Expandable neural networks for efficient modeling of various amine scrubbing configurations for CO_2 capture

Yu-Da Hsiao, Chuei-Tin Chang [*]

Department of Chemical Engineering, National Cheng Kung University, Tainan, Taiwan

ARTICLE INFO

ABSTRACT

Modeling of improved amine scrubbers using artificial neural networks (ANNs) were carried out in this study. Instead of training models from scratch with the case-by-case method, the expandable neural networks were utilized to progressively increase the number of model parameters and change the model input/output structures in a step-wise fashion. Efficient and rapid transformation of any existing model to a new one can be made realizable. This proposed strategy has been successfully validated in several process modification scenarios. From the experimental results, the required sampling sizes to achieve the similar prediction accuracy of the corresponding baseline model were considerably smaller, and, furthermore, over 47% of total data acquisition time can be saved. Finally, the corresponding sensitivity analyses showed that the proposed models were physically interpretable and able to extract the correct process mechanisms in the sense that the gain scales and signs were consistent with those of their rigorous counterparts.

1. Introduction

Due to the increasing emission rates of greenhouse gases from the industries, the amine scrubbing process (which includes both absorber and stripper) is widely considered to be a feasible option for effective post-combustion CO_2 capture (PCC) (Khalifa et al., 2022). However, the commercial-scale amine scrubber for CO_2 capture is still not widely implemented today due to its high capital and operational costs. In addition, the current CO_2 supply chains and the downstream conversion technologies are still unreliable to induce industrial applications of CO_2 capture, utilization and sequestration (CCUS) (Álvarez et al., 2017; Hasan et al., 2014). In other words, the development of CO_2 capture technologies is still at the stage of pilot testing and demonstration (Rochelle et al., 2019; Wu et al., 2010; Zhuang et al., 2022).

In recent years, a large number of researches have focused upon developments of amine scrubbing systems via absorbent improvements and process modifications (Aghel et al., 2022; Ahn et al., 2013; Borhani and Wang, 2019; Cousins et al., 2011). These studies aimed to raise the system performances by enhancing absorption efficiency and/or reducing energy consumption. Most of them relied on the first-principle models, e.g., Aspen HYSYS, to facilitate rigorous analyses. These models were derived typically on the basis of the complicated physiochemical equations to ensure high-fidelity simulations. Although successful

implementations of simulation-based optimization studies have been reported (Oh et al., 2016), the use of first-principle models in numerical optimization applications may not always be effective due to the high complexity and nonlinearity (Chung and Lee, 2020). Furthermore, the corresponding optimization runs may even fail to converge (Alarie et al., 2021). Therefore, the data-driven surrogate models can be considered to be a viable alternative to effectively lower the optimization effort (Caballero and Grossmann, 2008; Kajero et al., 2017; McBride and Sundmacher, 2019).

The data-driven models (DDMs) have already been applied extensively to the analysis, optimization and monitoring of amine scrubbing processes. Nonlinear surrogate models were used for the systematic evaluations of steam consumption rate and equipment purchase cost needed to design and operate these plants (Chung and Lee, 2020). The multiple-regression technique was used to correlate and analyze the relationships among various critical parameters of amine scrubbing process (Wu et al., 2010; Zhou et al., 2009). A popular type of DDMs, that is, artificial neural networks (ANNs) have been applied in numerous studies to surrogate modeling of the typical amine scrubber (Goldstein et al., 2022; Li et al., 2015; Li et al., 2018; Nuchitprasittichai and Cremaschi, 2013; Sipöcz et al., 2011), and also been utilized to analyze and monitor the operations of real-world pilot plants (Wu et al., 2010; Zhuang et al., 2022). The scaled conjugate gradient (SCG) and

---

* Corresponding author at: Department of Chemical Engineering, National Cheng Kung University (NCKU), Tainan 701, Taiwan.
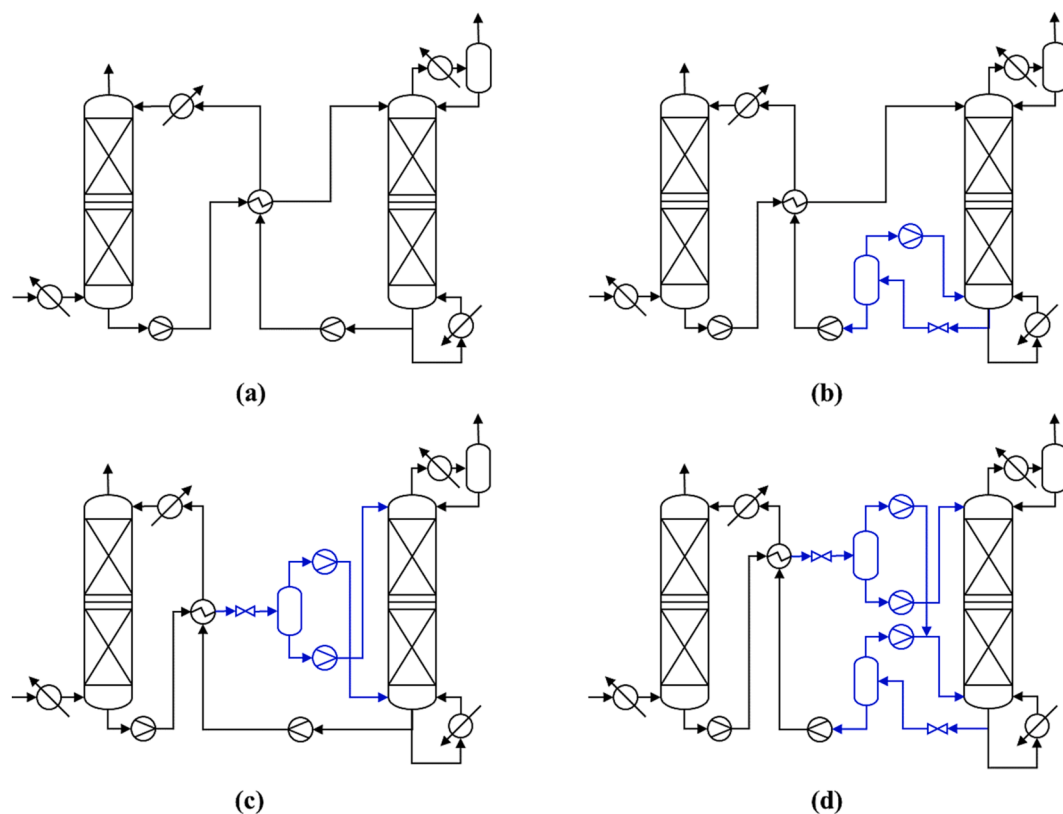 *E-mail address:* ctchang@mail.ncku.edu.tw (C.-T. Chang).

Levenberg-Marquardt (LM) algorithms were compared and adopted for modeling the simulative CO_2 capture plants by the feedforward neural networks (Sipöcz et al., 2011). The accurate ANN surrogate model may also replace the objective function in optimization problems, e.g., CO_2 capture cost minimization (Nuchitprasittichai and Cremaschi, 2013). The ANNs were also applied to model and simulate the CO_2 capture processes at different concentration levels, which were then compared with the simulation results of first-principle models (Goldstein et al., 2022; Henao and Maravelias, 2011). Since DDMs' predictions sometimes may be contaminated due to various uncertainty issues, the steady-state and dynamic surrogate models for estimating capture rate and level were constructed based on the bootstrap aggregated neural networks (BANNs) to enhance both accuracy and comprehensiveness (Li et al., 2015). Also, a similar modeling task was accomplished by using deep belief network (DBN) with many layers of restricted Boltzmann machine (RBM) to improve representation of its deep hierarchical features (Li et al., 2018). Aside from surrogate modeling, the neural network synthesis and the corresponding prediction sensitivity analysis approaches were integrated to identify the relationships between the key parameters of a CO_2 capture pilot plant (Wu et al., 2010). A dynamic hybrid-driven soft sensor which combines the chemical kinetic model and the long short-term memory (LSTM)-based model was also proposed to predict the CO_2 concentration profile within the absorption column (Zhuang et al., 2022).

Although ANN-based models have been extensively applied in the past to model the typical amine scrubber, its modified configurations have never been analyzed thoroughly. To construct the surrogate models for these modified configurations, the approach of training models from scratch individually is feasible but of course inefficient since the neural network parameters are randomly initialized, and a large number of samples should be collected from the corresponding high-fidelity simulation runs or pilot experiments. This modeling strategy should be considered to be computationally and economically expensive (Ochoa-Estopier et al., 2014). To resolve this issue of low model reusability (Goldstein et al., 2022), the transfer learning approaches are certainly appealing for a new model via utilizing the prior knowledge gained from previous learning from the process(es) with high similarities (Chuang et al., 2018; Hsiao et al., 2021; Lu et al., 2009). Nevertheless, since the input–output structures may vary if system configurations are modified, the modeling tasks may still be inefficient if multi-step flowsheet modifications take place at different stages of research and development. Therefore, in this study, the concept of progressive learning (Fayek et al., 2020), i.e., incremental learning or continual/lifelong learning (Parisi et al., 2019), was adopted to progressively construct new models for various amine scrubbing configurations by reusing and expanding the existing ANN models.

By following this improved novel modeling strategy, the number of input/output patterns and parameters of the existing models may be judiciously expanded to adapt to the new processes with only a small number of samples (Rusu et al., 2016; Yin et al., 2020; Yoon et al., 2018). In summary, a novel step-wise modeling procedure is proposed in this study for efficient construction of ANNs to characterize various modified configurations of the amine scrubbing process. More specifically, when compared with the base case, this approach is supposed to enable smoother and faster model transformations from the base structure to another improved ones with smaller sampling sizes and higher prediction accuracies.

2. Amine scrubbing process

2.1. Typical configuration

Fig. 1 depicts a typical amine scrubbing process, which consists of absorber, stripper, heat exchangers and pumps. The absorber is operated at the atmospheric pressure (i.e. 1 bar), while the stripper is pressurized to around 2 bar for energy-saving purpose. Before entering the scrubber,

Fig. 1. The flow diagram of the typical amine scrubbing process.

the particulate matter and impurities in flue gas from its source, i.e., SO_x and NO_x, are removed in pretreatment units (Korre et al., 2010). The pre-treated flue gas is then cooled down to a desired temperature and fed to the bottom of the absorber. The flue gas flow upward from bottom of the absorption column and counter-currently contacts with the down-flowing lean solvent introduced from the top. The chemical absorption reactions take place in the column together with complex heat and mass transfers. As a result, the CO_2 content is gradually removed from the flue gas. After scrubbing, the treated gas is eventually vented from the overhead. In practical operations, before dispersing to the ambient air, the treated gas should first go through a water wash section to recover amine. On the other hand, the CO_2 loading of amine solvent gradually rises up to a higher level as it flows downward, and finally leaves from the bottom. The rich solvent is regenerated in the stripper, where the captured CO_2 is desorbed and used as the final product. This CO_2 product is then sent to the compression train and dehydration section to facilitate distribution to other CO_2 user(s). Before entering the stripper from the top, it is a common practice to send the rich solvent to the lean/rich cross heat exchanger (LRHX) to recover the sensible heat of the hot lean solvent from the bottom of the stripper.

2.2. Modified configurations

Modifying process flowsheets is an important approach to improve the energy efficiency of amine scrubbing systems (Ahn et al., 2013; Cousins et al., 2011; Le Moullec et al., 2014). A variety of modified configurations have been proposed and extensively studied to enhance the commercial feasibility of PCC applications. Typically, these configurations strive to reduce the specific reboiler duty and/or total equivalent works. From the theoretical viewpoint, these modifications may be divided into three major categories, that is, absorber enhancement, heat integration and heat pumps (Le Moullec et al., 2014). The popular approaches for absorber enhancement, e.g., the intercooled absorber (ICA) or rich solvent recycle (RSR) (Dubois and Thomas, 2018; Le Moullec et al., 2014; Rezazadeh et al., 2017), aim to alter the chemical kinetics of absorption by manipulating the temperature profiles in the absorption column. Heat integration adjustments were used primarily to raise the thermodynamic reversibility by properly manipulating the mass flows for heat recovery, e.g., rich solvent split (RSS) (Le Moullec et al., 2014). On the other hand, the heat-pump-based modifications, e.g., mechanical vapor recompression (MVR) (Le Moullec et al., 2014; Li et al., 2022; Nguyen and Wong, 2021), were also attractive options for the reduction of specific reboiler duty.

Modeling of the MVR-based configurations is considered to be a complicated and challenging task since the number of both input and

Fig. 2. Various configurations of amine scrubbing process for post-combustion CO_2 capture: (a) base configuration, (b) lean vapor recompression (LVR), (c) rich vapor recompression (RVR), and (d) lean/rich vapor recompression (LRVR).

output variables are simultaneously increased as the configurations is derived from other simpler structure. For example, if the base configuration is altered to form the lean/rich vapor recompression (LRVR) process, two input variables (i.e., lean flash pressure and rich flash pressure) and one output variable (i.e., vapor recompression duty) should be added to the new model. Consequently, the aforementioned highly related processes, including lean vapor recompression (LVR), rich vapor recompression (RVR), and LRVR, were selected to verify the effectiveness of model transformations with the proposed method.

As shown in Fig. 2(b), the LVR process is the most popular MVR-based configuration found in the published literatures (Khalifa et al., 2022). In this configuration, the lean solvent from the pressurized stripping column is flashed in a vessel at a lower pressure (Le Moullec et al., 2014). The overhead vapor is compressed and recycled back to the bottom of the stripper, while the bottom liquid is sent to the LRHX to heat up the cold rich solvent. The LVR process may be applicable to accommodate several different solvents, e.g., aqueous monoethanolamine (MEA), piperazine (PZ), ammonia and their blends (Dubois and Thomas, 2018; Nguyen and Wong, 2021; Ullah et al., 2019). Similar to the LVR process, the RVR configuration, i.e., Fig. 2(c), utilizes the vapor recompression operation to the rich solvent stream from the bottom of absorber (Le Moullec et al., 2014; Li et al., 2022). After the recovering of sensible heat from the hot lean solvent at the LRHX, the rich solvent is fed to a flash drum, and the rich vapor from top of the drum is then recompressed and injected to the bottom of the stripping column to serve as an additional vapor source. The rich liquid solvent at lower temperature from the bottom of the drum is fed to the top of the stripper, which effectively reduces the condenser duty. In addition, some studied the integration of LVR and RVR processes to form the so-called LRVR structure (see Fig. 2(d)), in which ammonia was adopted as solvent (Nguyen and Wong, 2021). The corresponding results showed that a total of 26–54% energy saving can be achieved with the LRVR configuration if compared with other ammonia-based amine scrubbers.

2.3. Process simulations

In this study, Aspen HYSYS V10 was employed to rigorously simulate the aforementioned different amine scrubbing processes. The Acid Gas-Chemical Solvents (AGCS) package was selected to simulate the process systems including sour gas (i.e., H_2S and CO_2) and aqueous amines (Dubois and Thomas, 2018). Notice that the AGCS package supports the reaction kinetics for chemical absorptions of sour gases by various aqueous amine solvents or their blends. The aqueous lean solvent around 30 wt% MEA was used in this study. To simulate the absorber and stripper, HYSYS column modules were utilized, and the Modified HYSYS Inside-Out algorithm with the adaptive damping factor was selected as their solvers in the simulation runs (Oh et al., 2016). For all simulated processes, the recycled streams for lean solvent circulation were considered by adopting the HYSYS "Recycle" module. The minimum approach temperature of LRHX was set as 10 °C. In the cases of LVR and LRVR processes, the flashed lean vapor was recompressed and recycled back to the bottom of the strippers. This extra recycle loop requires an additional "Recycle" module to facilitate converge in the corresponding simulation runs (Dubois and Thomas, 2018).

2.4. Data acquisition

Four datasets for the base and the three modified configurations were collected by high-fidelity simulation runs in Aspen HYSYS with the help of automation techniques based on Component Object Model (COM) protocol (Santos Bartolome and Van Gerven, 2022) supported by the Python win32com package. Sufficient steady-state samples should be collected through repeated simulation runs. To ensure statistical diversity of the collected data (Kajero et al., 2017), the specifications for simulation runs were pseudo-randomly sampled by the Latin hypercube sampling (LHS) technique within relatively wide design ranges given in Table 1. For collecting 1000 samples, the total data acquisition time (in

Table 1
The design ranges of amine scrubbing CO_2 capture processes.

Absorber		Stripper	
Operating pressure	1.0 bar	Operating pressure	1.5–3.0 bar
Flue gas temperature	40–80 °C	Condenser temperature	40–80 °C
Flue gas flowrate	30–100 kg/h	Lean CO_2 loading	0.05–0.45
Flue gas CO_2 concentration	0.05–0.15	Lean flash pressure*	0.3–2.7 bar
Lean solvent temperature	40–60 °C	Rich flash pressure*	0.3–0.9 bar
Lean solvent flowrate	30–500 kg/h	**Momentum and heat transfer operations**	
Lean MEA concentration	0.27–0.33	Min. approach temperature	10 °C
Lean CO_2 loading	0.05–0.45	Adiabatic efficiency*	75%

*Parameters that only used in MVR-based scrubbers
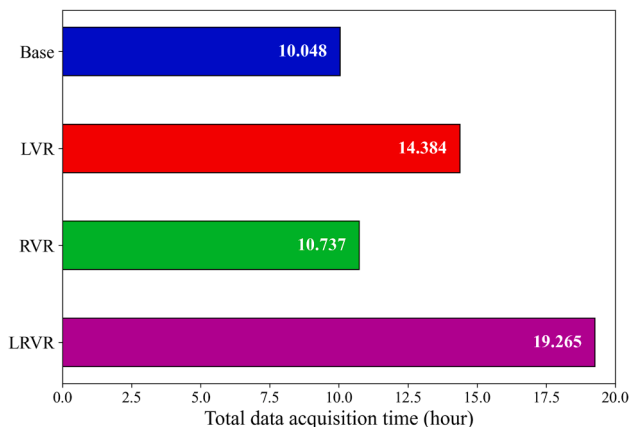
Fig. 3. The total data acquisition time (in hour) of each simulated configuration.

hour) of each simulation run is given in Fig. 3. In this study, a computer with Intel Core i7-7700 CPU 3.60 GHz was used for data acquisition. Since the complex rate-based model(s) and recycle loops need to be iteratively solved during each steady-state simulation run, the required data acquisition periods in all cases are quite long, which again confirms the necessity of building surrogate models. It can be clearly seen that the acquisition periods of the base and RVR configurations were significantly shorter when compared with those of the LVR and LRVR cases. This is due to the fact that each of the LVR and LRVR processes includes an additional recycle loop at the lean vapor recompression section (Dubois and Thomas, 2018). These observations show that adding the structural complexity of a process may greatly slow down the data acquisition speed. Therefore, enhancing reusability of ANN models via engineering insights may effectively reduce the computation effort for gathering data (Chuang et al., 2018; Goldstein et al., 2022; Lu et al., 2009).

3. Model migration methodology

Incremental learning (Fayek et al., 2020) (or continual/lifelong/progressive learning (Parisi et al., 2019; Yoon et al., 2018)) techniques have been applied to migrate the process knowledge embedded in the old model to the new one while at the same time avoid the catastrophic interferences (Parisi et al., 2019). Compared with the conventional transfer learning methods (Yosinski et al., 2014), incremental learning places more attentions on expanding and/or pruning the model architectures. As a type of connectionist models, it is highly flexible to customize inter-layer connections for block-modularizations and progressive expansion using deep neural networks (Terekhov et al., 2015). By increasing the number of hidden neurons, expandable neural network has shown its effectiveness to migrate the pre-trained models to new tasks with different input–output structures while preserve the prior knowledge gained from old tasks (Rusu et al., 2016; Yin et al., 2020; Yoon et al., 2018).

As mentioned previously, the data acquisition process is considered

to be the most time-consuming step for surrogate modeling (Nuchitprasittichai and Cremaschi, 2013). If the process configuration is modified due to the progress of on-going research, the existing model should be no longer suitable for characterizing the new systems. Furthermore, with the updated input–output relationships, the existing models cannot be directly migrated to describe the more upgraded process using the conventional transfer learning methods (Hsiao et al., 2021; Yosinski et al., 2014). Facing with this dilemma, one may be forced to train a new model from scratch again with the new datasets (Li et al., 2015; Sipöcz et al., 2011; Wu et al., 2010). On the other hand, by extracting the unique features of the expandable neural network architecture in the literature (Rusu et al., 2016; Yin et al., 2020), an innovative modeling approach has been developed in this study to address the aforementioned issues. In the following paragraphs, the mechanisms of the proposed model migration method are given in detail. As depicted in Fig. 4, the proposed flowchart for surrogate modeling of amine scrubbing systems may be divided into four steps, i.e., (1) process data acquisition, (2) base process modeling, (3) primary expansion and (4) secondary expansion(s). Both primary and secondary expansions are followed by new process modeling steps to train the new ANN models after expansions. The data acquisition part has been given in subsection 2.4. As a result, in the following subsections, the latter three steps will be introduced in detail.

3.1. Base process modeling

As mentioned before, a robust base model should be first trained with the conventional method. In this study, a fully-connected feedforward neural network (FNN) was adopted for this task. The generalized matrix–vector multiplication formulas in each layer of a FNN model can be expressed as follows.

$$\mathbf{h}_i^{(j)} = \phi\left(\mathbf{W}_i^{(j)}\mathbf{h}_{i-1}^{(j)} + \mathbf{b}_i^{(j)}\right) \tag{1}$$

where i and j are the indices of the hidden layers and the existing block (i.e., the base model) respectively; $\mathbf{h}_i^{(j)}$ denotes the hidden state vectors of layer i in block j; $\mathbf{W}_i^{(j)}$ and $\mathbf{b}_i^{(j)}$ represent the weight matrix and bias vector of layer i in block j respectively; ϕ denotes the activation operator, which is used to handle the scalar activations of each entity in an arbitrary vector. As illustrated in Fig. 4, after a robust base process modeling, one should implement the model migration procedure in two stages. The first is carried out in one step using the base model only, while in the second stage, multi-step modifications are incorporated. The details of these stages will be given in the successive subsections.

3.2. Primary expansion

As illustrated in Fig. 4(a), a new network *block* is laterally and layer-wisely connected to the existing one (i.e., base model) in the primary expansion stage. Catastrophic interference may occur and deteriorate the stored knowledge if the model parameters are updated with the new datasets (Parisi et al., 2019). To avoid this problem, the parameters of the existing block ($\mathbf{W}_i^{(j)}$ and $\mathbf{b}_i^{(j)}$) given in Equation (1) are kept frozen
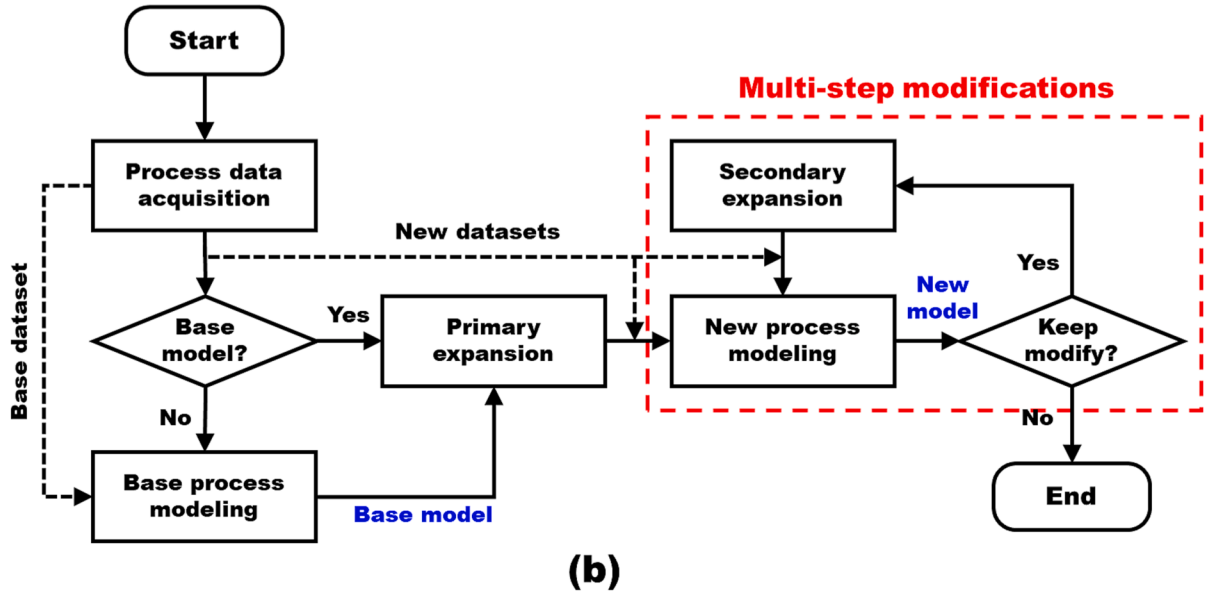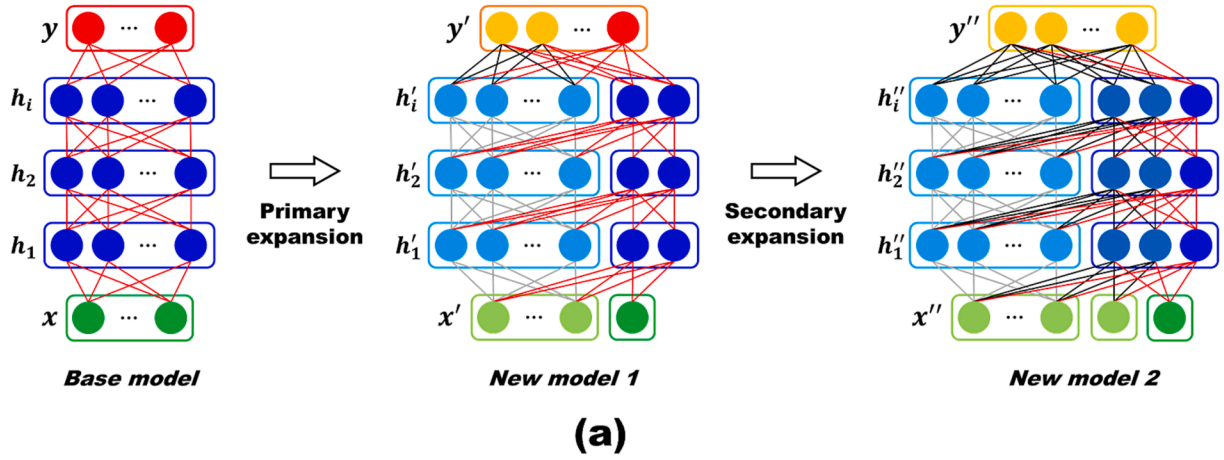
(a)

(b)

Fig. 4. The flowchart of multi-step expansions of base model.

during the expansion. Through this way, the prior knowledge embedded in the base model may be well preserved. On the other hand, in the primary expansion step, the parameters of the expanded block or lateral connections (i.e., inter-layer or inter-block connections colored in red as given in Fig. 4(a)) are end-to-end trained in a supervised manner with the new datasets. To be more specific, the matrix–vector multiplication in each layer of the existing block should still follow Eq. (1), while that of the expanded block can be expressed with Eq. (2).

$$\mathbf{h}_i^{(k)} = \phi\left(\mathbf{W}_i^{(k)}\mathbf{h}_{i-1}^{(k)} + \mathbf{W}_i^{(k;j)}\mathbf{h}_{i-1}^{(j)} + \mathbf{b}_i^{(k)}\right) \tag{2}$$

where k is the index of an expanded block, and $\mathbf{h}_i^{(k)}$ denotes the hidden state vectors of layer i in block k. Similar to those adopted in Eq. (1), $\mathbf{W}_i^{(k)}$ and $\mathbf{b}_i^{(k)}$ are the weight matrix and bias vector of layer i in block k respectively. Finally, notice that $\mathbf{W}_i^{(k;j)}$ is the newly initialized weight matrix of the lateral connection from layer $i-1$ in block j to layer i in block k, where the notation $k:j$ represents the lateral connection between block k and block j (Fayek et al., 2020; Rusu et al., 2016).

It is necessary to take a closer look at the weight matrices of the expanded block and lateral connection. As given in Eq. (3), $\mathbf{W}_i^{(k)}$ is a $m \times n'$ matrix, where m and n' are the dimensions of $\mathbf{h}_i^{(k)}$ and $\mathbf{h}_{i-1}^{(k)}$ respectively. On the other hand, as shown in Eq. (4), $\mathbf{W}_i^{(k;j)}$ is a $m \times n$ matrix, where n is the dimensions of $\mathbf{h}_{i-1}^{(j)}$.

$$\mathbf{W}_i^{(k)} = \begin{bmatrix} w_{1,1}^{(k)} & w_{1,2}^{(k)} & \cdots & w_{1,n'}^{(k)} \\ w_{2,1}^{(k)} & w_{2,2}^{(k)} & \cdots & w_{2,n'}^{(k)} \\ \vdots & \vdots & \ddots & \vdots \\ w_{m,1}^{(k)} & w_{m,2}^{(k)} & \cdots & w_{m,n'}^{(k)} \end{bmatrix}_i \tag{3}$$

$$\mathbf{W}_i^{(k;j)} = \begin{bmatrix} w_{1,1}^{(k;j)} & w_{1,2}^{(k;j)} & \cdots & w_{1,n}^{(k;j)} \\ w_{2,1}^{(k;j)} & w_{2,2}^{(k;j)} & \cdots & w_{2,n}^{(k;j)} \\ \vdots & \vdots & \ddots & \vdots \\ w_{m,1}^{(k;j)} & w_{m,2}^{(k;j)} & \cdots & w_{m,n}^{(k;j)} \end{bmatrix}_i \tag{4}$$

3.3. Secondary expansion

Since the model may be expanded several times together with the modifications of process configuration, the secondary expansions are implemented by enlarging the parametric capacity of the expanded block. As illustrated in Fig. 4, additional hidden neurons and their corresponding randomly initialized weights are added layer-by-layer to the expanded block, which has been pre-trained in the primary expansion stage. On the other hand, the parameters of the original base model (i.e., $\mathbf{W}_i^{(j)}$ and $\mathbf{b}_i^{(j)}$) are still kept intact during the expansion. Notice that a

proportional of parameters in the expanded block are already physically meaningful (i.e., $\mathbf{W}_i^{(k)}$ and $\mathbf{W}_i^{(k,j)}$). These values are expected to be better starting points for the training jobs. To facilitate better understanding, the secondary expansion is also expressed in a mathematical form and it is given below in Equation (5). It can be observed that the matrix–vector multiplication in layer i of the expanded block k can be formulated in a form which is similar to Equation (2).

$$\mathbf{h}_i^{'(k)} = \phi\left(\mathbf{W}_i^{'(k)}\mathbf{h}_{i-1}^{'(k)} + \mathbf{W}_i^{'(k,j)}\mathbf{h}_{i-1}^{'(j)} + \mathbf{b}_i^{'(k)}\right) \tag{5}$$

However, since the expanded block is enlarged again in the secondary expansion step with the larger numbers of input variables, output variables and hidden neurons, the dimensions of the corresponding weight matrix, bias vector and hidden state vector should also be modified. As given in Equation (6) and (7), the weight matrices of the expanded block and lateral connection are expanded accordingly. Sub-matrices $\mathbf{W}_i^{(k)}$ and $\mathbf{W}_i^{(k,j)}$ are identical to those in Equation (3) and (4) respectively, which are physically meaningful. If p nodes are added to layer i in block k, sub-matrices $\mathbf{V}_i^{(k)}$ and $\mathbf{V}_i^{(k,j)}$ (whose dimensions are $p \times n'$ and $p \times n$ respectively) are added to increase the number of rows in each weight matrices (see Equation (6) and (7)). On the other hand, if the layer $i-1$ in block k is appended with q additional nodes, sub-matrices $\mathbf{U}_i^{(k)}$ and $\mathbf{Z}_i^{(k)}$ (whose dimensions are $m \times q$ and $p \times q$ respectively) are added to increase the number of columns in the weight matrix of layer i in block k (shown in Equation (6)). Consequently, the dimensions of the weight matrices of the expanded block and lateral connection become $(m+p) \times (n'+q)$ and $(m+p) \times n$ respectively. If secondary expansion is carried out more than once, the abovementioned procedure should be repeated in each step.

$$\mathbf{W}_i^{'(k)} = \begin{bmatrix} \mathbf{W}_i^{(k)} & \mathbf{U}_i^{(k)} \\ \mathbf{V}_i^{(k)} & \mathbf{Z}_i^{(k)} \end{bmatrix} = \begin{bmatrix} w_{1,1}^{(k)} & w_{1,2}^{(k)} & \cdots & w_{1,n'}^{(k)} & u_{1,1}^{(k)} & u_{1,2}^{(k)} & \cdots & u_{1,q}^{(k)} \\ w_{2,1}^{(k)} & w_{2,2}^{(k)} & \cdots & w_{2,n'}^{(k)} & u_{2,1}^{(k)} & u_{2,2}^{(k)} & \cdots & u_{2,q}^{(k)} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ w_{m,1}^{(k)} & w_{m,2}^{(k)} & \cdots & w_{m,n'}^{(k)} & u_{m,1}^{(k)} & u_{m,2}^{(k)} & \cdots & u_{m,q}^{(k)} \\ v_{1,1}^{(k)} & v_{1,2}^{(k)} & \cdots & v_{1,n'}^{(k)} & z_{1,1}^{(k)} & z_{1,2}^{(k)} & \cdots & z_{1,q}^{(k)} \\ v_{2,1}^{(k)} & v_{2,2}^{(k)} & \cdots & v_{2,n'}^{(k)} & z_{2,1}^{(k)} & z_{2,2}^{(k)} & \cdots & z_{2,q}^{(k)} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ v_{p,1}^{(k)} & v_{p,2}^{(k)} & \cdots & v_{p,n'}^{(k)} & z_{p,1}^{(k)} & z_{p,2}^{(k)} & \cdots & z_{p,q}^{(k)} \end{bmatrix}_i \tag{6}$$

$$\mathbf{W}_i^{'(k,j)} = \begin{bmatrix} \mathbf{W}_i^{(k,j)} \\ \mathbf{V}_i^{(k,j)} \end{bmatrix} = \begin{bmatrix} w_{1,1}^{(k,j)} & w_{1,2}^{(k,j)} & \cdots & w_{1,n}^{(k,j)} \\ w_{2,1}^{(k,j)} & w_{2,2}^{(k,j)} & \cdots & w_{2,n}^{(k,j)} \\ \vdots & \vdots & \ddots & \vdots \\ w_{m,1}^{(k,j)} & w_{m,2}^{(k,j)} & \cdots & w_{m,n}^{(k,j)} \\ v_{1,1}^{(k,j)} & v_{1,2}^{(k,j)} & \cdots & v_{1,n}^{(k,j)} \\ v_{2,1}^{(k,j)} & v_{2,2}^{(k,j)} & \cdots & v_{2,n}^{(k,j)} \\ \vdots & \vdots & \ddots & \vdots \\ v_{p,1}^{(k,j)} & v_{p,2}^{(k,j)} & \cdots & v_{p,n}^{(k,j)} \end{bmatrix}_i \tag{7}$$

4. Numerical experiments

4.1. Case studies

For the design of amine scrubbing systems, it is well-known that a trade-off is present between the sour gas capture rate and energy consumption rate (Oh et al., 2016). Therefore, the multi-objective optimization methods may be applied to achieve an appropriate balance (Tikadar et al., 2020). Maximizing the sour gas capture rate is equivalent to minimizing the emission rate. Therefore, the CO_2 emission rate and reboiler duty were selected as the output variables of the surrogate models of all configurations, while the MVR duty was introduced as an extra output for all MVR models. The CO_2 emission rate was determined by the CO_2 content in the treated gas and its vent rate, while the MVR duty was the sum of power consumed by the vapor compressors. On the other hand, to mimic the input–output relationships of the rigorous models, nine variables, i.e., flue gas temperature, flue gas flowrate, flue gas CO_2 concentration, lean temperature, lean flowrate (i.e., solvent circulation rate), lean MEA concentration, lean loading, condenser temperature and stripper pressure, were selected as the inputs of all surrogate models (see Table 1). According to the modified process configuration, two additional variables, i.e., lean and rich flash pressures, were introduced as the extra inputs of MVR models.

To demonstrate the applicability and flexibility of the presented method, five scenarios were studied in this paper. As depicted in Fig. 5, all scenarios started from a robust surrogate model of the base configuration (see Fig. 1). As mentioned before, the structural modification scenarios may be divided into two stages. The primary expansion starts from the base model, while the secondary changes are then incorporated in the intermediate structures obtained in the primary stage. Three primary expansion scenarios can be evolved from the base model, i.e. (1) from base model to LVR model (Base-LVR), (2) from base model to RVR model (Base-RVR) and (3) from base model to LRVR model (Base-LRVR). Furthermore, two secondary expansion scenarios, i.e., (1) from Base-LVR model to LRVR model (Base-LVR-LRVR) and (2) from Base-RVR model to LRVR model (Base-RVR-LRVR), can also be identified from Fig. 5. In each expansion step, there may be additional input(s) and/or output(s), and they should be appended to the input and/or output layers of the existing models. In this study, several different model expansion routes have been analyzed and compared with the conventional modeling approach to demonstrate the effectiveness of the proposed model-building method.

4.2. Hyper-parameters

In this paper, multi-layer feedforward neural networks (FNNs) were selected for constructing the surrogate models of amine scrubbers. The open-source Keras software was utilized to build and train these FNN models. In most applications, FNNs with one or two hidden layers are sufficient to approximate an arbitrary function (Stathakis, 2009; Trenn, 2008). By prior cross validations of the model performances using the baseline method (training ANN model from scratch) with the collected datasets, it can be observed from Fig. 6 that the two-layer FNNs outperform their one-layer counterparts. The optimal numbers of hidden neurons of the base, LVR, RVR and LRVR configurations were also determined to be 10, 15, 15 and 20 respectively. As listed in Table 2, these hyper-parameters were adopted in the case studies.

To facilitate fair comparisons, the total number of hidden neurons in each layer of every expanded model was set to be identical to that in the corresponding layer of its baseline counterpart. For example, to obtain the LVR model through primary expansion from the base model (which is a two-layer FNN with 10 hidden neurons in each layer), 5 additional neurons should be placed into the expanded block. Thus, a total of 15 hidden neurons should be located in each layer (10 and 5 in the existing and expanded blocks respectively), which is consistent with the LVR model built by the baseline method. All abovementioned details are also
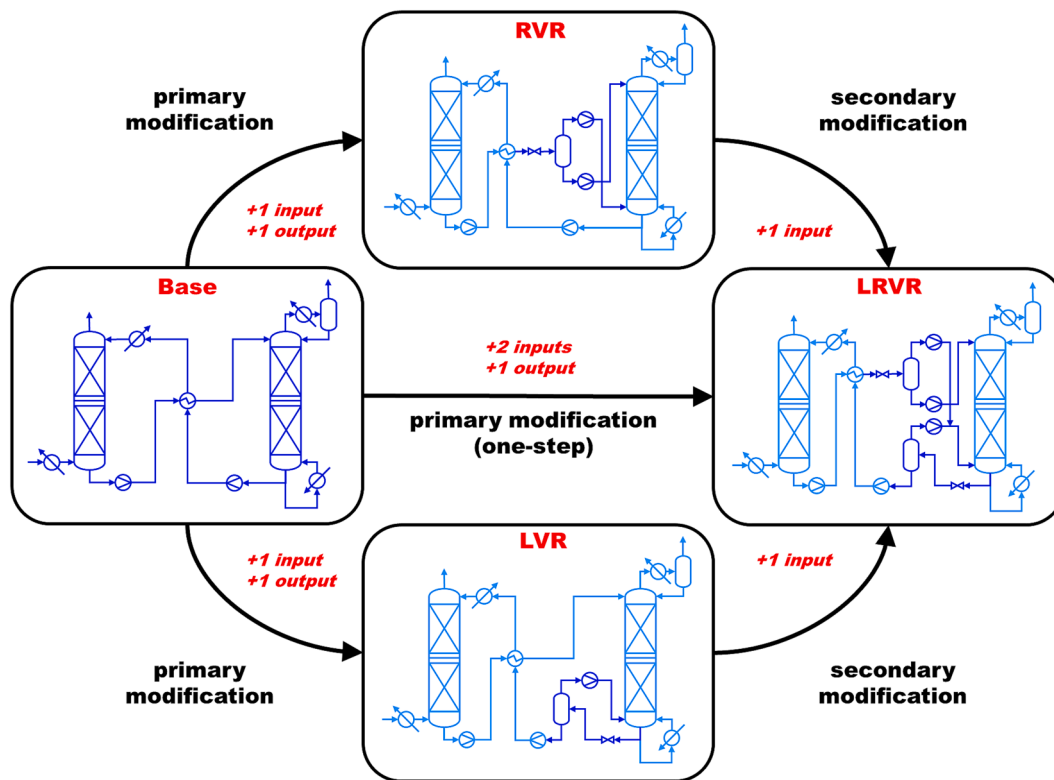
Fig. 5. One and two-step process modification scenarios.

given in Table 2. Since the existing blocks of the proposed models were frozen during the model expansion steps, notice that even with same number of hidden neurons, the number of trainable parameters of the proposed method are much smaller than those of its baseline counterpart.

The scaled exponential linear unit (SELU) was selected as the activation function for each hidden layer (Klambauer et al., 2017), and the linear function was used for the output layers. Since the SELU function was chosen for the activations of hidden neurons, the Lecun normal initializer was selected for random parameter initializations (LeCun et al., 2002), and the raw datasets were preprocessed through standard-deviation normalization (i.e., $z = (x - \mu)/\sigma$). The mean squared error (MSE) was adopted as the loss function for all output variables. The Adam optimizer was selected for gradient descent (Kingma and Ba, 2014), and the learning rate was set as 0.0005. The reason for using this slightly smaller learning rate than the default value (i.e., 0.001) is to avoid the potential problem of catastrophic interference caused by aggressive parameter updating (Kirkpatrick et al., 2017).

4.3. Model performance evaluations

For all modeling tasks, 1000 samples were acquired to be used as the testing sets. The remaining 100–1000 samples (according to the sampling size) were divided into two sets which respectively consists 70% and 30% of the data, and they were used as the training and validation sets respectively. In order to evaluate the model performances, the root mean squared error (RMSE) was adopted as the metric for quantifying the accuracy of the predicted model outputs. In addition, the R-squared (R^2) score was also used as an auxiliary metric. For the sake of completeness, definitions of these two metrics are given below:

$$RMSE = \sqrt{\frac{1}{N} \sum_{n=1}^{N} (\widehat{y}_n - y_n)^2} \tag{8}$$

$$R^2 = 1 - \frac{\sum_n (\widehat{y}_n - y_n)^2}{\sum_n (\overline{y} - y_n)^2} \tag{9}$$

5. Results and discussions

5.1. Pre-training of base model

The surrogate model of base configuration was pre-trained to serve as the base model for executing successive neural network architecture expansion. In order to provide correct and transferrable knowledge for progressive learning, it is important to first produce a highly accurate base model. As mentioned earlier, the base model was constructed with 2 hidden layers, and there were 10 hidden neurons in each layer. The base model was designed to predict the CO_2 emission rate and reboiler duty by nine selected input variables, e.g., lean loading, given in subsection 4.1. In this case study, 2000 samples of typical amine scrubbing process (Fig. 1) were collected from simulation runs for pre-training the base model, which was then tested with 1000 extra samples. As shown in Fig. 7, the testing parity plots imply that the base model was able to accurately inference both CO_2 emission rate and reboiler duty with given operating conditions. The R^2 scores of each output variable were both 0.997, where the corresponding RMSEs were 0.111 kg/h and 0.546 kW respectively. This well pre-trained base model was then used to serve as the starting model for the successive expansion steps.
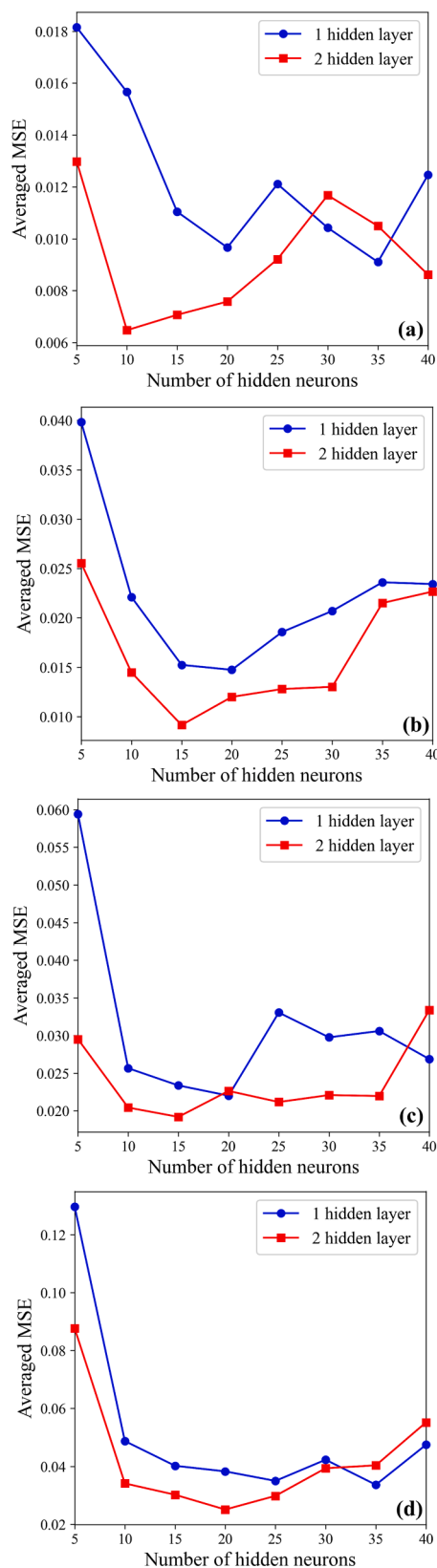
Fig. 6. Prior cross validation results for number of hidden neurons of various base models: (a) base process, (b) LVR process, (c) RVR process and (d) LRVR process.

Table 2
The hyper-parameters for different surrogate modeling cases.

Hyper-parameters	LVR	RVR	LRVR		
No. of input variables	10	10	11		
No. of output variables	3	3	3		
The baseline method					
No. of hidden layers	2	2	2		
No. of hidden neurons	15	15	20		
No. of parameters	453	453	723		
The proposed method					
Source model	Base	Base	Base	LVR	RVR
No. of hidden layers	2	2	2	2	2
No. of hidden neurons	10 + 5	10 + 5	10 + 10	10+(5 + 5)	10+(5 + 5)
No. of trainable parameters*	183	183	393	393	393

*Number of trainable parameters during model expansion.
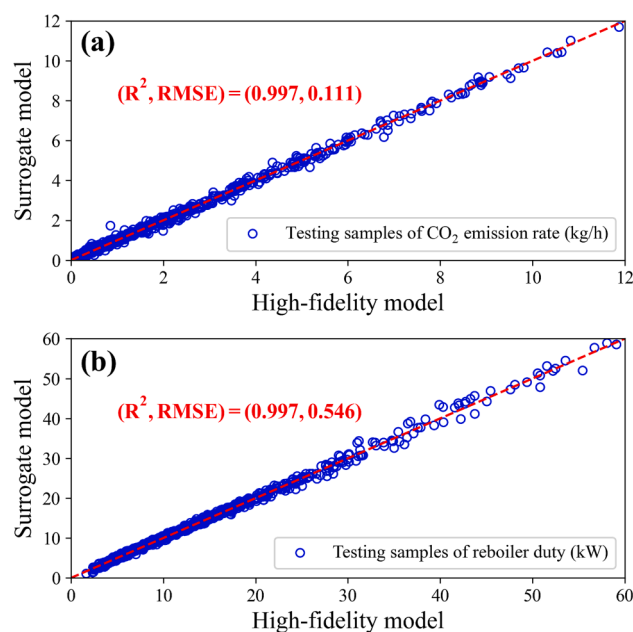
Fig. 7. Testing results of (a) CO_2 emission rate, (b) reboiler duty of the base model.

5.2. Primary expansions

5.2.1. Modeling of LVR process

The LVR model was expanded from the base model described in subsection 5.1. A new network block with 2 hidden layers and 5 hidden neurons at each layer was laterally connected with the existing one. New input and output variables, i.e., lean flash pressure and lean vapor recompression duty, were respectively appended to the input and output layers (see Fig. 4(a)). The new model were trained with 100–1000 samples (according to the sampling size), and tested with 1000 previously unused samples. The testing results of LVR models are given in Fig. 8. It can be seen that the LVR model is able to predict CO_2 emission rate and reboiler duty at highly accurate levels regardless of fact that relatively few modeling samples were used. Thus, it may be concluded that the knowledge stored in the base model was correctly integrated into the LVR model. More specifically, the interior features of the base block can be transferred for the new learning task. Thus, the new model accuracy can be enhanced with small dataset. It took only 450–500 samples for the proposed method to generate surrogate models with similar or even better performances than those of the baseline model trained with 1000 samples. The RMSEs of the proposed model trained with 500 samples were 0.178 kg/h, 1.109 kW and 0.046 kW for the CO_2
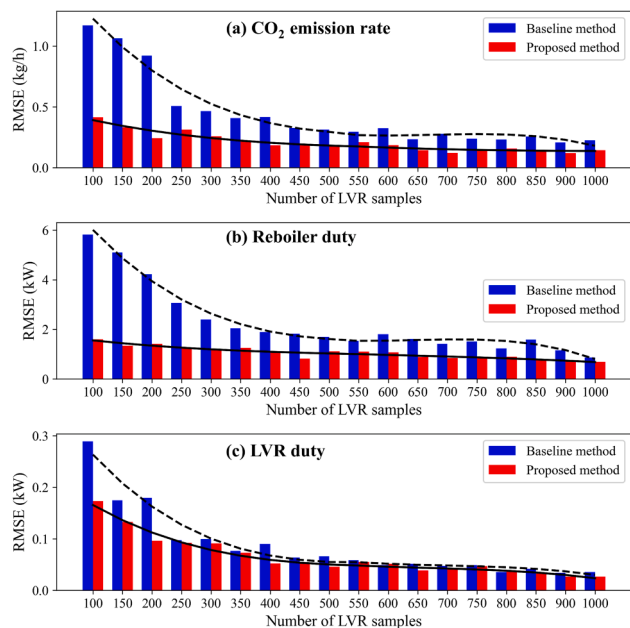
Fig. 8. Testing results of surrogate models of LVR configuration.

emission rate, reboiler duty and LVR duty respectively, which were similar to those of the baseline model trained with 1000 samples (i.e., 0.181 kg/h, 1.008 kW and 0.038 kW). The reduction of 500 modeling samples implies that around 50–55% of data acquisition time was saved.

5.2.2. Modeling of RVR process

The testing results of primary expansions from the base model to RVR models were given in Fig. 9. Similar to the network structures of LVR models, there was a two-layer neural network block with 5 hidden neurons at each layer laterally connected to the existing one. These models were also trained and tested with 100–1000 samples (according to the sampling size) and 1000 samples respectively. From the prediction results, it can be observed from Fig. 9 that the proposed modeling
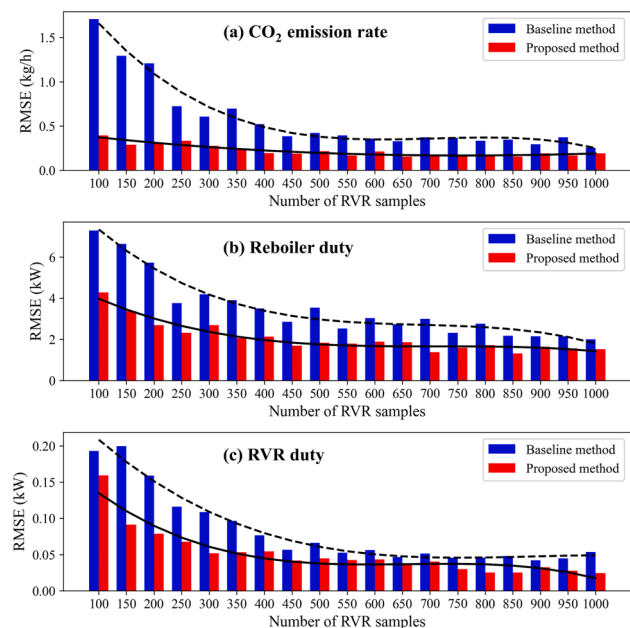
method is superior to its conventional counterpart. Also, the downward behaviors of RMSEs associated with the proposed models (solid lines) all dropped at much faster speeds than those of the baseline counterparts (dashed lines). These smooth trend lines were obtained by applying the Savitzky–Golay filter to the experiment results (Savitzky and Golay, 1964). With only 350 modeling samples, the proposed model was able to achieve better prediction accuracy than the baseline model trained with 1000 samples. The RMSEs of CO_2 emission rate, reboiler duty and RVR duty of the proposed model were found to be 0.242 kg/h, 2.059 kW and 0.053 kW respectively, which is much better performances than those of the baseline model also trained with 350 samples (i.e., 0.696 kg/h, 3.906 kW and 0.097 kW). Besides, the prediction results were also similar to those of its baseline counterpart trained with 1000 samples (i. e., 0.254 kg/h, 2.000 kW and 0.054 kW). The above observations imply that around 65.8% of data acquisition time may be saved for building the RVR model with similar prediction accuracies. This finding means that the data acquisition effort can be drastically lowered and also the modeling efficiency greatly enhanced. On the other hand, when similar modeling samples are available, 65.2%, 47.3% and 45.4% of predictive errors of the above-mentioned different outputs may be effectively reduced.

5.2.3. Modeling of LRVR process

There are two ways to produce LRVR model by properly using the proposed method, i.e., primary and secondary expansions (see Fig. 5). In this subsection, the case of direct primary expansion from base model is presented, while the cases concerning the secondary expansions will be discussed later in the next subsection. Different from the LVR and RVR configurations, the base model was laterally appended by a two-layer network block with 10 hidden neurons at each layer, but also trained and tested with 100–1000 samples (according to the sampling size) and 1000 samples respectively. As given in Fig. 10, the expandable neural networks worked well in executing transfer learning of either CO_2 emission rate or reboiler duty. The trend lines of the proposed method were much lower than their baseline counterparts. However, it can also be said that the improvement in prediction accuracy of the MVR duty may be quite limited, where the two trend lines in Fig. 10 are close to each other. Such results refute not only the claim of accuracy

Fig. 9. Testing results of surrogate models of RVR configuration.
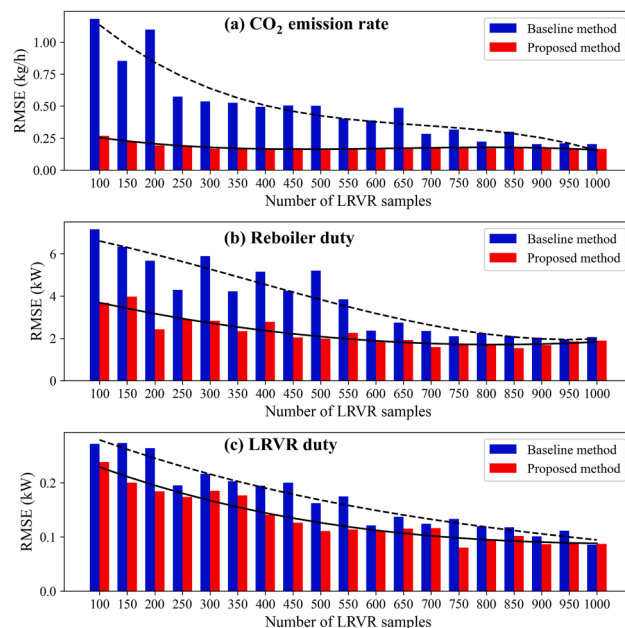
Fig. 10. Testing results of LRVR models obtained from primary expansion.

enhancement but also the hope of sampling size reduction. The proposed method required 750 samples to create a surrogate model with similar performance when compared with that of another model trained with 1000 samples by the baseline method. Specifically, the above observations sum up to the quantitative statistics that that 250 samples were reduced or around 23% of the data acquisition time was saved. The extent of modeling efficiency enhancement is much smaller than those achieved by LVR and RVR models.

5.3. Secondary expansions

As mentioned in the previous subsection, aside from primary expansion of the base model, the secondary expansion of LVR or RVR model is another option to construct LRVR model by using the proposed method. The model testing results of secondary expansions of LVR and RVR models to LRVR models are given in Fig. 11 and Fig. 12 respectively. In each case, five new nodes were added to every hidden layer of the expanded blocks, and these models were also trained and tested by 100–1000 samples (according to the sampling size) and 1000 samples respectively. For the Base-LVR-LRVR scenario, the LVR models trained respectively with 500 and 1000 samples were chosen to serve as the source models for expansion. On the other hand, the RVR models trained respectively with 350 and 1000 samples were used in the Base-RVR-LRVR scenario. The testing results of these scenarios were compared with those obtained by the baseline method.

From the testing results given in Fig. 11 and Fig. 12, when the modeling datasets were relatively small, it can be seen that the source models had no significant impacts on the model performances. However, as the number of samples increased, slightly differences can be detected. In particular, the source models trained with more samples resulted in slightly more accurate secondary models. If compared with the ones obtained by direct primary expansions, the LRVR models derived from the secondary expansions had better performances when similar numbers of modeling samples were used. This is reasonable since the initial parameters of the expanded block were at different levels for the cases of direct primary expansion and indirect secondary expansions. Notice that the expanded parameters of the former model were all randomly initialized, while those in the latter were initialized partially with the parameters gained from the previous expansion steps.
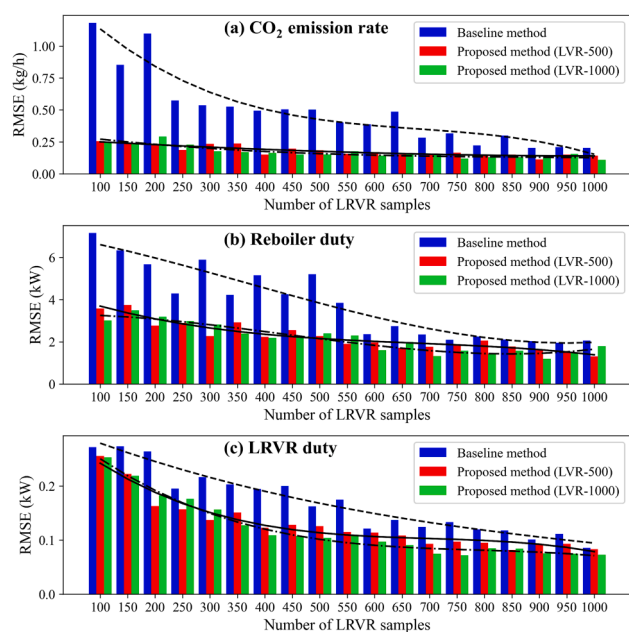
Fig. 11. Testing results of LRVR models from secondary expansions of LVR models.
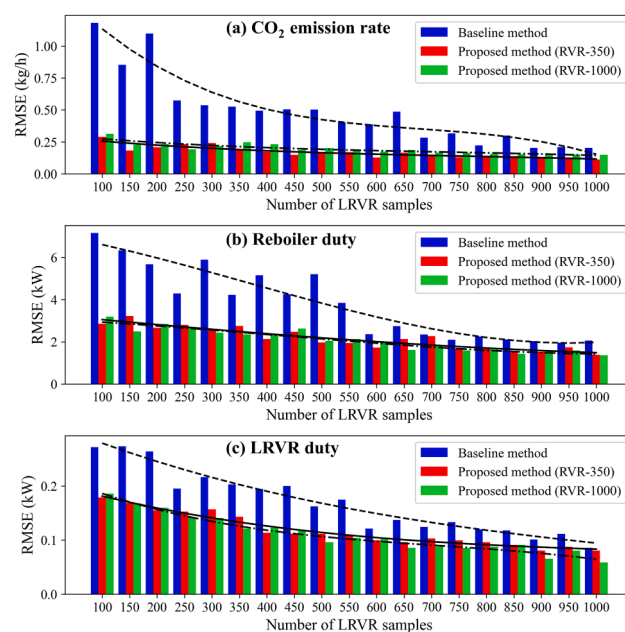
Fig. 12. Testing results of LRVR models from secondary expansions of RVR models.

5.4. Further discussions

5.4.1. Reduction of total data acquisition time

The modeling efficiency improvement by using the proposed method is further discussed below. A comprehensive comparison between the baseline and proposed methods is given in Table 3. It can be clearly observed that the performances of the paired baseline and proposed models were highly similar to each other. However, with prior knowledge extracted from the previous modeling tasks, the proposed models required fewer samples for training. This outcome suggests that the corresponding data acquisition time may be reduced. In the LVR and RVR models, 46.5% and 65.8% of data acquisition time may be saved due to the fact that the required sample sizes were reduced from 1000 to 500 and 350 respectively. On the other hand, for the LRVR models, by following alternative secondary expansion routes, 23–38.4% of the data acquisition time was reduced. It can be concluded that the models obtained with secondary expansion outperformed the one by direct primary expansion. Furthermore, these models may be used simultaneously, e.g., for the comparative studies of different process configurations. It is important to investigate the overall modeling efficiency in the lifecycle of process research and development. Therefore, the total data acquisition time was further summed and analyzed to show the effectiveness of the proposed method. From the results, it can be said that the best routes for obtaining the LVR, RVR and LRVR models were expansions via the sequences of Base-LVR, Base-RVR and Base-RVR-LRVR, and the corresponding data acquisition times were 7.699, 3.674 and 11.868 h respectively (see Table 3). In other words, these time savings represent 53.5%, 34.2% and 61.6% of those used to generate the baseline models. Finally, notice that the total acquisition time was 23.241 h, which is 52.36% of that needed to apply the baseline method.

5.4.2. Enhancement of model performance

Aside from reduction in data acquisition time, the enhanced model performances are presented in Table 4. With identical sample size (i.e., 1000 samples), the prediction accuracies of improved models can be compared with those of the baseline models. For all outputs of the LVR and RVR models, approximately 20–55% of prediction errors may be eliminated. On the other hand, 5–45% of prediction errors were eliminated with the improved LRVR model. Therefore, it may be concluded

Table 3
Data acquisition times for models with similar prediction performances.

Surrogate models		LVR	RVR	LRVR		
Parametric initialization	Baseline	Random	Random	Random		
	Proposed	Base	Base	Base	LVR-500	RVR-350
Number of modeling samples	Baseline	1000	1000	1000		
	Proposed	500	350	750	700	600
RMSEs of CO_2 emission rate (kg/h)	Baseline	0.181	0.254	0.204		
	Proposed	0.178	0.242	0.172	0.153	0.129
RMSEs of reboiler duty (kW)	Baseline	1.008	2.000	2.056		
	Proposed	1.109	2.059	1.713	1.772	1.734
RMSEs of MVR duty (kW)	Baseline	0.038	0.054	0.085		
	Proposed	0.046	0.053	0.080	0.093	0.099
Data acquisition time of each model (hour)	Baseline	14.384	10.737	19.264		
	Proposed	7.699	3.674	14.832	13.992	11.868
		−46.5%	−65.8%	−23.0%	−27.4%	−38.4%
Minimum total data acquisition time (hour)	Baseline	14.384 + 10.737 + 19.264 = 44.385 (100%)				
	Proposed	7.699 + 3.674 + 11.868 = 23.241 (52.36%)				

Table 4
Model performance enhancement with same amount of modeling samples.

Surrogate models		LVR	RVR	LRVR		
Parametric initialization	Baseline	Random	Random	Random		
	Proposed	Base	Base	Base	LVR-1000	RVR-1000
Number of modeling samples	Baseline	1000	1000	1000		
	Proposed	1000	1000	1000	1000	1000
RMSEs of CO_2 emission rate (kg/h)	Baseline	0.181	0.254	0.204		
	Proposed	0.145	0.192	0.166	0.109	0.126
		−19.9%	−24.4%	−18.6%	−46.6%	−38.2%
RMSEs of reboiler duty (kW)	Baseline	1.008	2.000	2.056		
	Proposed	0.694	1.528	1.896	1.788	1.357
		−31.2%	−23.6%	−7.8%	−13.0%	−34.0%
RMSEs of MVR duty (kW)	Baseline	0.038	0.054	0.085		
	Proposed	0.027	0.024	0.087	0.073	0.077
		−28.9%	−55.6%	+2.4%	−14.1%	−9.4%

that the modeling approach of gradual and progressive expansions of the surrogate ANNs should be preferred over the one-step expansion strategy since the models obtained in former case is superior to those created by the latter strategy in terms of both data acquisition time and prediction accuracy.

5.4.3. Sensitivity analysis

For realistic engineering applications, it is essential to make sure that the correct input–output relationships are captured by the surrogate models. Aside from the prediction accuracies, it is also important to examine the physical consistencies between process mechanisms and model behaviors (Hsiao et al., 2021). One of the most commonly used interpretation approaches for ANNs is the sensitivity analysis. In this study, the perturbation-based method was adopted for sensitivity analysis to compare the mechanistic behaviors of the high-fidelity model and those of the surrogate models based on different modeling strategies (Wu et al., 2010). The core concept of perturbation-based sensitivity analysis is similar to process gain, where a disturbance is applied to a selected input variable (i.e., manipulated variable), and the corresponding variations of output variables (i.e., controlled variables) are then analyzed (McIntosh and Mahalec, 1991). The sensitivity analyses of surrogate models have been done by recursively implementing local perturbations on different input variables. On the other hand, the first-order Taylor approximation was adopted to estimate the steady-state gains of the LRVR process, which was accomplished by multi-variate linear regressions of the transformed dataset consisted of Δx_i (where x_i represent the input variables, e.g., lean loading and stripper pressure) and Δy_k (where y_k denote the output variables, i.e., CO_2 emission rate, reboiler duty and MVR duty), where the proportional coefficients ($K_{i,k}$) of the fitted linear models in Equation (10), were considered as the

proxy steady-state gains of the high-fidelity models (McIntosh and Mahalec, 1991; Montavon et al., 2018).

$$\Delta y_j = \sum_{i=1}^{N} K_{ij} \Delta x_i \forall j = 1, 2, 3 \tag{10}$$

As depicted in Fig. 13, the results of sensitivity analyses of LRVR models from various sources were visualized as the heat maps. The values given in the hear maps are the averaged process gains (i.e., K_{ij}) of any paired x_i and y_j. It can be clearly seen that the heat maps were highly similar to the one of the first-order Taylor expansion, which implies that the surrogate models were able to correctly capture the critical sensitivity relationships as those of the high-fidelity model. For example, all models showed that there were six inputs (i.e., lean loading, lean MEA concentration, flue gas CO_2 concentration, stripper pressure, lean flash pressure and rich flash pressure) causing significant impact on the response in reboiler duty. The values of reboiler duty increases as the lean loading or stripper pressure decreases, and also increases as the lean MEA concentration or the flue gas CO_2 concentration increases. Furthermore, for the LRVR process, it is also crucial to correctly capture the relationships between the energy duties with the lean and rich flash pressures. It can be observed from the heat maps that all models showed the cause-and-effect connections that the increased flash pressures result in increased reboiler duty and decreased LRVR duty, which were also consistent with the background knowledge of the LRVR configuration.

Finally, it should be noted that the values shown on the heat maps were the averaged values resulting from each sensitivity analysis (Wu et al., 2010). Although there were minor differences among these analytic results, the key information extracted from these analyses is how a specific output variable may respond due to a change in certain input

Fig. 13. Results of sensitivity analyses of LRVR models from various source models.
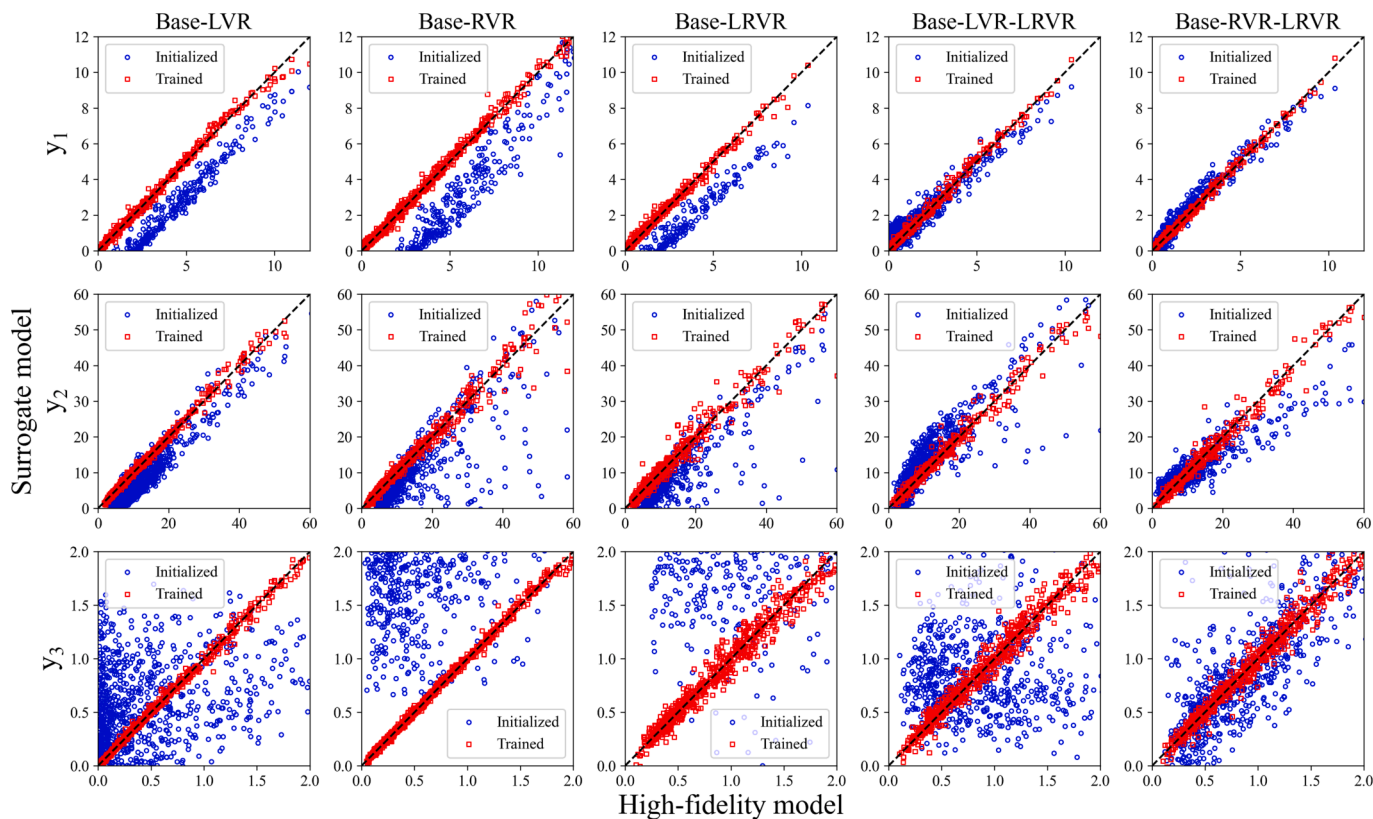

Fig. 14. Model testing results before and after training using the proposed method.

variable (Wu et al., 2010). Since the gain scales and gain signs of key input variables of various proposed surrogate models were consistent with the results of either first-order Taylor approximations or the baseline model, these data-driven surrogate models are physically interpretable and able to provide correct process mechanisms.

5.4.4. Effects of source models

To pinpoint the root cause of the effectiveness of the proposed method, the testing results of surrogate models originated from several different source models were depicted in Fig. 14. The blue round and red squared data points are the model prediction results before and after training respectively. Since the parameters of the proposed models were not completely randomly initialized, the predictions of the already known output variables before training were initially already at relatively high accurate levels but with only minor deviations from the true values. After the models were properly trained, the input–output relationships among the old outputs and new inputs may be captured by the expanded block, and the aforementioned predictive errors were then eliminated. On the other hand, in the primary expansion cases, since the additional output variable, i.e., MVR duty, was not learned by the base model, the parameters related to this new output were totally randomly initialized, which resulted in poor prediction performances. Indeed, it can be observed from Fig. 14 that for the primary expansion cases (column 1–3), the predicted values before training are far away from the dashed diagonal lines in the parity plots of MVR duties.

6. Conclusions and future perspectives

Accurate models of amine scrubbing configurations are essential for evaluating their techno-economic feasibility. In fact, most existing works were carried out mainly on the basis of rigorous simulation results. However, proper incorporation of high-fidelity simulators with numerical optimization frameworks are rarely found nowadays. This important task is usually hindered by the overwhelming computation efforts needed in the simulation studies, while any increase in process complexity inevitably makes this problem even more forbidding. On the other hand, although efficient construction of the data-driven models have already been attempted in the past to relieve the above practical burden, the data acquisition work is still very time-consuming. In this paper, a novel data-driven modeling method based on expandable neural networks has been proposed to construct the surrogate models for further reducing the aforementioned difficulties. By using this novel modeling procedure, the existing ANN models can be reused for the subsequent model expansion tasks. With the expandable neural networks, it becomes possible to incorporate new input/output features and hidden parameters according to the revamped system configurations. From the numerical experiment results obtained in case studies concerning four amine scrubbing configurations, over 47% of total data acquisition time can be saved, and, also, the model performances can be greatly enhanced. Furthermore, via the corresponding sensitivity analyses, it can be shown that the proposed surrogate models were able to capture existing process knowledge embedded in the raw data. Although the proposed modeling method was preliminarily shown to be effective, further extensions to more advanced CO_2 capture schemes, e.g., the complex stream splitting arrangements (Cho et al., 2015; Oh et al., 2016), the hybrid physical–chemical absorption process (Zhang et al., 2020) and even other chemical processes, etc., are still worthy of further investigations.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

References

Aghel, B., Janati, S., Wongwises, S., Shadloo, M.S., 2022. Review on CO_2 capture by blended amine solutions. Int. J. Greenhouse Gas Control 119, 103715.

Ahn, H., Luberti, M., Liu, Z., Brandani, S., 2013. Process configuration studies of the amine capture process for coal-fired power plants. Int. J. Greenhouse Gas Control 16, 29–40.

Alarie, S., Audet, C., Gheribi, A.E., Kokkolaras, M., Le Digabel, S., 2021. Two decades of blackbox optimization applications. EURO J. Comput. Optimiz. 9, 100011.

Álvarez, A., Bansode, A., Urakawa, A., Bavykina, A.V., Wezendonk, T.A., Makkee, M., Gascon, J., Kapteijn, F., 2017. Challenges in the greener production of formates/formic acid, methanol, and DME by heterogeneously catalyzed CO_2 hydrogenation processes. Chem. Rev. 117 (14), 9804–9838.

Borhani, T.N., Wang, M., 2019. Role of solvents in CO_2 capture processes: The review of selection and design methods. Renew. Sustain. Energy Rev. 114, 109299.

Caballero, J.A., Grossmann, I.E., 2008. An algorithm for the use of surrogate models in modular flowsheet optimization. AIChE J. 54 (10), 2633–2650.

Cho, H., Binns, M., Min, K.-J., Kim, J.-K., 2015. Automated process design of acid gas removal units in natural gas processing. Comput. Chem. Eng. 83, 97–109.

Chuang, Y.-C., Chen, T., Yao, Y., Wong, D.S.H., 2018. Transfer learning for efficient meta-modeling of process simulations. Chem. Eng. Res. Des. 138, 546–553.

Chung, W., Lee, J.H., 2020. Input–output surrogate models for efficient economic evaluation of amine scrubbing CO_2 capture processes. Ind. Eng. Chem. Res. 59 (42), 18951–18964.

Cousins, A., Wardhaugh, L.T., Feron, P.H.M., 2011. A survey of process flow sheet modifications for energy efficient CO_2 capture from flue gases using chemical absorption. Int. J. Greenhouse Gas Control 5 (4), 605–619.

Dubois, L., Thomas, D., 2018. Comparison of various configurations of the absorption-regeneration process using different solvents for the post-combustion CO_2 capture applied to cement plant flue gases. Int. J. Greenhouse Gas Control 69, 20–35.

Fayek, H.M., Cavedon, L., Wu, H.R., 2020. Progressive learning: A deep learning framework for continual learning. Neural Netw. 128, 345–357.

Goldstein, D., Heyer, M., Jakobs, D., Schultz, E.S., Biegler, L.T., 2022. Multilevel surrogate modeling of an amine scrubbing process for CO_2 capture. AIChE J. 68.

Hasan, M.M.F., Boukouvala, F., First, E.L., Floudas, C.A., 2014. Nationwide, regional, and statewide CO_2 capture, utilization, and sequestration supply chain network optimization. Ind. Eng. Chem. Res. 53 (18), 7489–7506.

Henao, C.A., Maravelias, C.T., 2011. Surrogate-based superstructure optimization framework. AIChE J. 57 (5), 1216–1232.

Hsiao, Y.-D., Kang, J.-L., Wong, D.-H., 2021. Development of robust and physically interpretable soft sensor for industrial distillation column using transfer learning with small datasets. Processes 9 (4), 667.

Kajero, O.T., Chen, T., Yao, Y., Chuang, Y.-C., Wong, D.S.H., 2017. Meta-modelling in chemical process system engineering. J. Taiwan Inst. Chem. Eng. 73, 135–145.

Khalifa, O., Alkhatib, I.I.I., Bahamon, D., Alhajaj, A., Abu-Zahra, M.R.M., Vega, L.F., 2022. Modifying absorption process configurations to improve their performance for post-combustion CO_2 capture – What have we learned and what is still missing? Chem. Eng. J. 430, 133096.

Kingma, D.P., Ba, J., 2014. Adam: A method for stochastic optimization. arXiv preprint.

Kirkpatrick, J., Pascanu, R., Rabinowitz, N., Veness, J., Desjardins, G., Rusu, A.A., Milan, K., Quan, J., Ramalho, T., Grabska-Barwinska, A., Hassabis, D., Clopath, C., Kumaran, D., Hadsell, R., 2017. Overcoming catastrophic forgetting in neural networks. PNAS 114 (13), 3521–3526.

Klambauer, G., Unterthiner, T., Mayr, A., Hochreiter, S., 2017. Self-normalizing neural networks. Advances in Neural Information Processing Systems 30.

Korre, A., Nie, Z., Durucan, S., 2010. Life cycle modelling of fossil fuel power generation with post-combustion CO2 capture. Int. J. Greenhouse Gas Control 4 (2), 289–300.

Le Moullec, Y., Neveux, T., Al Azki, A., Chikukwa, A., Hoff, K.A., 2014. Process modifications for solvent-based post-combustion CO_2 capture. Int. J. Greenhouse Gas Control 31, 96–112.

LeCun, Y., Bottou, L., Orr, G.B., Müller, K.R., 2002. Efficient backprop, Neural Networks: Tricks of The Trade. Springer, Berlin Heidelberg, Berlin, Heidelberg, pp. 9–50.

Li, T., Yang, C., Tantikhajorngosol, P., Sema, T., Liang, Z., Tontiwachwuthikul, P., Liu, H., 2022. Comparative desorption energy consumption of post-combustion CO_2 capture integrated with mechanical vapor recompression technology. Sep. Purif. Technol. 294, 121202.

Li, F., Zhang, J., Oko, E., Wang, M., 2015. Modelling of a post-combustion CO_2 capture process using neural networks. Fuel 151, 156–163.

Li, F., Zhang, J., Shang, C., Huang, D., Oko, E., Wang, M., 2018. Modelling of a post-combustion CO_2 capture process using deep belief network. Appl. Therm. Eng. 130, 997–1003.

Lu, J., Yao, K.e., Gao, F., 2009. Process similarity and developing new process models through migration. AIChE J. 55 (9), 2318–2328.

McBride, K., Sundmacher, K., 2019. Overview of surrogate modeling in chemical process engineering. Chem. Ing. Tech. 91, 228–239.

McIntosh, A.R., Mahalec, V., 1991. Calculation of steady-state gains for multivariable systems from closed-loop steady-state data. J. Process Control 1 (4), 178–186.

Montavon, G., Samek, W., Müller, K.-R., 2018. Methods for interpreting and understanding deep neural networks. Digital Signal Process. 73, 1–15.

Nguyen, H.L.Q., Wong, D.S.H., 2021. Integration of rich and lean vapor recompression configurations for aqueous ammonia-based CO_2 capture process. Chem. Eng. Res. Des. 169, 86–96.

Nuchitprasittichai, A., Cremaschi, S., 2013. An algorithm to determine sample sizes for optimization with artificial neural networks. AIChE J. 59 (3), 805–812.

Ochoa-Estopier, L.M., Jobson, M., Smith, R., 2014. The use of reduced models for design and optimisation of heat-integrated crude oil distillation systems. Energy 75, 5–13.

Oh, S.-Y., Binns, M., Cho, H., Kim, J.-K., 2016. Energy minimization of MEA-based CO_2 capture process. Appl. Energy 169, 353–362.

Parisi, G.I., Kemker, R., Part, J.L., Kanan, C., Wermter, S., 2019. Continual lifelong learning with neural networks: A review. Neural Netw. 113, 54–71.

Rezazadeh, F., Gale, W.F., Rochelle, G.T., Sachde, D., 2017. Effectiveness of absorber intercooling for CO_2 absorption from natural gas fired flue gases using monoethanolamine solvent. Int. J. Greenhouse Gas Control 58, 246–255.

Rochelle, G.T., Wu, Y., Chen, E., Akinpelumi, K., Fischer, K.B., Gao, T., Liu, C.-T., Selinger, J.L., 2019. Pilot plant demonstration of piperazine with the advanced flash stripper. Int. J. Greenhouse Gas Control 84, 72–81.

Rusu, A.A., Rabinowitz, N.C., Desjardins, G., Soyer, H., Kirkpatrick, J., Kavukcuoglu, K., Pascanu, R., Hadsell, R., 2016. Progressive neural networks. arXiv preprint.

Santos Bartolome, P., Van Gerven, T., 2022. A comparative study on Aspen Hysys interconnection methodologies. Comput. Chem. Eng. 162, 107785.

Savitzky, A., Golay, M.J.E., 1964. Smoothing and differentiation of data by simplified least squares procedures. Anal. Chem. 36 (8), 1627–1639.

Sipöcz, N., Tobiesen, F.A., Assadi, M., 2011. The use of artificial neural network models for CO_2 capture plants. Appl. Energy 88 (7), 2368–2376.

Stathakis, D., 2009. How many hidden layers and nodes? Int. J. Remote Sens. 30 (8), 2133–2147.

Terekhov, A.V., Montone, G., O'Regan, J.K., 2015. Knowledge transfer in deep block-modular neural networks. Conference on Biomimetic and Biohybrid Systems. Springer 268–279.

Tikadar, D., Gujarathi, A.M., Guria, C., 2020. Multi-objective optimization of industrial gas-sweetening operations using economic and environmental criteria. Process Saf. Environ. Prot. 140, 283–298.

Trenn, S., 2008. Multilayer perceptrons: Approximation order and necessary number of hidden units. IEEE Trans. Neural Netw. 19 (5), 836–844.

Ullah, A., Soomro, M.I., Kim, W.-S., 2019. Ammonia-based CO_2 capture parameters optimization and analysis of lean and rich vapor compression processes. Sep. Purif. Technol. 217, 8–16.

Wu, Y., Zhou, Q., Chan, C.W., 2010. A comparison of two data analysis techniques and their applications for modeling the carbon dioxide capture process. Eng. Appl. Artif. Intel. 23 (8), 1265–1276.

Yin, L., Gao, Q.i., Zhao, L., Wang, T., 2020. Expandable deep learning for real-time economic generation dispatch and control of three-state energies based future smart grids. Energy 191, 116561.

Yoon, J., Yang, E., Lee, J., Hwang, S.J., 2018. Lifelong learning with dynamically expandable networks. arXiv preprint.

Yosinski, J., Clune, J., Bengio, Y., Lipson, H., 2014. How transferable are features in deep neural networks. Advances in Neural Information Processing Systems 27.

Zhang, X., Song, Z., Gani, R., Zhou, T., 2020. Comparative economic analysis of physical, chemical, and hybrid absorption processes for carbon capture. Ind. Eng. Chem. Res. 59 (5), 2005–2012.

Zhou, Q., Chan, C.W., Tontiwachiwuthikul, P., Idem, R., Gelowitz, D., 2009. A statistical analysis of the carbon dioxide capture process. Int. J. Greenhouse Gas Control 3 (5), 535–544.

Zhuang, Y., Liu, Y., Ahmed, A., Zhong, Z., del Rio Chanona, E.A., Hale, C.P., Mercangöz, M., 2022. A hybrid data-driven and mechanistic model soft sensor for estimating CO_2 concentrations for a carbon capture pilot plant. Comput. Ind. 143, 103747.