Machine Learning Based
 Control System Design for BSM1 Wastewater
 Treatment Plant

> Yu Yang, Chengyan Ye Presenter: Chengyan Ye



General overview of the BSM1 plant



Ο

Figure 1: General overview of the BSM1 plant

BSM1 MATLAB SIMULINK OUTLINE



Variable	Value
N _{tot}	<18 g N.m ⁻³
COD _{tot}	<100 g COD.m ⁻³
$S_{\rm NH}$	<4 g N.m ⁻³
TSS	<30 g SS.m ⁻³
BOD_5	<10 g BOD.m ⁻³
1025	STO B DOD.III

Effluent quality limits

Effluent violations

95% percentile for effluent SNH (Ammonia95) = 10.6066 g N/m3 95% percentile for effluent TN (TN95) = 20.1673 g N/m3 95% percentile for effluent TSS (TSS95) = 20.9932 g SS/m3

The maximum effluent total nitrogen level (18 mg N/1) was violated during 2.1979 days, i.e. 31.3988% of the operating time. The limit was violated at 9 different occasions.

The maximum effluent ammonia nitrogen level (4 mg N/l) was violated during 4.6667 days, i.e. 66.6667% of the operating time. The limit was violated at 8 different occasions.

Effluent quality violations

- Problem: Three (Dry, Rain, Storm) scenarios have violations on effluent quality
- Goal: Improve the effluent quality (lower the violations) by using better control system design



Data Driven Modeling using PRBS signal

- Pseudorandom Binary Sequence(PRBS)
 - Binary Sequence (two values)
 - Random
 - Commonly using for testing and evaluating systems



Bad fitting with traditional data driven methods

- Input:
 - Internal Recycle Flow Rate
 - KLA5 (oxygen transfer coefficient)
- Output:
 - NO3-N(Nitrate) in the Second Anoxic Compartment (SNO2)
 - Dissolved Oxygen in the Last Aerated Compartment (SO5)
- Best fit range: 0-3 %

Machine Learning Based Control

+

0

- We model the control system rather than modelling the process
- Methodology
 - o Generate 15 rain and 15 storm scenarios
 - Generate random combinations
 of PI parameters and setpoints
 - o Run BSM1 simulation and collect the data
 - o Build the regression model
 - Use the regression model to search the optimal combinations of PI
 - parameters and setpoints

Generating Scenarios

Three dynamic influent (Dry, Rain, and Storm)

Rain influent: Dry influent + 1 rain event

Storm influent: Dry influent + 1 storm event + 1 rain event

Following the pattern, we modify when the event starts, how long it lasts, and the magnitude of the peak and trough.

 \mathbf{O}



Generating Scenarios

- Pre-control phase: 0-7s (first week)
- Control action: 7s 14s (second week)
- Influent soluble concentrations (Ss, SNH, SND, SI)
- Influent particulate concentrations (Хі, Хвн, Хва, Хѕ, Хмд)
- Methodology of generating rain scenarios
 - $\circ~$ Modify the influent flow
 - Modify the load concentrations follow the dilution calculation

Simulation and data collection

Generate random combinations of PI parameters and setpoints

Run simulations of each combination of PI parameters and setpoints (total 9 variables) with 30 scenarios Collect the jointed violation time of each simulation

3

Take the average violation time of 30 scenarios with each combination of PI parameters and setpoints

4

0

Machine Learning Based Regression model

Predictors: combination of PI parameters and setpoints (total 9 variables)

Response: the average violation time

Using the machine learning regression model, we can estimate the relationship between Predictors and Response

Once we built the model, we can use it to make prediction of new preditors

Modelling result



Violation time **default PI performance** (no sensornoise) Dry: 68.6429% Rain: 73.3631% Storm: 71.7262%



Simulation result

400 sets data

Range: 66.9296% to 80.8681% +

Machine Learning Based Modelling

Model 2.25

result

MOC	Jeis	
Sort	t by RSquared (V 🔻	it EBG
	2.25 Neural Network	R-Squared (Validation): 0.84
Last	change: Bilayered Neural	Network 9/9 features
	2.23 Neural Network	R-Squared (Validation): 0.83
Last	change: Medium Neural N	letwork 9/9 features
	2.17 Ensemble	R-Squared (Validation): 0.79
Last	change: Bagged Trees	9/9 features
슸	2.24 Neural Network	R-Squared (Validation): 0.78
Last	change: Wide Neural Net	work 9/9 features
슸	2.22 Neural Network	R-Squared (Validation): 0.72
Last	change: Narrow Neural Ne	etwork 9/9 features
슲	2.6 Tree	R-Squared (Validation): 0.69
Last	change: Medium Tree	9/9 features
☆	1 Tree	R-Squared (Validation): 0.66
Last	change: Fine Tree	9/9 features
슸	2.5 Tree	R-Squared (Validation): 0.66
Last	change: Fine Tree	9/9 features
슲	2.26 Neural Network	R-Squared (Validation): 0.65
Last	change: Trilayered Neural	Network 9/9 features
슲	2.18 Gaussian Proce	R-Squared (Validation): 0.56
Last	change: Squared Exponent	ntial GPR 9/9 features
슲	2.7 Tree	R-Squared (Validation): 0.56
Last	change: Coarse Tree	9/9 features
⊼] ≰	2 21 Gaussian Proce	P-Sourced (Validation): 0.56

· · · · · · · · · · · · · · · · · · ·	
Model 2.25: Neural Net Status: Trained	work
Training Results RMSE (Validation)	0.725
R-Squared (Validation)	0.84

Model 1 Summarv ×

it equales (remeaning)	
MSE (Validation)	0.52621
MAE (Validation)	0.41806
Prediction speed	~5200 obs/sec
Training time	4.2216 sec
Model size (Compact)	~8 kB

Model 2.1

Model Hyperparameters

Preset: Bilayered Neural Network Number of fully connected layers: 2 First layer size: 10 Second layer size: 10 Activation: ReLU Iteration limit: 1000 Regularization strength (Lambda): 0 Standardize data: Yes

Feature Selection: 9/9 individual features selecte

- PCA: Disabled
- Optimizer: Not applicable





Neural network

- Input layer: Predictors (9 variables)
- First layer: 10 neurons
- Second layer: 10 neurons
- Output layer: Response
- Each neuron has its own set of weights and a bias
- Each neuron output can be defined by 2 steps

 $\circ X_{1}^{*} W_{1} + X_{2}^{*} W_{2} + b$

 $\circ\,$ An activation function (ReLU)

```
\operatorname{ReLU}(x) = \begin{cases} x, & x > 0\\ 0, & x \le 0 \end{cases}
```

• Use the previous layer values to calculate its output

Continuing work

Use

Generate

Use the regression model to make predictions

parameters and setpoints (total 9 variables)

Generate more combination of PI

Search

Search the optimal combination with the lowest value of response

References

- (1) Alex, Jens & Benedetti, Lorenzo & Copp, Jb & Gernaey, Krist & Jeppsson, Ulf & Nopens, Ingmar & Pons, Marie-Noelle & Rieger, Leiv & Rosen, Christian & Steyer, J-P. (2008). Benchmark Simulation Model no. 1 (BSM1). Report by the IWA Taskgroup on Benchmarking of Control Strategies for WWTPs.
- (2) Vanhooren H., Nguyen K. (1996) Development of a simulation protocol for evaluation of respirometry-based control strategies. Report University of Gent and University of Ottawa.
- (3) Melcher, K. (n.d.). A friendly introduction to [deep] neural networks. KNIME. <u>https://www.knime.com/blog/a-friendly-introduction-to-deep-neural-networks</u>