Process modelling to facilitate model-based decision-making for resource recovery from urban wastewater - A grey-box approach applied to nanofiltration

Maria O. van Schaik, Iarima Silva Mendonca, Hans J. Cappon, Wei-Shan Chen, Huub H. M. Rijnaarts

1. File description

The zipped *Nf_greybox_model.zip* folder contains all files necessary to run the developed NF models. Those are organized into two folders and this readme file (*README-nf-greybox-model.pdf*). The *NF_parameter_estimation* folder contains the scripts and data for predicting the diffusion parameters of the nanofiltration model, and the *NF_optimization* folder contains the scripts for optimizing the nanofiltration process based on pressure (TMP) and membrane type. The files contained in each folder are the following:

I. Folder NF_parameter_estimation

MATLAB® files (.m) – Main script or function files

1. *NF_CrossValidation.m* – The main script for estimating the parameters. This script uses all the other MATLAB files as functions.
2. *NF_Calibration.m* – Parameter estimation function (with kfold=n cross validation)
3. *Flux_prediction.m* – Flux prediction based on intermediary parameters obtained during calibration
4. *SR_prediction.m* – Salt rejection prediction based on intermediary parameters obtained during calibration
5. *NF_Validation.m – Estimate Flux and SR with the* intermediary parameters (with kfold=n cross validation)
6. *Flux_response.m* – Flux prediction based on final parameters (average of intermediary) obtained during calibration
7. *SR_response.m* – SR prediction based on final parameters (average of intermediary) obtained during calibration
8. *T_correction_flux.m* – Corrects flux to the set temperature (25 °C)
9. *T_correction_SR.m* – Corrects SR to the set temperature (25 °C)
10. *NF_SR.m* – Used by the SR estimation functions to predict the removal
11. *objectiveNFsr.m* – Calculate and minimizes the error between real value and prediction
12. *NF_Test.m* – Estimate flux and SR based on the final parameters

MATLAB® files (.mat) - Files that store workspace variables

1. *Membranes.mat* – Contains name, material and MWCO of membranes
2. *TMP_bounds.mat* – Contains upper and lower bounds of TMPs used to estimate the parameters. This is the pressure range that the estimated parameters are supposed to be used.
3. *p_values.mat* – The estimated parameters, A and Bs.

Excel® files (.xlsx) – Spreadsheets for input and output data

1. *NF_90_norm.xlsx* – Normalized data (TMP, flow and salt rejection) for Dow NF90 membrane.
2. *NF_270_norm.xlsx* – Normalized data (TMP, flow and salt rejection) for Dow NF270 membrane.
3. *T_correction.xlsx* – Values of water viscosity for different temperatures. Used to correct temperature to 25 °C.
4. *NF_Estimation.xlsx* – Saved results obtained for parameters, predictions and errors.

NF_Estimation_example.xlsx – Saved results obtained for parameters, predictions and errors of this study (NF270 and NF90).

5. *NF_normalized_template.xlsx* – (Assistance file) used for normalizing the data.

II. Folder NF_optimization

MATLAB® files (.m) – Main script or function files

1. *NF_optimization.m* – The main script for NF optimization. This script uses all the other MATLAB files as functions.
2. *NF_opt_removal.m* – This function is used when the optimization is done in terms of removal (SR%).
3. *NF_opt_concentration.m* – This function is used when the optimization is done in terms of concentration of pollutants in the effluent.
4. *NF_MaxOptim.m* – This function uses the result of function *Calculate_SR_Flux.m* to group the possibilities of flux and salt rejection into one matrix. This matrix is later used to choose the best condition.
5. *NF_costOptim.m* – This function calculates costs associated with energy (pumps) and with each required membrane area (flux and water recovery dependent). Those costs are associated in one matrix that is later used to choose the best condition.
6. *Calculate_SR_Flux.m* – This function calculates the salt rejection and flux for each membrane and TMP. It uses the parameters estimated in the NF cross-validation (folder I).
7. *Graphic_subplots.m.m* – This function is used to plot the graphical results.

Excel® files (.xlsx) – Spreadsheets for input and output data

1. NF_influent_optimization.xlsx – File containing the influent characteristics.

2. Using the code

All files are stored in the zipped folder *Nf_greybox_model.zip*. To see and use the files, simply unzip (extract) the folder.

2.1. Parameter estimation

I. Open the file *NF_CrossValidation.m,* located in the folder *NF_parameter_estimation*, in MATLAB. Input from the user is only necessary in this main script where the data is loaded. The script is built for solving the parameters for two membranes, NF270 and NF90, and contains the specification of material and MWCO:

```
Data preparation

Insert number and name of membranes being analyzed
n_memb = 2;
names_memb = ["NF270" ; "NF90"];
material = ["TFC"; "TFC"];
MWCO = ["150-300"; "200-400"];
```

II. The next step is to load the data to the file. The script currently take the data from two Excel files: *NF_90_norm.xlsx* and *NF_270_norm.xlsx.* In order to add more data points to be computed, the modification has to be made in these Excel files.
The user can also replace those to files with a different set of data (even new membrane). For this, update the script with the name of the new file in the *readmatrix* command:

```
Prepare data
Load measured data

if i == 1 % First membrane
input_flux = readmatrix('NF_270_norm.xlsx','Sheet',1);
input_COD = readmatrix('NF_270_norm.xlsx','Sheet',2);
input_N = readmatrix('NF_270_norm.xlsx','Sheet',3);
input_P = readmatrix('NF_270_norm.xlsx','Sheet',4);
end
```

The script make use of normalized values (simply the current value divided by the maximum). If you'd like to add more data and those need to be normalized, you can use the file *NF_normalized_template.xlsx* file to calculate the normalized values to be added to the input sheet.

The Excel sheet has to be organized in a way that the sheets follow the order: Sheet 1 for Flux, 2 for COD, 3 for TN and 4 for TP. Check either *NF_90_norm.xlsx* and *NF_270_norm.xlsx* to see the proper way of organizing the data.

Please note that if you'd like to add an extra membrane (higher then 2 membranes), this would have to be manually modified in the script.

III. The user must also provide to the script the maximum flux and TMP contained in the data set for each membrane. When using the *NF_normalized_template.xlsx,* the maximum values are also displayed.

```
% Insert operational parameters
Flux_max = [185.6; 86.8];  % maximum flux for each membrane
TMP_max = [20; 20];        % maximum TMP for each membrane
```

IV. After all information is provided, you can RUN the code.

V. The result, i.e., parameters estimated, the predictions and errors, will be saved in the excel file *NF_Estimation.xlsx.* If you'd like to see the results of the membranes in this study, you can check the file NF_Estimation_example.xlsx.

VI. Running the script will also plot several graphs: boxplots of the estimated parameters and the prediction x actual data comparison. All plots are made for each membrane.

VII. By the end of the script, the parameters estimated are saved in the file *p_values.mat* and are then used in the optimization script.

2.2. Optimization model

I. Open the file *NF_Optimization.m,* located in the folder *NF_Optimization.* Input from the user is only necessary in this main script where the data is loaded. This script will perform the optimization (membrane and TMP selection) based on the information provided on the *p_values.mat* saved on the previous script. The estimated parameters and TMP ranges are contained in this file.

II. In the file *NF_Optimization.m,* the user must provide the influent data, i.e., flow in L/h and concentration of COD, TN and TP in mg/L. This is done through the Excel file *NF_influent_optimization.xlsx*. If you would like to change any of the influent data, it should be done in this respective Excel file.

III. After providing the input data, the script asks for a certain target in order to perform the optimization. This is first set through the variable *T*. *T* can receive a value of 0, 1 or 2; meaning the following:
  - T = 0, for no targets; the script then calculates the best combination of SR and costs: highest pollutant rejection at lower cost.
  - T = 1, for targets in removal percentages.
  - T = 2, for targets as desired effluent concentrations (mg/L).

IV. If the user would like to set a target, that is, *T* has received a value of either 1 or 2, the targets have now to be set specifically per pollutant (COD, TN and TP). The script requires now two inputs from the user.

The first is the target itself, that should be placed in the variable *target_#PollutantName#* (*e.g. target_COD*). According to the value given to variable T, the target value has to be either between 0 and 1 (for percentage removal) OR concentration (in mg/L).

The second input, called *limit_#PollutantName#* (*e.g. limit_COD*), determines if the pollutant should be removed or pass through the membrane.

If the pollutant must be removed: it must be limited as 'above the target'
If the pollutant must pass the membrane: it must be limited as 'below the target'
For this, use:
  - Above the target, then, *limit_#PollutantName#* = 1
  - Below the target, then, *limit_#PollutantName#* = 0

This is useful if the user would like to know the operational conditions to separate pollutants in different streams. See the example below:

```
T = 1;
%COD
target_COD = 0.7;
limit_COD = 1;
%TN
target_TN = 0.4;
limit_TN = 0;
```

According to the information added, the target *T* is set for removal percentages (T=1), in which at least 70% of COD should be retained and at least 40% of TN should pass through the membrane.

V. A target for water recovery must also be set (value between o and 1). If no target is set, the default value of 70% recovery will be used. This variable is used to calculate the area necessary to achieve this recovery.

VI. Finally, the user can choose either to plot the graphic results or not. This is defined by the variable *p*: 1 = yes, 0 = no

VII. After all information is provided, you can RUN the code.

VIII. Besides the plots, the model will provide the following information:

- Optimized parameters: TMP, membrane and A (water permeability).
- Effluent concentration
- The salt (pollutant) rejection
- Water recovery and flux
- Energy, area and associated costs.