper are the following:

- Introduction of LSTM as the predictive model in MPC to capture nonlinearities
- The combined structure of LSTM and MPC is new to RO desalination control

The remainder of the paper is outlined as follows. In Section 2 the model of the RO desalination plant and some scenarios for assessing the performance of the control system in closed loop will be described. Following this, in Section 3, a

section about the methods and materials will be given, in which the method of system identification using a LSTM and the problem formulation for the MPC using the identified LSTM as the prediction model will be described. Finally, the results of the system identification and the closed loop simulations control performance and discussions will be given in Section 4.

2. RO Desalination Plant Model and Control Scenarios

In this section, the model of the RO desalination plant and some scenarios for assessing the performance of the control system in closed loop will be described.

2.1. RO Desalination Plant Model

A RO desalination plant shown in **Figure 1** is used as the nonlinear plant on which the LSTM-based Model predictive control algorithm is applied to control the nonlinear process. The configuration of the system includes two tanks: a feed tank, and another tank for draining permeate. Furthermore, the plant includes reverse osmosis unit and a high pressure pump. A high-pressure pump is used to pump the water from the feed tank to the pressure (P_f) into the RO unit. From the inflows and outflows of the feed tank, it is obvious that the feed water total dissolved solids (feed TDS)—feed water concentration (C_f) is changing constantly, because some TDS leave with permeate (C_p), some TDS are lost due to adhesion on the membrane surface and some TDS (C_{in}) enter the system with the filling water (Q_{in}) for the feed water tank. The permeate concentration (C_p) and the brine concentration (C_b) and the total permeate quantity (Q_p) and brine (Q_b) at the outlet of the membrane module define the operating conditions of the RO unit itself and they can be controlled by adjusting the feed pressure at the RO unit inlet.

From **Figure 1**, the mass and the salt balances for the feed tank are given by the following equations:

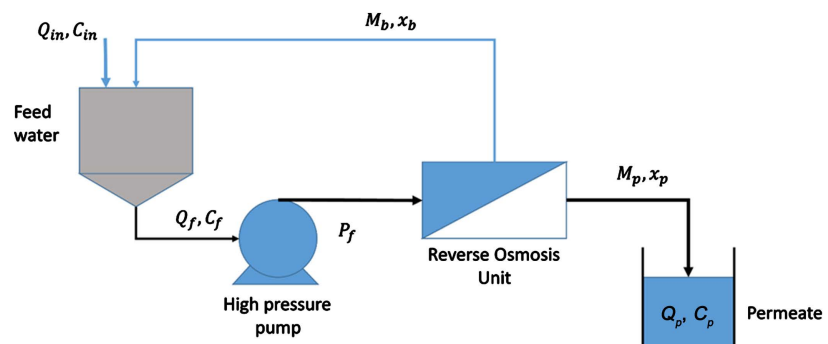$$\frac{\mathrm{d}Q_f}{\mathrm{d}t} = -M_p \tag{1}$$

Figure 1. Reverse osmosis desalination plant, Q_f —Feedwater flow rate, x—Concentration [g/L], x_b —Brine salinity [g/L], x_f —Feed salinity [g/L], x_p —Permeate salinity [g/L], and M_b —Brine flow rate [L/min], M_p —Permeate flow rate [L/min].

$$\frac{\mathrm{d}Q_f x_f}{\mathrm{d}t} = -M_p x_p \tag{2}$$

Expanding Equation (2) gives,

$$Q_f \frac{\mathrm{d}x_f}{\mathrm{d}t} + x_f \frac{\mathrm{d}Q_f}{\mathrm{d}t} = M_p x_p \tag{3}$$

Substituting Equation (1) into Equation (3), Equation (4) is obtained to:

$$Q_f \frac{\mathrm{d}x_f}{\mathrm{d}t} + x_f\left(-M_p\right) = M_p x_p \tag{4}$$

$$\frac{\mathrm{d}x_f}{\mathrm{d}t} = \left(-M_p x_p + x_f M_p\right)\big/Q_f \tag{5}$$

Finally, the feed tank can be described by Equations (1) and (5).

The same can be done to characterize the permeate tank to get the following two equations of mass and salt balances:

$$\frac{\mathrm{d}Q_p}{\mathrm{d}t} = M_p \tag{6}$$

$$\frac{\mathrm{d}Q_{pt} x_{pt}}{\mathrm{d}t} = M_p x_p \tag{7}$$

Substituting as we did previously, the two equations that describe the permeate tank can be obtained to Equations (6) and (8)

$$\frac{\mathrm{d}x_{pt}}{\mathrm{d}t} = \left(M_p x_p + x_{pt} M_p\right)\big/Q_{pt} \tag{8}$$

The differential Equations (1), (5), (6) and (8) coupled with theories of El-Dessouky and Ettouney in Equations (9)-(14) can be used to describe the RO system and give a complete characterization of the plant [24].

The salt passage rate through the reverse osmosis membrane $M_s\,[\mathrm{kg/s}]$ can be expressed as in Equation (9).

$$M_s = K_s A_{em} T_s \left(\beta C - C_p\right) \times 1000 \tag{9}$$

$$C = \left(M_f C_f + M_b C_b\right)\big/\left(M_f + M_b\right) \tag{10}$$

where $K_s\left[\mathrm{m^3/m^2 \cdot s}\right]$ is the salt permeability coefficient at the reference temperature $T_{ref}\,[^\circ\mathrm{C}]$, the total membrane area is denoted by $A_{em}\left[\mathrm{m^2}\right]$, β is the concentration polarization factor, T_s is the temperature correction factor for salt permeability, $C\,[\mathrm{ppm}]$ is the net concentration, and $C_p\,[\mathrm{ppm}]$ is the permeate concentration.

The permeate flow rate (water passage rate) $M_p\,[\mathrm{kg/s}]$ required in Equation (1) given by Equation (11) is a function of the membrane differential pressure $\Delta P\,[\mathrm{kPa}]$ and the net osmotic pressure $\Delta\pi\,[\mathrm{kPa}]$.

$$M_p = K_w A_{em} T_m \left(\Delta P - \beta\Delta\pi\right)\rho_w \tag{11}$$

where $K_w\left[\mathrm{m^3/m^2 \cdot s \cdot kPa}\right]$ is the water permeability coefficient at the reference

temperature $T_{ref}\left[°C\right]$, T_m is the temperature correction factor for water permeability, and $\rho_w\left[kg/m^3\right]$ is the permeate density.

The concentration of the permeate $C_p\left[ppm\right]$ and the brine $C_b\left[ppm\right]$ are given by Equations (12) and (14), respectively:

$$C_p = K_s C_b \left/ \left(\frac{J}{\exp\left(\frac{J}{k}\right)} + K_s \right) \right. \tag{12}$$

$$J = \frac{\Delta P - \Delta \pi}{\eta \left(R_m + R_c\right)} = \frac{Q_p}{A_{em}} \tag{13}$$

$$C_b = \frac{C_f M_f - C_p M_p}{M_b} \tag{14}$$

where $J\left[m/s\right]$ is the permeate flux, $k\left[m/s\right]$ is the mass transfer coefficient, $\eta\left[kPa\cdot s\right]$ is the seawater dynamic viscosity, $R_c\left[m^{-1}\right]$ is the cake layer resistance, and $R_m\left[m^{-1}\right]$ is the intrinsic membrane resistance.

2.2. Control Scenarios

Since this study has the main focus on process control, the system performance should be evaluated in closed loop control, where the system will be tracking a setpoint. So, the objective of the controller is to bring the RO desalination system quickly and smoothly to target set-point of the permeate flow rate and keep the permeate concentration under $M_{p,d}$ by adjusting the feed pressure. Furthermore, it is important to compare the closed-loop performance of the LSTM-based MPC against a classical non-linear MPC controller that utilizes the true RO desalination plant model, as described in Equations (1)-(14), directly.

3. Methods and Materials

The procedure for using an LSTM as the predictive model in the MPC comprises of several steps starting from 1) generating a dataset by acquiring data from the system using perturbations of the manipulated variables, here in our case, the feed pressure; 2) dividing the dataset into training and validation sets and training the LSTM on the training dataset while testing the network on the validation dataset for early stopping. There are some hyperparameters which need to be selected to find the best performance. This can be done manually, whereby several network configurations are trained and the best performing network selected, or one can use Bayesian optimization to find the parameters automatically [25]; 3) integrating the LSTM with the best performance with the MPC and 4) finally run closed loop simulations with LSTM-based MPC to evaluate its control performance.

3.1. Internal Model Using Long Short Term Memory Network

The task of system identification is main focus of this section and comprises of

approximating the RO desalination system as described by Equations (1)-(14). The p-step ahead prediction issue is supposed to be of vital important interest for the control using MPC. Deep neural networks are universal function approximators and can be used to capture the nonlinear dynamics of systems. They are relatively simple to obtain and evaluate in real-time. To them belongs the LSTMs that can better capture temporal dependencies in the dynamical system. Especially for predictive control, the LSTMs are particularly useful. They can be used to make the required p-step ahead predictions of state variables, based on the fact that the prediction for time-step p depends solely on the current state and all control actions in time-step $k \in \{0, \cdots, p-1\}$. The time-step $p-1$ predictions used in the time-step p prediction are equally dependent on the current state and all control behaviour in time-step $k \in \{0, \cdots, p-2\}$, etc.

Figure 2 shows the LSTM structure for the p-step ahead prediction problem. It is made out of repeating cells with four interacting components forming each layer, and in our case each cell represents a time-step, so that the state of the cell representing time-step $k \in \{0, \cdots, p-1\}$ serves as the input for a cell representing time-step $k+1$. Each cell contains user-specifiable N number of hidden nodes that encode the state representation. These cells use several gating functions, like the "forget", "input" and "output" gating functions, that serve to modulate the propagation of signals between cells. This cell structure avoids the gradient vanishing or exploding problem.

The basic LSTM cell structure (Figure 2(b)) is fully mathematically described in the appendix of [19]. It has three inputs denoted by h_{k-1}, C_{k-1} and u_k and two outputs given by h_k and C_k. At any given time step k, $h_k \in \mathbb{R}^N$ is the hidden state, C_k is the cell state, u_k is the current input.

The first layer is a sigmoid layer $\sigma : \mathbb{R}^N \to [0,1]^N$ which has two inputs h_{k-1} and u_k. h_{k-1} represent the hidden state of the previous cell. This is called the forget gate $f_k \in [0,1]$ because its output decides which information of the previous cell is to be included.

$$f_k = \sigma\left(W_f \cdot [h_{k-1}, u_k]' + b_f\right) \qquad (15)$$
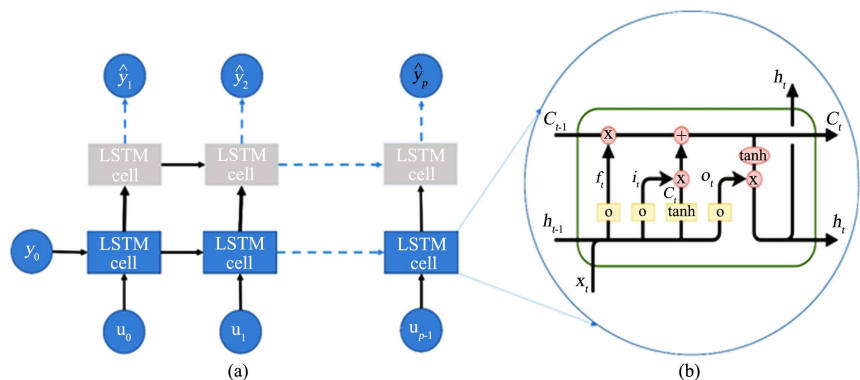

Figure 2. (a) LSTM structure for the p-step ahead prediction problem and (b) LSTM internal model structure with the three gates—forget gate, input gate and output gate.

The second layer is also a sigmoid layer and represents the input gate $i_k \in [0,1]$ that decides which new information is to be added to the cell. It takes two inputs h_{k-1} and x_k. A vector $\tilde{C}_k$ of the new candidate values is created by the tanh layer.

$$i_k = \sigma\left(W_i \cdot [h_{k-1}, u_k]' + b_i\right) \tag{16}$$

$$\tilde{C}_k = \tanh\left(W_C [h_{k-1}, u_k]' + b_c\right) \tag{17}$$

The two layers are then composed to determine the information to be stored as the cell state. The operator * denotes point-wise multiplication. The point-wise multiplication of the input gate and the vector $\tilde{C}_k$ of the new candidate values $(i_k * \tilde{C}_k)$ gives the amount of information to be added to the LSTM cell state. This result is added with the result of the forget gate f_k multiplied with previous cell state $(f_k * C_{k-1})$ to produce the current cell state C_k.

$$C_k = f_k * C_{k-1} + i_k * \tilde{C}_k \tag{18}$$

Finally, the output of the LSTM cell is calculated using a sigmoid and a tanh layer, whereby the sigmoid layer determines the part of the cell state which will be present in the output whereas tanh layer shifts the output in the range of [−1, 1]. The results of the two layers undergo point-wise multiplication to produce the output h_k of the cell.

$$o_k = \sigma\left(W_o \cdot [h_{k-1}, u_k]' + b_o\right) \tag{19}$$

$$h_k = o_k * \tanh(C_k) \tag{20}$$

$$y_k = \Phi(h_k) \tag{21}$$

where $\hat{y}_k \in \mathbb{R}^2$ is the cell output which corresponds to the state vector prediction for time-step k. h_0 is initialised in this study by using y_0.

The regressors required to predict $\hat{y}_k \in \{1, \cdots, p\}$ are henceforth represented by $\phi_k := \{y_0, u_0, \cdots, u_{k-1}\}$, and they are introduced into the LSTM in a fashion illustrated in Figure 2(a). Equation (22) below serves as a shorthand to describe the LSTM:

$$\hat{y}_k = \hat{f}_{LSTM}(\phi_k), k \in \{1, \cdots, p\} \tag{22}$$

An LSTM is characterized by the values of the weights and biases for the different gates Forget gate (W_f, b_f), Input gate (W_i, b_i), Output gate (W_o, b_o), W_c, b_c, and for all layers, and these values constitute the set of parameters. These parameters are learnt from training data by minimizing the predictive error of the model on the training set as determined through a user-specified loss function. The learning process is performed through the back-propagation through time (BPTT) algorithm that estimates the gradient of the loss function as a function of the weights, and an optimization algorithm that uses the calculated gradient to adjust the existing weights. The adaptive moment estimation algorithm (Adam) [26] is an example of an optimization algorithm that is widely

used. In the BPTT, the weights are initialized, the information is passed through the different gates, the output h_k and current cell state C_k are calculated, the gradient through back propagation through time at time step k are calculated using chain rule and finally using all gradients, the weights associated with input gate, output gate, and forget gate are updated.

3.2. Data Acquisition

For training the LSTM, a dataset which covers the whole operating range of the RO desalination plant was collected by perturbation of the manipulated variable, the feed pressure and recording the dynamic system response. A pre-defined sequence of the manipulated variable, $P_{f,k}, k \in \{0, \cdots, T_K - 1\}$ is introduced into the system and the dynamic response $M_{p,k}, k \in \{0, \cdots, T_K - 1\}$, $Q_{p,k}, k \in \{0, \cdots, T_K - 1\}$ is recorded. Such a signal for the feed pressure and the dynamic response for permeate flow rate M_p and permeate concentration x_p, total permeate quantity Q_p and permeate concentration C_p are shown in **Figure 3** and **Figure 4(a)** and **Figure 4(b)**, respectively. T_K denotes the final time-step for the perturbation experiment. The perturbation is sampled at Δt. $Q_{p,k}$, is the measured system output at time-step k after $P_{f,k-1}$, has been applied to the system for a period of Δt. These correspondences of the input and output variables are referred to in machine learning terminology as labels, and the data set is thereafter constructed from both the experimental sequences and their associated labels. For the p-step ahead prediction problem, each data point thus takes the form $\{y_k, u_k, u_{k+1}, \cdots, u_{k+p-1}\}$ with the associated label $\{y_{k+1}, \cdots, y_{k+p}\}$, $k \in \{0, \cdots, T_K - p\}$. $T_K - p$ data points can thus be extracted from each experimental sequence.

The input to the system, the feed water concentration is an uncertainty. Therefore, Gaussian noise was added to its signal before it was used to excite the system (**Figure 3**).

Using the normal approach in machine learning, before training the LSTM, the labeled dataset is split into three parts with one part for training (data used for adapting the network weights), one part for validation and the last part for testing.
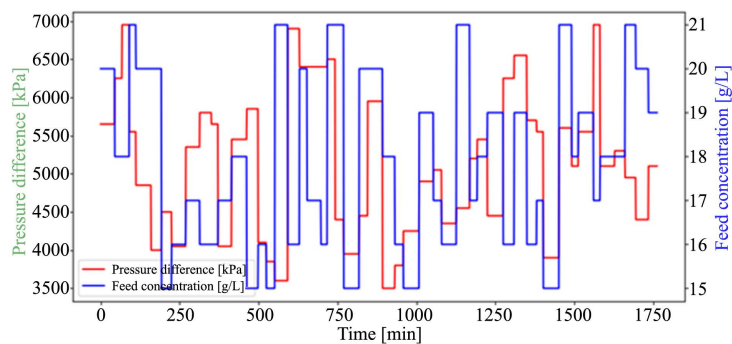

Figure 3. Perturbating the manipulated variable, feed pressure. The feed concentration is modeled as a disturbance.
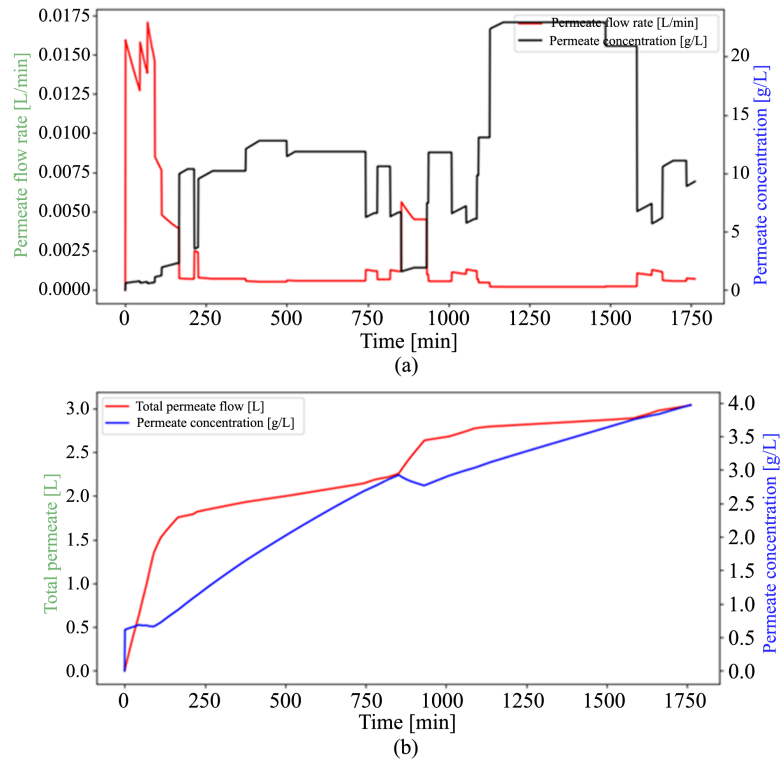
Figure 4. Dynamic system response to the perturbation signal in **Figure 3**. (a) The permeate flow rate and the permeate concentration; (b) The total permeate produced and the total permeate tank concentration.

3.3. Nonlinear Model Predictive Control Problem

The structure of the model predictive controller for a RO desalination system is shown in **Figure 5**. Briefly explained, the model predictive controller (MPC) decides m control moves for the future, $\{u_0, \cdots, u_{m-1}\}$, that minimizes an objective function over a finite prediction horizon of p steps by utilizing the dynamic system predictions for those p steps, $\{\hat{y}_1, \cdots, \hat{y}_p\}$. Typically, the objective function is chosen to penalize large control effort, which means higher power consumption for the actuator, and discrepancies between the state vector and the set-point at each time instance. Constraints on input and output may also be factored into the MPC formulation. Since MPC performance depends on the quality of the system's predictions, a reasonably accurate model obtained through system identification is crucial.

The Equations (23)-(26) below describe the MPC problem

$$\min_{\{\Delta u_0, \Delta u_1, \cdots, \Delta u_{m-1}\}} \left\{ \sum_{k=1}^{p} \left(\hat{y}_k - y_k^*\right)' Q_y \left(\hat{y}_k - y_k^*\right) + \sum_{k=0}^{m-1} \Delta u_k' Q_u \Delta u_k \right\} \quad (23)$$

s.t.

$$\hat{y}_k = \hat{f}_{LSTM}\left(\phi_k\right), k \in \{1, \cdots, p\} \quad (24)$$

$$u_k \in \left[u_{\min,k}, u_{\max,k}\right], k \in \{0, \cdots, m-1\} \quad (25)$$

$$\Delta u_k \in \left[\Delta u_{\min,k}, \Delta u_{\max,k}\right], k \in \{0, \cdots, m-1\} \quad (26)$$
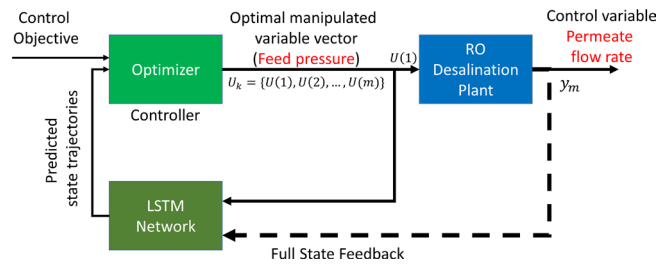
Figure 5. Schematic representation of a model predictive controller with full state feedback.

where $p \in \mathbb{Z}_+$ is the prediction horizon, $m \in \{1, \cdots, p\}$ the control horizon, $\hat{y}_k \in \mathbb{R}^2$ the prediction of the state vector for the discrete-time step k obtained from the LSTM, $\hat{f}_{LSTM}$ described in Equation (16), $y_k^* \in \mathbb{R}^2$ the set-point at time-step k, $u_k \in \mathbb{R}^2$ the manipulated vector for time-step k, $\Delta u_k \cong u_k - u_{k-1}$ the discrete-time rate of change of the manipulated vector which corresponds to the control action size at time-step k, $(Q_y, Q_u) \in (\mathbb{R}^{2\times2})^2$ symmetric positive semi-definite weight matrices, and $(u_{\min,k}, u_{\max,k}, \Delta u_{\min,k}, \Delta u_{\max,k}) \in (\mathbb{R}^2)^4$ the lower and upper limits for Δu and the rate of change of Δu at time-step k.

Within this formulation, no changes within actuator position are assumed beyond the time-stage $m-1$, *i.e.*, $\Delta u_{m-1} = \Delta u_{m+k} = 0, k \in \{0, \cdots, p-m-1\}$.

In general this problem of optimization is not convex and therefore does not have special structures suitable for global optimality. Therefore, this is a Non-Linear Programming (NLP) problem, and it can be solved with modern off-the-shelf solvers. For every step of the time, this problem is solved to yield the optimal control chain for that time-step, $\{\Delta u_0^*, \cdots, \Delta u_{m-1}^*\}$. The first element, Δu_0^*, is applied to the system until the next instant of sampling, where the problem is again resolved to yield another optimal control sequence. This process is then repeated in the form of a moving horizon. The complete procedure of the model predictive control is shown in **Table 1**.

4. Results and Discussions

The results will be discussed in two parts, the first part is about the results of the system identification and the second part describes the closed loop results for the MPC.

4.1. Model Identification Results

To measure the LSTM model predictive capability, we used the mean absolute error (MAE), the root mean square error (RMSE) as well as the correlation coefficient ρ. The model is implemented in Python environment on a PC with Intel (R) Core (TM) E5-2620 CPU, 62 GB memory. The training for 10 epochs took 1.45 s and the prediction for the test data of 1871 data points, about 0.02 s and did not show significant improvement after five epochs. **Figures 6(a)-(d)** show the validation MAE loss functions for the permeate flow rate, permeate concentration, total permeate flow and the total permeate concentration for 10 epochs.
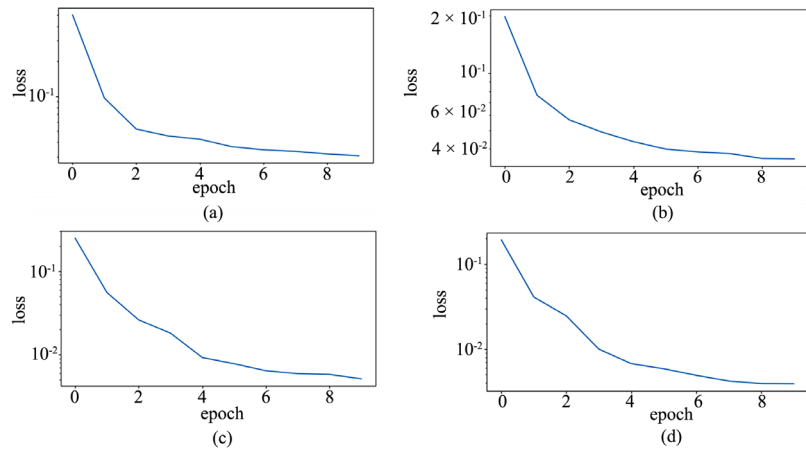
Figure 6. Loss functions of the (a) permeate flow rate M_p; (b) permeate concentration X_p; (c) total permeate flow quantity Q_p; (d) total permeate concentration C_p.

Table 1. NMPC algorithm.

Given	Model f, initial conditions $x(0)$, prediction horizon p, control horizon m, sampling time Δt, and weighting matrices Q and R
Step 1	At the current sampling time t_k, set $x(t_{k-1}) \leftarrow (t_k)$
Step 2	Solve Equations (23)-(26) for a sequence of m optimal input variables $\{U(1), U(2), \cdots, U(m)\}$
Step 3	Set $u(t_k) \leftarrow U(1)$ and inject the input to the plant
Step 4	At t_{k+1}, obtain the plant measurement y_m
Step 5	Corresponding to y_m, estimate the states $x^*(t_{k+1})$
Step 6	set $t_k \leftarrow t_{k+1}$
Step 7	Shift the prediction horizon p forward and repeat Step 1

A sharp drop in the MAE in the first a few iterations is shown. The training cycles stopped after 10 epochs with a smallest validation MAE value for the permeate flow rate, permeate concentration, total permeate flow and the total permeate concentration of 0.030, 0.0355, 0.0.0052 and 0.0039, respectively.

The hyperparameters in the prediction model such as the learning rate, batch size, dropout filtersize etc., need to be explored carefully to achieve the best prediction results. We utilize Bayesian optimization to search for these hyperparameters efficiently. From the Bayesian optimization, the best LSTM for system identification was found with the key parameters shown in **Table 2**.

Table 3 illustrates the model performance of the proposed method. Benefiting from the temporal convolutional architecture, dilated convolution and the residual unit, the method achieves remarkable predictive accuracy for the permeate flow rate, total permeate flow, permeate concentration and the total permeate concentration. The smaller the RMSE of the model on the test data, the better its general predictive power.

Table 2. Key hyperparameters of the LSTM.

Hyperparameter	Value
Number of LSTM layer	3
Number of Hidden layers	50
Number of dense layers	1
Dropout rate	0.2
Batch size	250
Learning rate	0.001

Table 3. Model performance on the test data based on correlation coefficient and root mean square error (RMSE).

Parameter	Root Mean square Error	Correlation Coefficient
Permeate flow rate [L/min]	0.0083	0.996
Permeate concentration[g/L]	0.0012	0.983
Permeate flow quantity [L]	0.0023	0.996
Total permeate tank concentration [g/L]	0.0052	0.983

Figure 7(a), Figure 8(a), Figure 9(a) and Figure 10(a) reveal a good fit of the LSTM to the training data for the permeate flow rate, total permeate flow, permeate concentration and the total permeate concentration, respectively, and testifies to the model's ability to reflect highly dynamic outputs from highly dynamic training data. The validation to determine the predictive capability of the model on a different data set was performed and Figure 7(b), Figure 8(b), Figure 9(b) and Figure 10(b) show that the model succeeded in capturing the general trends for previously unseen test data for the permeate flow rate, total permeate flow, permeate concentration and the total permeate concentration, respectively.

4.2. LSTM-Based MPC Closed-Loop Control Performance

The MPC controller in this study was implemented in Python version 3.6.5 through the scipy.optimize.minimize function, and the sequential least squares quadratic programming (SLSQP) algorithm was selected as the option for this solver.

The parameters for the MPC controller were set as shown in Table 4 and its main objective was to track a target set point trajectory as fast and as smooth as possible. The LSTM-based system was compared to a system which uses the true model of the RO desalination system and the results will be discussed in the following.

The response graphs in Figure 11 show that the LSTM-based MPC strategy successfully tracks the signal showing the robustness and successful set point tracking ability of the controller employed to RO desalination system. To be able to compare the performance of the two controllers quantitatively, we designed a
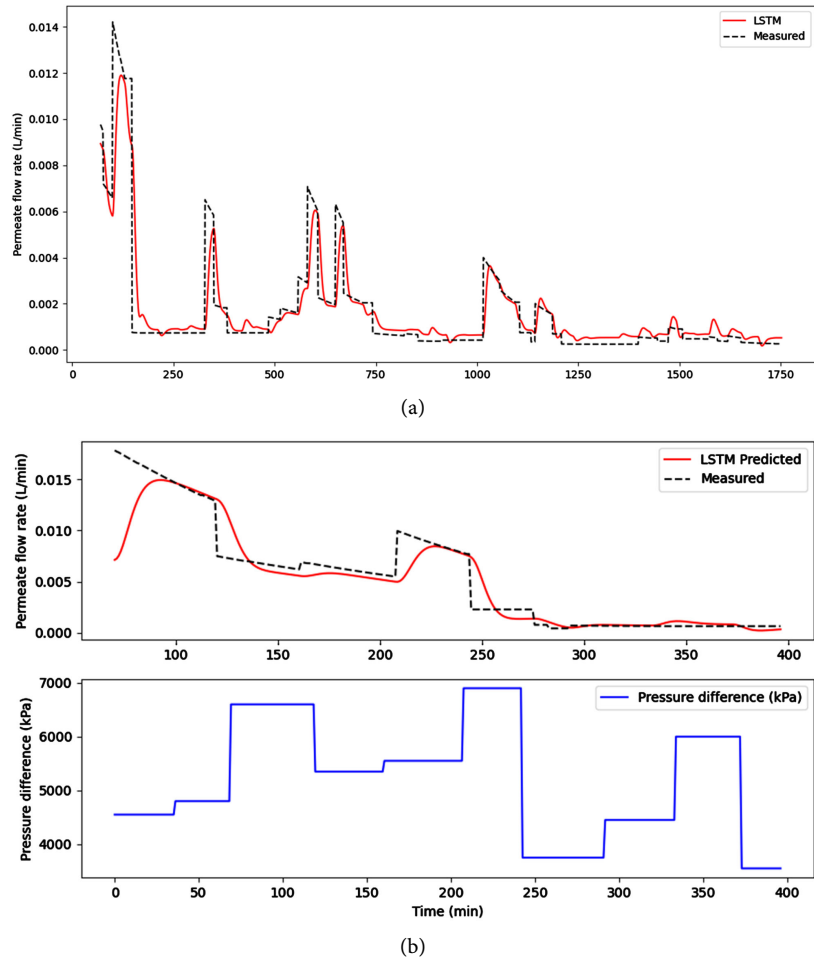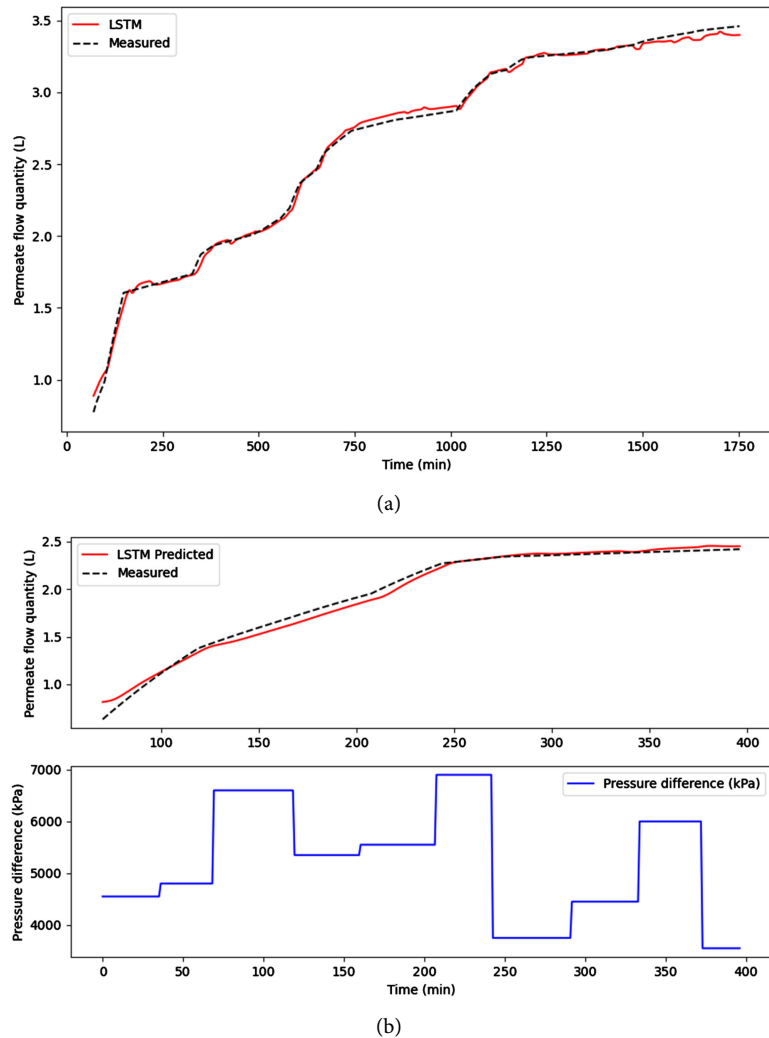
(a)

(b)

Figure 7. System identification performance of the optimised LSTM for the permeate flow rate, M_p. (a) Training performance of the optimised LSTM; (b) Validation performance of the optimised LSTM on test data.

Table 4. Parameter settings for the model predictive controller.

Parameter	Description	Value
ns	Simulation Length	100 min
p	Prediction Horizon	30 steps
m	Control Horizon	10 steps
$u_{min,k}$	Lower and upper bounds for the control action at time-step k	2500 kPa
$u_{max,k}$		7000 kPa
$\Delta u_{min,k}$	The rate of change of the control action at time-step k	10 kPa
$\Delta u_{max,k}$		100 kPa
Q_y	The weight matrices for the controller, Q_y and Q_u	1
Q_u		20
$M_{p,d}$	Upper limit of the permeate flow concentration	3 g/L

(a)

(b)

Figure 8. System identification performance of the optimised LSTM for the total permeate flow, M_p. (a) Training performance of the optimized LSTM; (b) Validation performance of the optimised LSTM on test data.

performance metric I given in Equation (28). This metric gives an indication of how good the LSTM-based MPC is compared to the MPC, which uses the full RO desalination system model as the predictive model.

$$J = \sum_{k=1}^{ns} \left(\left(\hat{y}_k - y_k^* \right)' Q_y \left(\hat{y}_k - y_k^* \right) + \Delta u_k' Q_u \Delta u_k \right) \quad (27)$$

$$I = \left(1 - \frac{J_{\text{LSTM}} - J}{J} \right) * 100\% \quad (28)$$

For the target set point trajectory shown in **Figure 11**, the performance metric I for the LSTM-based MPC was $I = 98.7\%$ which shows slight deviations but a very good performance.

The results of the permeate concentration in **Figure 12** shows that the model predictive controller could achieve close set point tracking (**Figure 11**), while staying in the required constraints of the permeate concentration.
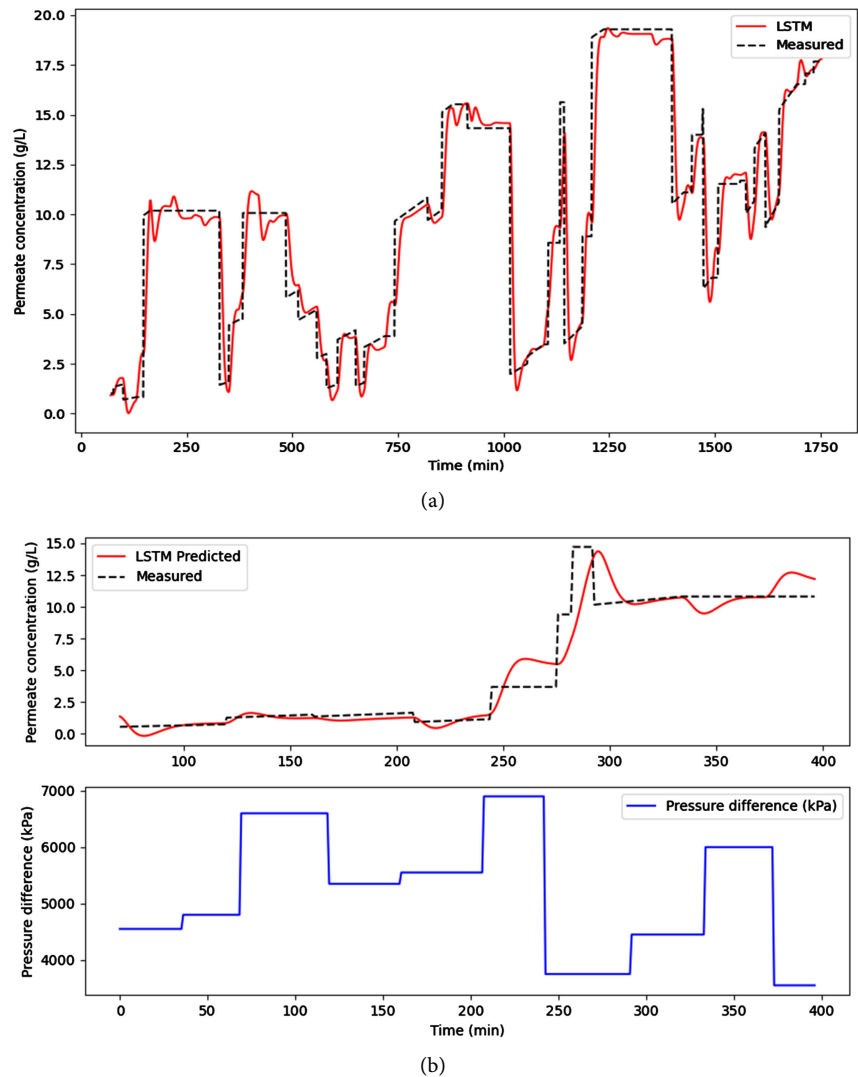
(a)

(b)

Figure 9. System identification performance of the optimised LSTM for the permeate concentration, X_p. (a) Training performance of the optimized LSTM; (b) Validation performance of the optimised LSTM on test data.

5. Conclusion

A nonlinear model predictive controller for RO desalination systems has been presented. To take model uncertainties, constraints, nonlinear dynamics into account, the system utilizes an LSTM Network as the predictive model. The LSTM can capture complex nonlinear dynamic behavior and provide long-range predictions even in the presence of disturbances. The main aim was to control the permeate flow rate obeying the constraints on the permeate concentration by manipulating the feed pressure. The LSTM based MPC was tested on reference signals which exhibits, the possible nonlinear process dynamics occurring inside a real RO desalination plant. It can be seen from the response graphs that the NMPC strategy successfully tracks the reference signal. These results illustrate and prove the tracking ability of LSTM-based MPC controller. Almost offset free and very close set point tracking is obtained using the strategy.
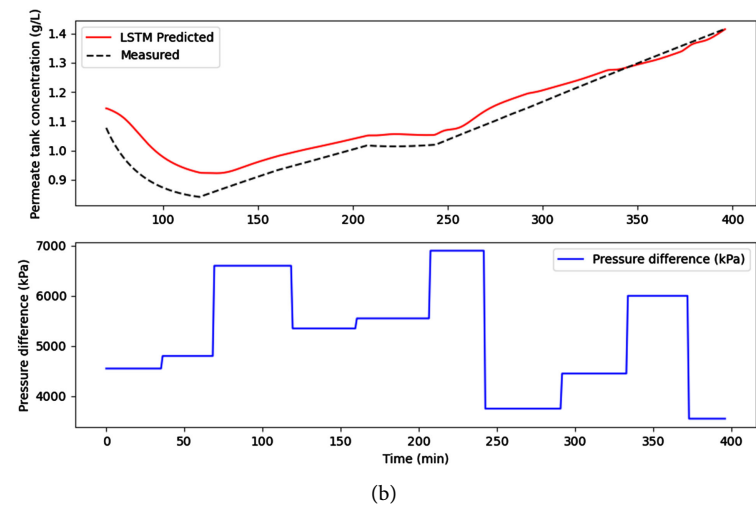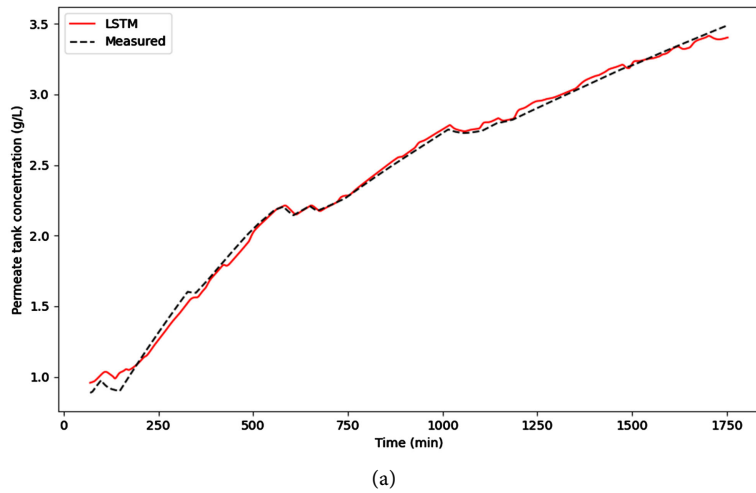
(a)

(b)

Figure 10. System identification performance of the optimised LSTM for the total permeate concentration, C_p. (a) Training performance of the optimized LSTM; (b) Validation performance of the optimised LSTM on test data.
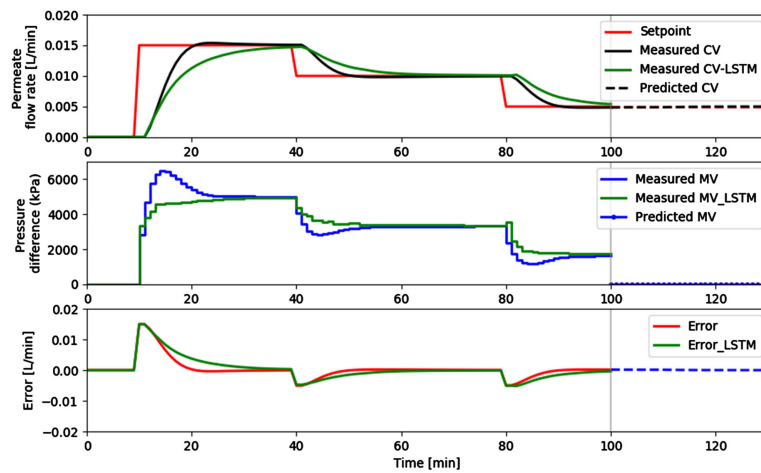

Figure 11. Closed loop results of the two model predictive controllers. Measured CV is the result of the MPC with the true model and measured CV-LSTM is the result of the MPC with the LSTM as predictive model.
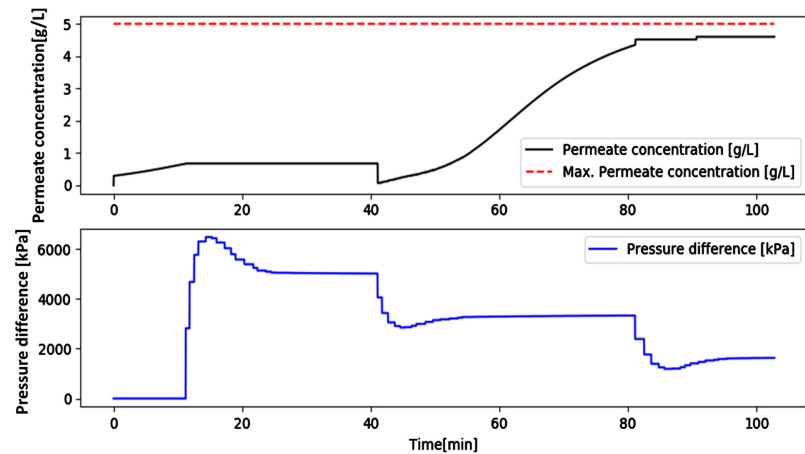
Figure 12. Closed loop results for the permeate concentration, where the red line is the maximum allowed concentration.

Acknowledgements

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] Liu, C.L., Pan, J. and Chang, Y.F. (2016) PID and LQR Trajectory Tracking Control for an Unmanned Quadrotor Helicopter: Experimental Studies. 2016 35*th Chinese Control Conference*, Chengdu, 10845-108503.
https://doi.org/10.1109/ChiCC.2016.7555074

[2] Janghorban, I., Ifaei, P., Rashidi, Z. and Yoo, C.K. (2016) Control Performance Evaluation of Reverse Osmosis Desalination System Based on Model Predictive Control and PID Controllers. *Desalination and Water Treatment*, **57**, 1-8.
https://doi.org/10.1080/19443994.2016.1191776

[3] Lu, H., Liu, C., Coombes, M., Guo, L. and Chen, W.-H. (2016) Online Optimisation-Based Backstepping Control Design with Application to Quadrotor. *IET Control Theory & Applications*, **10**, 1601-1611.
https://doi.org/10.1049/iet-cta.2015.0976

[4] L'Afflitto, A., Anderson, R.B. and Mohammadi, K. (2018) An Introduction to Nonlinear Robust Control for Unmanned Quadrotor Aircraft: How to Design Control Algorithms for Quadrotors Using Sliding Mode Control and Adaptive Control Techniques [Focus on Education]. *IEEE Control Systems Magazine*, **38**, 102-121.
https://doi.org/10.1109/MCS.2018.2810559

[5] Sassi, K. and Mujtaba, I. (2010) Simulation and Optimization of Full Scale Reverse Osmosis Desalination Plant. *Computer Aided Chemical Engineering*, **28**, 895-900.
https://doi.org/10.1016/S1570-7946(10)28150-6

[6] Abbas, A. (2006) Model Predictive Control of a Reverse Osmosis Desalination Unit. *Desalination*, **194**, 268-280. https://doi.org/10.1016/j.desal.2005.10.033

[7] Manenti, F., Nadezhdin, I.S., Goryunov, A.G., Kozin, K.A., Baydali, S.A., Papasidero, D., Rossi, F. and Potemin, R.V. (2015) Operational Optimization of Reverse Osmosis Plant Using MPC. *Chemical Engineering Transactions*, **45**, 247-252.

[8] Bieker, K., Peitz, S., Brunton, S., Kutz, J. and Dellnitz, M. (2019) Deep Model Predictive Control with Online Learning for Complex Physical Systems.

[9] Kargar, M. and Mehrad, R. (2020) Robust Model Predictive Control for a Small Reverse Osmosis Desalination Unit Subject to Uncertainty and Actuator Fault. *Water Supply*, **20**, 1229-1240. https://doi.org/10.2166/ws.2020.043

[10] Feng, G., Lai, C. and Kar, N.C. (2017) Expectation-Maximization Particle-Filter- and Kalman-Filter-Based Permanent Magnet Temperature Estimation for PMSM Condition Monitoring Using High-Frequency Signal Injection. *IEEE Transactions on Industrial Informatics*, **13**, 1261-1270. https://doi.org/10.1109/TII.2016.2591509

[11] Zhao, J. and Mili, L. (2018) Sparse State Recovery versus Generalized Maximum-Likelihood Estimator of a Power System. *IEEE Transactions on Power Systems*, **33**, 1104-1106. https://doi.org/10.1109/TPWRS.2017.2715561

[12] Thune, P. and Enzner, G. (2017) Maximum-Likelihood Approach with Bayesian Refinement for Multichannel-Wiener Postfiltering. *IEEE Transactions on Signal Processing*, **65**, 3399-3413. https://doi.org/10.1109/TSP.2017.2692731

[13] Tang, J., Deng, C. and Huang, G.-B. (2016) Extreme Learning Machine for Multilayer Perceptron. *IEEE Transactions on Neural Networks and Learning Systems*, **27**, 809-821. https://doi.org/10.1109/TNNLS.2015.2424995

[14] Ławryńczuk, M. and Tatjewski, P. (2010) Nonlinear Predictive Control Based on Neural Multi-Models. *Applied Mathematics and Computer Science*, **20**, 7-21. https://doi.org/10.2478/v10006-010-0001-y

[15] Shrivastava, P. and Trivedi, A. (2011) Control of Nonlinear Process Using Neural Network Based Model Predictive Control. *International Journal of Engineering Science and Technology*, **3**, 257.

[16] Antão, R., Antunes, J., Mota, A. and Escadas Martins, R. (2020) Model Predictive Control of Non-Linear Systems Using Tensor Flow-Based Models. *Applied Sciences*, **10**, 3958. https://doi.org/10.3390/app10113958

[17] Fairbank, M., Li, S., Fu, X., Alonso, E. and Wunsch, D. (2014) An Adaptive Recurrent Neural-Network Controller Using a Stabilization Matrix and Predictive Inputs to Solve a Tracking Problem under Disturbances. *Neural Networks*, **49**, 74-86. https://doi.org/10.1016/j.neunet.2013.09.010

[18] Li, S., He, J., Li, Y. and Rafique, M.U. (2017) Distributed Recurrent Neural Networks for Cooperative Control of Manipulators: A Game-Theoretic Perspective. *IEEE Transactions on Neural Networks and Learning Systems*, **28**, 415-426. https://doi.org/10.1109/TNNLS.2016.2516565

[19] Wong, W., Chee, E., Li, J.L. and Wang, X.N. (2018) Recurrent Neural Network-Based Model Predictive Control for Continuous Pharmaceutical Manufacturing. *Mathematics*, **6**, 242. https://doi.org/10.3390/math6110242

[20] Ling, Z.-H., Ai, Y., Gu, Y. and Dai, L.-R. (2018) Waveform Modeling and Generation Using Hierarchical Recurrent Neural Networks for Speech Bandwidth Extension. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, **26**, 883-894. https://doi.org/10.1109/TASLP.2018.2798811

[21] Pan, Y. and Wang, J. (2012) Model Predictive Control of Unknown Nonlinear Dynamical Systems Based on Recurrent Neural Networks. *IEEE Transactions on In-*

dustrial Electronics, **59**, 3089-3101. https://doi.org/10.1109/TIE.2011.2169636

[22] Cheng, L., Liu, W., Hou, Z.G., Yu, J. and Tan, M. (2015) Neural Network Based Nonlinear Model Predictive Control for Piezoelectric Actuators. *IEEE Transactions on Industrial Electronics*, **62**, 7717-7727. https://doi.org/10.1109/TIE.2015.2455026

[23] Liu, W.C., Cheng, L., Zhou, C., Hou, Z.G. and Tan, M. (2016) Neural-Network Based Model Predictive Control for Piezoelectric-Actuated Stick-Slip Micro-Positioning Devices. *IEEE International Conference on Advanced Intelligent Mechatronics*, Banff, 1312-1317. https://doi.org/10.1109/AIM.2016.7576951

[24] El-Dessouky, H.T. and Ettouney, H.M. (2002) Fundamentals of Salt Water Desalination. Elsevier, Amsterdam.

[25] Karimanzira, D., Renkewitz, H., Shea, D. and Albiez, J. (2020) Object Detection in Sonar Images. *Electronics*, **9**, 1180. https://doi.org/10.3390/electronics9071180

[26] Kingma, D.P. and Ba, J. (2014) Adam: A Method for Stochastic Optimization.