DATA-DRIVEN MODEL-FREE SYSTEM IDENTIFICATION OF BATCH DISTILLATION COLUMN USING XGBOOST MACHINE LEARNING ALGORITHM

SISTEM IDENTIFIKASI TANPA MODEL BERBASIS DATA UNTUK KOLOM DISTILASI BATCH MENGGUNAKAN ALGORITMA PEMBELAJARAN MESIN XGBOOST

Hayati Amallia[1,2], Dimitri Mahayana[1], Pranoto Hidaya Rusmin[1]
[1] School of Electrical Engineering and Informatics, Institut Teknologi Bandung, Bandung, Indonesia
[2] Direktorat SNSU – TK, Badan Standardisasi Nasional, Tangerang Selatan, Indonesia
Email: hayatiamaliahayati@gmail.com

ABSTRACT

The distillation process is crucial in the chemical industry, enabling material separation, purification, and waste product disposal. Distillation columns, including the batch type, are widely used in industries due to their ability to produce raw materials for various applications. However, modeling and controlling batch distillation columns pose challenges due to their nonlinear and dynamic behavior. This paper presents a novel data-driven approach for system identification using XGBoost, an advanced gradient boosting algorithm, eliminating the need for explicit model equations. The proposed methodology leverages the power of XGBoost to learn the underlying system behavior directly from data. The paper overviews the methodology, including data preprocessing, feature engineering, training the XGBoost model, and evaluating its performance. Techniques such as cross-validation and input feature delay tuning are also discussed to ensure robustness and optimal model performance. The approach's effectiveness is demonstrated through various case studies and some comparisons. The results highlight the capability of the proposed model-free system identification methodology using XGBoost in accurately capturing the dynamics of batch distillation systems, with the smallest Mean Absolute Error (MAE) measuring 0.219 on the training dataset and 0.45 on the testing dataset, utilizing 10 input features comprising the current reflux valve ratio, delayed reflux valve ratio from 1 to 4-time steps ($n_a = 4$), and delayed ethanol concentration from 1 to 5-time steps ($n_b = 5$).

Kata Kunci: *distillation columns; data-driven; system identification; XGBoost; model free*

ABSTRAK

Proses distilasi memiliki peran yang penting dalam industri kimia diantaranya yaitu untuk pemisahan bahan, proses pemurnian, dan pembuangan produk limbah. Kolom distilasi, khususnya tipe batch, banyak digunakan dalam industri karena kemampuannya untuk menghasilkan bahan-bahan baku untuk berbagai keperluan. Namun, pemodelan dan pengendalian kolom distilasi tipe batch mempunyai banyak tantangan karena perilaku nonlinier dan dinamisnya. Makalah ini menyajikan pendekatan baru berbasis data untuk identifikasi sistem menggunakan XGBoost, algoritma gradient boosting *yang canggih, yang menghilangkan kebutuhan akan persamaan model eksplisit. Metodologi yang diusulkan memanfaatkan kelebihan XGBoost untuk mempelajari perilaku sistem langsung dari data. Makalah ini memberikan gambaran tentang metode tersebut, termasuk pra-pemrosesan data, fiturfitur, pelatihan model XGBoost, dan evaluasi kinerjanya. Teknik seperti cross-validation dan penalaan delay fitur input juga dibahas untuk memastikan kekokohan dan kinerja model yang optimal. Keefektifan pendekatan ini ditunjukkan melalui beberapa studi kasus dan beberapa perbandingan. Hasil dari penelitian menyoroti kemampuan metode identifikasi sistem tanpa model yang diusulkan menggunakan XGBoost untuk menggambarkan dengan akurat dinamika sistem distilasi tipe batch dengan Mean Absolute Error (MAE) paling kecil adalah sebesar 0,219 pada dataset pelatihan dan 0,454 pada dataset pengujian ketika digunakan 10 fitur input yang mencakup nilai rasio katup refluks pada waktu saat ini, nilai rasio katup refluks dengan penundaan sebanyak 1 hingga 4 langkah waktu ($n_a = 4$), dan nilai konsentrasi ethanol dengan penundaan sebanyak 1 hingga 5 langkah waktu ($n_b = 5$).*

Keywords: *kolom distilasi; berbasis data; identifikasi sistem; XGBoost; tanpa model*

1. INTRODUCTION

One of the processes in the chemical industry that has received much attention in recent years is the distillation process (Bravo, 2019). Distillation is usually used for material separation, purification, and disposal of waste products which the process is carried out by utilizing the boiling points difference of the materials being treated and using the principle of condensation (Orozco et al., 2017). Distillation is considered one of the most essential processes in the chemical industry because the products produced through this process are widely used as raw materials to produce other materials. This distillation process uses a particular instrument called a distillation column.

There are two types of distillation columns, i.e., continuous type and batch type. In the continuous-type distillation column, the input material is given continuously to produce the distillate continuously and ceaselessly. Usually, this type of distillation column is used by large-scale industries to produce distillate in large quantities. The second type, the batch-type distillation column, can only produce a limited amount of distillate because the input material is not supplied continually. This type generally does not have a feeding tray so the input material is only given once in one distillation process and is fed at the beginning of the process before the distillation process is taken place. Therefore, industries using the batch type of distillation column are usually medium or small-scale industries because it is more efficient and can reduce production costs (Diwekar, 2011).

Due to its significant role in the chemical industry and the necessity of its products as raw materials for other industries, numerous studies have been conducted regarding the Batch Distillation Column (BDC). The main focus of these studies generally revolves around two key issues: how to model the distillation system under nonlinear conditions and the strong influence of the dynamic behavior of chemical processes, as well as how to design a control system capable of producing the desired concentration distillate in a BDC (Ogunnaike & Ray, 1994). Furthermore, this distillation system is renowned for its difficulty in mathematical modeling due to its high degree of nonlinearity. Nevertheless, System Identification is critical in comprehending and modeling complex dynamic systems. Conventional approaches to system identification often rely on model-based techniques, which necessitate prior knowledge of system dynamics and assumptions about the underlying equations. However,

obtaining an accurate mathematical model proves challenging or unfeasible in numerous real-world scenarios due to the system's complexity or lack of comprehensive information. In order to tackle this challenge, data-driven approaches have garnered significant attention, wherein machine learning algorithms can directly extract valuable insights and patterns from observed data.

In 2017, a linear modeling approach and four advanced control systems using predictive control, fuzzy logic control, gain scheduling, and hybrid control were carried out by (Zou et al., 2017) to be applied to the distillation column with better results when compared to the PID controller. Research to develop a controller with a linear approach was also carried out by (Mendis et al., 2019) and (Zou et al., 2006) but by applying a different controller, i.e., Linear Model Predictive Controller compared to Nonlinear Model Predictive Controller and Linear Model Predictive Controller (MPC) compared to the PI controller, respectively. Furthermore, 2017, (Orozco et al., 2017) used a nonlinear approach to compare the performance of the distillation column model for two conditions of relative volatility in the binary BDC, i.e., constant volatility and relative volatility. (Maulidda et al., 2018) Furthermore, (Rohman et al., 2018) attempted to model a distillation column using nonlinear and linear approaches, respectively, on a batch-type distillation column owned by Honeywell Laboratory at Sekolah Teknik Elektro dan Informatika-Institut Teknologi Bandung (STEI-ITB). Based on these studies, modeling using a linear approach was considered better than a nonlinear one. In the following years, (Ayu et al., 2019), (Nahrendra et al., 2019), and (Putri, 2021) designed several controllers to be applied to the BDC, i.e., using robust PI/PID controller, Data-Driven Control (DDC), and optimal control with a linear-quadratic approach, respectively.

This paper aims to introduce a novel data-driven model-free system identification methodology based on XGBoost, an advanced gradient boosting algorithm that eliminates the need for explicit model equations. By leveraging the power of XGBoost, this methodology enables accurate system identification and prediction by directly learning the underlying system behavior from data. The paper comprehensively overviews the proposed data-driven model-free system identification methodology using XGBoost. The key steps involve data preprocessing, feature engineering, training the XGBoost model, and evaluating the model's performance.

Cross-validation and input feature delay tuning are also discussed to ensure robustness and optimal model performance.

2. BATCH DISTILLATION COLUMN IN HONEYWELL LABORATORY OF STEI - ITB

This paper aims to introduce a novel data-driven model-free system identification methodology based on XGBoost, an advanced gradient boosting algorithm that eliminates the need for explicit model equations. By leveraging the power of XGBoost, this methodology enables accurate system identification and prediction by directly learning the underlying system behavior from data. The paper comprehensively overviews the proposed data-driven model-free system identification methodology using XGBoost. The key steps involve data preprocessing, feature engineering, training the XGBoost model, and evaluating the model's performance. Cross-validation and input feature delay tuning are also discussed to ensure robustness and optimal model performance.

Figure 1. Batch Distillation Column in Honeywell Laboratory of STEI - ITB

Located at the upper part of the column, there is a reflux valve that regulates the vapor flow to condenser 1 and condenser 2. Condenser 2 converts vapor into a distilled liquid product known as distillate. The distillate then moves toward a storage container integrated with an MQ3 sensor to measure its concentration level. Additionally, condenser 2 plays a role in converting vapor into reflux liquid to be returned to the heater pot. The piping and instrumentation diagram of the BDC is depicted in Figure 2.

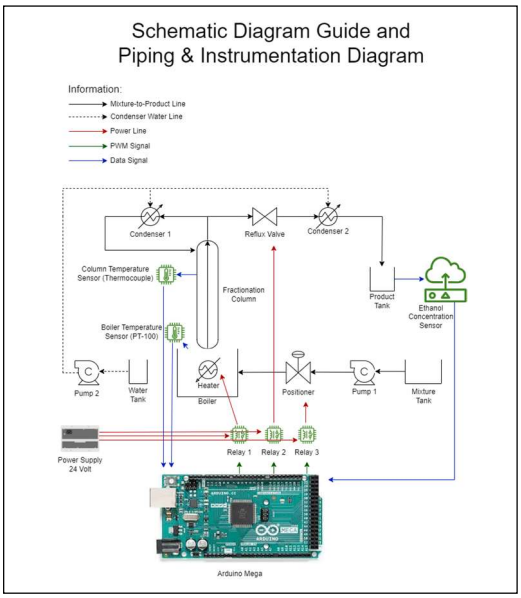


Figure 2. Piping and Instrumentation Diagram (P&ID) of BDC

The reflux valve positioned at the upper section of the column allows for adjustment of the valve opening using a ratio (θ) ranging from 0 to 1. θ of 0 indicates the fully closed position of the reflux valve. At the same time, θ of 1 signifies the fully open position. In the 0 position, all the vapor generated in the heater pot is directed solely to condenser 1, which is converted into reflux and returned to the pot. However, the vapor flow is bifurcated when the reflux valve is set to a non-zero position. Some vapor flows to condenser 1, transforming into reflux and returning to the heater pot. In contrast, the remaining portion is directed to condenser 2, where it is converted into distillate, the end product of the distillation process. Within the modeling system framework employed in this research, the reflux valve is an input parameter that can be manipulated to attain a specific concentration of the distillate concentration output.

The modeling of the BDC in the Honeywell laboratory of STEI-ITB has been conducted multiple times. The first research was conducted in 2017 by (maulidda, 2017), who modeled the non-linear BDC based on component mass balance, vapor-liquid equilibrium, and other physical characteristics. In this study, a mathematical model was derived from non-linear state space equations and implemented for systems with reflux valves set at openings of 1 and 0.5. Simulation results using Matrix Laboratory (MATLAB) software were then compared to the actual system behavior for validation purposes. During validation, the simulation results matched the trend of the actual system experiment, wherein the concentration decreased from approximately 96% to around 20% at a specific time, regardless of whether the valve opening was set at 1 or 0.5. However, there were differences in the transient behavior between the simulation and actual experiment graphs due to several assumptions made during the mathematical modeling process. The Root Mean Square Error (RMSE) obtained from this research was

significant, with a value of 45.4525 for θ of 1 and 31.0251 for θ of 0.5.

Subsequent research related to BDC modeling was conducted in 2018 by (Rohman et al., 2018), utilizing a black-box modeling scheme based on real input and output data from experiments. This study continued the previous white-box modeling method in 2017, which still had a significant RMSE. Compared to the previous research, modeling with a linear approach in this study yielded better fitness for the actual experimental data. The resulting mathematical model for θ of 1, with the sensor placed at the product tank inlet, exhibited better fitness and shorter delay time than the θ of 0.5. This black-box modeling approach was considered more practical and facilitated controller design, although it did not address nonlinearities such as temperature effects. Nevertheless, the performance of the BDC system would still be enhanced by implementing a closed-loop control system compared to an open-loop system.

In 2019, two studies were conducted to design a BDC as a Cyber-Physical System (CPS) to address the challenges of Industry 5.0 technology. These studies were carried out by (Ayu et al., 2019) and (Nahrendra et al., 2019). One of the challenges faced in network-based systems like this is the presence of delays caused by network traffic. In (Ayu et al., 2019), modeling was performed using a black-box modeling scheme, resulting in a model in the form of a First Order Plus Delay Time (FOPDT) model. The CPS-related research was then continued by (Nahrendra et al., 2019), who designed an adaptive control system using a Data-Driven Control (DDC) approach. This study modified the previous CPS and applied recurrent neural network (RNN) for model identification. The common aspect in both studies was comparing the response obtained using the designed controller with the response obtained using conventional PID control. Both studies yielded favorable responses when applied to the CPS, compared to conventional PID control.

The latest research related to BDC modeling in the Honeywell ITB laboratory was conducted by (Putri, 2021) in 2021, employing Non-Linear ARX Neural Network (NARX-NN) modeling, an approximation to linear form using ARMA Neural Network (ARMA-NN) and a direct approach (ARMA-direct method). According to this study, the identification of BDC is best performed using NARX-NN with a linear approximation using ARMA-NN.

3. LEARNING WITH EXTRA GRADIENT BOOSTING (XGBOOST)

Extra Gradient Boosting (XGBoost) is one of the ensembles learning algorithms that utilizes decision trees and incorporates boosting in the learning process. Unlike random forest, which uses the bagging concept and directly creates complete trees, the boosting concept in XGBoost begins the learning process by creating a single leaf rather than a stump or a complete tree. Like any algorithm, XGBoost has its own set of advantages and disadvantages. Some advantages include its optimization for speed in handling large datasets with many features. XGBoost incorporates built-in L1 (LASSO) and L2 (ridge) regularization techniques, preventing overfitting and improving the model's generalization to unseen data.

Additionally, it performs tree pruning during training, eliminating unnecessary branches and contributing to a more robust and straightforward model. The algorithm also features an efficient mechanism for handling missing data, reducing the need for extensive pre-processing of datasets. Conversely, XGBoost's disadvantages are primarily related to computational complexity. Despite its efficiency, XGBoost can be computationally intensive, especially when dealing with large datasets and complex models, necessitating substantial computing resources. Moreover, XGBoost has several hyperparameters that require tuning for optimal performance, and finding the right combination may involve significant experimentation. Without proper parameter tuning, there is a risk of overfitting, particularly with small datasets. Furthermore, like other ensemble methods, XGBoost is considered a "black-box" model, posing challenges in explaining the underlying decision-making process to non-experts.

The trees in XGBoost are formed to minimize the objective function, as equation [1] indicates. The first term in this function is the loss function. The second term is the regularization parameter, where y_i is the observed value of the i-th instance, $\hat{y}_i^{(t-1)}$ is the predicted value of the i-th instance in the previous iteration, $f_t(x_i)$ is the output value from the i-th tree or, in some literature ((Chen & Guestrin, 2016), (Pei et al., 2022)), referred to as the weight (ω_j), and $\Omega(f_t(x_i))$ is the regularization function defined by equation [2 (Chen & Guestrin, 2016). Second-order Taylor approximation is used to find the optimal solution for the loss function term, as shown in equation [3 , where g_i and h_i are

indicated by equation [4 and [5 , respectively.

$$\mathcal{L}^{(t)} = \sum_{i=1}^{n} l\left[y_i, \hat{y}_i^{(t-1)} + f_t(x_i)\right] + \Omega(f_t(x_i)) \quad [1]$$

$$\Omega\left(f_t(x_i)\right) = \gamma T + \frac{1}{2}\boldsymbol{\lambda} f_t^2(x_i) \quad [2]$$

$$\mathcal{L}^{(t)} = \sum_{i=1}^{n} \left[l\left(y_i, \hat{y}_i^{(t-1)}\right) + g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i)\right] + \gamma T + \frac{1}{2}\boldsymbol{\lambda} f_t^2(x_i) \quad [3]$$

$$g_i = \left[\frac{d}{d\hat{y}_i^{(t-1)}} l\left(y_i, \hat{y}_i^{(t-1)}\right)\right] f_t(x_i) \quad [4]$$

$$h_i = \frac{1}{2}\left[\frac{d^2}{d(\hat{y}_i^{(t-1)})^2} l\left(y_i, \hat{y}_i^{(t-1)}\right)\right] f_t^2(x_i) \quad [5]$$

Next, after performing partial derivative operations on equation [3 concerning f_t and setting it equal to zero, the optimal value of $f_t(x_i)$ is obtained, as shown in equation [6], and the corresponding value of $\mathcal{L}^{(t)}$ at that optimal value is indicated by equation [7].

$$f_t(x_i) = -\frac{\sum_{i=1}^{n}[g_i)]}{\sum_{i=1}^{n}[h_i] + \boldsymbol{\lambda}} \quad [6]$$

$$\mathcal{L}^{(t)} = -\frac{1}{2}\frac{(\sum_{i=1}^{n}[g_i])^2}{\sum_{i=1}^{n}[h_i] + \boldsymbol{\lambda}} + \gamma T \quad [7]$$

In forming the tree structure, XGBoost must split the leaf to create new branches. Assuming there is a left branch (L), a right branch (R), and the root (N). The loss reduction after the splitting process is indicated by equation [8. This loss reduction is used to evaluate the candidate tree structures and serves as an indicator of whether pruning should be performed or not. Pruning is carried out when the obtained $\mathcal{L}_{split}$ value is negative, while pruning is not performed if the $\mathcal{L}_{split}$ value is positive.

$$\mathcal{L}_{split} = \left[\left[\frac{(\sum_{i=1}^{n}[g_i])^2}{\sum_{i=1}^{n}[h_i] + \boldsymbol{\lambda}}\right]_L + \left[\frac{(\sum_{i=1}^{n}[g_i])^2}{\sum_{i=1}^{n}[h_i] + \boldsymbol{\lambda}}\right]_R - \left[\frac{(\sum_{i=1}^{n}[g_i])^2}{\sum_{i=1}^{n}[h_i] + \boldsymbol{\lambda}}\right]_n\right] - \gamma \quad [8]$$

The learning algorithm in XGBoost employs a greedy algorithm. When considering all possible splits on all input features, XGBoost utilizes what is known as the exact greedy algorithm. However, learning using the exact greedy algorithm can be time-consuming in cases where the number of features is enormous. Therefore, for scenarios where XGBoost needs to learn from a large number of input features, XGBoost divides these features into quantiles. The splitting is then performed based on these quantiles.

In addition to the greedy algorithm XGBoost uses, several other techniques are employed to accelerate the learning process. The first technique is called

parallel learning, which involves dividing the feature dataset so that multiple computers can simultaneously participate in the learning process. The dataset is collected into a histogram, divided into multiple quantiles with varying data sizes, and assigned equal weights. This process is known as Weighted Quantile Sketch. The second technique is cache-aware access, which allows XGBoost to rapidly store the Hessians and gradients in the cache to calculate similarity scores and other values. The last technique is the Block for Out of Core Computation Technique, where XGBoost stores some data on the hard drive by compressing the data when the dataset is too large to fit into the cache.

4. EXPERIMENTS

System modelling essentially involves finding the relationship between system inputs and outputs. In machine learning-based modelling, such as XGBoost used in this research, XGBoost must know the historical information regarding the input and output of the system in order to model it accurately.

In mathematical terms, this modelling process involves finding f in equation [9]. No explicit mathematical model is generated in the case of XGBoost system modelling in this research. It means that any specific equation does not represent the form of f in equation [9]. The resulting XGBoost model, after achieving the minimal Mean Average Error (MAE), can be directly utilized for other purposes, such as controller design. Therefore, the system identification in this research is referred to as model-free system identification.

$$y(k) = f(x(k), x(k-1), \ldots\ldots, x(k-n_a), y(k-1), y(k-2), \ldots\ldots, y(k-n_b)) \quad [9]$$

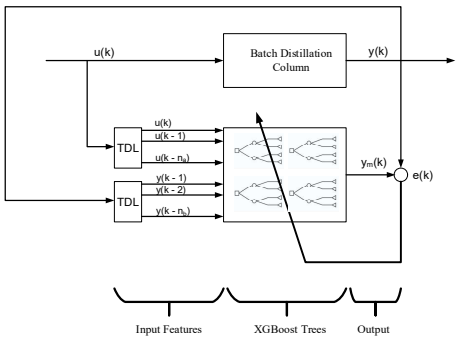
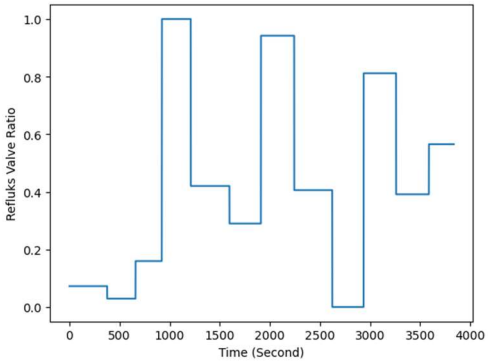


Figure 3. The Modelling Scheme

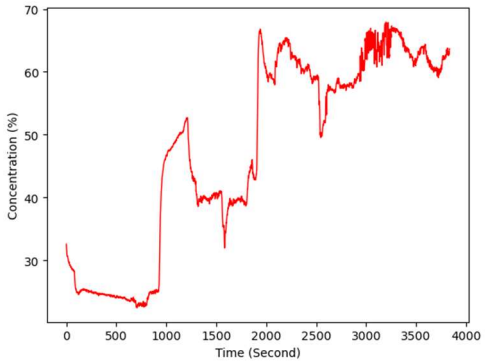
The modelling structure employed in this research, as depicted in Figure 3, consists of three components: the input features, XGBoost trees, and the output. *u(k)* represents the reflux valve ratio (θ), which indicates the duration of the reflux valve opening during a specific period, as

defined in equation [10]. The value of *u(k)* was fed into the BDC plant, resulting in ethanol with a specific concentration, denoted as *y(k)*, utilized as the target in the learning process. For modelling purposes, the values of *u(k)* and *y(k)* were input into respective Tapped Delay Lines (TDL) to be delayed from delay 0 for *u(k)* and from delay 1 for *y(k)* until the specific highest delay values of *u(k)* and *y(k)* (called by n_a and n_b, respectively), and serve as input features in the learning process. The output from XGBoost was denoted as *ym(k),* and together with *y(k),* it was used to calculate the loss function. The tree structure of XGBoost was tuned to minimize this loss function.

$$u(k) = \theta = \frac{t_{open}}{T_{refluks}} \times 100\% \qquad [10]$$

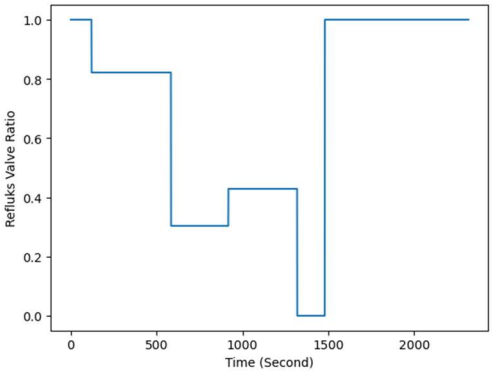


(a)

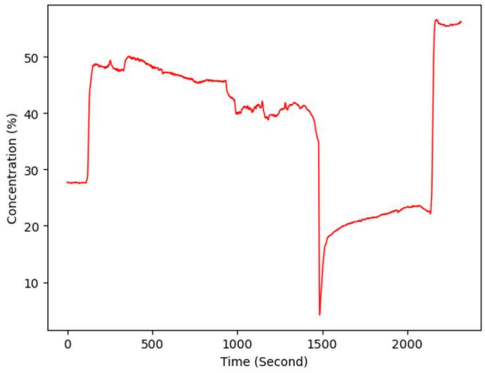


(b)

Figure 4. Input (a) and Target (b) of Learning Dataset



(a)



(b)

Figure 5. Input (a) and Target (b) of Testing Dataset

The value of *u(k)* used in this research ranges from 0 to 1, while *y(k)* ranges from 0 to 100. The datasets of *u(k)* and *y(k)*, prior to passing through the tapped

delay lines (TDL) for the training and testing processes, are illustrated in Figure 4 and Figure 5, respectively. These datasets were directly taken from the BDC plant in the Honeywell Lab of STEI-ITB, utilizing an open-loop system configuration as depicted in Figure 6. The reflux valve ratio *u(k)* value is generated with random amplitude and period using the Arduino IDE software. In contrast, the ethanol concentration value *y(k)* produced by the BDC plant is read using an MQ3 sensor and sent back to the Arduino IDE software for recording. An input-output module for data acquisition was developed from scratch using an Arduino Mega and several components integrated into a Printed Circuit Board (PCB), as illustrated in Figure 7.

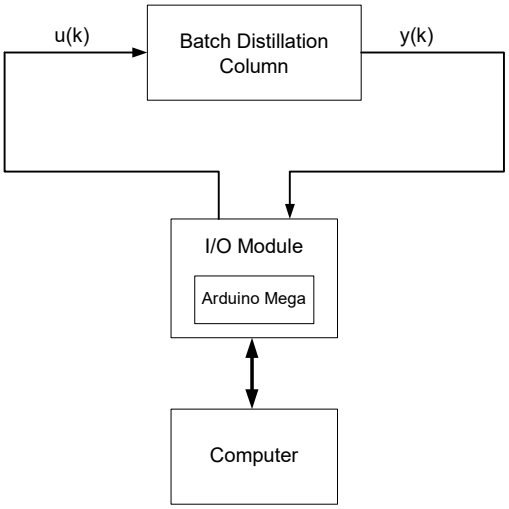


Figure 6. Data Acquisition Scheme with Open-Loop System

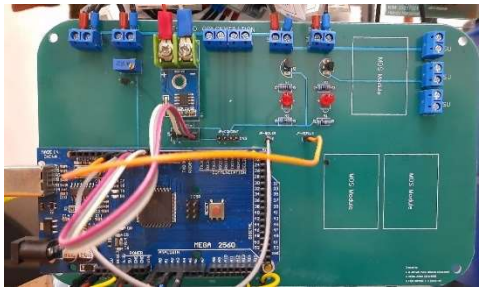


Figure 7. Data Acquisition Module

The hyperparameters used for training the XGBoost model are presented in Table 1 (xgboost developers, 2023). The values of n_a and n_b vary according to Table 2, and the training process is conducted using these maximum delay values. In this paper, the XGBoost model will be trained with varying input features depending on the values of na and nb. In this case, input features are calculated as $n_a + n_b + 1$.

The resulting trained model is then tested to learn from two datasets: the training and testing datasets, which are entirely distinct from the training dataset. Both datasets' Mean Absolute Error (MAE) is recorded and compared for each value of na and nb. Furthermore, the optimal values of n_a and n_b will be determined to achieve the minimum MAE. Henceforth, the Mean Absolute Error (MAE) obtained from the XGBoost model learning on the training and testing datasets will be referred to as the training MAE and testing MAE, respectively.

Table 1. Hyperparater Used for Training Process

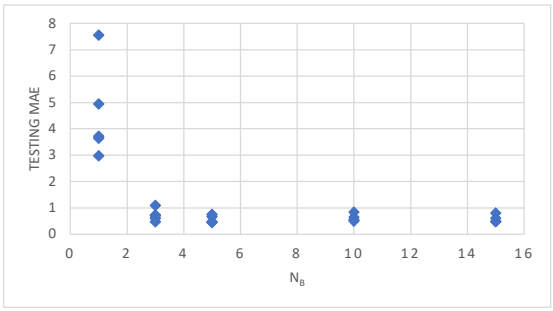
Hyperparamater	Value
max_depth	30
learning_rate	0.003
n_estimator	7000
objective	reg:squarederror
booster	gbtree
base_score	0.8
gamma	100
reg_lambda	100
scale_pos_weight	1
subsample	0.9
colsample_bytree	0.7

Table 2. Variation of n_a and n_b values

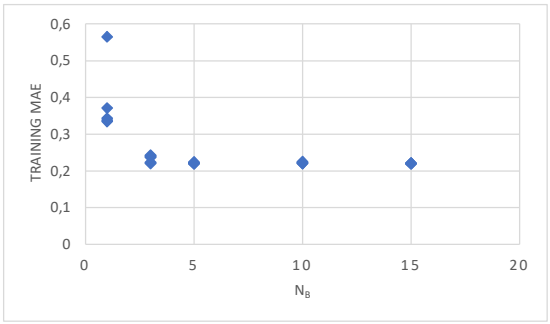
n_a	1					3					5					10					15				
n_b	1	3	5	10	15	1	3	5	10	15	1	3	5	10	15	1	3	5	10	15	1	3	5	10	15

5. RESULTS AND DISCUSSION

Firstly, the MAE values resulting from the learning process of the testing and training datasets against the values of n_a and n_b are plotted as shown in Figure 8 and Figure 9. From these four plots, it is evident that the MAE trend with respect to n_a does not provide a clear indication of the influence of n_a on the obtained MAE. Plotting against n_b is deemed to provide a clearer representation to start exploring the optimal values of n_a and n_b. In the graphs in Figure 8, the MAE values gradually decrease from large values to smaller values and eventually plateau at nearly the same value after $n_b = 5$, both for testing MAE and training MAE. Through these plots, it can be determined that the optimal value for n_b is 5 for the two datasets.

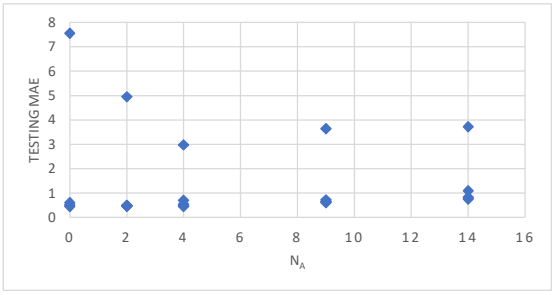


(a)

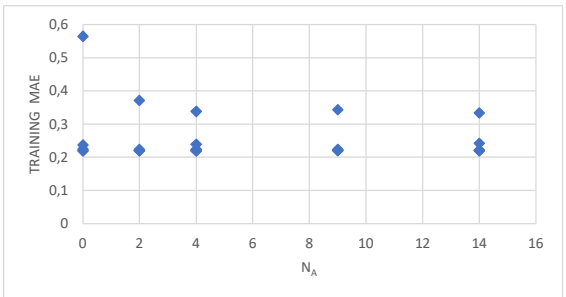


(b)

Figure 8. The Graph of Testing MAE (a) and Training MAE (b) Against n_b



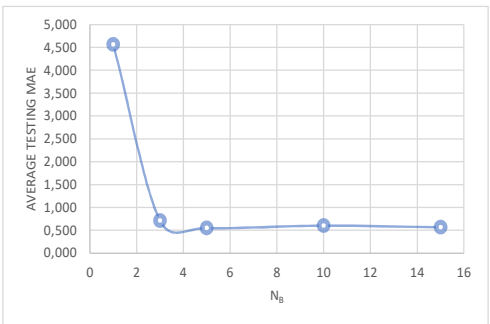
(a)



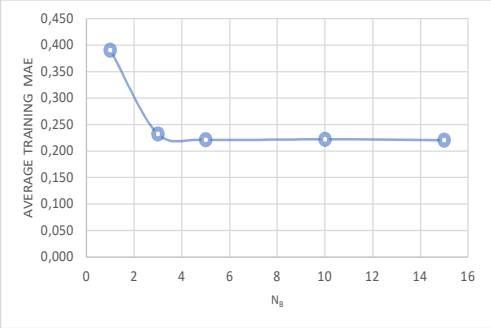
(b)

Figure 9. The Graph of Testing MAE (a) and Training MAE (b) Against n_a

To further investigate, all testing and training MAE values for n_b 1, 3, 5, 10, and 15, regardless of the n_a values, are averaged and plotted in the graph shown in Figure 10. Figure 10 shows that the optimal n_b value is 5, with an average MAE value of 0.551 for the testing dataset and an average MAE value of 0.221 for the training dataset. As the n_b value increases, the obtained MAE does not show significant differences compared to the MAE at n_b = 5, both for the average testing MAE and training MAE. Next, to determine the optimal value, the testing MAE and training MAE values for all n_a, but only when n_b = 5, are plotted in the graphs shown in Figure 11. The graphs show that when n_a is 0, 2, and 4, the testing MAE values are similar, approximately at 0.45. However, for the training dataset, at n_a = 4, the MAE value is small, 0.219, and there is a massive gap between the MAE values and other values.

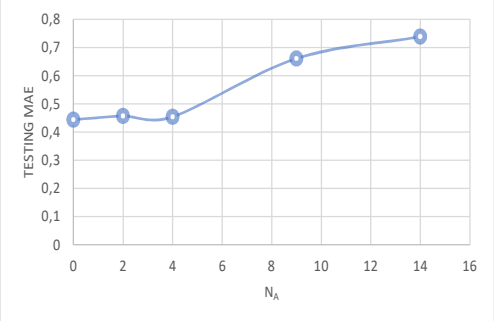


(a)

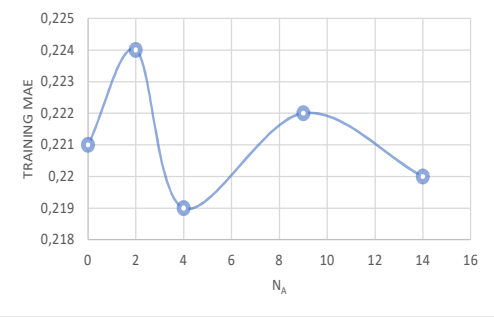


(b)

Figure 10. The Graph of Average Testing MAE (a) and Average Training MAE (b) Against n_a



(a)



(b)

Figure 11. The Testing MAE (a) and Training MAE (b) Values for All n_a, Specifically When $n_b = 5$

The graph of training time for the learning process with different combinations of n_a and n_b can be seen in Figure 12. From the graph, it can be observed that as the number of combinations of n_a and n_b increases, the tendency of training time also increases. It also can be seen from the graph that increasing n_a and n_b may lead to smaller MAE values, but it has to consider the trade-off with training time. If the difference in MAE values is insignificant when n_a and n_b are increased, choosing an optimal value is desired to achieve a suitable combination of n_a and n_b that minimizes MAE while maintaining a relatively fast training time.

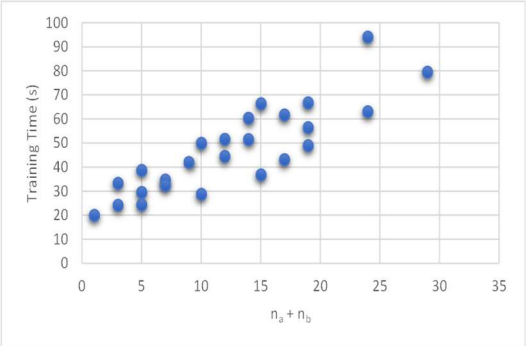


Figure 12. The Training Time

Based on Figure 13, the output response can be observed with varying values of n_a and n_b, where the response that closely approximates the actual data graph is when $n_a = 4$ and $n_b = 5$. When $n_a = 0$ and $n_b = 1$, there is a significant gap between the XGBoost model response and the actual data. However, when $n_a = 2$ and $n_b = 3$, $n_a = 9$ and $n_b = 10$, and $n_a = 14$ and $n_b = 15$, the difference between the XGBoost model output response and the actual data is not very significant, although their MAE remains more

extensive than the response with $n_a = 4$ and $n_b = 5$.

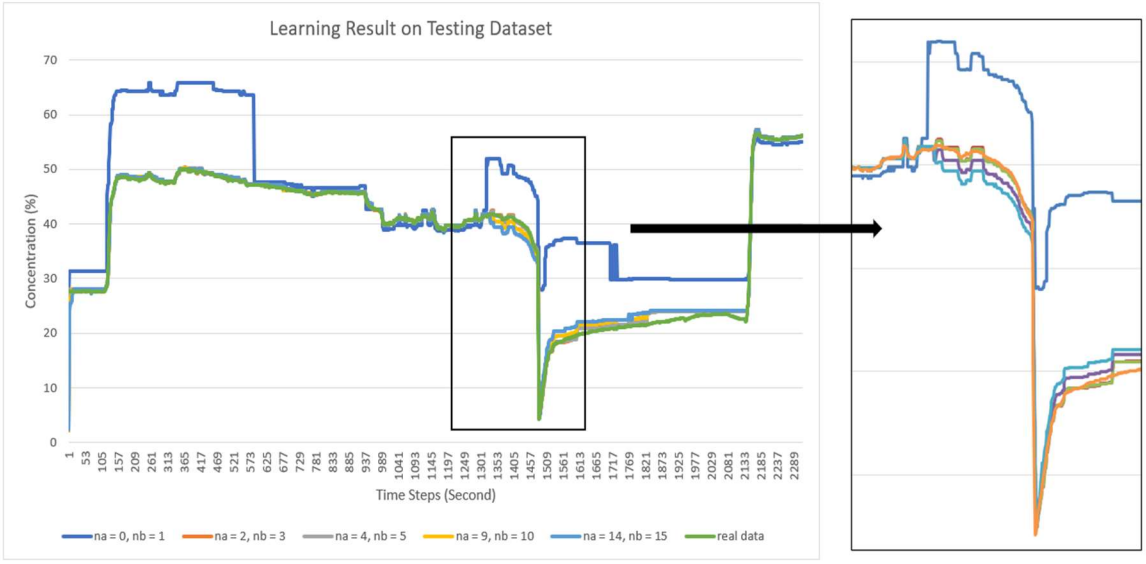


Figure 13. Learning Result of Various Value of n_a and n_b

In addition to plotting the varying values of n_a and n_b, the graph with $n_b = 5$ combined with several values of n_a is also plotted in Figure 14. Based on Figure 14, the difference in model output response for all values of n_a and n_b has a slight deviation compared to the actual data. However, the most petite MAE response is observed at $n_a = 4$ and $n_b = 5$.

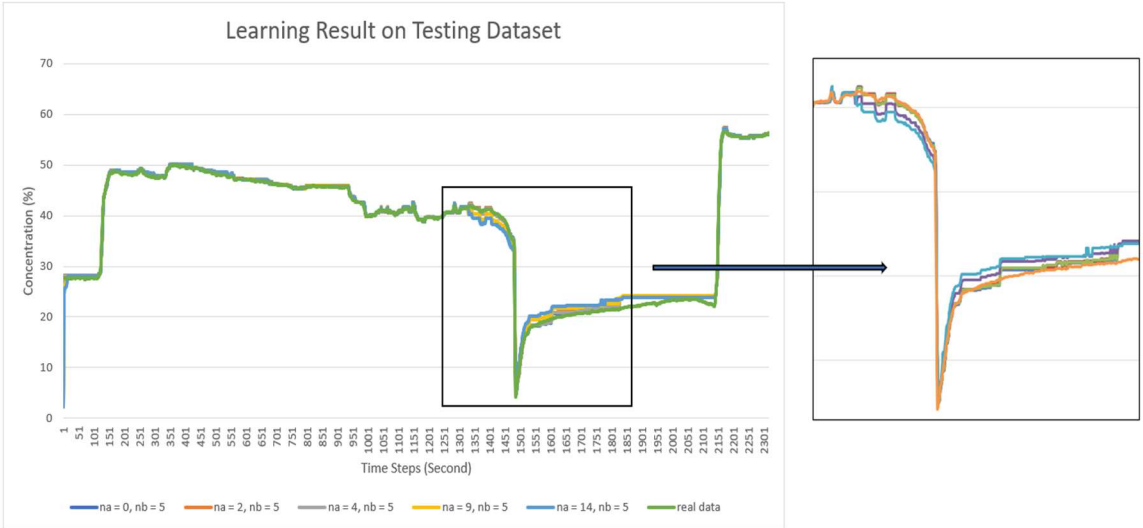


Figure 14. Learning Result of Various Value of n_a with same value of $n_b = 5$

6. CONCLUSIONS

Using XGBoost for data-driven system identification offers a promising alternative to conventional model-based approaches. By leveraging the capabilities of XGBoost, researchers and practitioners can accurately capture and model complex system dynamics solely from data without relying on prior knowledge of system equations. Based on the experimental results in this study, the optimal performance, with a Mean Absolute Error (MAE) value of 0.219 on the training dataset and 0.45 on the testing dataset, was achieved when utilizing a structure with 10 input features. These features include the current reflux valve ratio, delayed reflux valve ratio from 1 to 4-time steps ($n_a = 4$), and delayed ethanol concentration from 1 to 5-time steps ($n_b = 5$). This paper concludes by emphasizing the potential of XGBoost in enabling accurate and efficient system identification in real-world applications and motivating further research and exploration in this field.

7. REFERENCES

Ayu, W. S., Rusmin, P. H., & Hidayat, E. M. I. (2019). Robust PID Control Design in CPS-based Batch Distillation Column. *International Conference on Electrical Engineering, Computer Science and Informatics (EECSI)*, 95–100. https://doi.org/10.23919/EECSI48112.2019.8977083

Bravo, J. L. (2019). Distillation Technology : What's Next ? *CEP Magazine*, *August*. https://www.aiche.org/resources/publications/cep/2019/august/distillation-technology-whats-next

Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, *13-17-Augu*, 785–794. https://doi.org/10.1145/2939672.2939785

Diwekar, U. (2011). *Batch Distillation: Simulation, Optimal Design, and Control, Second Edition* (2nd ed.). CRC Pess.

Maulidda, R., Rusmin, P. H., Rohman, A. S., Idris Hidayat, E. M., & Mahayana, D. (2018). Modeling and

Simulation of Mini Batch Distillation Column. *Proceedings of 2017 5th International Conference on Instrumentation, Communications, Information Technology, and Biomedical Engineering, ICICI-BME 2017*, *November*, 62–67. https://doi.org/10.1109/ICICI-BME.2017.8537724

Mendis, P., Wickramasinghe, C., Narayana, M., & Bayer, C. (2019). Adaptive Model Predictive Control with Successive Linearization for Distillate Composition Control in Batch Distillation. *MERCon 2019 - Proceedings, 5th International Multidisciplinary Moratuwa Engineering Research Conference*, 366–369. https://doi.org/10.1109/MERCon.2019.8818777

Nahrendra, I. M. A., Rusmin, P. H., & Hidayat, E. M. I. (2019). Adaptive Control of Cyber-Physical Distillation Column using Data Driven Control Approach. *Proceedings of the International Conference on Electrical Engineering and Informatics*, (July), 93–98. https://doi.org/10.1109/ICEEI47359.2019.8988821

Ogunnaike, B. A., & Ray, W. H. (1994). *Process Dynamics, Modeling, and Control* (Illustrate). Oxford University Press.

Orozco, G., Cortés, B., Heras, M., Téllez, A., & Anzurez, J. (2017). Analysis and Comparison of Distillation Column Models Considering Constant and Variable Relative Volatility. *2016 IEEE International Autumn Meeting on Power, Electronics and Computing, ROPEC 2016*, *Ropec*. https://doi.org/10.1109/ROPEC.2016.7830590

Pei, X., Mei, F., & Gu, J. (2022). The Real-Time State Identification of the Electricity-Heat System based on Borderline-SMOTE and XGBoost. *IET Cyber-Physical Systems: Theory and Applications*, 1–11. https://doi.org/10.1049/cps2.12032

Putri, A. N. (2021). Pemodelan Dan Perancangan Kendali Optimal Sistem Kolom Distilasi Tipe Batch Dengan Pendekatan Penjejakan Kuadratis Linier. *Tesis Program Magister, Institut Teknologi Bandung*.

Rohman, A. S., Rusmin, P. H., Maulidda, R., Hidayat, E. M. I., Machbub, C., & Mahayana, D. (2018). Modelling of The Mini Batch Distillation Column. *International Journal on*

Electrical Engineering and Informatics, *10*(2), 350–368. https://doi.org/10.15676/ijeei.2018.10.2.11

xgboost developers. (2023). *xgboost*.

Zou, Z., Wang, Z., Meng, L., Yu, M., Zhao, D., & Guo, N. (2017). Modelling and Advanced Control of a Binary Batch Distillation Pilot Plant. *Proceedings - 2017 Chinese Automation Congress, CAC 2017*, *2017-Janua*, 2836–2841. https://doi.org/10.1109/CAC.2017.8243259

Zou, Z., Yu, D., Hu, Z., Guo, N., Yu, L., & Feng, W. (2006). State Space Modeling and Predictive Control of a Binary Batch Distillation Column. *Proceedings of the World Congress on Intelligent Control and Automation (WCICA)*, *2*, 6252–6256. https://doi.org/10.1109/WCICA.2006.1714285