IISE Transactions
Operations Engineering & Analytics
Supply Chain & Logistics
Taylor & Francis
INSTITUTE OF INDUSTRIAL & SYSTEMS ENGINEERS

IISE Transactions

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/uiie21

MMP Net: A feedforward neural network model with sequential inputs for representing continuous multistage manufacturing processes without intermediate outputs

Hojin Cho, Kyeongbin Kim, Kihyuk Yoon, Jaewook Chun, Jaeyong Kim, Kyeongmin Lee, Junghye Lee & Chiehyeon Lim

IISE TRANSACTIONS
https://doi.org/10.1080/24725854.2023.2242434

MMP Net: A feedforward neural network model with sequential inputs for representing continuous multistage manufacturing processes without intermediate outputs

Hojin Cho[a*], Kyeongbin Kim[b*], Kihyuk Yoon[a*], Jaewook Chun[c], Jaeyong Kim[c], Kyeongmin Lee[c], Junghye Lee[d], and Chiehyeon Lim[a,e]

[a]Department of Industrial Engineering, Ulsan National Institute of Science and Technology, Ulsan, Republic of Korea; [b]LG Electronics, Seoul, Republic of Korea; [c]SK Enmove, Ulsan, Republic of Korea; [d]Technology Management, Economics, and Policy Program, Seoul National University, Seoul, Republic of Korea; [e]Graduate School of Artificial Intelligence, Ulsan National Institute of Science and Technology, Ulsan, Republic of Korea

ABSTRACT
Machine learning models that are used for the prediction and control of production can improve quality and yield. However, developing models that are highly accurate and reflective of real-world processes is challenging. We propose a feedforward neural network model specifically designed for continuous Multistage Manufacturing Processes (MMPs) without intermediate outputs. This model, which is termed "MMP Net," can accurately represent the control mechanism of continuous MMPs. Whereas existing studies on learning MMPs assume an intermediate output data, the MMP Net does not require such an unrealistic assumption. We use the MMP Net to develop prediction models for the lubricant base oil production process of a world-leading lubricant manufacturer. Evaluation results show that the MMP Net is superior to other deep neural network and machine learning models. Consequently, the MMP Net was actually implemented in a real factory in 2022 and is expected to save 900,000 dollars per year for each production line. We believe that our work can serve as a basis to develop customized machine learning solutions for improving continuous MMPs.

ARTICLE HISTORY
Received 30 June 2022
Accepted 27 June 2023

KEYWORDS
Multistage manufacturing process; neural network; petrochemical manufacturing system; quality prediction; real-world application

1. Introduction

In any manufacturing process, the proper control of variables relevant to each stage is the key to managing the quality and production yield during the transformation of raw material inputs into final product outputs (Kano and Nakagawa, 2008; Park *et al.*, 2022). Successful control requires the accurate prediction of output variables (Arif *et al.*, 2013). The increasing availability of data on the input, control, and output variables of manufacturing processes has led to the application of machine learning techniques to the prediction of output variables and optimization of process control (Niaki and Davoodi, 2009; Amini and Chang, 2018; Zhu *et al.*, 2019; Wu *et al.*, 2020).

The complexity of modern product systems directs manufacturers to adopt Multistage Manufacturing Processes (MMPs) rather than single-stage ones (Shi and Zhou, 2009; Kim *et al.*, 2018). Any MMPs, whether continuous or discrete, involve the simultaneous control of multiple variables in each stage; control variables in a stage affect the intermediate output from that stage. Notably, control variables in a stage naturally affect the intermediate outputs in the next stages (Zou and Tsung, 2008) but not vice versa (i.e., control variables in a stage do not affect the intermediate outputs from previous stages). The quality and production yield of final products depend on the outputs from the intermediate stages (Li *et al.*, 2007; Zhang *et al.*, 2021). Therefore, the application of machine learning to MMPs has mainly involved multi-model frameworks, in which different learning models are used for different stages (Jin and Shi, 2012; Bera and Mukherjee, 2016; Yin *et al.*, 2018; Lee, Kim, and Kim, 2021), whereas single-model frameworks have been used for single-stage processes (Chen *et al.*, 2008; Wang *et al.*, 2009; Sun *et al.*, 2018). Multi-model frameworks predict the outputs of each intermediate stage in a forward-stepwise manner (i.e., the predicted output of a previous stage is sent as the input to the next stage, as shown in the upper part of Figure 1).

This approach is inapplicable to cases in which only the control and environment variables of each stage ($\mathbf{x}_n$ in Figure 1) are known, whereas the intermediate outputs (y_n

CONTACT Kihyuk Yoon dreamistar@unist.ac.kr; Junghye Lee junghye@snu.ac.kr; Chiehyeon Lim chlim@unist.ac.kr.

*First authors (equal contribution).

Supplemental data for this article is available online at https://doi.org/10.1080/24725854.2023.2242434.

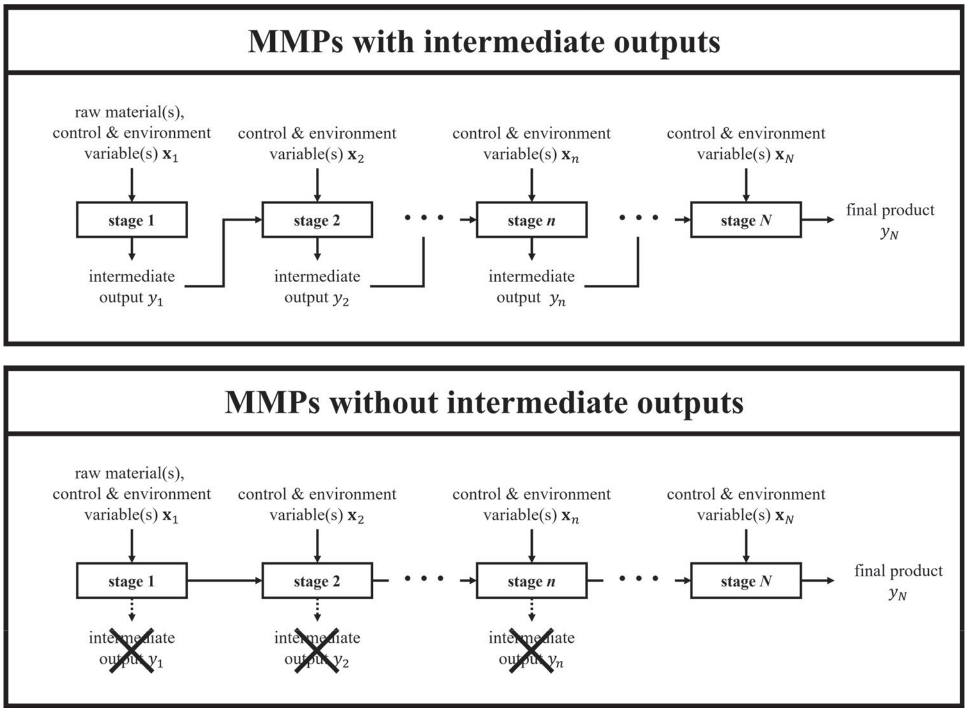
MMPs with intermediate outputs
raw material(s), control & environment variable(s) $\mathbf{x}_1$
control & environment variable(s) $\mathbf{x}_2$
control & environment variable(s) $\mathbf{x}_n$
control & environment variable(s) $\mathbf{x}_N$
stage 1
stage 2
stage n
stage N
final product y_N
intermediate output y_1
intermediate output y_2
intermediate output y_n
MMPs without intermediate outputs
raw material(s), control & environment variable(s) $\mathbf{x}_1$
control & environment variable(s) $\mathbf{x}_2$
control & environment variable(s) $\mathbf{x}_n$
control & environment variable(s) $\mathbf{x}_N$
stage 1
stage 2
stage n
stage N
final product y_N
intermediate output y_1
intermediate output y_2
intermediate output y_n

Figure 1. MMPs with intermediate outputs (upper) and without intermediate outputs (bottom).

in Figure 1) have no available data, especially in continuous MMPs. Continuous manufacturing is a manufacturing system where raw materials are continuously inputted and processed without interruption, such as liquid production processes in petrochemical and drug manufacturing. The fundamental objective of establishing and operating a continuous manufacturing system is to increase production yield by minimizing the time between different stages. Therefore, continuous MMPs often do not involve inspection after intermediate stages, whereas inspection of intermediate outputs from discrete MMPs in automobile assembly and semiconductor manufacturing is easy. Consequently, in the typical context of continuous MMPs where intermediate output data cannot be collected, the sequential process control mechanism has a composite structure, as shown in the Figure 1 (bottom) and the following formula. $\mathbf{x}_n$ in the figure and formula indicates the control and environment variables of each stage:

$$y_N = stage_N(\ldots(stage_3(stage_2(stage_1(\mathbf{x}_1),\ \mathbf{x}_2),\ \mathbf{x}_3),\ldots),\ \mathbf{x}_N) \tag{1}$$

In the above scenarios, the multi-model framework is infeasible. Thus, a single model specialized for continuous MMPs without the need for data on intermediate outputs can be useful for predicting and optimizing manufacturing processes. However, to the best of our knowledge, such a model is not available. In fact, we could not find a single study on machine learning applications to continuous MMPs that do not measure intermediate outputs. We believe that this indicates the difficulty of predicting the final output variables for such scenarios because existing single-model frameworks are not designed to represent the dynamics of multiple stages.

In this work, we propose MMP Net, a FeedForward Neural Network (FFNN) model that accurately represents the “control mechanism of continuous MMPs” in manufacturing industries (Figure 2). The MMP Net sequentially injects special inputs into the hidden layers of an FFNN. Given that the MMP Net is a single model, no intermediate output information is necessary. The sequence of input features in continuous MMPs can be considered intuitively, and the dynamics in multiple stages can be represented systematically. We validated the MMP Net through quality predictions for a real-world lubricant base oil manufacturing, which is a typical continuous, nonstop process that is a very high value-added business in the petrochemical industry (Yu, 2012).

Previous studies have combined existing machine learning techniques to address specific manufacturing processes, but the results are often inapplicable to other cases. In this respect, the main contributions of this study to industrial and manufacturing systems engineering are summarized as follows. First, we originally developed the MMP Net, which is an end-to-end single model for solving machine learning problems of continuous MMPs without intermediate outputs by representing the control mechanism. Whereas existing studies on learning MMPs assume the existence of rich intermediate output data or simply adopt an existing model, the MMP Net does not require such an unrealistic assumption and its structure is specifically designed for continuous MMPs with no intermediate outputs. As a result, the MMP Net takes advantage of the multi-model framework (i.e., reflecting the sequential characteristics of MMP) and overcomes its weaknesses (i.e., not applicable to cases without intermediate outputs) through the injected input layers. Second, we show the validity of the proposed MMP Net through its real-world application on the optimization of a

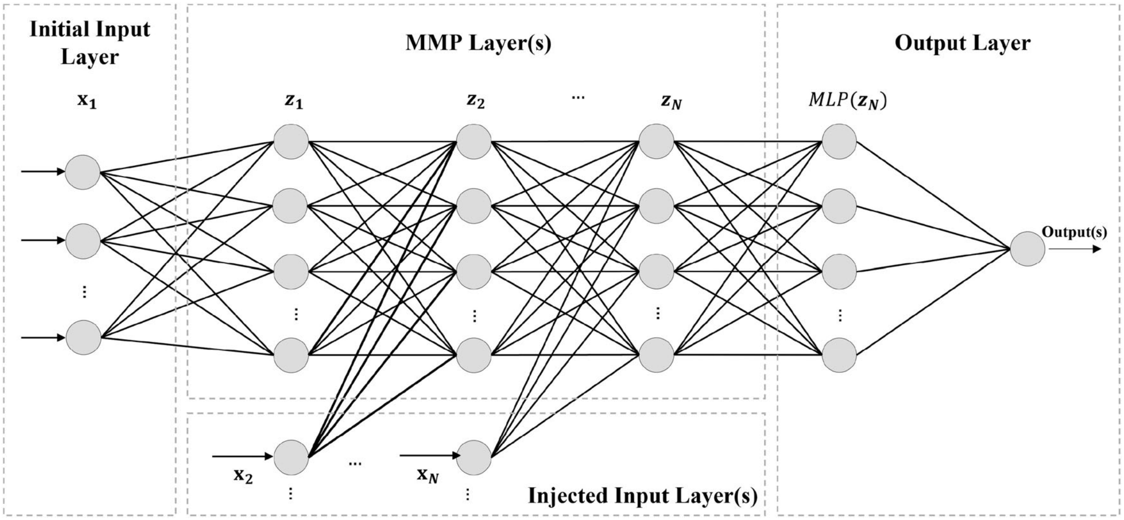

Figure 2. Architecture of MMP Net.

petrochemical production. In terms of accuracy, the trained MMP Net solution is superior to other solutions that are constructed based on the representative existing machine learning and deep learning models. The solution provides accurate predictions that can sufficiently support or replace humans; engineers can control the factory in optimal ways by referring to prediction outcomes. Given this superiority, the MMP Net was actually implemented in a factory in 2022 and is expected to generate an annual savings of 900,000 USD per production line.

The remainder of this article is organized as follows. Section 2 reviews the related works, and Section 3 explains the architecture and algorithm of the MMP Net. Section 4 describes the real-world application of the MMP Net and validates its utility. Section 5 discusses the methodological and managerial contributions, as well as limitations of the present study, and our plans for future work.

2. Related studies

To address the problems of data-driven manufacturing process control, case-specific approaches have been proposed to improve the prediction performance by considering the particular characteristics of various manufacturing processes. For example, in the semiconductor field, wafer defects can be categorized into single- and multi-defect patterns. Detecting both patterns is important for product quality control. However, detecting them simultaneously presents a challenge because traditional shallow-learning models, which are widely used to solve the wafer-defect problem, cannot classify defects accurately when the datasets contain multi-defect patterns. To cope with this problem, Tello *et al.* (2018) proposed a deep learning model that identifies and classifies single- and mixed-defect patterns using a gain-based splitter with a convolutional neural network. In addition, other studies have used dimensionality reduction and feature selection techniques for variables in manufacturing to solve its classification and regression problems. However, although several variables play an important role in the output, previous models cannot reflect the physical and chemical mechanisms of manufacturing. Therefore, Yuan *et al.* (2018) developed a deep learning-based model that trains the network layer by layer with a variable-wise weighted stacked autoencoder to reflect the mechanism of a debutanizer column process through feature representation.

The abovementioned studies have shown that reflecting the characteristics of the target process is effective in learning its data. Thus, multi-model approaches have been presented to address the structural characteristics of MMPs, which take place in sequential stages (Sales-Setién *et al.*, 2018), wherein the variables affect the operations of subsequent stages (Zou and Tsung, 2008); as manufacturers often have difficulties in producing outputs that meet the desired quality standards, scholars have attempted to overcome this difficulty by applying data-driven approaches in MMPs to predict product quality from process variables (Tian *et al.*, 2017; Sharp *et al.*, 2018; Wang and Tsung, 2021). The nature of MMPs results in the use of multi-model frameworks, in which machine learning models are developed for each individual stage and connected by intermediate outputs (Lee, Kim, and Kim, 2021); the input features of each stage are considered separately. For example, a simplified-solution approach (Bera and Mukherjee, 2016) that assigns a linear regression model to each stage is suggested for the multi-stage multiple response optimization problem. Similarly, Lee, Kim, and Kim, (2021) proposed a multi-model framework with decision trees to address the data-driven control problem of MMPs. Meanwhile, Jiang *et al.* (2014) proposed a machining error propagation network to reflect the correlation among complicated interactions in different MMP stages. Liu *et al.* (2018) developed a framework for multi-stage basic oxygen furnace steel-making process using a Hybrid Kernel Least Squares Support Vector Machine (HKLSSVM) as the base model for each stage. The proposed multi-model approach showed high performance compared with single-stage model approaches that only consider the error of the final output, such as an SVM, relevance vector machine, least squares SVM (LSSVM; Valyon and Horváth, 2009), and single-stage HKLSSVM.

However, the fundamental challenge is that the above multi-model approaches require intermediate output data. Although such data are available in several cases, the prediction errors accumulate across different models in a forward-stepwise manner with existing multi-model approaches, leading to the unreliable estimation of search spaces for process control optimization. To overcome these problems, Arif *et al.* (2013) proposed a framework with multiple Principal Component Analysis (PCA) and iterative dichotomizer algorithms, which attempt to solve and reflect the nature of MMPs with a single- rather than a multi-model approach. In this framework, PCs are obtained for the variables of the initial stage through PCA, and the variables in subsequent stages are added to derive the next PCs. This process is repeated until the last stage is reached. The obtained PCs of the last stage are used as the input of the iterative dichotomizer algorithms that predict the product quality. With this framework, the characteristics of MMPs can be captured by sequentially preprocessing the variables of each stage using several PCAs. However, the structural characteristics of MMPs without intermediate outputs are still not reflected in the actual learning process. Recently, Zhang *et al.* (2021) proposed a deep learning approach for quality prediction in MMPs. This model focuses on the dependency relationships among multiple machines through a path-enhanced bidirectional graph attention network, but it also requires intermediate outputs. In conclusion, to the best of our knowledge, no study on generally applicable methods has reflected the continuous MMP without intermediate outputs in a single model.

Having a reference model for a specific real-world industrial or social problem is useful and necessary. Examples of such models include the deep learning-based newsvendor problem solving framework (Oroojlooyjadid *et al.*, 2020), the deep learning framework for remaining useful life estimation of complex systems (Kim and Liu, 2020), the Transformer-based multivariate stock movement prediction model (Yoo *et al.*, 2021), the controlled sequence generation model for the diet planning problem, which had not been addressed as a sequence generation problem but a combinatorial optimization problem (Lee, Kim, Lim, Kim, Kim and Jung, 2021), and other recent studies (Hao *et al.*, 2022; Seo *et al.*, 2022; Li *et al.*, 2023). Given that accurate machine learning models are crucial for industrial intelligence (Yang *et al.*, 2019; Chen *et al.*, 2023), developing a reference model for learning continuous MMPs in various industries will significantly help bridge the gap between the unique mechanism of such processes and modern machine learning models and techniques. This reference model would offer a high-performance basis for researchers and practitioners aiming to improve their continuous MMPs. The proposed MMP Net described in the next section would serve as such a reference model that can represent continuous MMPs without intermediate outputs in industries.

3. MMP Net

3.1 Architecture of MMP Net

The MMP Net has two types of input layers, namely, the initial and the injected (Figure 2). These input layers perform the same role of receiving information, but have different forward positions. The initial input layer forwards the initial input to the first hidden layer, and the injected input layer concatenates these data with the hidden layer and forwards the intermediate input to the next hidden layer. The injected input layer injects variables in the same manner as in the actual process. Consequently, the model can represent the sequential control mechanism of continuous MMPs.

Before describing the MMP Net architecture in detail, we define our notations as follows. Let $\mathbf{x}_1 \in \mathbb{R}^{d_1}$, $\mathbf{x}_n \in \mathbb{R}^{d_n}$, $\boldsymbol{z}_1 \in \mathbb{R}^{r_1}$, and $\boldsymbol{z}_n \in \mathbb{R}^{r_n}$ be the initial input, n^{th} injected input, initial hidden layer, and n^{th} hidden layer, respectively, for $n = 2, \ldots, N$ (where the N is the last stage). The weights of the initial input layer, the n^{th} injected input layer, and the n^{th} hidden layer in the MMP Net are denoted by $\boldsymbol{W}_1 \in \mathbb{R}^{d_1 \times r_1}$, $\boldsymbol{W}_n^I \in \mathbb{R}^{d_n \times r_n}$, and $\boldsymbol{W}_n^V \in \mathbb{R}^{r_{n-1} \times r_n}$, respectively. The nonlinearity of the features is indicated by the use of a nonlinear activation function (denoted by $\sigma(\cdot)$), such as rectified linear unit (ReLU).

The configuration of features for a certain layer (depending on the input of the previous layer) is represented as

$$\boldsymbol{z}_1 = \sigma(\boldsymbol{W}_1^{\mathrm{T}} \mathbf{x}_1), \tag{2}$$

$$\boldsymbol{z}_n = \sigma(\boldsymbol{W}_n^{V^T} \boldsymbol{z}_{n-1} + \boldsymbol{W}_n^{I^T} \mathbf{x}_n), \ \forall n = 2, \ldots, N, \tag{3}$$

$$y_N = MLP(\boldsymbol{z}_N), \tag{4}$$

where $MLP(\cdot)$ is one or more fully connected layers to obtain the output of the MMP Net. In MMP Net, the input can be one or more of the following (depending on the layer position): (i) initial input layer, (ii) injected input layer(s), and (iii) hidden layer(s). The MMP Net architecture can adapt to changes based on data characteristics, because the number and location of injected input layers, number of hidden layers, and the number of latent features per hidden layer are user-defined parameters.

The proposed MMP Net emulates the sequential process control mechanism of continuous MMPs by injecting the control and environment variable(s) of intermediate stages into the hidden layers. Therefore, the proposed MMP Net can operate with the composite structure of sequential MMP control mechanisms, which was described in the Introduction section with Equation (1). In view of the forward pass in training the neural network, the first-stage data flow through the first hidden layer(s) and produce the latent vector $\boldsymbol{z}_1$, which represents the characteristics of the first stage of the MMP under consideration. Subsequently, $\boldsymbol{z}_1$ is merged with the second-stage data to generate the latent vector $\boldsymbol{z}_2$, which represents the characteristics of the first and second stages of the MMP under consideration. This mechanism exactly matches the sequential process control mechanism of continuous MMPs, in which the raw material inputs become the first intermediate outputs after passing through the process control in the first stage, and the first intermediate outputs become the second intermediate outputs after passing through the process control in the second stage.

3.2 Training algorithm of MMP Net

Algorithm 1: MMP Net training algorithm for regression example

```
Input: $\boldsymbol{X}_1$ : initial input; $\boldsymbol{X}_n$ : $n^{\text{th}}$ intermediate input
       $\eta$ : learning rate
       $\sigma(\cdot)$ : activation function
       $max\_iter$: maximal number of iterations
       $\boldsymbol{\Theta} = [\boldsymbol{\theta}_1, ..., \boldsymbol{\theta}_n, ..., \boldsymbol{\theta}_N, \boldsymbol{\theta}_{MLP}]$ where,
       $\boldsymbol{\theta}_n = \{W_n^I, W_n^V\}$, $\forall n = 2, ..., N$ is a set of the parameters of MMP Net
Output: $\hat{\boldsymbol{y}}$ : predicted output
Procedure MMP Net training
  Initialize: $\boldsymbol{\Theta}^{(0)}$
  $iter \leftarrow 0$
  repeat
    for $n = 1, ..., N$ do
      if $n = 1$ then
        $\boldsymbol{Z}_1 = \sigma(\boldsymbol{W}_1^T \boldsymbol{X}_1)$
      else
        $\boldsymbol{Z}_n = \sigma(\boldsymbol{W}_n^{V\,T} \cdot \boldsymbol{Z}_{n-1} + \boldsymbol{W}_n^{I\,T} \cdot \boldsymbol{X}_n)$
      end if
  $\hat{\boldsymbol{y}} = MLP(\boldsymbol{Z}_n)$
  $\mathcal{L} = \|\hat{\boldsymbol{y}} - \boldsymbol{y}\|_F^2$
  $\boldsymbol{\Theta}^{(iter+1)} = \boldsymbol{\Theta}^{(iter)} - \eta \times \nabla\mathcal{L}_{\boldsymbol{\Theta}^{(iter)}}$
  until $iter < max\_iter$
  return $\boldsymbol{\Theta}$
end function
```

To create a prediction model based on MMP Net, we need to estimate the parameters of the MMP Net. Given data $\mathcal{D} = \{\boldsymbol{X}_1 \in \mathbb{R}^{d_1 \times s}, \boldsymbol{X}_n \in \mathbb{R}^{d_n \times s}, \boldsymbol{y} \in \mathbb{R}^s\}$, where s is the sample size, the training process of the MMP Net with the objective function of minimizing the regression loss between the predicted and the actual values is represented as

$$\left(\boldsymbol{W}_1{}^*,\ \boldsymbol{W}_n^{\mathrm{I}\,*}, \boldsymbol{W}_n^{\mathrm{V}*},\ \boldsymbol{W}_{MLP}{}^*\right) = \underset{\boldsymbol{W}_1,\ \boldsymbol{W}_n^I,\ \boldsymbol{W}_n^V,\ \boldsymbol{W}_{MLP}}{\operatorname{argmin}} \|\hat{\boldsymbol{y}} - \boldsymbol{y}\|_F^2,\ \forall n = 2, ..., N, \tag{5}$$

where $\|\cdot\|$ denotes the Frobenius norm, and $\hat{\mathrm{y}}$ is the estimated value of $\boldsymbol{y}$. The size changes according to the number of intermediate inputs. The training process for the MMP Net is presented in Algorithm 1. Given the initial and intermediate inputs defined by the domain knowledge of the actual process structure and available dataset, the weights (including the bias) are randomly initialized. The initial input layer is propagated to the initial hidden layer. Subsequently, the injected input layer $\boldsymbol{X}_n$ and the previous hidden layer $\boldsymbol{Z}_{n-1}$ are concatenated and propagated to the next hidden layer $\boldsymbol{Z}_n$. Finally, the estimates of outputs are computed through the $MLP(\cdot)$ in (4). The differences (i.e., losses) between the actual and estimated values are calculated, and the parameters are updated to minimize the loss based on a gradient descent algorithm.

We can predict the outputs of new data based on the trained MMP Net (i.e., $\hat{f}(\boldsymbol{x})$, where the optimal parameters $\boldsymbol{W}_1{}^*$, $\boldsymbol{W}_n^{\mathrm{I}\,*}$, $\boldsymbol{W}_n^{\mathrm{V}\,*}$, and $\boldsymbol{W}_{MLP}{}^*$ are mounted). The prediction process is the same as the forward propagation of Algorithm 1.

Given this process, the input layers of the MMP Net are only connected to each other with the features of the hidden layers $\boldsymbol{z}_n$, whereas the intermediate outputs connect the inputs of each model in the multi-model approaches reviewed in Section 2. As such, the MMP Net has an advantage compared with existing multi-model approaches; being a single model end-to-end learning framework, the MMP Net has less risk of losing the information of variables from earlier stages.

3.3 Relevance with the control mechanism of continuous MMPs

3.3.1 Backward pass of MMP Net

In continuous MMPs, the influences of control variables in a stage do not affect previous stages. To capture the sequential process control mechanism of continuous MMPs in the MMP Net, it is essential to ensure that the injected input(s) and corresponding weight(s) do not affect the latent vectors and gradients of the weights in subsequent layers. The backward pass of the MMP Net (we assume activation function σ as ReLU and omit it for simplification) is represented as follows:

$$\begin{aligned}\frac{\partial\mathcal{L}}{\partial\boldsymbol{z}_n} &= \frac{\partial\boldsymbol{z}_{n+1}}{\partial\boldsymbol{z}_n}\cdot\frac{\partial\mathcal{L}}{\partial\boldsymbol{z}_{n+1}}\\ &= \boldsymbol{W}_{n+1}^V\cdot\frac{\partial\mathcal{L}}{\partial\boldsymbol{z}_{n+1}},\end{aligned} \tag{6}$$

$$\begin{aligned}\frac{\partial\mathcal{L}}{\partial\boldsymbol{z}_{n+1}} &= \frac{\partial\boldsymbol{z}_{n+2}}{\partial\boldsymbol{z}_{n+1}}\cdot\frac{\partial\mathcal{L}}{\partial\boldsymbol{z}_{n+2}}\\ &= \boldsymbol{W}_{n+2}^V\cdot\frac{\partial\mathcal{L}}{\partial\boldsymbol{z}_{n+2}},\end{aligned} \tag{7}$$

$$\begin{aligned}\frac{\partial\mathcal{L}}{\partial\boldsymbol{W}_n^V} &= \frac{\partial\boldsymbol{z}_n}{\partial\boldsymbol{W}_n^V}\cdot\frac{\partial\mathcal{L}}{\partial\boldsymbol{z}_n}\\ &= \boldsymbol{z}_{n-1}\cdot\frac{\partial\mathcal{L}}{\partial\boldsymbol{z}_n},\end{aligned} \tag{8}$$

$$\begin{aligned}\frac{\partial\mathcal{L}}{\partial\boldsymbol{W}_n^I} &= \frac{\partial\boldsymbol{z}_n}{\partial\boldsymbol{W}_n^I}\cdot\frac{\partial\mathcal{L}}{\partial\boldsymbol{z}_n}\\ &= \mathbf{x}_n\cdot\frac{\partial\mathcal{L}}{\partial\boldsymbol{z}_n}.\end{aligned} \tag{9}$$

During the backward pass, the gradient of the latent vector $\boldsymbol{z}_n$ is the production of weight $\boldsymbol{W}_{n+1}^V$ and $\frac{\partial\mathcal{L}}{\partial\boldsymbol{z}_{n+1}}$, whereas that of the latent vector $\boldsymbol{z}_{n+1}$ is the production of weight $\boldsymbol{W}_{n+2}^V$ and $\frac{\partial\mathcal{L}}{\partial\boldsymbol{z}_{n+2}}$ in (6) and (7). The gradient of the latent vector does not incorporate injected input(s) and the corresponding weight(s) of subsequent layers. In addition, the gradient of the weights $\boldsymbol{W}_n^V$ and $\boldsymbol{W}_n^I$ in (8) and (9) does not include the subsequent injected input(s). Thus, the learning weights in a layer of the MMP Net is not affected by the control and environment of the subsequent stages. Thus, the MMP Net can represent the sequential process control mechanism of continuous MMPs.

3.3.2 Limitation of conventional FFNNs in continuous MMPs

The MMP Net is a feedforward neural network (FFNN)-based architecture, but it has a distinct difference with conventional FFNNs. Conventional FFNNs cannot operate with the composite structure described in Figure 1 and Equation (1), and thus cannot demonstrate the sequential process control mechanism of continuous MMPs. For instance, a one-layer FFNN model involves the equation $y = \boldsymbol{W}^T\mathbf{x}$, and the variables in all stages are included in the equation (i.e., $\mathbf{x} = concat([\mathbf{x}_1, ..., \mathbf{x}_N])$. Consequently, a one-layer FFNN model processes the data in all stages simultaneously in a mixed manner; thus, this model cannot accommodate the sequential characteristics of continuous MMPs. This limitation also applies to the use of multilayer FFNN models.

3.3.3 Limitation of conventional recurrent neural network (RNN)-type models in continuous MMPs

RNN-type models, such as conventional RNN, Long Short-Term Memory (LSTM), and Gated Recurrent Unit (GRU), employ a set of weights for each step, which are sequentially updated across all steps. In the context of MMPs, the unit of step can be the stage, and the RNN can learn the sequential structure of the MMPs by treating each stage as a step. However, RNN-type models cannot reflect the control mechanism of continuous MMPs because the weights (W^{hidden} and W^{input}) of such models are shared across all steps (i.e., all stages of MMPs). For example, when data from continuous MMPs without intermediate outputs are used to train an RNN, the weights are initially estimated using the Stage 1 data and then updated as the Stage 2 data are inputted. This process repeats until the data from all stages are inputted. Thus, RNNs estimate weights in the training process by aggregating the data across the stages sequentially but do not estimate weights per different stages, and the use of trained weights in the inference process does not appropriately represent the sequential mechanism of continuous MMPs.

4. Validation of MMP Net

We applied and evaluated the MMP Net to a real-world lube base oil production, a chemical continuous MMP in which the intermediate outputs are not observed. As shown in Figure 3, the feed passes from Stage 1 to Stage 4 in sequence before being transformed into the final product (i.e., lube base oil). Herein, the feed is unconverted oil, which is what remains of crude oil after producing petroleum products such as gasoline and diesel. Stages 1 and 2 involve catalyst towers; a chemical reaction occurs when the feed enters these stages. Stages 3 and 4 distill the feed by adjusting the tower temperature, pressure, reflux, and steam.

The quality of lube base oil is an important factor, because it is commonly used as a raw material to make lubricant-based products. However, maintaining lube base oil quality (as measured by the pour point and kinematic viscosity) consistently with production standards in real time is extremely difficult and costly in the continuous manufacturing process, due to the time and effort required in its measurement. The pour point is the minimum temperature, accurate to 1 °C, at which the oil loses its flow characteristics. Kinematic viscosity refers to a measurement of resistance to oil flow, accurate to 0.1 centistokes. Furthermore, the production involves numerous variables, and tracking intermediate product quality is infeasible due to high heat and pressure. This condition causes difficulties and complications when attempting to apply the multi-model approach to the lube base oil MMP. To solve this problem, we developed a soft analyzer that estimates the product quality in real time based on the prediction with the MMP Net.

4.1 Data description

We used the company's lube base oil production data, which consist of three types of data: process operation, feed specification, and product quality. Process operation data were collected by sensors at 10-minute intervals during production. These sensors recorded data on process environments and control throughout

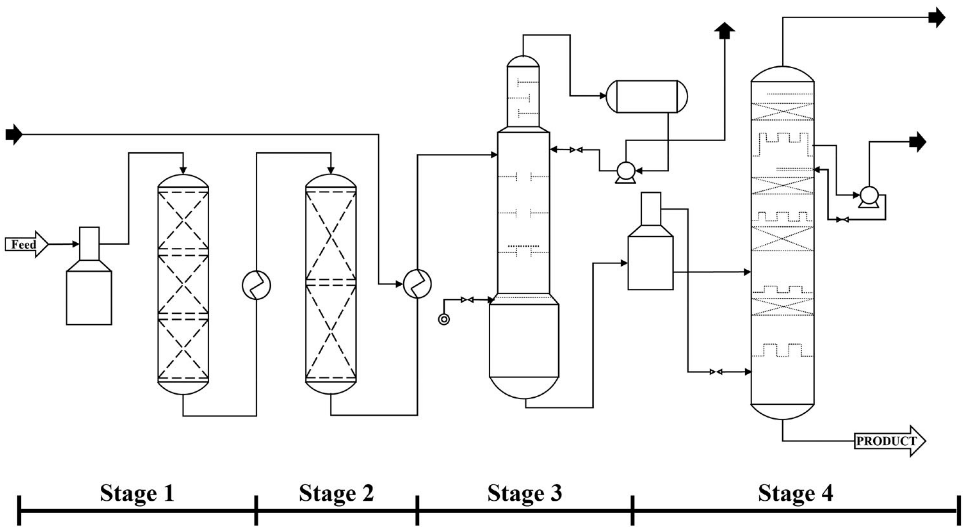

Figure 3. Multistage process flow diagram of lube base oil production.

the manufacturing in real time. The data consist of 5751 observations and 18 variables, which are categorized as Stage 1 (Reaction 1), Stage 2 (Reaction 2), Stage 3 (Distillation 1), and Stage 4 (Distillation 2). Feed specification and product quality data were measured by five characteristic properties and two characteristic properties, respectively. Both were collected by manual chemical experiments once or twice a day. Thus, the experimental values were interpolated by referring to the knowledge of domain experts (i.e., process engineers and operators), because the experimental data were scarce compared with process operation data. Specifically, the experimental values of feed specification were connected as the most recently measured experimental value was copied and pasted until a new experimental value is measured. This is because the experimental values of feed specification do not change considerably. The experimental values of product quality were connected by referring to the trend of kinetics-based quality sensor data. Although the kinetics-based quality sensor data could be collected in real time, its measurement errors were not accurate; hence, the chemical engineers who worked for the manufacturer did not use it. However, the trend information it provided was considered reliable. Therefore, we used the connected data as interpolated labels for the real-time prediction performance of the proposed MMP Net and other baseline models. The variables of each category are presented in Table 1.

4.2 Data preprocessing

For data preprocessing, we initially eliminated outliers that were identified by domain experts on the manufacturing process. These outliers were caused by interruptions or sensor errors. We transformed all variables to have a mean of zero and a standard deviation of one. Next, time lag alignment between different stages was performed. As shown in Figure 4(a), one significant challenge in using the data collected from continuous MMPs is that the control variables and input processed by the controls do not match the row because the data were collected in real time. Specifically, the control data of the i^{th} stage (i.e., Stage i) and of Stage i+1 at time t are attributed to different inputs. In other words, the operations and output data located in the same row do not match directly. To generate a direct relationship between the variables and outputs in the production process, we must place all data on the same oil in the same row. Therefore, after an extensive discussion with domain experts, a time lag was calculated and adjusted by considering the oil flow rate and the pipe width (Figure 4(b)).

4.3 Modeling

Figure 5 shows the MMP Net architecture for learning the lube base oil production. The proposed framework has four stage modules and output regression modules. Each stage module consists of a hidden layer, Batch Normalization (BN) layer, activation layer, and a dropout. BN (Luo *et al.*, 2018) and dropout (Wager *et al.*, 2013) were utilized for regularization to solve the uncertainty propagation in the real-world dataset. The regression module has a simple MLP structure. Each stage module inputs variables corresponding

Table 1. Variables in lube base oil production.

Variable category	*Variables*
Feed specification	Kinematic viscosity, viscosity index, sulfur, nitrogen, pour point
Stage 1 (Reaction 1)	Feed flow, catalyst life index, catalyst average temperature, and H_2 flow
Stage 2 (Reaction 2)	Catalyst average temperature
Stage 3 (Distillation 1)	O.H. temperature, O.H. pressure, reflux, tower temperature, steam flow
Stage 4 (Distillation 2)	O.H. temperatures, O.H. pressure, reflux, steam flow, inflow temperature, viscosity chemical analyzer, pour point chemical analyzer
Product quality (label)	Kinematic viscosity, pour point

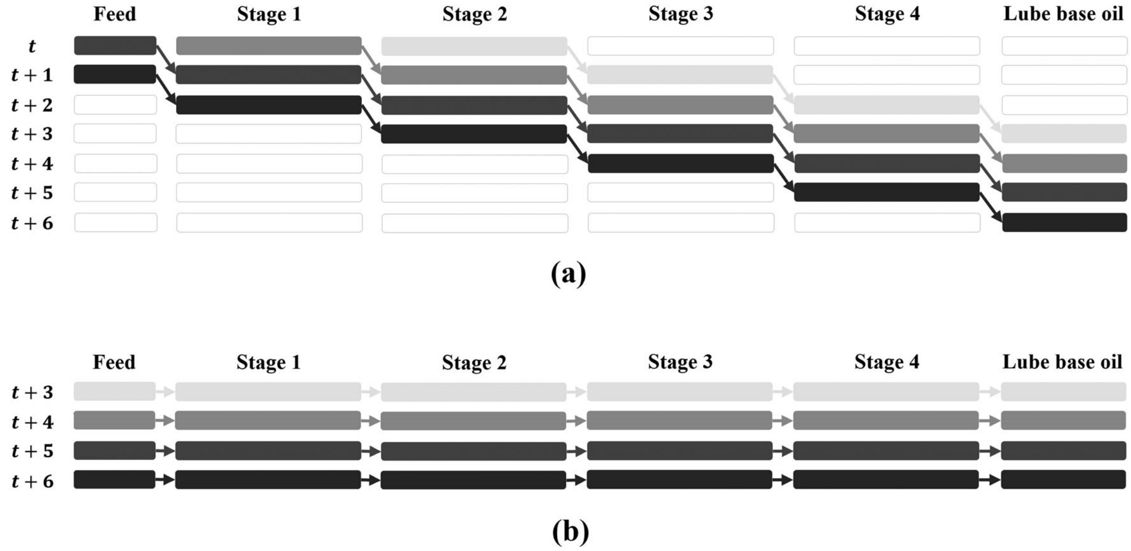

Figure 4. Time lag alignment of each stage variables and output in lube base oil production process. (a) before and (b) after applying a time lag alignment based on the output.

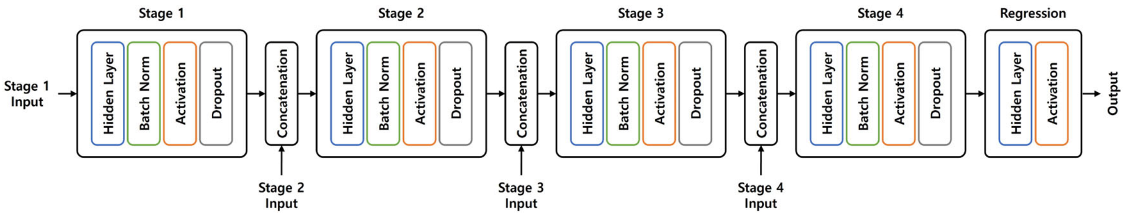

Figure 5. MMP Net architecture for lube base oil production process.

to each stage and concatenates it with the previously processed variables. For example, feed specification variables and Reaction I variables are the input into the Stage 1 module. The output of the Stage 1 module and Reaction II variables were concatenated and inputted into the Stage 2 module.

The epoch, learning rate, batch size, dropout proportion, activation function, and optimizer were set as 100, 0.001, 128, 10%, ReLU, and Adam, respectively. For the tuning of the hidden size in the MMP Net, we used the grid search method to find the optimal value that minimizes the Mean Square Error (MSE) of the validation set as follows: hidden size in [50, 100, 150, 200]. The number of hidden layers and size in the regression module were set as one and half of the hidden size in MMP, respectively. The parameters of each model were determined during training as values with minimum validation loss using the early stopping method.

4.4 Experimental setup

To evaluate the performance of the MMP Net, we compared the model with baseline models: linear regression, Least Square Support Vector Regressor (LSSVR), random forest, LightGBM, XGBoost, one-layer FFNN, multilayer FFNN, RNN, LSTM, and GRU. Our experiment involves two types of FFNN: FFNN without BN and FFNN with BN. The FFNN without BN is a conventional FFNN without BN and dropout. Additionally, a Multilayer FFNN (with BN) was also tested, which has the same depth, width, and user-defined parameters with the MMP Net to check the performance of the injected layers and conduct fair comparison with the MMP Net. RNN, LSTM, and GRU models, which can capture sequential patterns, were experimented by inputting each stage data sequentially. Herein, embedding layers were added in these models to match the same input dimension (i.e., we matched all stage dimensions through embedding layers). Linear regression was compared in that it can represent a state space model generalizable for continuous MMPs considering the states defined at the process stage level, not the time level. Specifically, in view of the control mechanism of continuous MMPs without intermediate outputs shown in Figure 1 and Equation (1), a state space model can be conceptually defined as $y_N = \mathbf{A}_N^T(\ldots(\mathbf{A}_3^T(\mathbf{A}_2^T(\mathbf{A}_1^T\mathbf{x}_1 + \mathbf{x}_2) + \mathbf{x}_3)\ldots) + \mathbf{x}_N)$, where $A_i \in \mathbb{R}^{d_i \times d_{i+1}}\ \forall i = 1, 2, \ldots, (N-1)$ and $A_N \in \mathbb{R}^{d_N \times 1}$ are the i^{th} weight of the model. This eventually becomes the same as a linear regression model.

The experiment was repeated five times using random seeds ranging from zero to four (except for LSSVR, which does not utilize a random seed). The dataset was divided chronologically into training, validation, and test sets in a ratio of 80:10:10. For the tuning of user-defined parameters, the grid search method was used to find the optimal values that minimize the MSE of the validation set. Appendix A summarizes the best user-defined parameters identified through the grid search.

We used MSE, Root Mean Squared Error (RMSE), and Mean Absolute Error (MAE) as the evaluation metrics of product quality prediction, as shown as follows:

$$\mathrm{MSE} = \frac{1}{n}\sum\nolimits_{i=1}^{n}\left(\hat{y}_i - y_i\right)^2, \tag{10}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum\nolimits_{i=1}^{n}\left(\hat{y}_i - y_i\right)^2}, \tag{11}$$

$$\mathrm{MAE} = \frac{1}{n}\sum\nolimits_{i=1}^{n}\left|\hat{y}_i - y_i\right|, \tag{12}$$

where $\hat{y}_i$ and y_i represent the predicted and actual values, respectively; and n is the number of samples.

4.5 Results

Table 2 compares the performance of the MMP Net with other baseline models for predicting pour point and kinematic viscosity. Each value in the table represents the average value and standard deviation of the five repeated experiments (except for LSSVR, which does not utilize a random seed). Although the MMP Net outperforms the baseline models, the multilayer FFNN (with BN)* shows lower performance than the MMP Net, indicating that the existence of injected input layers positively affects the performance. Interestingly, while the RNN-type models show lower performance than the MMP Net, they are inferior to the linear regression and one-layer FFNN models. This finding suggests that updating and using a single set of weights may be unsuitable for reflecting the control mechanism of continuous MMPs and learning the data from such processes. Lastly, the high performance of the linear regression may indicate that the aforementioned general stage space model may partially reflect the control mechanism of continuous MMPs and be useful for learning data from such processes.

Figures 6 and 7 show the MMP Net prediction results for pour point and kinematic viscosity, respectively. The values of these figures are the original scale values reconverted from the standardized values. The MMP Net follows the label trend and can predict results close to the label, indicating that the MMP Net can be used as a real quality prediction analyzer for continuous MMPs.

Table 2. Performance comparison.

	Pour point			*Kinematic viscosity*		
	MSE	*RMSE*	*MAE*	*MSE*	*RMSE*	*MAE*
Linear Regression	**0.2376 ± 0.0298**	**0.4865 ± 0.0299**	**0.3808 ± 0.0332**	0.0003 ± 0.0000	0.0170 ± 0.0008	0.0134 ± 0.0011
LSSVR	0.8970	0.9471	0.7342	0.0006	0.0242	0.0191
RandomForest	0.4494 ± 0.0059	0.6704 ± 0.0044	0.6037 ± 0.0051	0.0012 ± 0.0001	0.0351 ± 0.0023	0.0299 ± 0.0019
LightGBM	0.3891 ± 0.0000	0.6238 ± 0.0000	0.5256 ± 0.0000	0.0007 ± 0.0000	0.0256 ± 0.0000	0.0224 ± 0.0000
XGBoost	0.3991 ± 0.0000	0.6317 ± 0.0000	0.5400 ± 0.0000	0.0007 ± 0.0000	0.0265 ± 0.0000	0.0210 ± 0.0000
1-layer FFNN (without BN)	0.4430 ± 0.1472	0.6422 ± 0.1331	0.4965 ± 0.0983	**0.0002 ± 0.0000**	**0.0149 ± 0.0010**	**0.0118 ± 0.0009**
Multi-layer FFNN (without BN)	0.4385 ± 0.1383	0.6521 ± 0.1152	0.5030 ± 0.0979	0.0004 ± 0.0001	0.0200 ± 0.0026	0.0169 ± 0.0025
1-layer FFNN (with BN)	0.4971 ± 0.1231	0.7003 ± 0.0820	0.5503 ± 0.0618	0.0003 ± 0.0001	0.0168 ± 0.0017	0.0140 ± 0.0016
Multi-layer FFNN (with BN)	0.2502 ± 0.0597	0.4963 ± 0.0626	0.3866 ± 0.0357	0.0005 ± 0.0002	0.0232 ± 0.0034	0.0198 ± 0.0031
Multi-layer FFNN (with BN)*	0.2530 ± 0.0603	0.4995 ± 0.0598	0.3857 ± 0.0367	0.0004 ± 0.0001	0.0205 ± 0.0036	0.0168 ± 0.0034
RNN	0.4533 ± 0.0516	0.6722 ± 0.0373	0.4914 ± 0.0329	0.0004 ± 0.0000	0.0196 ± 0.0012	0.0158 ± 0.0005
LSTM	0.5281 ± 0.1142	0.7228 ± 0.0751	0.5259 ± 0.0779	0.0003 ± 0.0002	0.0174 ± 0.0045	0.0136 ± 0.0031
GRU	0.4466 ± 0.1511	0.6584 ± 0.1148	0.4414 ± 0.0655	0.0004 ± 0.0001	0.0202 ± 0.0029	0.0174 ± 0.0026
MMP Net	**0.2226 ± 0.0250**	**0.4711 ± 0.0264**	**0.3678 ± 0.0174**	**0.0002 ± 0.0000**	**0.0118 ± 0.0012**	**0.0097 ± 0.0012**

*While the user-defined parameters of all the other neural network models are optimized, the depth and width of this network are kept the same as those of the MMP Net

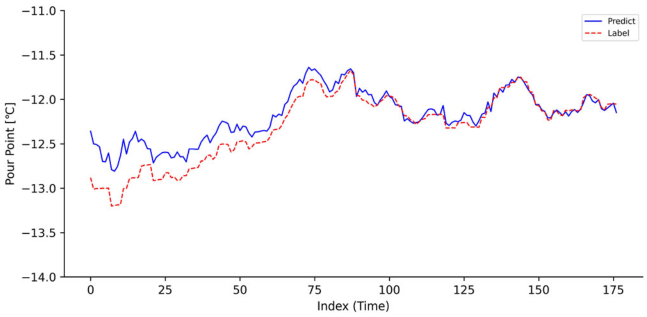

Figure 6. Pour point quality prediction with MMP Net.

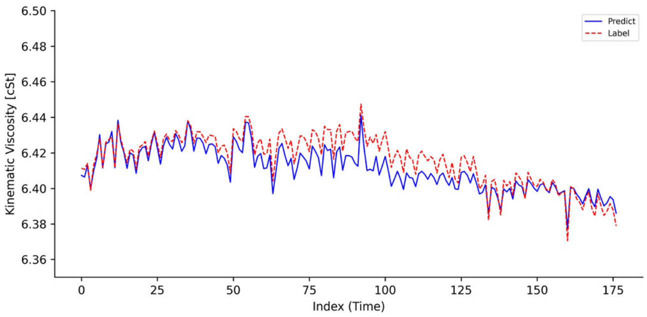

Figure 7. Kinematic viscosity quality prediction with MMP Net.

We believe that the superiority of the MMP Net over the baseline models can be attributed to its ability to reflect the control mechanism of continuous MMPs. The injected input layer of the MMP Net reflects the variable sequence in the model. In our real-world application of the MMP Net, the first-stage data flow through the first hidden layer(s) and produce the first-stage latent vector, which represents the control in the first stage of the MMP under consideration. Subsequently, the first-stage latent vector is merged with the second-stage data to generate the second-stage latent vector, which represents the control in the first and second stages of the MMP under consideration. This training mechanism of the MMP Net exactly matches the sequential process control mechanism of continuous MMPs, in which the raw material inputs become the first intermediate outputs after passing through the process control in the first stage and the first intermediate outputs become the second intermediate outputs after passing through the process control in the second stage. This simple and end-to-end single model learning approach can reduce the risk of losing the information of variables in each stage and propagate the gradients backward by reflecting the real control mechanism of continuous MMPs.

4.6 Solution implementation in the factory

Traditionally, the control of lube base oil production has been based on chemical experiments on the product quality and supplemented by a chemical analyzer. If the product specification is experimentally out of the normal range, then changes occur in the process control variables, such as the temperature and flow of certain stages. Such "manual" control has been conservative, because no trustworthy information can be consulted in real time. If the performance of the chemical analyzer is inaccurate, then the operators and engineers must wait and rely on future experiments to verify the result of their control. This process takes approximately 8 hours on average to complete and is expensive (Figure 8(a)). Given the uncertainty about the current product specification and quality, controlling the process or responding to unexpected changes has been extremely difficult.

The practical motivation of this study originated from the challenges faced by the lube oil manufacturer, whereas the academic motivation comes from the lack of a single model for the MMP Net. Given its superiority over existing methods, the soft analyzer solution based on the MMP Net was implemented in an actual lubricant factory in 2022. As demonstrated in the aforementioned results, the prediction performance of the soft analyzer with the MMP Net was extensively evaluated for real use by the operators and engineers in the factory and is validated superior to those of other baseline machine learning models and current chemical analyzers. With this soft analyzer solution, operators and engineers can monitor the process in real time and fine-tune it as necessary (Figure 8(b)).

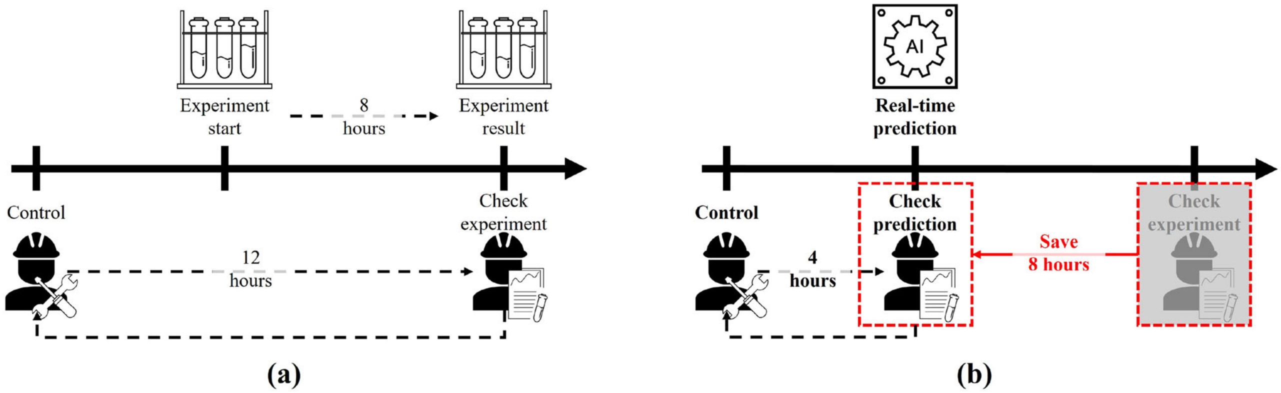

Figure 8. Control of operators is reflected in the product specification after approximately 4 hours: (a) Control based on experimentation needs 12 hours to obtain a result and is performed once or twice in a day; (b) The soft sensor can predict the product specification in real-time.

5. Concluding remarks

In this article we propose the MMP Net, a FFNN model that accurately represents the "control mechanism of continuous MMPs" in manufacturing industries, such as liquid production processes in petrochemical and drug manufacturing. The injected input layers enable the MMP Net to reflect the sequential characteristics of the MMP of interest and eliminate the accumulated error from which existing multi-model approaches typically suffer. The effectiveness of the MMP Net is quantitatively validated based on its application to real-world lubricant base oil manufacturing, showing a superior performance than those of other models. Based on the experimental results, the MMP-Net-equipped soft analyzer was actually implemented in a lubricant factory. The yield enhancement by controlling the quality of pour point and the kinematic viscosity is expected to improve by 0.285% and 0.365%, respectively, resulting in a total savings of 900,000 USD per year.

In modern machine learning application studies, reflecting the inherent characteristics of the problem is the key to learning the real data collected from the original problem source. For example, the characteristics can be reflected in designing the loss function, such as physics-informed loss function design (Raissi *et al.*, 2019), chemistry-informed feature engineering (Kim *et al.*, 2023), and the customization of the neural network architecture with expert knowledge (Gao *et al.*, 2018). Our proposed MMP Net is in line with this hybrid approach by reflecting the characteristics of manufacturing processes with multiple and sequential stages by using a novel architecture. As shown in the results, the basic MMP Net outperforms other well-known existing models in the prediction problem of MMPs. Moreover, to serve various purposes, the proposed framework can be flexibly combined with other deep learning techniques, such as a convolutional layer, transfer learning, and attention mechanism. In conclusion, we believe that the MMP Net illustrates how the gap between neural network models and the real manufacturing can be bridged successfully.

Nonetheless, this study has several limitations that need consideration in future work. For example, in our real-world application of the MMP Net, the time lag adjustment was implemented based on domain knowledge when the columns and rows of data were adjusted according to the lube base oil production. However, this adjustment can potentially introduce errors. A statistical approach can be devised to find an optimal case of time lag adjustment. We also identified potential research topics that could systematically address the data uncertainty issues of continuous MMPs without intermediate outputs. These topics include the integration of multirow learning approaches (e.g., convolution operation) and transfer learning techniques with the proposed MMP Net to address uncertainties in time series inputs to the network and catalyst-based chemical reactions in continuous MMPs in petrochemical industries, respectively. The combination of multirow learning approaches that can extract temporal features (e.g., dilated causal convolution) with MMP Net may be able to address another limitation of the MMP Net, which is its inability to consider nonstationary dependencies between time series data. However, even with these limitations, we believe that implementing and extending the MMP Net can reduce the prediction error rate or increase the production yield not only in chemical manufacturing, such as the lubricant case that is examined here, but also in other production optimization problems of continuous MMPs, such as for steel, drug, and food manufacturing. We plan to apply the MMP Net to such processes in the future.

Acknowledgments

The authors would like to sincerely thank the editors and anonymous reviewers for their helpful and insightful comments and suggestions that have resulted in a much-improved version of this manuscript.

Funding

This work was supported by the following institutes and grants: National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2021R1I1A4A01049121; NRF-2021S1A5A2A03065747; NRF-2022R1A6A3A13072975; NRF-2022S1A5A8054010; NRF-2023S1A5A8080351), Institute of Information & Communications Technology Planning & Evaluation (IITP) grants funded by the Korean government (MSIT) (No. 2020-0-01336: Artificial Intelligence Graduate School Program - UNIST, No. 2021-0-02068, Artificial Intelligence

Innovation Hub), and project funds (1.220081.01, 1.220073.01, 1.230058.01, 1.230019.01) of UNIST (Ulsan National Institute of Science & Technology).

ORCID

Hojin Cho http://orcid.org/0000-0001-9035-7545
Kyeongbin Kim http://orcid.org/0000-0001-6910-4754
Kihyuk Yoon http://orcid.org/0000-0003-0492-2865
Junghye Lee http://orcid.org/0000-0003-3221-1526
Chiehyeon Lim http://orcid.org/0000-0001-6112-9674

References

Amini, M. and Chang, S. (2018) A review of machine learning approaches for high dimensional process monitoring, in *Proceedings of the IISE Annual Conference and Expo*, pp. 390–395.

Arif, F., Suryana, N. and Hussin, B. (2013) A data mining approach for developing quality prediction model in multi-stage manufacturing. *International Journal of Computer Applications*, **69**(22), 5–40.

Bera, S. and Mukherjee, I. (2016) A multistage and multiple response optimization approach for serial manufacturing system. *European Journal of Operational Research*, **248**(2), 444–452.

Chen, W.C., Tai, P.H., Wang, M.W., Deng, W.J. and Chen, C.T. (2008) A neural network-based approach for dynamic quality prediction in a plastic injection molding process. *Expert Systems with Applications*, **35**(3), 843–849.

Chen, X., Deng, T., Shen, Z.J.M. and Yu, Y. (2023) Mind the gap between research and practice in operations management. *IISE Transactions*, **55**(1), 32–42.

Gao, X., Jin, S., Wen, C.K. and Li, G.Y. (2018) ComNet: Combination of deep learning and expert knowledge in OFDM receivers. *IEEE Communications Letters*, **22**(12), 2627–2630.

Hao, X., Liu, C., Liu, M., Zhang, C. and Zheng, L. (2022) Solving a real-world large-scale cutting stock problem: A clustering-assignment-based model. *IISE Transactions*, **55**(11), 1160–1173.

Jiang, P., Jia, F., Wang, Y. and Zheng, M. (2014) Real-time quality monitoring and predicting model based on error propagation networks for multistage machining processes. *Journal of Intelligent Manufacturing*, **25**(3), 521–538.

Jin, R. and Shi, J. (2012) Reconfigured piecewise linear regression tree for multistage manufacturing process control. *IIE Transactions*, **44**(4), 249–261.

Kano, M. and Nakagawa, Y. (2008) Data-based process monitoring, process control, and quality improvement: Recent developments and applications in steel industry. *Computers & Chemical Engineering*, **32**(1-2), 12–24.

Kim, J., Lee, Y. and Kim, H. (2018) Detection and clustering of mixed-type defect patterns in wafer bin maps. *IISE Transactions*, **50**(2), 99–111.

Kim, M. and Liu, K. (2020) A Bayesian deep learning framework for interval estimation of remaining useful life in complex systems by incorporating general degradation characteristics. *IISE Transactions*, **53**(3), 326–340.

Kim, Y., Lim, C., Lee, J., Kim, S., Kim, S. and Seo, D.H. (2023) Chemistry-informed machine learning: Using chemical property features to improve gas classification performance. *Chemometrics and Intelligent Laboratory Systems*, **237**, 104808.

Lee, C., Kim, S., Lim, C., Kim, J., Kim, Y. and Jung, M. (2021) Diet planning with machine learning: Teacher-forced REINFORCE for composition compliance with nutrition enhancement, in *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 3150–3160.

Lee, D.H., Kim, S.H. and Kim, K.J. (2021) Multistage MR-CART: Multiresponse optimization in a multistage process using a classification and regression tree method. *Computers & Industrial Engineering*, **159**, 107513.

Li, J., Freiheit, T., Hu, S.J. and Koren, Y. (2007) A quality prediction framework for multistage machining processes driven by an engineering model and variation propagation model. *Journal of Manufacturing Science and Engineering*, **129**(6), 1088–1100.

Li, Y., Du, J. and Jiang, W. (2023) Reinforcement learning for process control with application in semiconductor manufacturing. *IISE Transactions*, in press and online available.

Liu, C., Tang, L., Liu, J. and Tang, Z. (2018) A dynamic analytics method based on multistage modeling for a BOF steelmaking process. *IEEE Transactions on Automation Science and Engineering*, **16**(3), 1097–1109.

Luo, P., Wang, X., Shao, W. and Peng, Z. (2019) Towards understanding regularization in batch normalization, in *Proceedings of the 7th International Conference on Learning Representations*, OpenReview.net, New Orleans, LA, USA, pp. 1–23

Niaki, S.T.A. and Davoodi, M. (2009) Designing a multivariate–multistage quality control system using artificial neural networks. *International Journal of Production Research*, **47**(1), 251–271.

Oroojlooyjadid, A., Snyder, L.V. and Takáč, M. (2020) Applying deep learning to the newsvendor problem. *IISE Transactions*, **52**(4), 444–463.

Park, S., Kim, K. and Kim, H. (2022) Prediction of highly imbalanced semiconductor chip-level defects using uncertainty-based adaptive margin learning. *IISE Transactions*, **55**(2), 147–155.

Raissi, M., Perdikaris, P. and Karniadakis, G.E. (2019) Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, **378**, 686–707.

Sales-Setién, E., Penarrocha-Alós, I. and Abellán-Nebot, J.V. (2018) Estimation of nonstationary process variance in multistage manufacturing processes using a model-based observer. *IEEE Transactions on Automation Science and Engineering*, **16**(2), 741–754.

Seo, H., Shin, J., Kim, K.H., Lim, C. and Bae, J. (2022) Driving risk assessment using non-negative matrix factorization with driving behavior records. *IEEE Transactions on Intelligent Transportation Systems*, **23**(11), 20398–20412.

Sharp, M., Ak, R. and Hedberg Jr, T. (2018). A survey of the advancing use and development of machine learning in smart manufacturing. *Journal of Manufacturing Systems*, **48**, 170–179.

Shi, J. and Zhou, S. (2009) Quality control and improvement for multistage systems: A survey. *IIE Transactions*, **41**(9), 744–753.

Sun, C., Ma, M., Zhao, Z., Tian, S., Yan, R. and Chen, X. (2018) Deep transfer learning based on sparse autoencoder for remaining useful life prediction of tool in manufacturing. *IEEE Transactions on Industrial Informatics*, **15**(4), 2416–2425.

Tello, G., Al-Jarrah, O.Y., Yoo, P.D., Al-Hammadi, Y., Muhaidat, S. and Lee, U. (2018) Deep-structured machine learning model for the recognition of mixed-defect patterns in semiconductor fabrication processes. *IEEE Transactions on Semiconductor Manufacturing*, **31**(2), 315–322.

Tian, W., Jin, R., Huang, T. and Camelio, J.A. (2017) Statistical process control for multistage processes with non-repeating cyclic profiles. *IISE Transactions*, **49**(3), 320–331.

Valyon, J. and Horváth, G. (2009) A sparse robust model for a Linz–Donawitz steel converter. *IEEE Transactions on Instrumentation and Measurement*, **58**(8), 2611–2617.

Wager, S., Wang, S. and Liang, P.S. (2013) Dropout training as adaptive regularization, in *Proceedings of the 26th International Conference on Neural Information Processing Systems,* Curran Associates, Inc., Stateline, NV, USA, pp. 351–359.

Wang, D., Liu, J. and Srinivasan, R. (2009) Data-driven soft sensor approach for quality prediction in a refining process. *IEEE Transactions on Industrial Informatics*, **6**(1), 11–17.

Wang, K. and Tsung, F. (2021) Hierarchical sparse functional principal component analysis for multistage multivariate profile data. *IISE Transactions*, **53**(1), 58–73.

Wu, X., Chen, J., Xie, L., Chan, L.L.T. and Chen, C.I. (2020) Development of convolutional neural network based Gaussian process regression to construct a novel probabilistic virtual metrology in multi-stage semiconductor processes. *Control Engineering Practice*, **96**, 104262.

Yang, H., Kumara, S., Bukkapatnam, S.T. and Tsung, F. (2019) The internet of things for smart manufacturing: A review. *IISE Transactions*, **51**(11), 1190–1216.

Yin, X., He, Z., Niu, Z. and Li, Z.S. (2018) A hybrid intelligent optimization approach to improving quality for serial multistage and multi-response coal preparation production systems. *Journal of Manufacturing Systems*, **47**, 199–216.

Yoo, J., Soun, Y., Park, Y.C. and Kang, U. (2021) Accurate multivariate stock movement prediction via data-axis transformer with multi-level contexts, in *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, Association for Computing Machinery Virtual Event, Singapore, pp. 2037–2045.

Yu, J. (2012) Online quality prediction of nonlinear and non-Gaussian chemical processes with shifting dynamics using finite mixture model based Gaussian process regression approach. *Chemical Engineering Science*, **82**, 22–30.

Yuan, X., Huang, B., Wang, Y., Yang, C. and Gui, W. (2018) Deep learning-based feature representation and its application for soft sensor modeling with variable-wise weighted SAE. *IEEE Transactions on Industrial Informatics*, **14**(7), 3235–3243.

Zhang, D., Liu, Z., Jia, W., Liu, H. and Tan, J. (2021) Path enhanced bidirectional graph attention network for quality prediction in multistage manufacturing process. *IEEE Transactions on Industrial Informatics*, **18**(2), 1018–1027.

Zhu, Q.X., Zhang, X.H., Wang, Y., Xu, Y. and He, Y.L. (2019) A novel intelligent model integrating PLSR with RBF-kernel based extreme learning machine: Application to modelling petrochemical process. *IFAC-PapersOnLine*, **52**(1), 148–153.

Zou, C. and Tsung, F. (2008) Directional MEWMA schemes for multistage process monitoring and diagnosis. *Journal of Quality Technology*, **40**(4), 407–427.

Appendix A

Table 3. (Pour point) Best user-defined parameters for deep learning models.

	Learning			*Architecture*				
Model	*# of epochs*	*Learning rate*	*Batch size*	*# of layers*	*Hidden size*	*activation*	*dropout*	*# of parameters*
MMP Net	100	0.001	128	5 (4 stage & 1 hidden layer)	150	ReLU	0.1	84,151
1-layer FFNN (without BN)	100	0.001	128	1	150	ReLU	–	3751
1-layer FFNN (with BN)	100	0.001	128	1	50	ReLU	0.1	1351
Multilayer FFNN (without BN)	100	0.001	128	4	150	ReLU	–	71,701
Multilayer FFNN (with BN)	100	0.001	128	5	100	ReLU	0.1	43,901
Multilayer FFNN (with BN)*	100	0.001	128	5	150	ReLU	0.1	84,151
RNN	100	0.001	128	5	50	–	–	26,901
LSTM	100	0.001	128	1	50	–	–	21,801
GRU	100	0.001	128	5	100	–	–	305,801

*While the user-defined parameters of all the other neural network models are optimized, the depth and width of this network are kept the same as those of the MMP Net.

Table 4. (Pour point) Best user-defined parameters for machine learning models.

Model	*User-defined parameters*
LSSVR	C: 5 / gamma: 0.1
XGBoost	learning_rate: 0.1 / n_estimators: 500 / max_depth: 10
LightGBM	n_estimators: 1000 / max_depth: 10
RandomForest	n_estimators: 1000 / max_depth: 20

Table 5. (Kinematic viscosity) Best user-defined parameters for deep learning models.

	Learning			*Architecture*				
Model	# of epochs	*Learning rate*	*Batch size*	*# of layers*	*Hidden size*	*activation*	*dropout*	*# of parameters*
MMP Net	100	0.001	128	5 (4 stage & 1 hidden layer)	200	ReLU	0.1	147,201
1-layer FFNN (without BN)	100	0.001	128	1	200	ReLU	–	5001
1-layer FFNN (with BN)	100	0.001	128	1	200	ReLU	0.1	5401
Multilayer FFNN (without BN)	100	0.001	128	4	50	ReLU	–	8901
Multilayer FFNN (with BN)	100	0.001	128	4	50	ReLU	0.1	9301
Multilayer FFNN (with BN)*	100	0.001	128	5	200	ReLU	0.1	147,201
RNN	100	0.001	128	3	150	–	0.1	140,101
LSTM	100	0.001	128	5	100	–	0.1	406,801
GRU	100	0.001	128	1	150	–	0.1	140,101

*While the user-defined parameters of all the other neural network models are optimized, the depth and width of this network are kept the same as those of the MMP Net.

Table 6. (Kinematic viscosity) Best user-defined parameters for machine learning models.

Model	*User-defined parameters*
LSSVR	C: 80 / gamma: 0.1
XGBoost	learning_rate: 0.01 / n_estimators: 500 / max_depth: None
LightGBM	n_estimators: 100 / max_depth: None
RandomForest	n_estimators: 100 / max_depth: 20